



Consiglio Nazionale delle Ricerche

**Scheduling di Traffico Internet  
sulla Rete di Accesso UMTS  
con Interfaccia Radio TD-CDMA**

Marco Conti, Enrico Gregori, Graziano Bini

Nota Tecnica  
CNUCE-B4-2000-027

**CNUCE**

**Pisa**



---

# Indice

---

<b>Introduzione</b> .....	xi
<b>CAPITOLO 1 Introduzione a UMTS</b>	
I.1 Introduzione ai sistemi radiomobili di terza generazione .....	1
I.2 Breve storia della tecnica di accesso CDMA .....	6
I.3 Scelta della tecnica di accesso multiplo .....	8
<i>I.3.1 I principali schemi di accesso</i> .....	8
<i>I.3.2 Concetti fondamentali del CDMA</i> .....	10
I.4 W-CDMA e TD-CDMA .....	18
<i>I.4.1 La tecnica W-CDMA</i> .....	19
<i>I.4.2 La tecnica TD-CDMA</i> .....	20
I.5 Alcune caratteristiche del sistema IMT-2000/UMTS .....	21
<i>I.5.1 Ricevitore RAKE</i> .....	21
<i>I.5.2 Multiuser Detection (Joint Detection)</i> .....	23
<i>I.5.3 Controllo di potenza (Power Control)</i> .....	24

<i>V.2.1 Analisi statica</i> .....	131
<i>V.2.2 Analisi dinamica</i> .....	140
<b>CAPITOLO 6 Conclusioni</b>	
VI.1 Proposta di algoritmi di scheduling .....	151
VI.2 Ulteriori studi .....	159
<b>Lista degli acronimi</b> .....	161
<b>Bibliografia</b> .....	163

---

# INTRODUZIONE

---

L'inarrestabile crescita che ultimi anni ha interessato il settore della telefonia mobile è per molti versi sorprendente. I livelli di utenza e i tassi di penetrazione raggiunti in alcune nazioni come l'Italia dai sistemi radiomobili hanno stupito anche, e forse soprattutto, gli addetti ai lavori.

Le più recenti previsioni indicano che entro il 2003, a livello mondiale, ci saranno più di un miliardo di telefoni mobili in esercizio, il che comporterebbe un superamento del numero di linee fisse telefoniche previste per quella data (cosa peraltro già avvenuta in alcune realtà nazionali come l'Italia).

A questa sbalorditiva crescita si accompagna l'esplosione sul mercato di un altro settore in rapido sviluppo: *Internet*. Su tale fronte ogni mese si registrano circa 18 milioni di nuovi utenti, con un traffico dati che raddoppia ogni 6 mesi circa, al punto che si può ormai asserire, senza tema di smentita, che Internet sta diventando il mezzo più importante per raccogliere e distribuire informazione a livello mondiale.

La telefonia mobile e la trasmissione dati tramite Internet rappresentano quindi, senza ombra di dubbio, i due più grandi successi di fine secolo nel campo delle telecomunicazioni, e, da quanto si può osservare, questo sembra solo il punto di partenza per il nuovo millennio.

In questo scenario si inserisce il nuovo settore di comunicazioni mobili multimediali di terza generazione che permetterà di far confluire in un'unica realtà i passi evolutivi di cui sopra, esaltando per la telefonia mobile e per Internet, le pur promettenti prospettive di crescita separate.

Il sistema **UMTS/IMT-2000** nasce pertanto in una fase di convergenza tra il mondo delle telecomunicazioni e quello dell'*Information Technology*. Il suo sviluppo si presenta come un'ambiziosa sintesi di queste due storie evolutive, soprattutto sul piano dei servizi.

L'introduzione della trasmissione dati tramite Internet rappresenta il primo passo verso nuovi servizi multimediali che costituiscono l'obiettivo finale dei sistemi radiomobili di terza generazione. Tale servizio è già offerto dai sistemi radiomobili di seconda generazione, come il GSM, e a breve sarà disponibile anche su un *backbone* a commutazione di pacchetto nel sistema, cosiddetto di seconda generazione e mezzo, GPRS. Ma sia il sistema GSM che quello GPRS presentano in tale ambito delle grosse limitazioni. I problemi fondamentali sono due: da un lato le velocità di trasferimento sono limitate dalla scarsità delle risorse radio messe a disposizione, dall'altro le ridotte dimensioni e la scarsa capacità di elaborazione dei terminali mobili rende critico il problema della rappresentazione dell'informazione su tali apparati.

L'evoluzione verso UMTS/IMT-2000 mira, tra le altre cose, a superare queste due limitazioni.

In questo report si è data particolare enfasi al primo problema, studiando nel dettaglio le tecniche di accesso radio introdotte nei sistemi di terza generazione.

L'introduzione di nuovi servizi che richiedono bit rate sempre più elevati rende critica la gestione delle risorse radio. In ogni sistema radiomobile difatti la parte di accesso radio costituisce il *bottleneck* per la capacità trasmissiva, per cui, per i nuovi sistemi, si rende necessaria, oltre l'allocazione di nuove bande frequenziali, anche una nuova tecnica di

accesso multiplo e nuovi algoritmi di gestione delle risorse radio. Sullo studio di questi ultimi è incentrato il presente lavoro.

La capacità dell'intero sistema radiomobile è limitata dalla capacità della tratta radio e quest'ultima è strettamente legata all'efficienza degli algoritmi di *scheduling* adottati.

Prima di poter proporre possibili algoritmi di scheduling per la trasmissione dati tramite Internet ed effettuare una analisi simulativa delle loro prestazioni, è stato però necessario un lungo studio delle caratteristiche del sistema UMTS/IMT-2000.

Delle due principali soluzioni proposte dagli organismi di standardizzazione *ETSI* e *ITU* per la tecnica di accesso multiplo si è scelto di analizzare quella cosiddetta **TD-CDMA**. Tale interfaccia radio è costituita da una soluzione ibrida a divisione di tempo e di codice che effettua la suddivisione della capacità per le due direzioni di traffico nel dominio del tempo. Ciò attribuisce la possibilità di modificare il rapporto tra le capacità assegnate all'up e al down link, rendendo particolarmente adatta questa soluzione per la gestione di traffici asimmetrici, quale è tipicamente quello Internet, dove il livello di asimmetria (definito come il rapporto tra i bit trasmessi sul down link e quelli trasmessi sull'up link) può raggiungere valori di svariate decine o centinaia nell'ambito della stessa sessione.

Si è quindi cercato di comprendere e delineare l'attuale stato dell'arte degli studi sulla rete di accesso radio terrestre del sistema UMTS (**UTRAN**, *UMTS Terrestrial Radio Access Network*) in modalità **TDD** (*Time Division Duplex*). Il punto di partenza sono stati i documenti prodotti dal **3GPP** (*Third Generation Partnership Project*), grazie ai quali è stato fatto un quadro dell'attuale stato di avanzamento nella produzione delle specifiche tecniche riguardanti l'interfaccia radio UMTS. Da questi si è partiti per la definizione di un modello di simulazione della rete di accesso radio UMTS. I numerosi punti ancora non specificati nelle normative e quelli lasciati liberi all'implementazione sono stati definiti introducendo ipotesi di funzionamento il più semplici possibili, senza però compromettere l'efficienza degli algoritmi proposti. Tutte le ipotesi comunque sono state introdotte nel completo rispetto di quanto riportato nei documenti del 3GPP.

## Struttura del report

Mostriamo adesso sinteticamente l'organizzazione data a questo lavoro.

I primi due capitoli rappresentano una vasta introduzione al mondo UMTS.

In particolare nel *primo capitolo* è stata data enfasi al problema della tecnica di accesso multiplo adottata nei sistemi wireless di terza generazione. Dopo un breve excursus storico sulle tecniche di accesso a divisione di codice sono presentate le due soluzioni per l'interfaccia radio UMTS, quella con *Frequency Division Duplex*, detta *W-CDMA* e quella basata sul *Time Division Duplex*, *TD-CDMA*. Alla fine del capitolo vengono analizzate, nel contesto del nuovo sistema di accesso a divisione di codice, problematiche tipiche di un sistema radiomobile, come l'*handover* e il controllo di potenza. Inoltre vengono spiegati concetti fondamentali e peculiari del nuovo sistema, come il ricevitore *RAKE*, la *Joint Detection* e l'organizzazione gerarchica in celle sovrapposte del sistema UMTS, detta *Hierarchical Cell Structure* (HCS).

Nel *secondo capitolo* vengono analizzate quelle specifiche tecniche dell'UMTS che, dallo studio dei documenti del 3GPP, appaiono più stabili e possono essere considerate definitive con ragionevole certezza. Si parte da una descrizione generica dell'architettura dell'interfaccia radio, per passare poi alla presentazione dettagliata degli strati protocollari che in essa operano, in particolare dello strato fisico, che ad oggi risulta la sezione meglio definita da parte del 3GPP. Insieme alla struttura della trama e all'elenco dei canali fisici, di trasporto e logici, vengono spiegate le tecniche di modulazione, di *spreading* e di *scrambling* utilizzate nell'UTRAN TDD. Oltre allo strato fisico vengono illustrati le funzioni dei due sottostrati del *data link layer*: lo strato *MAC* e lo strato *RLC*. Un paragrafo specifico è stato dedicato alle possibili infrastrutture di rete previste per i sistemi di terza generazione.

Il *terzo capitolo* spiega le problematiche su cui si è voluto proporre un contributo originale. Vengono mostrate diverse proposte di tecniche di scheduling, nate da un graduale sviluppo delle idee portanti del nostro lavoro. In particolare sono affrontati due problemi: la ripartizione della capacità tra le due direzioni di trasmissione in base alla simmetria del

traffico da gestire e l'adattabilità del nostro algoritmo di scheduling a variazioni delle condizioni di interferenza tra utenti di una stessa cella, tipiche del sistema CDMA.

Nel *quarto capitolo* è descritta l'architettura del simulatore sviluppato in linguaggio *RESQ* per l'analisi delle prestazioni degli algoritmi proposti. Vengono giustificate le assunzioni adottate nel modello di rete UTRA e spiegate, senza entrare nel dettaglio implementativo, le principali caratteristiche del simulatore, come la realizzazione delle procedure di attivazione e rilascio del canale e la tecnica di scheduling. In conclusione di capitolo sono presentati alcuni risultati introduttivi finalizzati a validare il funzionamento del simulatore e dare una rappresentazione del tipo di statistiche prese in esame nella nostra analisi.

Il *quinto capitolo* presenta, attraverso i risultati delle simulazioni, la valutazione delle performance conseguibili con le nostre proposte. Nella prima parte vengono confrontate la tecnica statica e quella dinamica di ripartizione dei time slot tra up e down link. Nella seconda parte vi è una doppia analisi del comportamento degli algoritmi che adattano la capacità del canale alle caratteristiche di interferenza tra gli utenti che condividono il mezzo radio. Ad una prima analisi statica, in cui vengono considerate, una alla volta, differenti funzioni di interferenza, segue un'analisi dinamica in cui la funzione di interferenza viene fatta variare durante la simulazione.

Infine nel *sesto capitolo* vengono tracciate le conclusioni del lavoro svolto. Alla luce dei risultati ottenuti dalle simulazioni viene analizzata una proposta di scheduling per la trasmissione dati tramite Internet che, nell'ottica della massima *fairness* tra gli utenti, sia in grado di adattarsi sia a fluttuazioni del bilanciamento del traffico tra le direzioni di trasmissione, sia alle variazioni delle condizioni interferenziali tra gli utenti di una stessa cella. Il fine di questo algoritmo è massimizzare il throughput della trasmissione dati, in modo da sfruttare il più efficientemente possibile le risorse radio disponibili. Al termine delle conclusioni vengono illustrati gli spazi di ricerca che questo lavoro lascia ad ulteriori studi.



---

# CAPITOLO

# 1

---

## INTRODUZIONE AD UMTS

*In questo primo capitolo verranno analizzati i concetti fondamentali che caratterizzano lo standard IMT-2000/UMTS, con particolare riferimento alla tecnica di accesso multiplo di tipo CDMA specifica di tale sistema.*

### **I.1 Introduzione ai sistemi radiomobili di terza generazione**

La capillare diffusione dei sistemi di telefonia cellulare verificatasi negli ultimi anni è per molti versi sorprendente. La crescita dei sistemi radiomobili ha stupito anche e forse soprattutto gli addetti ai lavori. Basti pensare come all'inizio degli anni '80 gli analisti stimavano per il 2000 circa un milione di utilizzatori, mentre alla fine del 1997 questi erano già oltre 150 milioni e che attualmente circa 400 milioni di persone nel mondo utilizzano questi sistemi, con livelli di penetrazione che in alcune nazioni già oggi superano il 50% della popolazione.

I primi prodotti introdotti sul mercato agli inizi degli anni '80, e cioè i sistemi AMPS (Advanced Mobile Phone Service) negli Stati Uniti, NMT (Nordic Mobile Telephone) nei Paesi Nordici, TACS (Total Access Communication System) nel Regno Unito, sono serviti da trampolino di lancio per una vertiginosa crescita del settore sfociata alcuni anni dopo nella presentazione del GSM, nato come sistema standard europeo.

Il GSM ha rappresentato un importante successo europeo, una delle poche aree in cui l'Europa ha raggiunto una leadership rispetto agli Stati Uniti sia dal punto di vista tecnologico che commerciale. Oggi il GSM, presente in oltre 60 paesi, è il sistema di telefonia mobile più diffuso.

Tuttavia le previsioni sulla domanda di nuovi servizi di telecomunicazioni con terminali mobili hanno messo in luce l'esigenza di predisporre sistemi che dai primi anni del 2000 siano in grado di fornire efficientemente servizi per comunicazioni multimediali, che necessitano di velocità di trasmissione dati certamente più elevate rispetto agli attuali standard.

Per soddisfare questa domanda da alcuni anni presso diversi enti sono stati avviati studi preliminari con l'obiettivo di individuare le tecniche più appropriate per realizzare sistemi radiomobili di "terza generazione".

Il mondo delle comunicazioni radiomobili si è negli ultimi anni concentrato attorno allo sforzo di standardizzare un sistema universale per le reti radiomobili di terza generazione, che proprio in questi anni dovrebbe prendere corpo.

Questo processo è stato sviluppato parallelamente negli Stati Uniti e in Europa dai principali istituti di standardizzazione nel campo delle telecomunicazioni.

Sono nati così due standard: l'IMT-2000 (*International Mobile Telecommunication-2000*), proposto dall'organismo internazionale ITU (*International Telecommunication Union*) e l'UMTS (*Universal Mobile Telecommunications System*) elaborato in Europa dall'ETSI (*European Telecommunication Standard Institute*).

Un ampio dibattito a livello internazionale si è svolto attorno alla scelta dello schema di accesso multiplo per tali reti radiomobili di terza generazione.

---

Recentemente intense ricerche hanno proposto il sistema di accesso multiplo a divisione di codice (*Code Division Multiple Access*) per l'interfaccia aerea di **IMT-2000/UMTS**. Appare oggi chiaro che il CDMA è il più forte candidato per i sistemi di comunicazione personali wireless di terza generazione.

Le richieste incalzanti di servizi con sempre più alti data rate e di efficienza spettrale sempre migliore sono stati i principali driver che hanno portato alla definizione dei nuovi sistemi radiomobili.

Già dal 1985 l'ITU ha iniziato a sviluppare l'IMT-2000, precedentemente chiamato **FPLMTS** (*Future Public Land Mobile Telephone System*), mentre in ETSI la standardizzazione dell'UMTS è iniziata nel 1990 con la creazione del comitato tecnico SMG5.

I principali obiettivi per l'interfaccia aerea IMT-2000/UMTS possono essere riassunti come segue:

- completa copertura con piena mobilità a bit rate di 144 Kb/s (preferibilmente 384 Kb/s),
- limitata copertura con bassa mobilità a bit rate di 2 Mb/s,
- alta efficienza spettrale nei confronti dei sistemi esistenti,
- alta flessibilità finalizzata all'introduzione di nuovi servizi.

I bit rate previsti sono stati scelti in accordo con quelli del sistema ISDN (*Integrated Service Digital Network*), infatti ISDN prevede servizi a 144, 384, 1920 Kb/s rispettivamente con i canali 2B+D, H0 e H12.

Al momento risulta comunque difficile stabilire i servizi offerti dal nuovo sistema e non è detto che essi si baseranno strettamente su quelli dell'ISDN. I data rate offerti nei sistemi commerciali saranno determinati essenzialmente dalla domanda del mercato. La figura mostra le relazioni tra bit rate e mobilità nei sistemi di seconda e terza generazione.

Il maggiore problema nell'analisi dei sistemi di terza generazione è la vastità degli obiettivi proposti. A seconda dei driver considerati principali si possono incontrare soluzioni molto diverse.

---

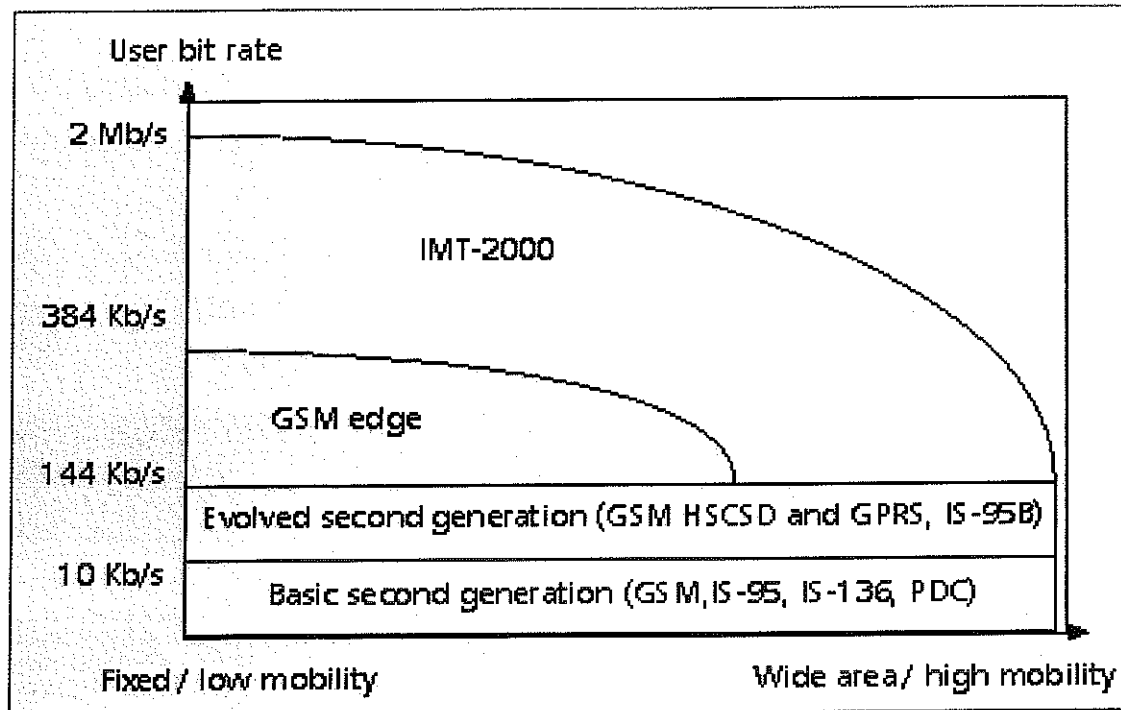


Figura 1. Bit rate di utente vs copertura e mobilità

Inoltre lo sviluppo dei sistemi di seconda generazione non è assolutamente uniforme, si va da una completa maturità con alti tassi di penetrazione in Europa, Stati Uniti e Giappone, fino a bassissime penetrazioni nei paesi in via di sviluppo, dove i sistemi cellulari hanno appena iniziato il loro ciclo di vita. E' chiaro quindi che questa disomogeneità causa differenti necessità e approcci che inevitabilmente portano a diversificare le soluzioni per IMT-2000/UMTS. C'è anche da tenere presente l'influenza che i problemi di compatibilità verso il basso esercitano sulle soluzioni tecnologiche.

Attualmente è comunque evidente che il futuro dei sistemi cellulari di terza generazione risiede nella proposta IMT-2000/UMTS.

Il rapido sviluppo nel processo di standardizzazione avutosi negli ultimi anni va attribuito soprattutto all'iniziativa giapponese. Agli inizi del 1997 l'ARIB (*Association for Radio Industry and Business*), ente responsabile della standardizzazione radio del Giappone, decide di procedere ad una dettagliata standardizzazione del CDMA a larga banda. La tecnologia giapponese accelera la standardizzazione in Europa e negli Stati Uniti. Durante il

1997 vengono concordati i parametri comuni nelle proposte per il CDMA a larga banda europeo e giapponese. Comunemente ci si inizia a riferire all'interfaccia aerea come **WCDMA** (*Wideband CDMA*).

Nel gennaio '98 l'ETSI arriva a scegliere il WCDMA come schema di interfaccia aerea per **UTRA** (*UMTS Terrestrial Radio Access*). Tale scelta è successivamente operata in Asia e dagli operatori americani.

Nascono due soluzioni alternative che, secondo lo standard, devono coesistere nel panorama delle reti di terza generazione. Al WCDMA basato sul *Frequency Division Duplex* (**FDD mode**), cioè sulla divisione in frequenza delle tratte in up-link e in down-link, si affianca il **TD-CDMA** basato sul *Time Division Duplex* (**TDD mode**) che separa le due tratte nel dominio del tempo.

Negli Stati Uniti nel 1998, il comitato TR45.3 del **TIA** (*Telecommunications Industry Association*) stabilisce un sistema detto *cdma2000* tipo WCDMA compatibile verso il basso con il sistema **IS-95** (uno dei primi sistemi con tecnica di accesso multiplo di tipo CDMA a banda stretta, *narrowband CDMA*).

I fattori che stanno influenzando le scelte tecniche per i sistemi di terza generazione sono svariati e possono essere di natura tecnica, politica e commerciale.

I fattori tecnici coinvolgono argomenti quali il soddisfacimento dei data rate e delle performance richiesti dai vari servizi.

I fattori politici includono il raggiungimento di accordi tra i vari corpi di standardizzazione internazionali e nazionali e tengono conto dei vari punti di partenza di differenti regioni e paesi.

Per quel che riguarda gli aspetti commerciali, da un lato gli ingenti investimenti effettuati nel sistema esistente rendono obbligatorio un approccio di compatibilità verso il basso; dall'altro nuove opportunità di investimento o la possibilità di cambiare l'attuale situazione potrebbero motivare a spingersi verso un approccio differente.

---

## I.2 Breve storia della tecnica di accesso CDMA

Vediamo brevemente come si è giunti storicamente all'introduzione della tecnica di accesso multiplo a divisione di codice nei sistemi radiomobili.

Il concetto di "*spread spectrum*" prende origine in campo militare e nei sistemi di navigazione. Tale tecnica nasce per combattere il jamming intenzionale e la possibilità di intercettazione, ma trova successivamente largo impiego nelle applicazioni in ambiente cellulare attraverso canali dispersivi.

Nel 1949 John Pierce scrive un memorandum tecnico in cui descrive un sistema di multiplexing nel quale i segnali opportunamente codificati condividono sovrapposti nel tempo e in frequenza un mezzo comune senza sincronizzazione.

Tale sistema può essere classificato nell'ambito dei sistemi di accesso multiplo come *time hopping spread spectrum* (vedi fig.4, pag 18).

Claude Shannon e Robert Pierce introducono le idee base del CDMA nel 1949 descrivendone gli effetti di ripartizione dell'interferenza.

Nel 1950 De Rosa e Rogoff propongono per la prima volta un sistema spread spectrum di tipo *direct sequence* e introducono l'equazione di *guadagno di processo* (par. I.3.2).

Nel 1956 Price e Green registrano il brevetto del ricevitore "*RAKE*" *antimultipath*. Utilizzando un segnale a spettro espanso, i segnali che arrivano attraverso differenti cammini di propagazione possono essere separati e combinati dal RAKE receiver (par. I.5.1).

Magnuski ha invece il merito di aver individuato per primo nel 1961 il problema *near-far*, tipico problema dell'ambiente cellulare in cui, se tutti gli utenti trasmettono con la stessa potenza, il segnale debole ricevuto da un utente lontano viene coperto dall'interferenza prodotta dagli utenti vicini che condividono lo stesso canale (par. I.5.2).

I primi a proporre la tecnica di spread spectrum in ambiente cellulare sono stati Cooper e Nettleton nel 1978.

---

Negli anni '80 la **Qualcomm** ha studiato le tecniche di accesso multiplo a divisione di

Pioneer Era	
1949	John Pierce: time hopping spread spectrum
1949	Claude Shannon and Robert Pierce: basic ideas of CDMA
1950	De Rosa-Rogoff: direct sequence spread spectrum
1956	Price and Green: antimultipath "RAKE" patent
1961	Magnuski: near-far problem
1970s	Several developments for military field and navigation systems
Narrowband CDMA Era	
1978	Cooper and Nettleton: cellular application of spread spectrum
1980s	Investigation of narrowband CDMA techniques for cellular applications
1986	Formulation of optimum multiuser detection by Verdu
1993	IS-95 standard
Wideband CDMA Era	
1995	Europe: FRAMES FMA2
	Japan: Core-A
	USA: cdma2000
	Korea: TTA I TTA II
2000s	Commercialization of wideband CDMA systems

Tabella 1. Sviluppo storico del CDMA

codice a sequenza diretta (*DS-SS-CDMA*). Questi studi hanno condotto alla commercializzazione di un sistema di comunicazione cellulare basato su tecnica di spread spectrum sotto forma di CDMA a banda stretta (*narrowband CDMA*), standardizzato con il nome di **IS-95** nel 1993. L'effettiva operatività commerciale dell'IS-95 è iniziata nel 1996.

Intense ricerche sono poi state effettuate attorno al *multiuser detection* (**MUD**) da quando nel 1986 Verdu ha formulato un MUD ottimo per canale AWGN (Additive White Gaussian Noise). Vedremo nel paragrafo I.4.5 come le tecniche di multiuser detection servano a limitare l'interferenza intracella tipica dei sistemi cellulari a divisione di codice.

Durante tutti gli anni '90 tecniche CDMA a larga banda (*wideband CDMA*) con bande di 5 MHz o più sono state studiate intensamente in tutto il mondo e diversi sistemi sperimentali sono stati costruiti e testati. Tra questi ci sono *FRAMES Multiple Access* in Europa, *Core-A* in Giappone, lo schema *WCDMA* che armonizza le soluzioni europea e giapponese, *cdma2000* negli USA, e gli schemi del *TTA I* e *II* (*Telecommunication Technology Association*) in Corea.

Una rappresentazione schematica della storia del CDMA porta a dividere la sua evoluzione in tre periodi: il periodo pionieristico, quello della banda stretta e quello della larga banda, come mostrato nella tabella 1.

## **I.3 Scelta della tecnica di accesso multiplo**

Nella standardizzazione dei sistemi radiomobili di terza generazione uno degli aspetti cruciali è la condivisione del mezzo radio da parte degli utenti. Da un lato c'è il problema della scarsità delle risorse radio disponibili per l'accesso alla rete, dall'altro quello dei nuovi servizi che richiedono bit rate sempre più elevati (oltre alla voce e ai dati a bassi bit rate, nel IMT-2000/UMTS si vogliono garantire servizi come video telefono, video conferenza e Internet, i quali necessitano di trasferimenti ad elevate velocità se si vogliono ottenere qualità accettabili). Tutto questo rende decisiva dal punto di vista tecnologico la scelta della tecnica di accesso multiplo che deve consentire di poter sfruttare in maniera efficiente le risorse radio disponibili, garantendo una elevata efficienza spettrale.

### **I.3.1 I principali schemi di accesso**

Con il termine **accesso** si intendono tutte le modalità secondo le quali gli utenti accedono alla risorsa radio disponibile e la condividono. Queste tecniche da un lato devono mostrarsi efficaci, ma al tempo stesso devono anche minimizzare i costi di installazione e di gestione, le dimensioni degli apparati, resistere alle condizioni ostili di propagazione ed interferenza.

Le principali tecniche di accesso utilizzate per le applicazioni radiomobili sono tre:

- La tecnica **SCPC/FDMA** (*Single Channel Per Carrier / Frequency Division Multiple Access*) è utilizzata nella maggior parte dei sistemi radiomobili analogici. Consiste nel suddividere lo spettro delle frequenze in sottobande ognuna associata ad un singolo canale. La sottobanda dovrà essere larga a sufficienza per convogliare una portante modulata con un solo segnale di utente, identificato dalla sua frequenza portante. Il vantaggio evidente è quello di una
-

---

grande semplicità e robustezza sia della tecnica in sé, sia degli apparati che la costituiscono. Tuttavia rende necessaria la presenza di un apparato di modulazione-demodulazione per ogni canale e l'utilizzo di filtri particolarmente selettivi. Nel caso particolare delle applicazioni radiomobili le portanti radio non vengono associate agli utenti in modo rigido. In realtà viene fatto un riassegnamento ogni volta che si instaura una nuova comunicazione; tale associazione permane fino a quando la connessione è in atto nell'ambito della stessa cella (può invece variare se la stazione mobile passa sotto il controllo di un'altra cella, cioè nel caso di handover).

- La tecnica TDMA (*Time Division Multiple Access*) è utilizzata soprattutto nei sistemi radiomobili di seconda generazione di tipo numerico. Si utilizza, ad esempio, congiuntamente alla tecnica FDMA nel sistema europeo GSM, nell'americano ADC (*American Digital Cellular*), e nel giapponese JDC (*Japanese Digital Cellular*). Permette di trasmettere nello stesso canale diversi segnali di utente, ognuno occupante brevi intervalli temporali secondo i dettami di una trama accordata. Dunque si associa ad ogni canale di utente uno di questi intervalli di trama, caratterizzandolo in maniera univoca. Con questa tecnica si può così gestire un'unica catena di trasmissione e ricezione e sfruttare al meglio le potenze in gioco. Richiede una selettività dei filtri (che in questo caso sono delle porte temporali) meno spinta che nel caso SCPC/FDMA, perché la banda associata alla singola portante radio è più grande dato che in questo caso non vengono realizzati sottosezionamenti della risorsa spettrale complessiva. Tuttavia la potenza di picco dei trasmettitori deve venire regolata secondo il numero totale di canali presenti. Pertanto le stazioni mobili devono potere erogare queste potenze di picco, anche se solo per la breve trama di competenza di ogni sottocanale. A questa tecnica, nella tratta in down link, si può associare la TDM (*Time Division Multiplexing*), con cui si modula una sola portante con un unico
-

flusso numerico su cui vengono multiplate nel tempo tutte le informazioni da trasmettere.

- La tecnica **CDMA** (*Code Division Multiple Access*) è utilizzata per il sistema americano e se ne prevede l'uso per i sistemi di terza generazione. Gli utenti vengono separati gli uni dagli altri tramite l'uso di codici ortogonali e il segnale di utente viene disperso su una banda molto maggiore di quella strettamente necessaria. L'allargamento dello spettro veniva un tempo utilizzato solo in ambito militare per limitare la probabilità di intercettazione e fronteggiare i disturbatori intenzionali (*jammer*); ora viene applicato anche in campo civile con due tecniche diverse a seconda della complessità tecnologica.

### I.3.2 Concetti fondamentali del CDMA

In una tecnica di accesso multiplo CDMA a ciascun utente è assegnata un'unica sequenza di codice che esso usa per codificare il suo segnale informativo. Il ricevitore conoscendo la sequenza di codice dell'utente, dopo averlo ricevuto decodifica il segnale recuperando i dati originali. Questo è reso possibile dalla bassa crosscorrelazione esistente tra il codice dell'utente desiderato e quello degli altri utenti.

La banda del segnale codificato risulta essere molto più larga di quella del segnale informativo in quanto il processo di codifica allarga lo spettro del segnale. E' per questo che la modulazione è anche detta modulazione "*spread-spectrum*" (a spettro espanso) e il CDMA è spesso chiamato "*spread spectrum*" multiple access (SSMA).

E' proprio l'allargamento spettrale del segnale trasmesso che conferisce al CDMA la sua capacità di accesso multiplo. Risulta quindi importante conoscere le tecniche di allargamento dello spettro e le proprietà di tali segnali.

Il rapporto tra la banda trasmessa ( $B_t$ ) e quella del segnale informativo ( $B_i$ ) è chiamato **guadagno di processo o di espansione**,  $G_p$ , del sistema a spettro espanso:

$$G_p = \frac{B_t}{B_i}$$

---

(Nella terminologia IMT-2000/UMTS tale guadagno è detto *Spreading Factor* (SF)).

Il segnale ricevuto viene correlato con una replica generata in modo sincrono del codice di spreading usato in trasmissione. E' quindi assunto che il ricevitore conosca il codice usato in trasmissione per modulare i dati.

In virtù della codifica e del conseguente allargamento spettrale, i segnali SS (spread spectrum) presentano delle proprietà specifiche rispetto ai segnali a banda stretta. Le più importanti dal punto di vista dei sistemi di comunicazioni sono presentate di seguito.

- *Capacità di accesso multiplo*. Se più utenti trasmettono contemporaneamente segnali a spettro espanso, ai ricevitori tali segnali giungono sovrapposti nel tempo e in frequenza. Nonostante ciò il ricevitore è ancora in grado di recuperare il segnale informativo di un singolo utente a patto che esso abbia utilizzato una sequenza di codice unica e con una crosscorrelazione con le sequenze di codice degli altri utenti sufficientemente bassa.

In queste condizioni infatti correlando il segnale ricevuto (somma di tutti i segnali trasmessi contemporaneamente) con la sequenza di codice di un particolare utente solo il suo segnale verrà decodificato (despread) tornando ad occupare la banda stretta del segnale informativo originale, mentre tutti gli altri segnali rimarranno codificati sulla larga banda.

Nella banda di informazione quindi la potenza del segnale desiderato è molto più grande di quella degli interferenti e, se non ci sono troppi interferenti, è possibile estrarre il segnale desiderato.

La figura 2 illustra il principio appena esposto. I segnali a banda stretta 1 e 2 vengono codificati in trasmissione e subiscono un allargamento spettrale (fig. 2.a). In ricezione essi giungono temporalmente sovrapposti. Dopo l'operazione di despreading (moltiplicazione per il codice dell'utente desiderato), il segnale 1 torna nella banda stretta, mentre il segnale 2 resta disperso sulla banda larga. L'interferenza del segnale 2 sul segnale 1 sarà tanto più piccola, quanto maggiore è il suo allargamento spettrale, cioè quanto maggiore è il guadagno di espansione.

---

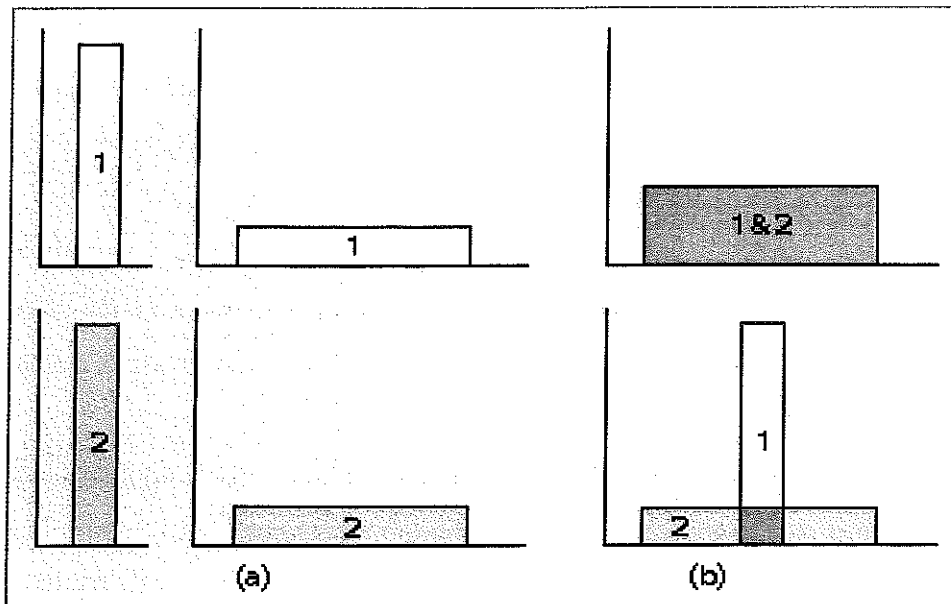


Figura 2. Principio di accesso multiplo per segnali spread-spectrum

- *Protezione contro le interferenze da cammini multipli.* In un canale radio non vi è mai un solo cammino tra trasmettitore e ricevitore. A causa delle riflessioni e delle rifrazioni un segnale sarà sempre ricevuto attraverso una serie di cammini differenti. In ricezione si ha quindi la sovrapposizione dei segnali provenienti da diversi cammini che sono tutti copie del segnale trasmesso, ma con differenti ampiezze, fasi, ritardi e angoli di arrivo. La somma di tutti questi segnali che giunge al ricevitore produrrà un fenomeno di interferenza di tipo costruttivo su alcune frequenze e distruttivo su altre, il che nel dominio del tempo equivale ad una dispersione del segnale.

La modulazione a spettro espanso può combattere questo tipo di interferenza da cammini multipli (multipath interference). Se infatti la sequenza di codice utilizzata ha una funzione di autocorrelazione ideale allora essa presenta un picco nell'origine ed è nulla al di fuori dell'intervallo  $[-T_c; T_c]$ , dove  $T_c$  è la durata di un simbolo di chip. Questo significa che se il segnale desiderato è ricevuto insieme ad una sua versione ritardata di più di  $2T_c$ , demodulando coerentemente, la versione ritardata sarà trattata come un segnale interferente e solo una piccola parte della sua potenza cadrà nella banda informativa.

Questa proprietà è denominata “macrodiversità” e può essere vantaggiosamente sfruttata mediante il ricevitore RAKE (par. I.5.1).

- *Privacy*. Il processo di despreading in ricezione e il conseguente recupero dei dati trasmessi può essere effettuato solo se si conosce la sequenza di codice in trasmissione.
- *Reiezione dell'interferenza di un segnale a banda stretta*. Se il segnale a spettro espanso riceve interferenza da un segnale a banda stretta, in ricezione il processo di despreading allarga il segnale interferente la cui potenza risulta dispersa su una banda molto più larga di quella del segnale informativo, riducendone quindi l'interferenza sulla banda del segnale informativo. Questa, insieme alla successiva, è una delle proprietà che rendono le modulazioni a spettro espanso attraenti dal punto di vista delle applicazioni militari, infatti un disturbo intenzionale a banda stretta (jamming) trasmesso sulla banda informativa del segnale viene reittato dal processo di despreading in ricezione (capacità anti-jamming).

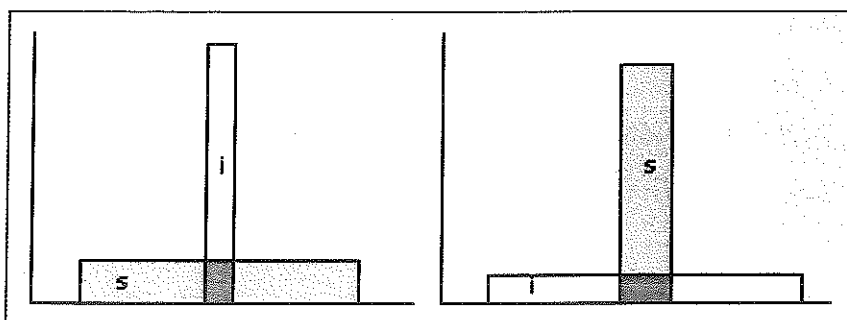


Figura 3. Reiezione di un segnale interferente a banda stretta

- *Bassa probabilità di intercettazione*. Il segnale trasmesso a spettro espanso ha una densità di potenza molto più bassa del segnale informativo ed è per questo difficilmente intercettabile da un ascoltatore nemico.

Esistono diverse tecniche di modulazione che generano segnali a spettro espanso. Le due più importanti sono:

- Il *DS-CDMA* (Direct Sequence CDMA) permette di utilizzare contemporaneamente la stessa portante sulla stessa banda in frequenza, sfruttando la separazione introdotta da sequenze pseudocasuali associate a ciascun canale di utente. I segnali di informazione vengono moltiplicati dalla propria sequenza di codice ad una velocità di cifra molto maggiore di quella che si avrebbe con la frequenza di funzionamento usuale, e conseguentemente dispersi su una banda molto più larga di quella che il canale di informazione richiederebbe.
- Il *FH-CDMA* (Frequency Hopping CDMA) consiste invece nel suddividere il segnale trasmesso in intervalli di tempo e modulare ciascuno di essi con una diversa frequenza portante, scelta all'interno della banda disponibile in accordo ad una sequenza di codice di tipo pseudo-aleatorio. Si distinguono due tipi di FH-CDMA in base alla velocità di salto della portante. Se la velocità di salto (hopping rate) è molto più grande della velocità di simbolo allora si parla di *fast frequency hopping* (F-FH). In questo caso la frequenza portante varia un certo numero di volte durante la trasmissione di un unico simbolo ed un unico bit è trasmesso su più portanti. Nel caso contrario si parla invece di *slow frequency hopping* (S-FH) in cui più simboli sono trasmessi sulla stessa portante.

Le due tecniche differiscono sostanzialmente per quanto riguarda l'occupazione in frequenza. Mentre un segnale DS-SS occupa l'intera banda disponibile durante la trasmissione, un segnale FH-SS occupa soltanto una piccola porzione di tale banda la cui disposizione nello spettro frequenziale cambia però nel tempo.

Le modulazioni DS-CDMA e FH-CDMA presentano particolari vantaggi e svantaggi.

Per quanto riguarda la DS-CDMA tra i vantaggi possiamo contare:

- la facilità di codifica del segnale effettuata attraverso una semplice moltiplicazione,
-

- la generazione di una sola frequenza portante che semplifica il sintetizzatore,
- la possibilità di demodulazione coerente del segnale,
- la non necessaria sincronizzazione tra utenti.

Tra gli svantaggi invece annoveriamo:

- la difficoltà nell'acquisizione e nel mantenimento della sincronizzazione tra il codice generato localmente dal ricevitore e quello del segnale trasmesso,
- la limitazione della banda occupata ad un massimo di 10-20 MHz a causa del fatto che, congiuntamente alla indisponibilità di larghe bande frequenziali continue, per una ricezione corretta, l'errore di sincronizzazione tra la sequenza di codice generata localmente e quella del segnale ricevuto deve essere molto piccolo (sotto una frazione del tempo di chip),
- la difficoltà di realizzare efficienti algoritmi di power control necessari a gestire l'effetto near-far, cioè a fare in modo che tutti gli utenti siano ricevuti dalla BS con la stessa potenza media.

Per la FH-CDMA consideriamo invece i seguenti vantaggi:

- la sincronizzazione è più semplice. Come nel DS-CDMA anche nel FH-CDMA l'errore di sincronizzazione deve essere mantenuto sotto una frazione del tempo di salto (hop time). In realtà però in quest'ultimo sistema l'allargamento spettrale non è ottenuto aumentando la frequenza di salto, ma il numero di frequenze di salto (hop-set); per cui il tempo di salto risulta molto più grande del tempo di chip di un sistema DS-CDMA ed è per questo che il sistema tollera un errore di sincronizzazione più elevato,
  - non è necessario che le bande occupate dal segnale in salti consecutivi siano contigue, in quanto il sintetizzatore di frequenza può facilmente saltare su qualunque parte dello spettro. Combinato con una più semplice sincronizzazione ciò permette di conseguire una occupazione di banda più elevata,
  - la probabilità che due utenti trasmettano nello stessa banda allo stesso tempo è piccola. Ciò rende meno problematico l'effetto near-far, poichè utenti lontani e
-

vicini alla stazione base trasmettono con elevata probabilità su frequenze differenti e quindi non interferiscono tra loro. Potendo offrire la possibilità di trasmettere su bande più larghe i sistemi FH offrono anche una maggiore riduzione dell'interferenza da segnali a banda stretta rispetto ai sistemi DS.

Gli svantaggi di tale tecnica invece sono:

- la necessità di un sintetizzatore di frequenza più sofisticato,
- l'interruzione del segnale durante il cambio di frequenza dovuta al fatto che una improvvisa variazione del segnale durante il salto produrrebbe un incremento della banda occupata,
- la difficoltà nella realizzazione di una demodulazione coerente a causa della problematicità del mantenimento delle relazioni di fase durante il salto di frequenza.

E' interessante notare che, fissata la banda complessivamente disponibile  $B$  e il tempo di simbolo  $T_s$  di un sistema, il numero di canali ortogonali che si possono avere al massimo vale  $N=2T_sB$  indipendentemente dalla tecnica di accesso che viene scelta.

Sfruttando il teorema di Shannon sulla capacità del canale AWGN (Additive White Gaussian Noise) è possibile confrontare pertanto la capacità di ciascuno schema, partendo dall'ipotesi che siano presenti  $K$  utenti fra di loro ortogonali, ognuno dei quali ha a disposizione una potenza  $P$ .

Nel caso FDMA ognuno degli utenti presenti nel sistema ha a disposizione una banda  $B/K$ , pertanto la sua capacità risulta:

$$C_i = \frac{B}{K} \log_2 \left[ 1 + \frac{P}{(B/K)N_0} \right] \quad [\text{bit / s / Hz}]$$

In maniera del tutto analoga, nel caso sia stato scelto un sistema TDMA, ogni utente trasmette per una frazione del tempo totale disponibile pari ad  $1/K$ , ma utilizza l'intera

---

banda disponibile, con una potenza media  $PK$ ; questa volta la capacità per utente risulta essere:

$$C_i = \frac{1}{K} B \log_2 \left[ 1 + \frac{PK}{BN_0} \right] \text{ [bit / s / Hz]}$$

che è identica al caso FDMA a parità di energia per bit.

Nel caso si consideri infine uno schema di accesso del tipo CDMA, si può supporre, se si considera un caso di sistema con  $K$  codici perfettamente ortogonali, che ad ogni utente venga riservata una banda equivalente pari a  $B/K$ , con una potenza di rumore ridotta di un fattore  $K$ . Il risultato è una legge identica a quella del caso FDMA.

Pertanto nella scelta dello schema di accesso da utilizzare si deve tenere conto del fatto che, in un sistema cellulare, la capacità è limitata principalmente dall'interferenza degli utenti che riutilizzano lo stesso canale nelle altre celle. Inoltre, nel caso CDMA, va tenuto in conto che non è possibile utilizzare codici ortogonali nella tratta in uplink, a causa della mancanza di sincronismo tra i segnali ricevuti dalla stazione base, per cui l'interferenza proviene anche dagli utenti della stessa cella, e in maniera preponderante rispetto a quella esterna che è invece mitigata invece dalla distanza.

Oltre a quelle citate esistono modulazioni ibride in cui due o più tecniche possono essere utilizzate insieme per combinarne i vantaggi o combatterne gli svantaggi.

Inoltre è anche possibile combinare il CDMA con altre tecniche di accesso multiplo come ad esempio la TDMA (TD-CDMA).

Nella figura successiva sono classificati i principali tipi di CDMA. Nei CDMA puri oltre ai già citati DS e FH viene annoverato anche il Time Hopping (TH), che esegue nel tempo le stesse operazioni che FH esegue in frequenza.

---

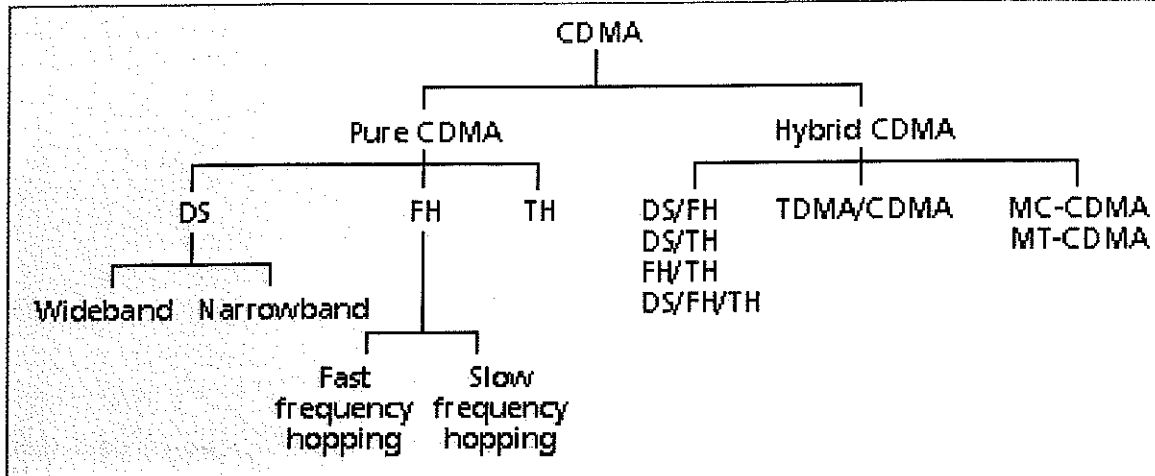


Figura 4. Classificazione delle principali tecniche di accesso CDMA

## I.4 W-CDMA e TD-CDMA

Analizzate le principali tecniche di accesso CDMA, siamo ora in grado di presentare le soluzioni proposte per realizzare l'interfaccia radio dei sistemi radiomobili di terza generazione.

Le proposte dei vari gruppi di standardizzazione sono confluite nelle due tecniche seguenti, entrambe con accesso di tipo DS-CDMA:

- W-CDMA
- TD-CDMA

In un primo momento è stata proposta anche una tecnica del tipo OFDM (Orthogonal Frequency Division Multiplexing), la cui efficienza è garantita soprattutto in ambito microcellulare di tipo indoor, ma allo stato attuale non sembra che questa ipotesi abbia avuto successo.

La tecnica **W-CDMA** (*Wideband – Code Division Multiple Access*) è quella che permette di fornire con maggior flessibilità servizi cellulari su vaste aree, sia per la trasmissione della voce sia per trasmissioni dati fino a 2 Mbit/s. D'altra parte l'analisi della domanda ha messo in evidenza una significativa e crescente richiesta di servizi con requisiti

di trasmissione asimmetrici per i due versi di trasmissione; per questo tipo di servizi la tecnica **TD-CDMA** (*Time Division – Code Division Multiple Access*) si presta certamente meglio dell'altra. Essa infatti prevede l'utilizzo della stessa frequenza portante nei due versi di trasmissione, ma si trasmette in intervalli di tempo diversi, cosicché dimensionando opportunamente gli intervalli, si possono sbilanciare le capacità di trasporto nelle due direzioni.

E' stata pertanto predisposta l'adozione della tecnica W-CDMA per le trasmissioni simmetriche nelle bande appaiate e invece l'impiego della soluzione TD-CDMA per le

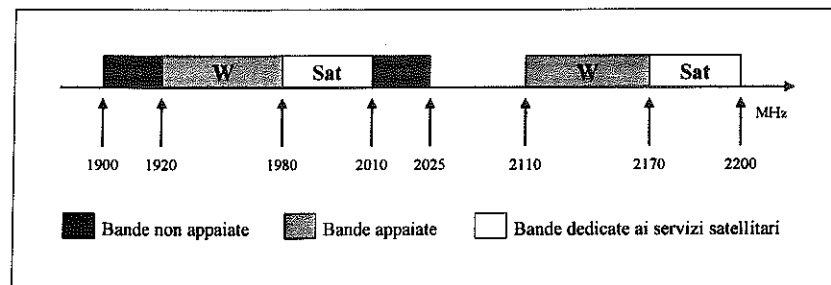


Figura 5. Frequenze assegnate in Europa all'UMTS

trasmissioni asimmetriche nelle bande non appaiate. La figura 5 mostra l'allocazione delle frequenze in Europa: per la soluzione W-CDMA è oggi prevista una coppia di bande ciascuna dell'ampiezza di 60 MHz, mentre per la TD-CDMA sono assegnate due porzioni di banda che coprono complessivamente 35 MHz.

Diamo ora una breve descrizione delle due tecniche di accesso radio.

#### I.4.1 La tecnica W-CDMA

La soluzione W-CDMA prevede una tecnica di accesso DS-SS-SS-SS in cui i segnali di diversi utenti sono trasmessi contemporaneamente sullo stesso canale radio e alla stessa frequenza portante. Fra questi segnali si creano quindi interferenze reciproche; il processo di codifica e decodifica attuato sui segnali elimina tuttavia l'effetto delle interferenze.

A ciascun segnale da trasmettere sul canale radio è infatti associata, con un'operazione di

moltiplicazione, una sequenza numerica (codice) con una velocità di trasmissione (chip rate) molto maggiore della velocità dell'informazione da trasmettere (bit rate). Questa sequenza di codice è nota al ricevitore, che, con un'operazione di correlazione, può recuperare il contenuto dell'informazione.

Le sequenze di codice associate ai vari segnali che condividono lo stesso canale sono fra loro diverse e sono scelte in modo che siano "ortogonali": è cioè nulla la correlazione tra sequenze di codice diverse.

Come già osservato, il rapporto tra il "chip rate" del codice ed il "bit rate" dell'informazione è detto Spreading Factor (SF). Questo perché la banda occupata dal segnale codificato è maggiore di quella che avrebbe un segnale non codificato approssimativamente di un fattore pari allo SF, ma questa perdita di efficienza spettrale è in realtà più che compensata dalla possibilità di sovrapporre più segnali sullo stesso canale radio. Quanto più alto è lo SF, tanto maggiore è la robustezza all'interferenza, e quindi tanto maggiore è il numero di segnali che si possono sovrapporre sullo stesso canale.

La soluzione adottata in ambito IMT-2000/UMTS è detta "a larga banda" (wideband) perché il chip rate è molto maggiore di quello dei sistemi CDMA attualmente in esercizio. Sono infatti previsti tre possibili valori pari a 4096 kchip/s, 8192 kchip/s e 16384 kchip/s.

E' da notare che pur mantenendo fisso il chip rate, la capacità di trasporto dell'informazione può essere variata entro ampi margini variando lo SF (tra un minimo di 4 e un massimo di 512) e adattando proporzionalmente la potenza di trasmissione. Questa caratteristica conferisce alla tecnica W-CDMA una grande flessibilità nell'assegnazione dei bit rate a ciascun canale.

### I.4.3 La tecnica TD-CDMA

La soluzione TD-CDMA è una combinazione di una tecnica di accesso a divisione di tempo (come quella usata nel GSM) e di una a divisione di codice (come la W-CDMA).

Secondo la tecnica a divisione di tempo, la trasmissione su ogni portante radio è organizzata in trame suddivise in  $n$  intervalli temporali (time slot). A differenza del GSM,

---

ciascuno di questi intervalli invece di essere dedicato ad un unico collegamento può essere contemporaneamente impiegato da  $m$  collegamenti diversi, sovrapponendo in esso  $m$  segnali con una tecnica a divisione di codice come quella sopra descritta (in questo caso SF e quindi  $m$  può variare tra 1 e 16). Su ciascuna portante radio si possono così moltiplicare fino a  $n \times m$  canali.

Come accennato in precedenza, per realizzare canali asimmetrici nei due versi di trasmissione basta dedicare una parte degli intervalli di tempo ad un verso e quella rimanente al verso opposto.

Le tecniche utilizzate per realizzare la suddivisione delle due direzioni di trasmissione vengono dette *Frequency Division Duplex (FDD)* per la W-CDMA e *Time Division Duplex (TDD)* per la TD-CDMA.

Nel nostro studio verrà presa in considerazione esclusivamente la tecnica TDD. Infatti la trasmissione su canali asimmetrici, che la TDD permette di realizzare in maniera molto semplice e flessibile, è caratteristica di molti dei servizi disponibili oggi in Internet (basti pensare ad esempio al trasferimento di un file). Sarà pertanto presentata una descrizione esauriente dell'implementazione di questa tecnica nell'interfaccia radio UMTS.

## **I.5 Alcune caratteristiche del sistema IMT-2000/UMTS**

Vediamo ora i concetti fondamentali per la realizzazione di un sistema CDMA a sequenza diretta e per la sua applicazione nei sistemi di terza generazione. Quanto successivamente esposto, se non esplicitamente indicato, è valido sia per l'interfaccia radio di tipo W-CDMA, sia per quella TD-CDMA, in quanto entrambe utilizzano una codifica a sequenza diretta.

### **I.5.1 Ricevitore RAKE**

La forma d'onda di un segnale a spettro espanso si adatta bene ad un canale con cammini multipli. In un canale con cammini multipli, quale è tipicamente un canale radio, il segnale

---

trasmesso originariamente è riflesso dagli ostacoli come costruzioni e montagne e giunge al ricevitore sotto forma di più copie con differenti ritardi. Il canale risulta quindi essere selettivo in frequenza e una parte del segnale risulterà affetta da fading.

Se i segnali arrivano ritardati l'uno dall'altro di più di un intervallo di chip, allora il ricevitore è in grado di distinguerli. Infatti dal punto di vista di un singolo segnale ricevuto, gli altri segnali possono essere visti come interferenza ed essere soppressi nel processo di despreading (vedi par. I.3.2, protezione da interferenza da cammini multipli). Un ulteriore vantaggio può essere ottenuto se i segnali provenienti da cammini multipli sono combinati nel ricevitore RAKE.

Il ricevitore RAKE consiste in un banco di correlatori, ciascuno adattato al segnale proveniente da un diverso cammino multiplo. Dopo il despreading effettuato dai correlatori i segnali sono combinati assieme opportunamente.

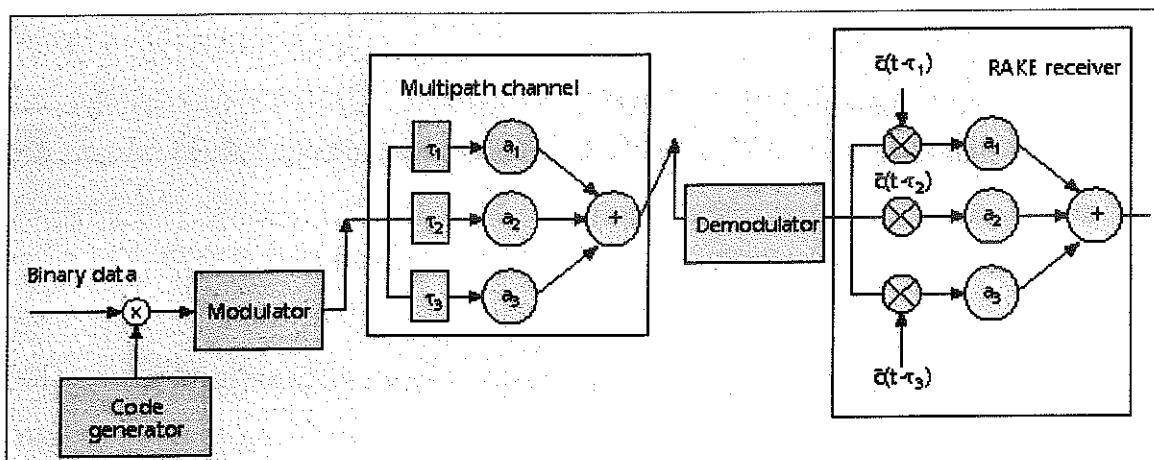


Figura 6. Principio di funzionamento del ricevitore RAKE

Il principio di funzionamento del ricevitore RAKE è illustrato in figura 6. Dopo lo spreading e la modulazione il segnale è trasmesso attraverso il canale dispersivo, che può essere modellato come un insieme di linee di ritardo. In figura abbiamo tre componenti multicammino con differenti ritardi ( $t_1$ ,  $t_2$ ,  $t_3$ ) e fattori di attenuazione ( $a_1$ ,  $a_2$ ,  $a_3$ ), ciascuna corrispondente ad un differente cammino di propagazione. Il ricevitore RAKE presenta un ramo di ricezione per ciascuna componente. In ciascun ramo il segnale ricevuto è correlato

con una sequenza di codice allineata temporalmente con il ritardo subito nel cammino dal segnale. Dopo il despreading i segnali sono opportunamente pesati e combinati. Nell'esempio in figura, prima di effettuarne la somma, ciascun segnale è pesato con un fattore pari all'attenuazione subita nel relativo cammino (maximal ratio combining).

Il movimento del mobile rende problematico il dimensionamento dei rami del RAKE receiver in quanto produce un cambiamento dell'ambiente di scattering, con conseguente modifica del profilo del canale, cioè del numero di cammini multipli e dei relativi ritardi e fattori di attenuazione. E' necessario quindi stimare dinamicamente tale profilo ed adattare conseguentemente i rami del ricevitore.

### I.5.2 Multiuser Detection (Joint Detection)

Come visto, i ricevitori CDMA sono basati sul principio di funzionamento del ricevitore RAKE, in cui i segnali degli altri utenti sono considerati interferenza. La capacità di un sistema CDMA a sequenza diretta è quindi limitata dall'interferenza. In pratica questo significa che quando un nuovo utente o interferente entra nella cella, gli altri utenti percepiscono una degradazione della qualità del servizio, che può scendere sotto la soglia di accettabilità. Più la cella è in grado di resistere all'interferenza, più utenti possono essere serviti. L'interferenza complessiva dovuta all'accesso multiplo risulta essere la somma dell'interferenza intra- ed inter-cella.

Gli algoritmi di MUD (*Multiuser detection*), detti anche *joint detection* o *interference cancellation (IC)*, mirano a ridurre l'effetto dell'interferenza intracella da accesso multiplo e quindi ad incrementare la capacità del sistema. Il MUD in realtà effettua sostanzialmente una cancellazione dell'interferenza intracella, per cui la capacità del sistema è limitata dall'efficienza di tale algoritmo e dalla interferenza intercella.

In aggiunta al miglioramento della capacità, il MUD allevia anche il problema del near-far tipico dei sistemi DS-CDMA. Una utente vicino alla stazione base può bloccare il traffico complessivo della cella utilizzando una potenza di trasmissione troppo alta. Se l'utente è rilevato per primo e il suo segnale informativo decodificato sottratto con tecniche

---

MUD dal segnale in ingresso alla stazione base, gli altri utenti non percepiranno l'interferenza.

Poichè l'algoritmo ottimo di MUD è molto complesso ed è praticamente impossibile implementarlo per un numero ragionevole di utenti, sono stati sviluppati una serie di algoritmi subottimi.

I ricevitori subottimi possono essere divisi in due categorie: i *rivelatori lineari* e i *cancellatori di interferenza*.

I rivelatori lineari applicano una trasformata lineare all'uscita dei filtri adattati, cercando di eliminare l'interferenza da accesso multiplo (interferenza dovuta alla correlazione tra i codici di utenti diversi). Esempi di tali rivelatori lineari sono il decorrelatore e il rivelatore lineare a minimo errore quadratico medio (*LMMSE Linear Minimum Mean Square Error*).

Nel cancellatore invece l'interferenza da accesso multiplo è prima stimata e quindi sottratta dal segnale ricevuto. Esempi di tali algoritmi sono il cancellatore parallelo (*Parallel Interference Cancellation*) e quello seriale (*Serial Interference Cancellation*).

In conclusione, l'algoritmo di joint detection mira all'aumento della capacità di un sistema DS-CDMA attraverso la riduzione dell'interferenza intercella. Questo è valido per entrambe le soluzioni IMT-2000/UMTS, cioè sia per sistemi WCDMA che per sistemi TD-CDMA. In particolare però questo algoritmo risulta più critico nei sistemi TD-CDMA, in quanto a causa dei piccoli rapporti di spreading utilizzati, se è necessario allocare molti utenti per frame, risultano indispensabili tecniche di riduzione dell'interferenza intercella (in questo caso l'interferenza da accesso multiplo è sperimentata da utenti che condividono lo stesso slot).

### I.5.3 Controllo di Potenza (Power Control)

La necessità di un controllo di potenza è uno dei punti più critici nella tratta di up link di un sistema DS-CDMA. Questo problema nasce a causa dell'interferenza prodotta dagli utenti che accedono contemporaneamente al mezzo usando la stessa banda e lo stesso intervallo di tempo. A causa dei meccanismi di propagazione, a parità di potenza di

---

trasmissione, il segnale ricevuto alla stazione base da un terminale di utente vicino sarà più forte di quello ricevuto da un altro terminale situato al limite della cella. Conseguentemente gli utenti distanti saranno dominati in potenza da quelli più vicini. Questo effetto è chiamato *near-far*. Per raggiungere capacità considerevoli bisogna fare in modo che tutti i segnali dei vari utenti arrivino alla stazione base, indipendentemente dalla distanza, con la stessa potenza media. Gli algoritmi di controllo di potenza si occupano proprio di raggiungere questo risultato e perciò le loro performance influenzano pesantemente la capacità del sistema.

Nella tratta di down link il problema non si pone negli stessi termini, in quanto i segnali verso una stazione mobile si propagano tutti lungo lo stesso canale e quindi sono ricevuti con uguale potenza. Non è presente quindi un effetto *near-far*, ma un controllo di potenza viene comunque implementato per minimizzare l'interferenza fra celle diverse (interferenza intercella).

Oltre che per combattere l'interferenza tra utenti, i meccanismi di controllo di potenza migliorano anche le prestazioni del sistema nei confronti del fading del canale, compensandone i picchi. Se fosse possibile stimare perfettamente il fading introdotto dal canale, l'algoritmo di controllo della potenza trasformerebbe un canale con fading in un canale AWGN eliminandone completamente i picchi.

Esistono sostanzialmente due principi di power control: ad *anello aperto* e ad *anello chiuso*. La tecnica ad anello aperto misura le condizioni di interferenza sul canale e corregge la potenza di trasmissione in accordo a tali misure. Comunque poiché per il fading veloce non c'è correlazione tra le tratte di up link e di down link, le tecniche di controllo di potenza ad anello aperto regoleranno la potenza di trasmissione attorno ad un valore corretto solo in media. Sono quindi necessarie tecniche di controllo ad anello chiuso. Tali algoritmi consistono nel misurare in ricezione il rapporto segnale interferenza (SIR) e mandare comandi di controllo al trasmettitore per la modifica della potenza di trasmissione. L'obiettivo è raggiungere un prefissato SIR.

---

### I.5.4 Il Soft Handover

E' una tecnica che prevede che la stazione mobile possa essere connessa contemporaneamente a più stazioni base. Il soft handover è utilizzato nei sistemi CDMA per ridurre l'interferenza sulle altre celle e migliorare le performance sfruttando il concetto di macrodiversità.

Nei sistemi cellulari FDMA e TDMA tipicamente le celle vicine non usano le frequenze usate da una data cella (le celle che usano le stesse risorse frequenziali vengono spazialmente distanziate attraverso una opportuna pianificazione delle frequenze). Questo concetto è chiamato *riuso di frequenza*. In virtù del guadagno di processo tale separazione spaziale non è necessaria nei sistemi CDMA e può essere utilizzato un fattore di riuso di frequenza pari ad uno.

Solitamente una stazione mobile effettua l'handover quando la potenza del segnale di una cella vicina supera quella della cella attuale di una data soglia. Questo è il classico *hard handover*. Poichè nei sistemi CDMA le frequenze delle celle vicine sono le stesse della cella considerata, questo tipo di approccio causerebbe un'eccessiva interferenza intercella, che si ripercutirebbe in una degradazione delle capacità del sistema. Per evitare questa interferenza, un handover istantaneo dalla cella attuale alla nuova cella dovrebbe essere effettuato appena la potenza istantanea della nuova cella supera quella della cella attuale. Questo però non è in pratica realizzabile. Il meccanismo di handover dovrebbe sempre permettere alla stazione mobile di connettersi alla cella che riceve con la potenza più alta. Poichè nel soft handover la stazione mobile è contemporaneamente connessa a due o più stazioni base la sua potenza può essere controllata dalla stazione ricevuta con la potenza più alta. Una stazione mobile entra in stato di soft handover quando la potenza del segnale delle celle vicine supera una certa soglia, ma resta al di sotto della potenza del segnale della stazione base corrente.

La struttura di un segnale di un sistema CDMA ben si adatta all'implementazione del soft handover. Sia perchè nella tratta di up link, due o più stazioni possono ricevere lo stesso segnale grazie al fattore di riuso di frequenza pari ad uno, sia perchè nella tratta in down link

---

la stazione mobile può combinare coerentemente i segnali provenienti da più stazioni base aggiustando opportunamente i rami del ricevitore RAKE (in pratica vari segnali provenienti da stazioni base diverse vengono visti come componenti provenienti da cammini multipli aggiuntivi). Questo produce un ulteriore beneficio detto *macrodiversità* (il guadagno di diversità prodotto dalla ricezione di uno o più segnali aggiuntivi). Un canale separato detto pilota è solitamente usato per le misure di potenza necessarie al soft handover.

In down-link comunque il soft handover crea un aumento dell'interferenza nel sistema in quanto la nuova stazione base ora trasmette un segnale addizionale per la stazione mobile. E' possibile che la stazione mobile non sia in grado di sfruttare tutta l'energia trasmessa dalla stazione base a causa del limitato numero di rami del RAKE.

Quindi il guadagno complessivo dovuto al soft handover dipende dalla somma del guadagno ottenuto con la macrodiversità e delle perdite dovute all'incremento di interferenza.

In figura 7 è illustrato il principio di funzionamento del soft handover tra due stazioni base.

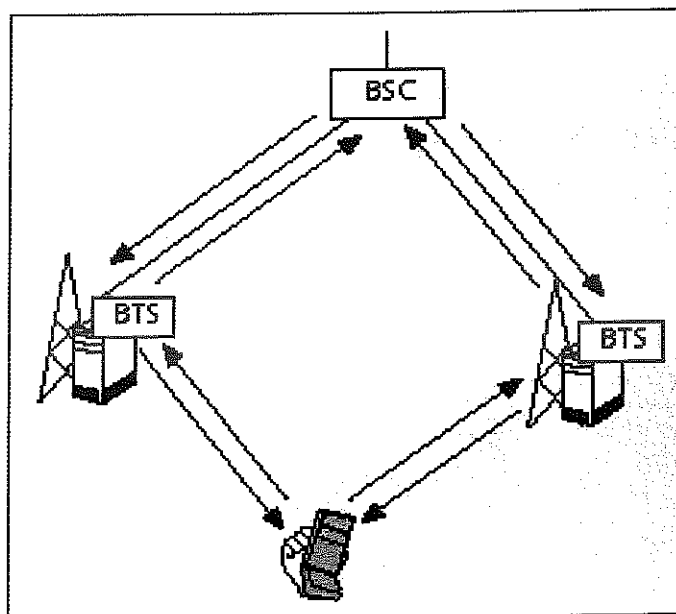


Figura 7. Principio del soft handover applicato su due stazioni base

In up link il segnale della stazione mobile è ricevuto contemporaneamente da due stazioni base, che dopo averlo demodulato e elaborato nel proprio ricevitore RAKE, passano il segnale direttamente ad un punto di combinazione, tipicamente una stazione base di controllo (BSC). In down link la stessa informazione è trasmessa da entrambe le stazioni base e la stazione mobile riceve i segnali come se provenissero da una stessa stazione attraverso cammini multipli differenti e li combina quindi nel ricevitore RAKE.

### I.5.5 Interfrequency Handover (Interlayer Handover)

In ambiente IMT-2000/UMTS, al fine di garantire la massima flessibilità del sistema, sono previsti 3 tipi di celle: la *picocella* (per ambienti indoor), la *microcella* (per ambienti urbani) e la *macrocella* (per ambienti suburbani o rurali). Esiste anche una *megacella* per la copertura via satellite. Tutte queste celle operano disgiuntamente, ma si sovrappongono nella copertura dello stesso territorio in una struttura gerarchica detta **HCS** (*Hyerarchical Cell Structure*) (Fig. 8).

Per evitare interferenze tra celle sovrapposte è prevista una separazione in frequenza, nel senso che ogni tipo di cella lavora su una banda assegnatale senza interferire con le altre.

La giustificazione di questo tipo di struttura prende origine dal fatto che lo scopo della rete IMT-2000/UMTS è quello di garantire una copertura universale di tutti i tipi di territori.

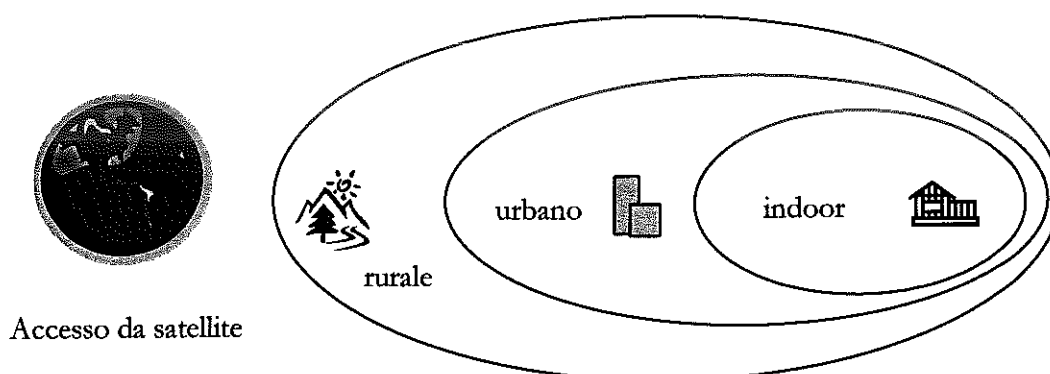


Figura 8. Struttura HCS con copertura pico-, micro-, macro- cellulare e accesso diretto da satellite

In una particolare zona coperta dalla rete cellulare la densità della popolazione di MS può essere molto alta. Per gestire gli elevati carichi di traffico generati in queste aree è necessario l'uso di piccole celle. D'altro canto la riduzione della dimensione delle celle introduce un aumento delle informazioni di segnalazione ed un aumento della probabilità di handover. Il sistema HCS prevede un compromesso tra l'aumento di capacità (attenuto a favore delle celle più piccole) ed una diminuzione della segnalazione e della probabilità di handover (ottenuto tramite l'utilizzo di celle più grandi).

L'implementazione del sistema HCS ha due principali impatti sulla gestione delle risorse radio:

- la divisione in modo appropriato delle risorse radio disponibili tra gli strati. E' previsto che la separazione venga effettuata nel dominio della frequenza in modo da evitare l'interferenza fra strati (interlayer interference),
- l'introduzione di una caratteristica specifica del sistema HCS che è l'handover tra strati (interlayer handover).

Lo studio delle procedure di handover tra strati sovrapposti coinvolge la modellizzazione degli spostamenti degli utenti, le strategie e le soglie di selezione dello strato.

La procedura di interlayer handover deve essere in grado di decidere attraverso alcuni parametri di input (come ad esempio la velocità del mobile), in quale strato un particolare utente dovrà essere servito. Quindi questa procedura deve avere una soglia predefinita coinvolta nelle decisioni. E' chiaro ad esempio che utenti fissi possono essere collocati in pico-celle, in quanto non producono mai fenomeni di handover, mentre la cella scelta da un utente mobile deve essere tanto più grande quanto maggiore è la sua velocità, per diminuire la probabilità di handover.

Per quanto riguarda il WCDMA, essendo la trasmissione di tipo continuo, non ci sono slot temporali liberi per effettuare la misura della potenza e della qualità del segnale in più strati contemporaneamente, mantenendo la connessione sulla frequenza portante attuale. Per risolvere tale problema sono stati proposti meccanismi di compressione o di sdoppiamento dei ricevitori.

---

Nella modalità a compressione, gli slot per le misure sono creati trasmettendo i dati di un frame, per esempio, con un più basso rapporto di spreading in un periodo di tempo più breve, in modo da avere a disposizione il resto del tempo per effettuare le misure sulle altre portanti.

Il ricevitore duale è in grado invece di ricevere altre frequenze senza compromettere la ricezione della portante attualmente in uso.

### I.5.6 DCA (Dynamic Channel Allocation)

Come vedremo nel prossimo capitolo, nella rete di accesso radio terrestre UMTS (UTRAN), tutte le stazioni base utilizzano gli stessi codici da assegnare ai propri utenti.

Poichè i codici OVSF assegnati da ogni stazione base ai propri utenti sono sempre gli stessi, è chiaro che stazioni base in visibilità ottica tra loro possono interferire a vicenda.

Supponiamo di avere due stazioni base adiacenti in cui sono presenti due utenti che trasmettono allo stesso tempo e con lo stesso codice. Si sperimenta interferenza in entrambe le direzioni di collegamento. Questa interferenza può essere di due tipi a seconda che i due mobili siano nello stesso stato oppure uno in trasmissione e l'altro in ricezione. In up link il segnale utile giunto alla stazione base dall'utente appartenente alla cella è sovrapposto al segnale interferente proveniente dall'altra stazione base o dall'altro mobile a seconda che quest'ultimo sia rispettivamente in ricezione o in trasmissione. Analogamente in down link il mobile può ricevere interferenza dalla stazione base adiacente o dall'altro mobile a seconda che quest'ultimo sia in ricezione o in trasmissione.

L'interferenza intercella è quindi dovuta al fattore di riutilizzo che nel sistema UMTS è posto pari ad 1, a differenza dei sistemi radiomobili di seconda generazione. Un fattore di riutilizzo uguale ad 1, equivale a dire che ogni stazione base utilizza tutte le risorse a disposizione (nel nostro caso codici utilizzati in ogni time slot) con conseguente aumento della capacità del sistema.

In realtà l'interferenza intercella è limitata dal processo di scrambling, che verrà analizzato nel capitolo successivo. Celle vicine hanno sicuramente codici di scrambling diversi (i

---

codici di scrambling sono dell'ordine del centinaio), che rendono quasi ortogonali i segnali provenienti da celle diverse. I codici di scrambling essendo in numero molto elevato non possono garantire una perfetta ortogonalità, per cui il fenomeno dell'interferenza intercella rimane presente.

Esistono particolari algoritmi di allocazione delle risorse, come ad esempio l'*Autonomous Reuse Partitioning* e il *Channel Segregation*, che tendono a minimizzare questo tipo di interferenza, facendo in modo che celle particolarmente interferenti (ad esempio con stazioni base in visibilità ottica tra di loro), utilizzino all'interno dello stesso slot codici differenti. La caratteristica di queste tecniche è che si basano tutte sul concetto di *decentralizzazione*, secondo il quale ciascuna stazione base prende le proprie decisioni autonomamente, senza pianificazioni a priori o comunicazioni tra BS (è una specie di riuso dei codici, come il riuso di frequenza nel sistema GSM, fatto però in maniera dinamica, senza pianificazione a priori e con efficienza notevolmente maggiore).

Il concetto di decentralizzazione è basato sulla necessità che ciascuna BS prenda le proprie decisioni autonomamente, senza pianificazioni e comunicazioni tra varie BS, in modo da ridurre al minimo la segnalazione tra le stesse.

Viene suggerito che per separare in modo trasparente le funzioni dello strato 2, gli algoritmi di CA e lo Scheduler devono essere considerati separatamente. L'algoritmo di allocazione non deve considerare problemi di scheduling. Esso conosce quante risorse sono disponibili, qual è la qualità presente su ciascuna risorsa e dispone il traffico su quella risorsa secondo l'ordine di arrivo dallo scheduling. Lo Scheduler dovrà dare priorità a certi tipi di traffico e quindi manderà all'algoritmo di allocazione solo quelle unità di dati che possono essere trasmesse.

#### 1.5.6.1 Channel Segregation

A titolo di esempio analizziamo il principio di funzionamento dell'algoritmo di channel segregation.

---

Il principale vantaggio di questo algoritmo è di evitare qualsiasi pianificazione delle risorse e di adattare l'allocazione delle risorse alla qualità del canale e al traffico. Esso presenta interessanti capacità di apprendimento autoadattive e si presta bene a schemi a commutazione di pacchetto.

L'algoritmo si basa sulla misura della qualità del canale che è valutata considerando il livello di interferenza sul canale stesso. Ciascuna BS crea una lista di priorità dei suoi time slot in base a tali misure. I passi di implementazione dell'algoritmo sono i seguenti:

- per ogni frame, ciascuna BS definisce i valori di priorità dei suoi time slot basandosi sui livelli di interferenza sperimentati dai differenti utenti negli slot. Una funzione di priorità esprime la relazione tra il valore di priorità assegnato allo slot ed il suo livello di interferenza.
- La lista di priorità dei time slot è aggiornata in ciascuna BS sulla base di una formula con finestra geometrica che tiene in conto sia i nuovi che i vecchi valori di priorità. Ciascun valore di priorità è pesato in base al suo tempo di vita, in modo che la sua influenza decresca con l'aumentare di tale tempo.
- Lo schema di allocazione delle risorse consiste nell'utilizzare per primi gli slot con più alta priorità.

L'efficienza dell'algoritmo è stata studiata confrontandone gli effetti con quelli di un sistema senza DCA. In un sistema senza DCA tutto il traffico in arrivo viene uniformemente ripartito su tutti gli slot a disposizione. Questo non permette di tenere in conto l'interferenza tra BS che può essere più o meno alta. Se ad esempio due BS si trovano in visibilità ottica, trasmettendo con gli stessi codici negli stessi intervalli di tempo (negli stessi slot) esse interferiranno a vicenda. Attraverso le misure di interferenza, l'algoritmo effettua una ripartizione dinamica dei codici. Alla fine si verifica che le celle in visibilità ottica o in condizioni di particolare interferenza reciproca vanno ad utilizzare per ogni slot codici diversi in modo da minimizzare l'interferenza intercella.

---

---

# CAPITOLO

# 2

---

## LE SPECIFICHE UMTS

*In questo secondo capitolo verranno analizzate quelle specifiche tecniche dell'UMTS che, dallo studio dei documenti del 3GPP, appaiono più stabili e possono essere considerate dunque definitive con ragionevole certezza.*

*Si partirà da una descrizione generica dell'architettura dell'interfaccia radio, per passare poi alla presentazione dettagliata degli strati protocollari che in essa operano, in particolare dello strato fisico, che ad oggi risulta la sezione meglio definita da parte del 3GPP.*

### **II.1 Introduzione a UTRA Time Division Duplex**

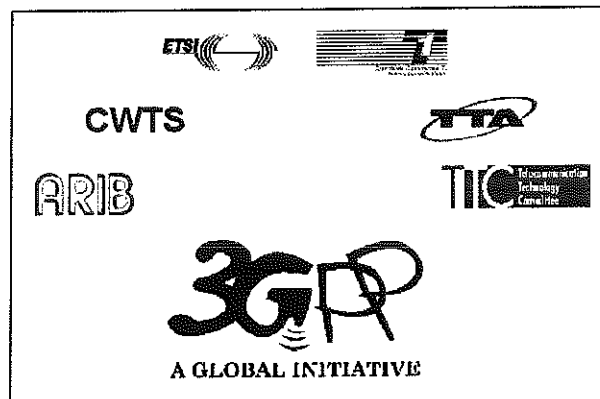
L'UTRA-TDD prevede una modalità di accesso multiplo basata sull'unione di tecniche di divisione di tempo e di codice detta TD-CDMA.

Si tratta di una tecnica di accesso multiplo TDMA con spreading. Gli utenti sono separati ortogonalmente all'interno dei vari time slot, e in ciascun time slot, un'ulteriore separazione è effettuata attraverso i codici di spreading.

Tutte le specifiche tecniche riportate nel seguito per definire la struttura del UTRA-TDD mode sono state ricavate dal lavoro del **3GPP** (*3rd Generation Partnership Project*). Questo gruppo di ricerca e standardizzazione, partendo dalle definizioni di ETSI e ITU, ha elaborato le principali soluzioni tecniche per i punti cruciali del sistema radiomobile di terza generazione.

Per la nostra indagine risulta di particolare interesse il lavoro del Gruppo RAN (Radio Access Network), da cui sono state elaborate le informazioni in merito ai canali fisici, alla mappatura in questi ultimi dei canali di trasporto, alla struttura del frame, ai processi di modulazione e di spreading e alle specifiche delle funzioni degli strati MAC e RLC.

Nel lavoro del 3GPP convergono non solo i contributi dei principali enti di standardizzazione mondiale (ETSI, ITU, ARIB, ecc.), ma anche quelli delle principali industrie manifatturiere presenti nell'attuale mercato delle telecomunicazioni. E' proprio l'apporto fornito da queste ultime che ha permesso di raggiungere una discreta definizione dei punti cardine del sistema che, sebbene ancora in fase di elaborazione, hanno ridotto l'incredibile vastità di problemi di definizione lasciati aperti dai report degli organismi di standardizzazione.



## II.2 Architettura logica dell'UTRAN

Un sistema UMTS è costituito, secondo lo standard proposto, da 3 parti distinte. Innanzitutto il CN (*Core Network*) che si occupa delle funzioni riguardanti l'instaurazione ed il controllo della comunicazione e la gestione dei dati riguardanti l'utente in generale.

Segue poi la rete di accesso radio, ovvero l'UTRAN (*UMTS Terrestrial Radio Access Network*) che si occupa invece dello scambio di informazioni tra l'utente mobile e la rete

fissa. Infine c'è la **MS** (*Mobil Station*) o **UE** (*User Equipment*) che è la stazione mobile rappresentativa del singolo utente.

L'UTRAN è formata da un insieme di *Radio Network Subsystem* (**RNS**) connessi alla Core Network attraverso l'interfaccia **Iu**.

Un RNS è formato a sua volta da un *Radio Network Controller* (**RNC**) e da una o più entità dette **Node B**. I Node B sono connessi al RNC attraverso l'interfaccia **Iub** che permette di trasmettere vari tipi di informazione. Innanzitutto quelle di applicazione radio rivolte ai segnali: si hanno così le risorse per la gestione dei Node B; poi quelle di partizione della trama dati ed infine quelle per il controllo della qualità. In definitiva l'interfaccia Iub permette di gestire le macro-diversità tra blocchi di dati, di prendere le decisioni per gestire il softhandover, di canalizzare i codici e gestire le risorse hardware dei Node B.

All'interno del UTRAN gli RNC di diversi RNS possono essere interconnessi tra loro attraverso l'interfaccia **Iur**. L'architettura logica dell'UTRAN è mostrata in Figura 1.

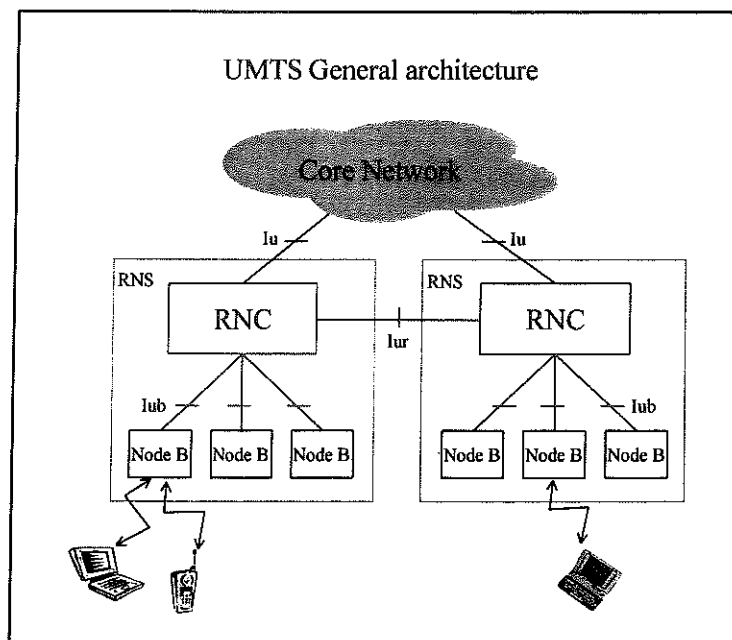


Figura 1. Architettura logica dell'UTRAN

La figura 2 mostra un possibile cammino evolutivo dal GSM all'UMTS, basato sulla configurazione del GSM di fase 2+. Qui, il sistema GSM è visto in una doppia veste: da un lato fornisce i servizi a circuito (voce e dati a basso bit rate) attraverso l'infrastruttura basata sugli apparati MSC e *Gateway MSC*; dall'altro fornisce i servizi dati a pacchetto per mezzo del sistema GPRS.

La configurazione illustrata in figura può rappresentare un primo passo verso l'UMTS, capace di risolvere principalmente problemi di capacità. Infatti qui l'accesso radio innovativo viene usato semplicemente per aumentare la capacità disponibile. La configurazione non permette di fornire servizi innovativi rispetto al GSM, in quanto le funzioni di rete fissa rimangono quelle del GSM e non sono aggiornate secondo i nuovi requisiti di servizio.

La soluzione di figura 3 risponde invece all'esigenza di fornire nel breve periodo servizi dati di tipo innovativo. L'infrastruttura innovativa dell'UMTS risulta integrata con il GSM. La Core Network UMTS realizza funzionalità a circuito e a pacchetto.

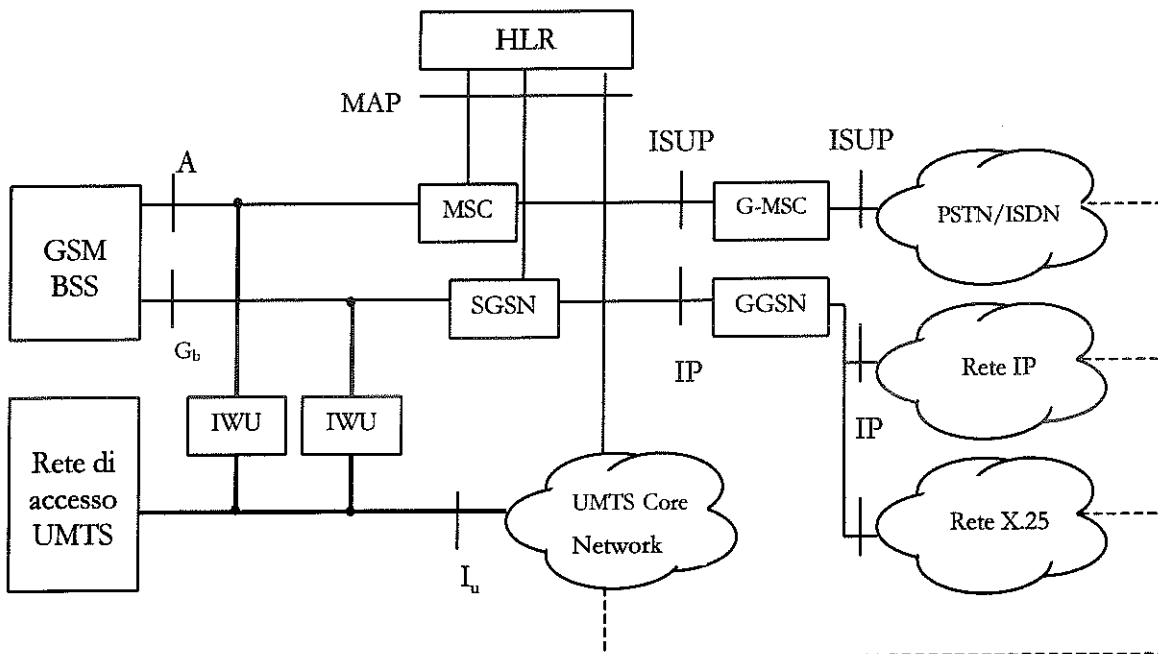


Figura 3. Migrazione GSM-UMTS per servizio

Le principali tendenze evolutive mirano a superare l'attuale divisione dei domini di rete, introducendo una soluzione completamente basata su IP, sia per il traffico voce che per quello dati.

Come già ripetutamente notato, la specifica UMTS è in continua evoluzione mediante la definizione e la successiva introduzione di Release annuali. Quanto descritto in questo paragrafo si riferisce all'infrastruttura di rete UMTS di Release 1999 (R99), completata ed approvata dal 3GPP nel 1999 e trasferita ai rispettivi enti di standardizzazione regionale.

In ambito 3GPP è sempre più concreta la possibilità di specificare una soluzione di rete UMTS per la Release 2000 (R2000) completamente basata sulla tecnologia IP. In questo caso, viene abbandonata la struttura a doppio backbone, per passare ad una infrastruttura di rete unica in cui anche il traffico voce viene gestito tramite commutazione di pacchetto.

Uno degli aspetti cruciali di una soluzione di rete UMTS basata completamente su IP è quella della qualità del servizio (QoS), intesa in termini di banda disponibile, ritardo complessivo end-to-end e jitter. Infatti il sistema dovrà garantire una qualità della voce comparabile con, o migliore di, quella fornita dalle attuali reti di telecomunicazioni di seconda generazione.

## II.4 Architettura protocollare dell'interfaccia radio

L'interfaccia radio può essere vista come una struttura a tre strati (fig. 2), ognuno dei quali comunica con quello adiacente tramite dei ben definiti protocolli:

- *strato fisico* (physic layer o L1)
- *strato di collegamento dati* (data link layer o L2)
- *strato di rete* (network layer o L3)

Lo strato 2 è a sua volta diviso in due sottostrati: controllo di accesso al collegamento (*Radio Link Control* o **RLC**) e controllo di accesso al mezzo (*Medium Access Control* o **MAC**)

Lo strato 3 e l'RLC sono divisi in piani di Controllo (*C-plane*) e di Utente (*U-plane*). Nel C-plane, lo strato 3 è ancora partizionato in due sottostrati, l'inferiore dei quali, detto

---

*Radio Resource Control (RRC)* interfaccia con lo strato 2. L'RRC assieme a tutti gli strati sottostanti formano il cosiddetto "strato di accesso". Lo strato superiore all'RRC nel C-plane infatti, che si occupa della gestione della mobilità (*Mobility Management, MM*) e del controllo di chiamata (*Call Control, CC*) è supposto non facente parte dello "strato di accesso" e non verrà pertanto preso in considerazione.

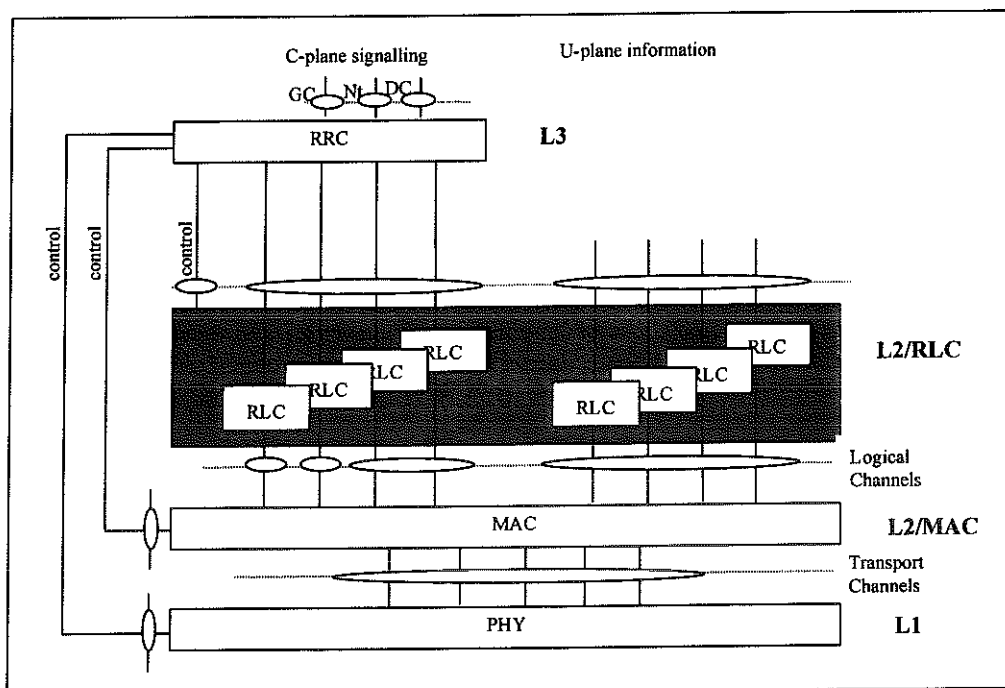


Figura 2. Architettura protocollare dell'interfaccia radio

La Figura 2 mostra appunto l'architettura protocollare dell'interfaccia radio. Ogni blocco rappresenta un protocollo, mentre i cerchi d'interfaccia rappresentano i *Service Access Point (SAP)* per la fornitura di servizi dai e ai vari strati. Il SAP tra il sottostrato MAC e lo strato fisico fornisce servizi detti *Canali di Trasporto (Transport Channel)*, mentre i SAP tra i sottostrati RLC e MAC forniscono servizi detti *Canali Logici (Logical Channel)*. Sia i Transport che i Logical Channel verranno descritti in seguito.

Nel C-plane l'interfaccia tra l'RRC ed il sottostrato superiore (MM, CC) è definita da tre SAP detti General Control (GC), Notification (N) e Dedicated Control (DC).

In Figura 2 vengono anche mostrate delle connessioni tra RRC e MAC e tra RRC e L1 ed

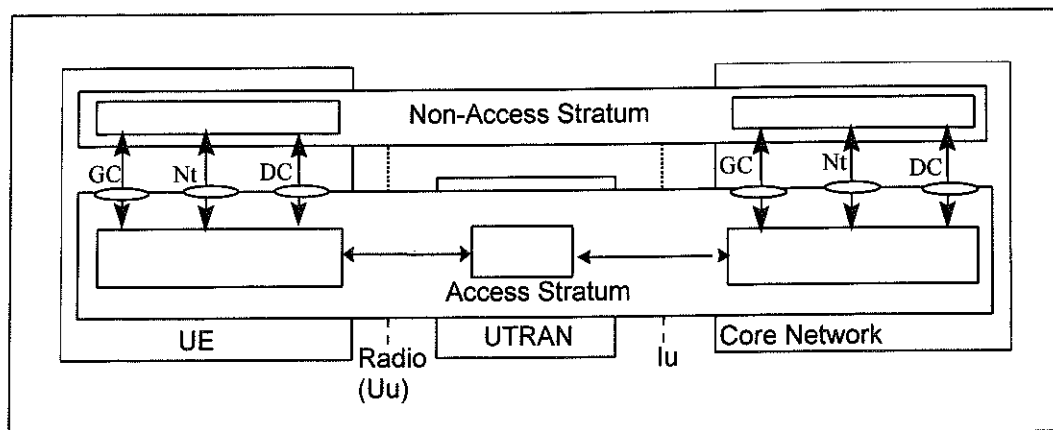


Figura 3. Architettura generale dell'UMTS

un'analogia interfaccia esiste tra l'RRC e l'RLC. Queste forniscono servizi locali di controllo interstrato, in pratica permettono all'RRC di controllare e gestire la configurazione degli strati sottostanti.

## II.5 Lo strato fisico

### II.5.1 I canali di trasporto

I canali di trasporto rappresentano i servizi offerti dallo strato 1 agli strati più alti. Un canale di trasporto è definito da come e con quali caratteristiche viene effettuato il trasferimento dati sull'interfaccia aerea.

Una prima classificazione generale dei canali di trasporto è quella che li divide in:

- *canali comuni*, che portano principalmente messaggi per la gestione della chiamata, della mobilità e delle risorse radio;
- *canali dedicati*, che portano l'informazione generata dal singolo utente.

Esiste un solo tipo di canale di dedicato (*Dedicated Channel*, **DCH**) ed è finalizzato al trasporto dell'informazione di utente o di controllo tra l'UTRAN e l'UE sia in up link che in down link.

Esistono invece una serie di canali di trasporto comuni utilizzati per portare informazioni di segnalazione necessarie per la gestione dell'accesso. I messaggi trasportati su questi canali possono essere di vario tipo: informazioni di tipo broadcast, richieste di accesso, concessione di accesso, messaggi di paging, pacchetti di utenti. A seconda della funzione e della struttura, i canali comuni si distinguono in:

*Broadcast Channel (BCH)* è un canale di trasporto in down link utilizzato dalla stazione base per comunicare agli utenti all'interno della cella informazioni specifiche del sistema e della cella, come la struttura del frame, l'allocazione dei canali fisici per l'accesso a contesa (time slots e codici che la stazione mobile può usare per inviare una richiesta di accesso), ecc.

*Paging Channel (PCH)* è un canale di trasporto in down link che è utilizzato per portare informazioni di controllo ad un utente mobile quando il sistema non conosce la cella in cui esso è situato. Attraverso questo canale si effettua una ricerca della stazione mobile verso la quale si vuole instradare una particolare chiamata.

*Forward Access Channel (FACH)* è un canale di trasporto in down link che è utilizzato per portare informazione di controllo ad utente mobile quando è conosciuta la sua attuale cella di residenza. Principalmente esso trasporta messaggi di assegnazione delle risorse radio, conferme di corretta ricezione o richieste di ritrasmissione. E' previsto dallo standard che il FACH possa anche portare piccoli pacchetti di informazioni di utente.

*Random Access Channel (RACH)* è un canale di trasporto in up link che trasporta le richieste per stabilire una connessione. Poichè la richiesta di accesso alla stessa stazione base è trasmessa da tutte le stazioni mobili appartenenti a quella cella su un unico canale fisico condiviso (anche se lo standard prevede dimensioni variabili per adattarsi a diverse condizioni) è necessario gestire il rischio di collisioni tra utenti, problema che viene risolto

---

adottando un idoneo protocollo di gestione della multiutenza. Anche in questo caso lo standard prevede che il RACH possa trasportare piccoli pacchetti di informazione di utente.

*Synchronisation Channel (SCH)* è il canale fondamentale in down link che permette alla stazione mobile di sincronizzarsi con la stazione base. Esso fornisce tutti i sincronismi necessari all'utente per poter comunicare con la base tra cui la locazione dei canali fisici contenenti il BCH (da cui si ricava il set di codici e slot caratterizzanti il canale fisico del RACH che l'utente utilizzerà per instaurare la connessione).

*Up link Shared Channel (USCH)* è un canale di trasporto in up link condiviso da più utenti tipicamente utilizzato per servizi dati a pacchetto NRT (No Real Time). Per tali servizi questo canale permette una efficiente allocazione per piccoli periodi di tempo.

*Down link Shared Channel (DSCH)* è l'equivalente del USCH nella direzione opposta.

Lo standard prevede anche altri canali di trasporto comuni per funzionalità aggiuntive dell'interfaccia radio che però non risultano rilevanti nel contesto di questo lavoro.

I canali fisici sono invece i mezzi di cui si servono i canali di trasporto per veicolare l'informazione.

Analizzando gli standard si può notare come i canali di trasporto sono pressoché equivalenti per le due soluzioni tecniche adottate. Questo è ovvio se si tiene conto del fatto che lo stesso standard UMTS/IMT-2000 prevede che le due soluzioni scelte per l'accesso multiplo (WCDMA e TD-CDMA) debbano poter coesistere all'interno della stessa rete. E' chiaro quindi che, per ragioni di compatibilità, i canali di trasporto devono essere gli stessi a prescindere dal fatto che, nei due casi, essi vengano mappati nei canali fisici con modalità completamente diverse.

Per quanto riguarda il sistema TDD da noi preso in esame, un canale fisico è un burst trasmesso in un particolare time slot allocato in un particolare frame radio. L'allocazione può essere continua, cioè lo stesso time slot viene allocato al canale fisico in ogni frame per

---

tutta la durata della connessione o discontinua, cioè il time slot è allocato solo in un subset di tutti i frame.

Un burst è la combinazione di una parte dati, un midambolo e un periodo di guardia. La sua durata è pari ad un time slot.

Più burst possono essere trasmessi allo stesso istante da un unico trasmettitore. In questo caso, la parte dati usa differenti codici di canalizzazione OVSF, ma lo stesso codice di scrambling. Per quanto riguarda le parti di midambolo (sequenza di training conosciuta dal ricevitore necessaria per la stima del canale e il controllo di potenza) di burst di utenti nello stesso slot, esse possono essere diverse anche se devono essere generate dallo stesso codice base.

La parte dati di un burst subisce il processo di spreading attraverso la combinazione della codifica di canalizzazione e di quella di scrambling. Il codice di canalizzazione è un codice OVSF, che può avere un fattore di spreading  $Q=SF=1, 2, 4, 8$  o  $16$ . Il data rate del canale fisico dipende dal fattore di spreading del particolare codice OVSF utilizzato.

La parte di midambolo del burst può essere di due tipi: midambolo corto di 256 chip o lungo di 512 chip. Quindi il data rate del canale fisico dipende anche dalla lunghezza del midambolo utilizzato.

Risulta quindi che un canale fisico, cioè un burst, è definito dall'allocazione di un codice di canalizzazione, di un time slot, di un frame radio e di una frequenza di lavoro. Il codice di scrambling e il codice di midambolo base (da cui con tecniche di traslazione si ricavano tutti i possibili midamboli da utilizzare in una cella) necessari all'utente per comunicare con la stazione base sono trasmessi in modalità broadcast (BCH) e lo standard prevede che possono essere costanti all'interno di ciascuna cella.

## II.5.2 La struttura del frame

Come mostrato in figura 4, il frame TDMA ha una durata di 10 ms ed è suddiviso in 15 time slot (TS). Ciascun TS contiene un numero fissato di chip pari a 2560.

---

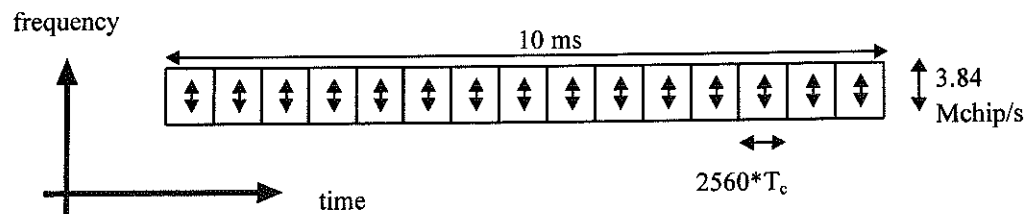


Figura 4. Struttura del frame TDD

Questo equivale a fissare un chip rate di 3.84 Mchip/s. Infatti detto  $T_c$  il periodo di chip ed  $R_c$  il chip rate si deve avere:

$$2560 * T_c * 15 = 10 \text{ ms};$$

da cui

$$R_c = \frac{1}{T_c} = 3.84 \text{ Mchip/s}$$

Lo standard non prevede a priori una fissata ripartizione dei TS tra up link e down link in modo da permettere con questa flessibilità di poter adattare il sistema TDD a differenti scenari di impiego. L'unica limitazione imposta è che per ogni configurazione possibile ogni frame deve avere almeno uno slot allocato per il down link ed uno per l'up link.

Esistono 4 possibili soluzioni al problema della ripartizione degli slot. La soluzione più flessibile è quella che lascia piena libertà ed è rappresentata da una configurazione con più punti di switch e allocazione asimmetrica (fig. 5.a).

Può essere presa in considerazione anche una soluzione con più punti di switch, ma simmetrica (fig. 5.b). Restano infine le soluzioni con allocazione simmetrica o asimmetrica di up e down link con un singolo punto di switch (figg. 5.c 5.d). In questo caso la scelta fatta è di far precedere sempre la parte di down link a quella di up link.

Gli schemi rappresentativi delle 4 soluzioni sono riportati di seguito.

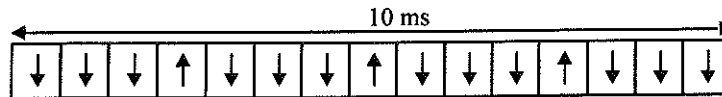


Figura 5.a. Configurazione con punti di switching multipli e ripartizione asimmetrica di down link e up link

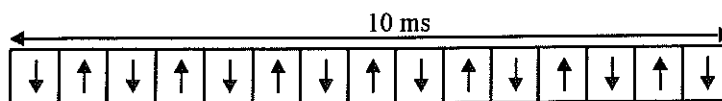


Figura 5.b. Configurazione con punti di switching multipli e ripartizione simmetrica di down link e up link

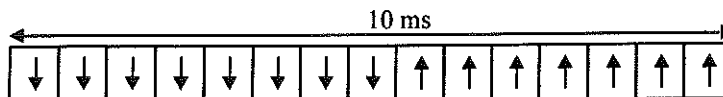


Figura 5.c. Configurazione con punto di switching singolo e ripartizione simmetrica di down link e up link

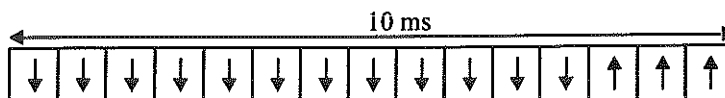


Figura 5.d. Configurazione con punto di switching singolo e ripartizione asimmetrica di down link e up link

Lo standard lascia comunque piena libertà nella ripartizione degli slot tra le due tratte e permette in particolare di effettuare una assegnazione dinamica frame per frame degli slot al down link e all'up link.

Per quanto riguarda le soluzioni asimmetriche, esse trovano la loro giustificazione in un contesto in cui sia presente una rilevante quantità di traffico di tipo sbilanciato come il traffico dati di tipo WEB in cui la tratta in discesa viene solitamente caricata molto più pesantemente di quella in salita.

---

### II.5.3 Spreading e modulazione

Prima di passare ad una descrizione dettagliata della struttura dei canali fisici su cui vengono veicolati i singoli canali di trasporto è necessario chiarire le operazioni di spreading effettuate sulla parte dati di un burst e le proprietà dei particolari codici utilizzati.

Abbiamo già visto che, sfruttando la loro proprietà di ortogonalità, è possibile assegnare più codici CDMA ad un singolo utente o a differenti utenti che trasmettono simultaneamente i loro burst nello stesso time slot e alla stessa frequenza. Vediamo come ciò viene reso possibile.

#### II.5.3.1 Modulazione

Il massimo numero possibile di codici CDMA utilizzati in uno slot, che deve essere minore o uguale a 16, dipende dai singoli fattori di spreading usati dagli utenti.

Come vedremo, ciascun burst di utente ha due parti che portano dati informativi, chiamati blocchi dati:

$$\underline{d}^{(k,i)} = (\underline{d}_1^{(k,i)}, \underline{d}_2^{(k,i)}, \dots, \underline{d}_{N_k}^{(k,i)})^T \quad i = 1, 2; k = 1, \dots, K.$$

dove  $K$  è il massimo numero di codici assegnati nello slot (e può variare con continuità da 1 a 16),  $N_k$  è il numero di simboli per campo dati dell'utente  $k$ -esimo. L'indice  $i$  indica il primo o il secondo blocco dati del burst (il blocco  $i=1$  viene trasmesso prima del midambolo, mentre quello con  $i=2$  dopo), l'indice  $k$  è rappresentativo del  $k$ -esimo utente nello slot a cui è associato un relativo codice. I simboli  $\underline{d}_n^{(k,i)}$  sono stati sottolineati per evidenziare che si tratta di quantità complesse.

Ciascuno degli  $N_k$  simboli di dati di ciascuno dei due campi del burst ha una durata di simbolo pari a  $T_s = Q_k * T_c$ , che, a parità di chip rate  $T_c$  (fissato a 3.84 Mchip/s), varia al variare del fattore di spreading  $Q_k$ , proprio del  $k$ -esimo utente e del codice a lui assegnato.

La modulazione dati è di tipo QPSK, cioè i simboli di dati  $\underline{d}_n^{(k,i)}$  sono generati a partire da due sequenze binarie di bit informativi opportunamente codificate (per il controllo dell'errore) e interallacciate

$$b_{l,n}^{(k,i)} \in \{0,1\} \quad l = 1,2; k = 1, \dots, K; n = 1, \dots, N_k; i = 1,2$$

attraverso le equazioni

$$\begin{aligned} \operatorname{Re}\{\underline{d}_n^{(k,i)}\} &= \frac{1}{\sqrt{2}}(2b_{1,n}^{(k,i)} - 1) \\ \operatorname{Im}\{\underline{d}_n^{(k,i)}\} &= \frac{1}{\sqrt{2}}(2b_{2,n}^{(k,i)} - 1) \quad k = 1, \dots, K; n = 1, \dots, N_k; i = 1, 2. \end{aligned}$$

che rappresentano le tipiche equazioni di una modulazione QPSK effettuata a partire dalle due sequenze binarie prima viste.

La modulazione QPSK permette, come al solito, di raddoppiare il bit rate rispetto al data rate.

### II.5.3.2 Spreading

Il processo di spreading dei simboli informativi presenti nella parte dati di un burst consta di due operazioni distinte: la *canalizzazione* e lo *scrambling*.

Innanzitutto ciascun simbolo della sequenza dati  $\underline{d}_n^{(k,i)}$  subisce un'operazione di spreading da parte di un codice di canalizzazione complesso  $\underline{c}^{(k)}$  di lunghezza  $Q_k \in \{1, 2, 4, 8, 16\}$ .

In seguito la sequenza risultante subisce il processo di scrambling da parte di un codice di lunghezza 16.

Analizziamo i codici di canalizzazione. Ciascun elemento di un codice di canalizzazione complesso  $\underline{c}^{(k)} = (c_1^{(k)}, c_2^{(k)}, \dots, c_{Q_k}^{(k)})$   $k=1, \dots, K$ ; deve essere scelto all'interno del seguente set complesso:

$$\underline{V}_c = \{1, j, -1, -j\}$$

in cui con  $j$  si denota l'unità immaginaria.

Il codice di canalizzazione complesso  $\underline{c}^{(k)}$  è generato dalla relativa sequenza binaria di codici  $\mathbf{a}_{Q_k}^{(k)} = (a_1^{(k)}, a_2^{(k)}, \dots, a_{Q_k}^{(k)})$  di lunghezza  $Q_k$  mostrata in figura 6 e allocata all'utente  $k$ -esimo.

La relazione tra gli elementi delle due sequenze è data da:

$$c_q^{(k)} = (j)^q \cdot a_q^{(k)} \quad a_q^{(k)} \in \{1, -1\}; q=1, \dots, Q_k.$$

Ne consegue che gli elementi dei codici di canalizzazione sono alternativamente reali e immaginari.

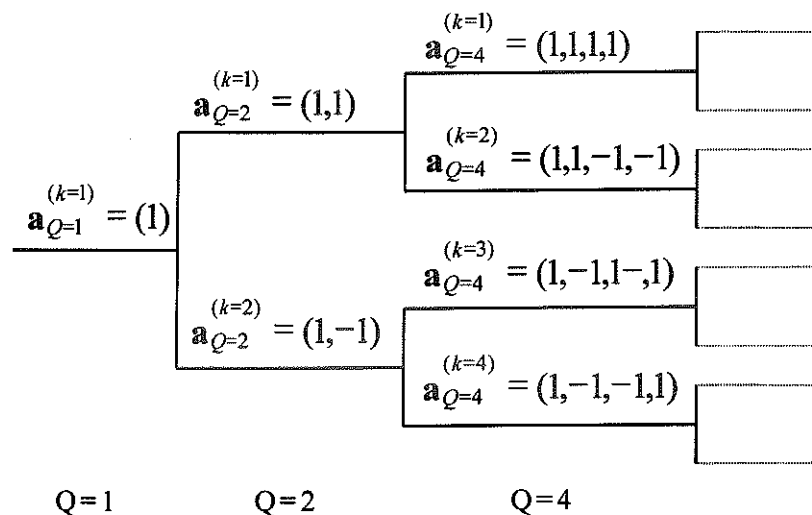


Figura 6. Albero per la generazione dei codici OVSF (Orthogonal Variable Spreading Factor) utilizzati per l'operazione di canalizzazione

Le sequenze sono rappresentate da codici **OVSF** (*Orthogonal Variable Spreading Factor*). Tali codici permettono di mescolare all'interno dello stesso time slot canali con differenti fattori di spreading preservandone l'ortogonalità.

I codici OVSF possono essere definiti a partire dall'albero di codice rappresentato in figura 6.

Ciascun livello nell'albero di codice definisce un fattore di spreading indicato in figura dal valore  $Q$ . Non tutti i codici dell'albero possono però essere usati contemporaneamente in uno stesso TS. Un codice può essere utilizzato in un dato TS se e solo se non sono attivi altri codici che, partendo dalla radice, attraversano lo stesso nodo dell'albero del codice considerato.

Questo significa che il numero di codici disponibili in un TS non è fisso, ma dipende dal rate e dal fattore di spreading di ciascun canale fisico.

Nel sistema TDD il massimo fattore di spreading è fissato a  $Q_{MAX}=16$ . Per cui si ottiene una limitazione sul numero massimo di codici (e quindi di utenti) in uno stesso slot pari a 16.

Vale la pena notare esplicitamente che il numero di codici in uno slot può variare con continuità da 1 a 16 e non è necessariamente una potenza del 2. Possiamo per esempio allocare 3 utenti con differenti bit rate nello stesso slot assegnando al primo uno dei due codici con fattore di spreading  $Q=2$  e agli altri due i due codici con fattore di spreading  $Q=4$  rimanenti nell'altro ramo dell'albero. E' chiaro che i due utenti con fattore di spreading  $Q=4$  trasmetteranno ad un bit rate pari alla metà di quello dell'utente con fattore  $Q=2$ .

Passiamo ora all'operazione di *scrambling*.

Per quanto detto prima è chiaro che i codici OVSF sono comuni a tutte le celle e quindi celle adiacenti possono usare, all'interno dello stesso slot, gli stessi codici. Per limitare quindi l'interferenza intercella è necessario procedere al secondo passo del processo di spreading.

---

Lo spreading dei dati da parte del codice di canalizzazione complesso di lunghezza  $Q_k$  è seguito da un'operazione di scrambling effettuata con una particolare sequenza  $v=(v_1, v_2, \dots, v_{Q_{MAX}})$  specifica della cella di lunghezza  $Q_{MAX}=16$ .

Dopo aver moltiplicato i simboli  $d_n^{(k,i)}$  per il codice di canalizzazione complesso  $c^{(k)}$  di lunghezza  $Q_k$ , ad ogni simbolo corrisponderanno  $Q_k$  chip di codifica. Per cui, raggruppando i simboli  $d_n^{(k,i)}$  in gruppi di  $Q_{MAX}/Q_k$ , si ottengono, dopo il processo di canalizzazione, gruppi di chip di lunghezza  $Q_{MAX}$  pari alla lunghezza della sequenza di scrambling. E' possibile a questo punto moltiplicare ad uno ad uno i chip di ciascun gruppo con i chip della sequenza di scrambling, come rappresentato nel seguente schema.

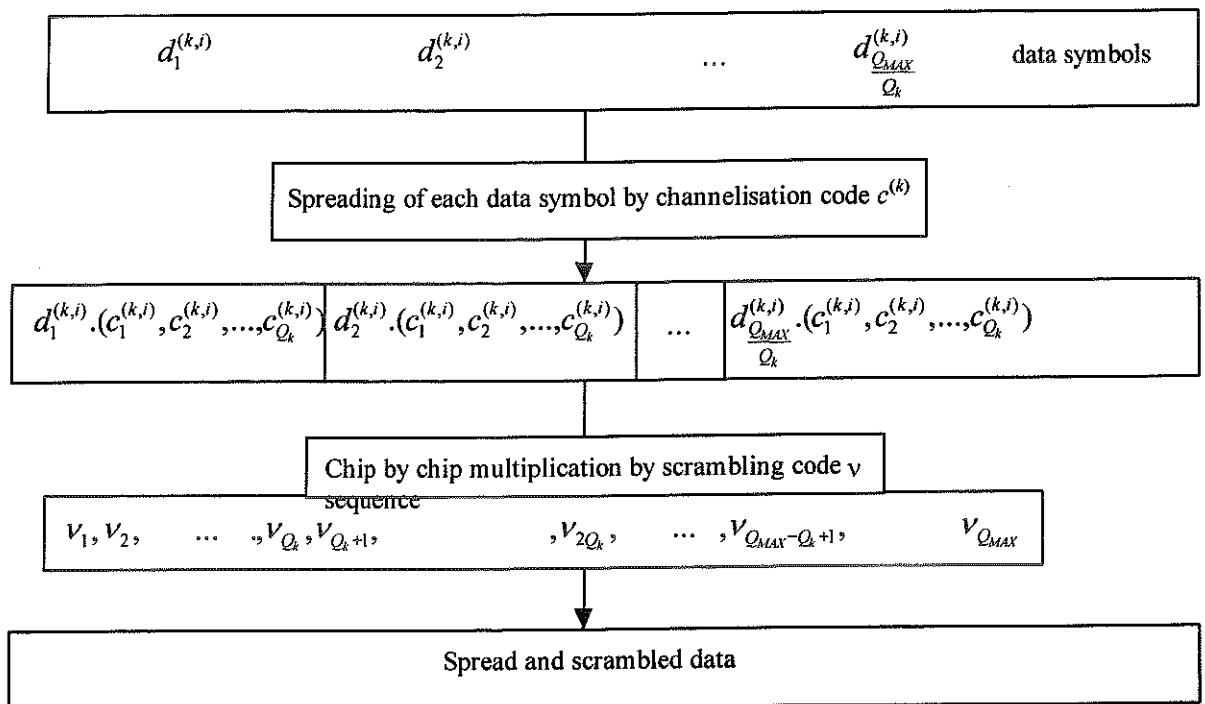


Figura 7. Schema dello spreading dei simboli dati

Concludiamo il discorso sullo spreading nei sistemi TDD osservando che la combinazione di un codice di canalizzazione specifico dell'utente e di uno di scrambling specifico della cella può essere vista come il risultato di un unico processo di codifica

effettuato con un codice caratteristico sia della cella che dell'utente. Tale codice è rappresentato da

$$\mathbf{s}^{(k)} = (s_p^{(k)}) \quad \text{con} \quad s_p^{(k)} = c_{1+[(p-1) \bmod Q_k]}^{(k)} \cdot v_{1+[(p-1) \bmod Q_{\max}]} \quad k=1, \dots, K, p=1, \dots, N_k Q_k.$$

Utilizzando un filtro di modulazione per il chip con risposta impulsiva a radice di coseno rialzato il segnale trasmesso nel primo blocco di dati antistante il midambolo è

$$\underline{d}^{(k,1)}(t) = \sum_{n=1}^{N_k} \underline{d}_n^{(k,1)} \sum_{q=1}^{Q_k} s_{(n-1)Q_k+q}^{(k)} \cdot C r_0(t - (q-1)T_c - (n-1)Q_k T_c)$$

mentre il secondo blocco dopo il midambolo è

$$\underline{d}^{(k,2)}(t) = \sum_{n=1}^{N_k} \underline{d}_n^{(k,2)} \sum_{q=1}^{Q_k} s_{(n-1)Q_k+q}^{(k)} \cdot C r_0(t - (q-1)T_c - (n-1)Q_k T_c - N_k Q_k T_c - L_m T_c).$$

dove  $L_m$  è il numero di chip del midambolo ( $L_m = 256$  o  $512$ ).

## II.5.4 I canali fisici

Abbiamo appena visto come la codifica del segnale attraverso il processo di spreading tipico dei sistemi CDMA permetta l'accesso simultaneo di utenti sovrapposti in frequenza e nel tempo.

Chiarita la dinamica e le principali caratteristiche dello spreading è possibile adesso prendere in rassegna la mappatura dei canali di trasporto presentati precedentemente nei canali fisici e analizzare la struttura di questi ultimi.

### II.5.4.1 Canale fisico dedicato (DPCH)

Un canale fisico dedicato è quello assegnato ad un utente in trasmissione o in ricezione per il suo trasferimento dati.

Lo standard prevede per tale canale un fattore di spreading fissato pari a 16 per il down link ed un fattore di spreading variabile da 16 ad 1 per l'up link.

Per supportare trasferimenti ad elevati bit rate in down link è prevista la possibilità di utilizzare da parte di un unico utente canali fisici multipli in parallelo sfruttando differenti codici di canalizzazione (trasmissione multicode). Si presume quindi la possibilità da parte di un stazione mobile di poter ricevere contemporaneamente più segnali codificati con codici diversi. E' inoltre prevista per un singolo slot in down link la possibilità di operare con un singolo codice, cioè con un fattore di spreading pari ad 1.

Per quanto riguarda l'up link invece, essendo lasciata libera la scelta dello SF, è possibile variare il bit rate fornito all'utente modificando il suo fattore di spreading. E' comunque previsto che un utente possa effettuare trasmissioni multicode usando però un massimo di due canali fisici per TS simultaneamente, il che equivale a dire che ad un utente possono essere assegnanti in trasmissione al massimo due codici di canalizzazione per slot. Anche in questo caso bisogna presumere che l'utente sia in grado di trasmettere contemporaneamente due segnali (ma non più di due) con codici OVSF differenti.

Per i canali fisici dedicati sono definiti due tipi di burst che differiscono per le dimensioni del midambolo. Ogni burst consiste di due campi di simboli dati, un midambolo ed un periodo di guardia (fig. 8). Indichiamo come burst di tipo 1 quello con il midambolo più lungo, pari a 512 chip e con burst di tipo 2 quello con midambolo di 256 chip.

Come già osservato una delle funzioni principali del midambolo è quella della stima del canale, per cui il burst di tipo 1 con midambolo più lungo risulta più adatto ad essere utilizzato nella tratta in up link, dove si deve poter arrivare a stimare fino a 16 differenti risposte impulsive del canale (il che accade quando alloco nello stesso slot 16 utenti diversi che utilizzano differenti codici di canalizzazione con SF=16). I 256 chip di midambolo del burst di tipo 2 risultano invece sufficienti per la stima del canale in down link o anche per uno slot in up link con meno di 4 utenti allocati. Si deve tenere conto però del fatto che in uno stesso slot non possono coesistere burst di tipo diverso.

Entrambi i tipi di burst presentano un periodo di guardia di lunghezza fissata pari a 96 chip (si ricordi che si può parlare indifferentemente di chip o di intervalli temporali in

---

quanto il chip rate è fissato a 3.84 Mchip/s). La presenza di tale periodo di guardia in ogni slot va giustificata in virtù dello scopo, sempre presente negli standard, di fornire la massima flessibilità possibile al sistema. Infatti, abbiamo già visto che la ripartizione degli slot tra le due tratte non è stabilita a priori e deve essere quindi previsto un tempo di switching tra due qualsiasi TS consecutivi per commutare i dispositivi dalla trasmissione alla ricezione o viceversa.

Nel burst di tipo 1 sono assegnati 512 chip per il midambolo e 96 per il periodo di guardia per cui dei 2560 chip complessivi del burst ne restano 1952 per la parte dati, ripartiti in due campi simmetrici di 976 chip ciascuno (fig 8.a).

Il burst di tipo 2 ha un parte dati leggermente più ampia a causa del midambolo più corto e presenta due campi dati da 1104 ciascuno (fig 8.b).

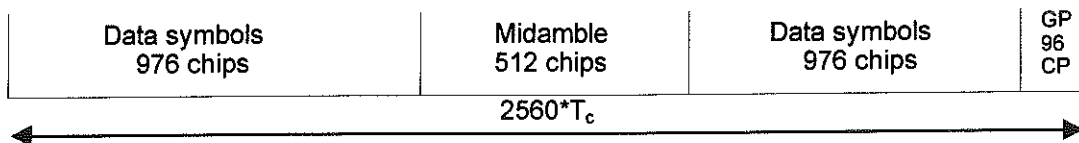


Figura 8.a. Formato di un burst di tipo 1 (midambolo lungo)

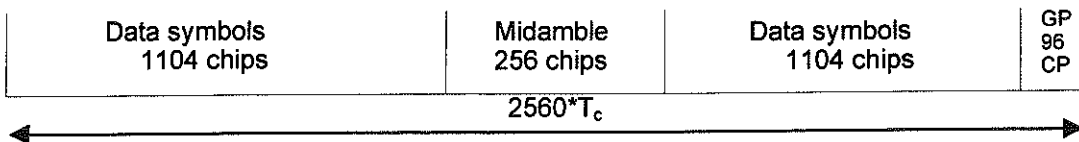


Figura 8.b. Formato di un burst di tipo 2 (midambolo corto)

Per quanto visto sulle tecniche di spreading è chiaro che fissare il numero di chip per burst assegnati alla parte dati non equivale a fissare il numero di simboli di dati presenti in ogni burst. Questo perchè chip e simboli di dati stanno in un rapporto variabile pari al fattore di spreading. Nella tabella 1 sottostante è indicato il numero di simboli per ciascuno dei due campi dati di entrambi i tipi di burst al variare del fattore di spreading.

Spreading factor (Q)	Number of symbols (N) per data field in Burst 1	Number of symbols (N) per data field in Burst 2
1	976	1104
2	488	552
4	244	276
8	122	138
16	61	69

Tabella 1. Numero di simboli per campo dati in burst di tipo 1 e 2

Bisogna a questo punto precisare che non tutti i simboli della parte dati vengono utilizzati effettivamente per il trasporto dell'informazione generata dall'utente, in quanto esistono due tipi di informazioni di controllo che, quando presenti, vengono trasmesse direttamente all'interno del campo dati. Si tratta del TFCI (Transport Format Control Indicator) e del TPC (Transmission Power Control).

Il TFCI funge da indicatore di trasporto del formato ed è opzionale. Può essere trasmesso in entrambi i tipi di burst sia in up link che in down link.

Il TPC contiene i comandi di potenza e può essere trasmesso in entrambi i tipi di burst solo in up link.

La loro trasmissione è negoziata al momento del call setup e può essere rinegoziata durante la chiamata.

La trasmissione di entrambi è fatta all'interno della parte dati del rispettivo canale fisico, in modo da non alterare la lunghezza e la struttura del midambolo. Questo significa che i bit di TFCI e TPC subiscono le stesse procedure di modulazione e di spreading subite dai bit di informazione.

La lunghezza del TFCI può variare tra 0, 4, 8, 16, 32 bit per burst, mentre i bit assegnati al TPC possono essere 0 o 2 a seconda che esso sia presente o meno.

Si deve tener presente che a causa della modulazione QPSK adottata non si può parlare indifferentemente di bit o di simboli, in quanto ad ogni simbolo corrispondono due bit informativi.

Riassumendo il numero di bit disponibili per l'utente in un burst (formato del time slot) dipende dal fattore di spreading, dalla lunghezza del midambolo e dal numero di bit del TFCI, in up link si aggiunge anche la possibile presenza dei bit del TPC.

## II.5.4.2 Canali fisici comuni

### Canale fisico comune di controllo primario

Nel *Primary Common Control Physical Channel (P-CCPCH)* è mappato il BCH.

La posizione di tale canale fisico, cioè il set di codici e slot che lo compongono, è conosciuta dall'utente attraverso il Canale di Sincronizzazione (SCH).

Poichè la quantità di informazioni trasmesse agli utenti in modalità broadcast attraverso il BCH (struttura del frame, posizione del PRACH, ecc.) è pressochè la stessa per ogni frame, non è necessario prevedere una variazione delle dimensioni del P-CCPCH da frame a frame.

(E' prevista la possibilità di richiedere ulteriori risorse per la trasmissione broadcast se il P-CCPCH non è sufficiente. In tal caso lo stesso P-CCPCH indica le risorse aggiuntive di un canale secondario S-CCPCH in cui devono essere mandate le informazioni broadcast aggiuntive).

Per tale canale fisico lo standard prevede l'utilizzo di un burst di tipo 1 senza TFCI e con un fattore di spreading fissato pari a 16. Il codice di canalizzazione utilizzato è sempre  $a_{Q=16}^{(k=1)}$ , il primo di quelli con SF=16 (nell'albero dei codici OVSF si trova in alto a destra).

### Canale fisico comune di controllo secondario

Nel *Secondary Common Control Physical Channel (S-CCPCH)* sono mappati i canali di trasporto PCH e FACH. Poichè la quantità di informazioni contenute in questi canali di trasporto non è fissa, ma può variare con le condizioni di traffico del sistema, è prevista l'allocazione dinamica di uno o più canali secondari in modo da poter adattarne la capacità alle effettive necessità della rete.

Per tali canali fisici secondari è previsto l'utilizzo di entrambi i tipi di burst ed anche del TFCI. Anche in questo caso il fattore di spreading è fissato pari a 16.

---

### Canale fisico di accesso casuale

Il canale di trasporto RACH è mappato in uno o più *Physical Random Access Channel* (PRACH) in up link. Anche in questo caso, come nel precedente, la quantità di informazioni trasportate sul RACH può variare da frame a frame in virtù di un aumento o una diminuzione delle domande di accesso. Potendo stimare in qualche modo il numero di richieste di accesso previste è possibile dimensionare opportunamente il PRACH in modo da trovare un compromesso tra probabilità di collisione degli accessi casuali e spreco di risorse radio.

Per tale canale è previsto un fattore di spreading variabile SF=8 o 16. Non è chiaro perchè si dovrebbe usare un canale con SF=8, al posto di due canali con SF=16, che dimezzerebbero la probabilità di collisione tra gli utenti. L'unica spiegazione è che le dimensioni di una richiesta di accesso potrebbero essere talmente elevate da non poter essere contenute nella parte dati di un burst con SF=16.

La locazione del PRACH all'interno della matrice delle risorse radio unitarie slot/codici è comunicata agli utenti attraverso il BCH, che contiene una apposita sezione per i parametri di configurazione del RACH, cioè il set di codici e slot utilizzabili per una richiesta di accesso.

A differenza degli altri canali fisici che utilizzano i burst di tipo 1 o 2, il formato del burst del PRACH è leggermente diverso in quanto presenta un periodo di guardia leggermente più lungo. Nel burst di un canale fisico di accesso casuale infatti il periodo di guardia raddoppia le sue dimensioni da 96 a 192 chip. I 96 chip in più vengono sottratti al secondo campo dati, rendendo i campi a cavallo del midambolo asimmetrici. Essendo il PRACH un canale di up link, il suo midambolo è lungo ed ha le solite dimensioni di 512 chip.

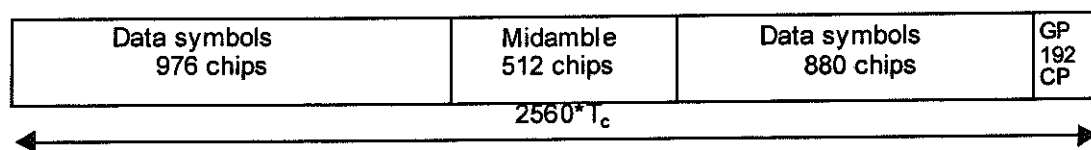


Figura 9. Formato di un burst del PRACH

Nella tabella seguente sono indicati il numero di simboli di ciascuno dei due campi dati di un burst PRACH al variare dello SF.

Spreading factor (Q)	Number of symbols in data field	
	1	2
8	122	110
16	61	55

Tabella 2. Numero di simboli per campo dati in un burst PRACH

### Canale fisico di sincronizzazione

Il *Physical Synchronisation Channel (PSCH)* consta di uno o al massimo due slot temporali in down link per frame. La posizione di tali slot deve essere fissata e, secondo lo standard, può al più variare in un periodo temporale molto lungo.

Lo standard prevede tre casi di allocazione dei canali fisici PSCH e P-CCPCH che riassumiamo di seguito:

1. PSCH e P-CCPCH allocati entrambi nell'unico time slot  $k$ -esimo, con  $k=0\dots16$ ;
2. PSCH allocato in due time slot: il  $k$ -esimo e il  $k+8$ -esimo, con  $k=0\dots6$  e P-CCPCH allocato nello stesso time slot  $k$ -esimo;
3. PSCH allocato in due time slot: il  $k$ -esimo e il  $k+8$ -esimo, con  $k=0\dots6$  e P-CCPCH allocato in un time slot separato  $i$ -esimo indicato all'interno del PSCH.

In tutti e tre i casi la posizione del P-CCPCH è nota una volta conosciuto il PSCH.

Il PSCH consiste in una sequenza di codice primaria e 3 sequenze di codice secondarie (scelte tra 16 possibili), tutte di lunghezza pari a 256 chip.

Le tre sequenze di codice secondarie sono modulate QPSK e trasmesse in parallelo con il codice di sincronizzazione primario.

A seconda di quali delle 16 sequenze di codice secondarie si utilizzano e del modo in cui esse sono modulate (quali sulla componente in fase e quali in quadratura) è possibile risalire al gruppo di codice e al timing del frame associato alla cella. Esistono per ognuno dei tre casi analizzati delle particolari tabelle di corrispondenza in base alle quali si associa ai codici di sincronizzazione secondari uno dei 32 gruppi di codice. Ciascuno dei 32 gruppi di

codice contiene 4 specifici codici di scrambling (che sono in totale 128) e ciascun codice di scrambling è associato con uno specifico codice di midambolo base sia breve che lungo.

A ciascun gruppo di codice è associato inoltre uno specifico tempo di offset,  $t_{\text{offset}}$ , che il sistema TDD utilizza per mantenere la sincronizzazione tra stazioni base. Infatti attraverso le informazioni fornite dal canale di sincronizzazione, la stazione mobile può ricavare l'estremo del frame dalla posizione della sequenza SCH e dalla conoscenza del  $t_{\text{offset}}$ .

### Canali fisici condivisi

Il Physical Up link Shared Channel (PUSCH) ed il Physical Down link Shared Channel (PDSCH) prevedono la stessa struttura di burst di un canale fisico dedicato (DPCH).

Per il PDSCH si pone però il problema di indicare all'utente che ci sono dati da decodificare su tale canale. Per risolverlo sono previsti tre metodi di segnalazione:

- usare il campo TFCI di un canale associato o del PDSCH;
- usare un midambolo specifico dell'utente sul DSCH ricavato dal set di midamboli ricavati per quella cella;
- utilizzare segnalazioni di strati più alti.

Quando si utilizza il metodo basato sul midambolo, l'utente deve decodificare il PDSCH solo se esso è stato trasmesso con il midambolo indicato per lui dal sistema.

Lo schema seguente riassume il modo in cui i canali di trasporto sono mappati sulle risorse fisiche.

---

Transport Channels	Physical Channels
DCH	Dedicated Physical Channel (DPCH)
BCH	Primary Common Control Physical Channel (P-CCPCH)
FACH PCH	Secondary Common Control Physical Channel (S-CCPCH)
RACH	Physical Random Access Channel (PRACH)
SCH	Physical Synchronisation Channel (PSCH)
USCH	Physical Up link Shared Channel (PUSCH)
DSCH	Physical Down link Shared Channel (PDSCH)

Figura 10. Mappatura dei canali di trasporto sui canali fisici

## II.6 Lo strato MAC

Lo strato MAC costituisce, nelle definizioni delle specifiche, la parte che presenta maggiore incompletezza. Molte delle sue componenti infatti sono ancora in fase di studio. Verranno qui presentate quelle parti che sembrano godere di maggior stabilità e che sono di interesse in questo lavoro.

### II.6.1 I canali logici

I Logical Channel sono i servizi forniti dallo strato MAC agli strati superiori. Riportiamo quelli di nostro interesse.

#### 1) *Control Channel*

I seguenti canali di controllo sono utilizzati per il trasferimento dell'informazione nel Control plane (C-plane):

- Synchronisation Control Channel (SCCH)
- Broadcast Control Channel (BCCH)
- Paging Control Channel (PCCH)
- Common Control Channel (CCCH)
- Dedicated Control Channel (DCCH)

## 2) Traffic Channel

I seguenti canali di traffico sono utilizzati per il trasferimento dell'informazione nello User plane (U-plane):

- Dedicated Traffic Channel (DTCH)
- Common Traffic Channel (CTCH)

Vengono eseguite le seguenti connessioni tra i Logical Channel ed i Transport Channel (descritti in precedenza):

- Il SCCH è connesso al SCH
- Il BCCH è connesso al BCH
- Il PCCH è connesso al PCH
- Il CCCH è connesso al RACH e al FACH
- Il DCCH e il DTCH possono essere connessi al RACH e al FACH, o a un DCH
- Il CTCH può essere connesso al FACH o al BCH (in fase di studio)

## II.6.2 Le funzioni del MAC

Le funzioni del protocollo MAC includono:

- Mappatura tra Logical Channel e Transport Channel
  - Gestione della priorità tra gli UE
  - Scheduling dei messaggi di broadcast, paging e notification
  - Identificazione degli UE sui Common Transport Channel
  - Multiplazione (demultiplazione) delle PDU di strato superiore consegnate allo (ricevute dallo) strato fisico nei Common Transport Channel
  - Multiplazione (demultiplazione) delle PDU di strato superiore consegnate allo (ricevute dallo) strato fisico nei Dedicated Transport Channel
  - Controllo del volume di traffico
-

- Trasporto della segnalazione degli strati superiori
- Mantenimento di una connessione di segnalazione MAC con l'entità MAC remota

Altre funzionalità sono in fase di studio.

### II.6.3 Le Protocol Data Unit (PDU)

Sono previste due tipologie di unità dati del protocollo MAC: la MAC-Data-PDU e la MAC-Control-PDU. La prima è utilizzata per il trasporto dei dati, la seconda per il trasporto della segnalazione. Ad oggi è stata definita, e non completamente, solo la MAC-Data-PDU (che verrà qui descritta), mentre la MAC-Control-PDU è ancora in fase di studio.

La MAC-Data-PDU è formata da un'intestazione (header) opzionale e da una MAC Service Data Unit (MAC-SDU), come mostrato in Figura 11. Sia l'header che la MAC-SDU

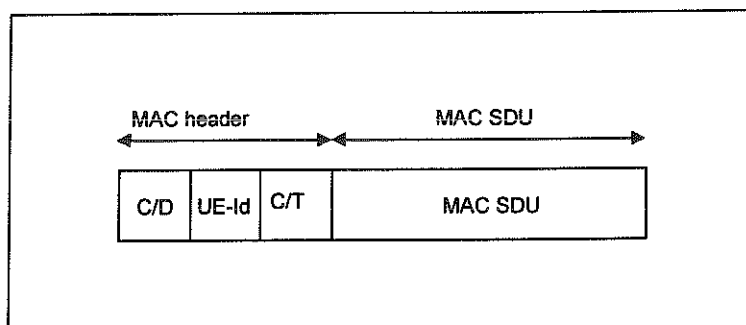


Figura 11. La MAC-Data-PDU

sono di dimensione variabile.

Il contenuto e la dimensione dell'header dipende dal tipo di Logical Channel, ed in qualche caso nessuno dei parametri dell'intestazione risulta necessario.

La dimensione della MAC-SDU dipende dalla dimensione dell'unità dati dello strato RLC (la RLC-PDU).

Di seguito sono descritti i parametri che compaiono nell'header della MAC-Data-PDU:

- Campo C/D

Il campo C/D è costituito da un singolo bit che identifica a quale classe appartiene il

---

Logical Channel che si sta trasportando sul FACH o sul RACH (ovvero se si sta trasportando un CCCH o un Dedicated Logical Channel).

C/D field	Designation
1	CCCH
0	DCCH or DTCH

Tabella 1. Il campo C/D

- Campo C/T

Il campo C/T identifica il Logical Channel quando più Logical Channel sono trasportati sullo stesso Transport Channel. La dimensione di questo campo può essere variabile.

C/T field (e.g. 4 bits)	Designation
0000	Logical channel 1
0001	Logical channel 2
...	...
1111	Logical channel 16

Tabella 2. Il campo C/T

- Campo UE-Id

Il campo UE-Id fornisce un identificativo del mobile

Come detto in alcuni casi nessuno dei parametri dell'intestazione risulta necessario. In particolare il loro utilizzo per il DTCH e DCCH è il seguente:

a) DTCH o DCCH mappati nel DCH, assenza di moltiplicazione:

L'intestazione MAC non è necessaria

b) DTCH o DCCH mappati nel DCH, moltiplicazione di più canali dedicati:

Solo il campo C/T è incluso nell'intestazione MAC

c) DTCH o DCCH mappati nel RACH/FACH:

I campi C/D e UE-Id sono inclusi nell'intestazione. Il campo C/T è incluso se è presente  
 multiplazione

d) DTCH o DCCH mappati nel RACH/FACH, ove DTCH o DCCH sono gli unici canali:

Solo il campo UE-Id è incluso nell'intestazione. Il campo C/T è incluso se è presente  
 multiplazione

L'utilizzo dei parametri per il CCCH è invece il seguente:

a) CCCH mappato nel RACH/FACH:

Solo il campo C/D è incluso nell'intestazione. L'inclusione del campo UE-Id è in fase di  
 studio.

b) CCCH mappato nel RACH/FACH, ove il CCCH è l'unico canale:

L'inclusione del campo UE-Id è in fase di studio.

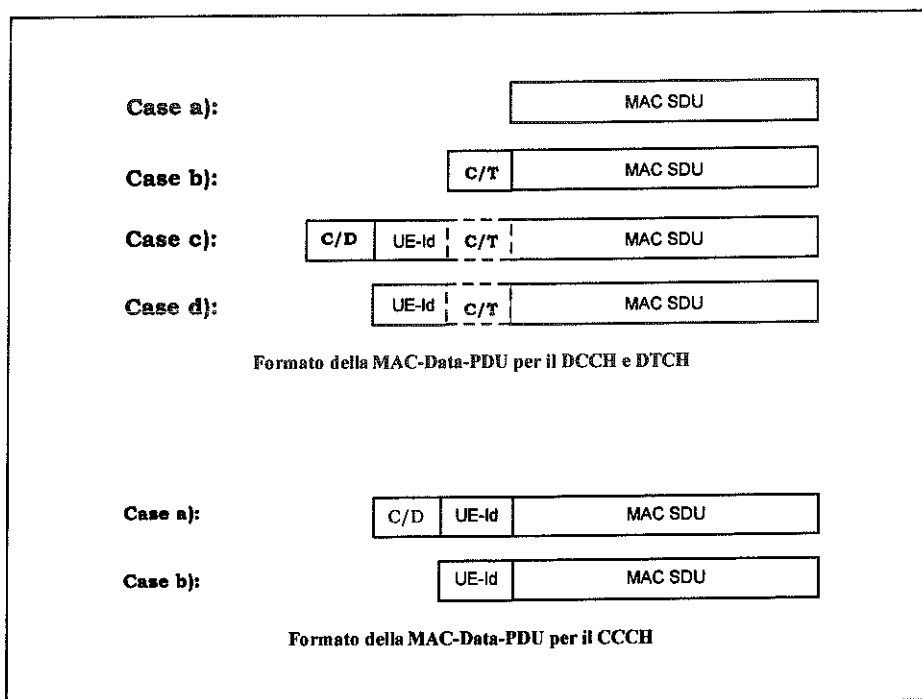


Figura 12. Formato della Mac-Data-PDU

## II.7 Lo strato RLC

### II.7.1 Tre modalità di trasmissione

Lo strato RLC prevede tre differenti modalità di trasmissione.

#### 1) Transparent data transfer

Questa modalità opera la trasmissione delle PDU dello strato superiore senza aggiungere ad esse alcuna informazione di strato RLC (Figura 13). Essa include eventualmente la funzione di segmentazione/riasseblamento.

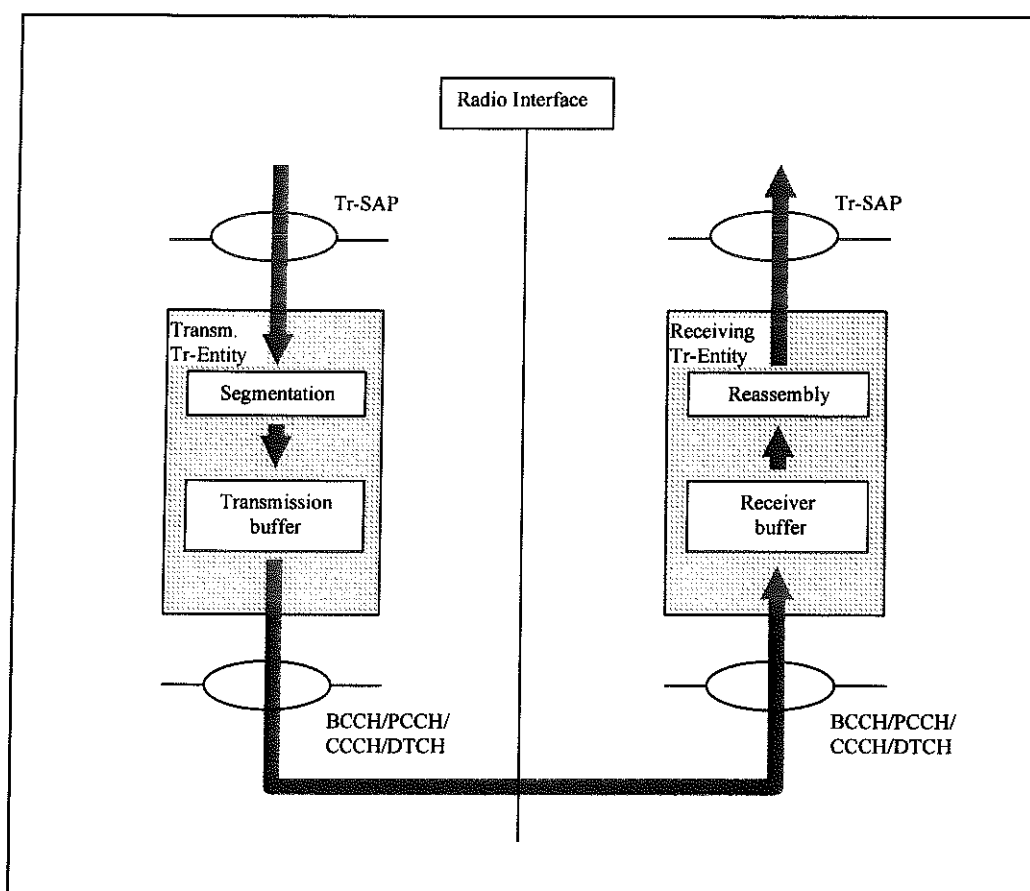


Figura 13. Modello di due entità RLC remote operanti in "transparent mode"

## 2) Unacknowledged data transfer

Questa modalità opera la trasmissione delle PDU di strato superiore senza garantire la consegna all'entità RLC remota (Figura 14). Essa presenta le seguenti caratteristiche:

- rivelazione di errore: l'entità RLC ricevente consegnerà allo strato superiore solo quelle SDU che non presentano errori.
- consegna unica: l'entità RLC consegnerà allo strato superiore ogni SDU soltanto una volta, utilizzando la funzione di rivelazione di duplicati
- consegna immediata: l'entità RLC ricevente consegnerà la SDU allo strato superiore appena essa viene ricevuta.

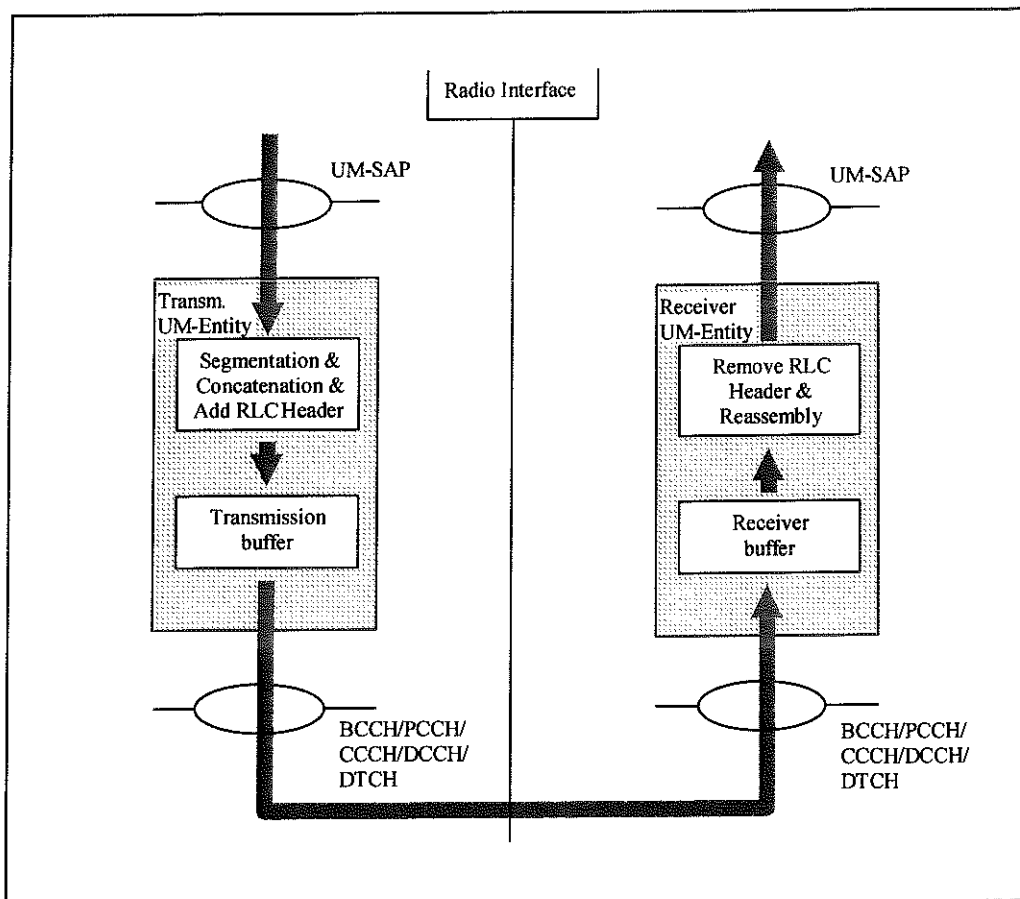


Figura 14. Modello di due entità RLC remote operanti in "unacknowledged mode"

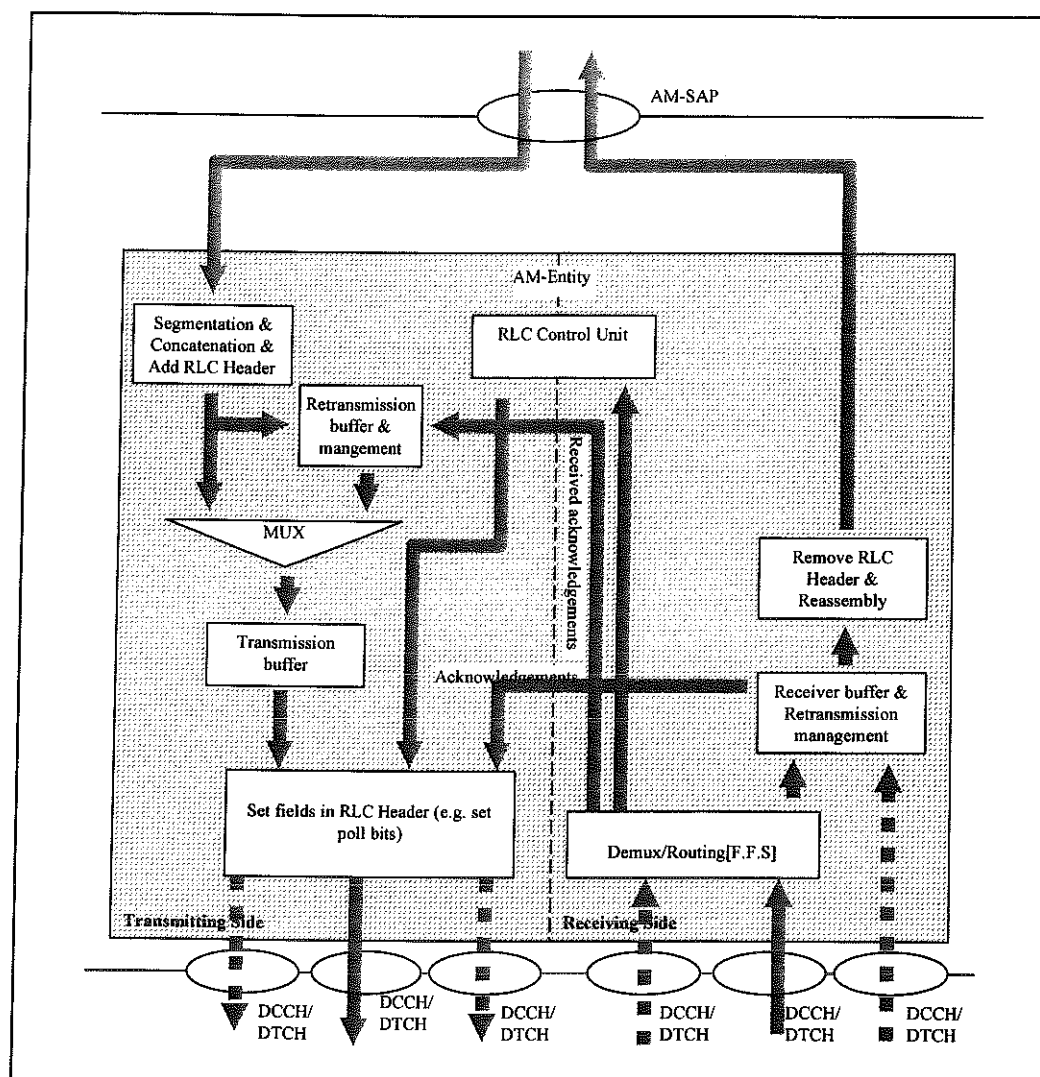


Figura 15. Modello di un'entità RLC operante in "acknowledged mode"

### 3) Acknowledged data transfer

Questa modalità opera la trasmissione delle PDU di strato superiore garantendo la consegna all'entità RLC remota (Figura 15). Per questa modalità è prevista sia la consegna in sequenza che la consegna fuori sequenza. In caso di consegna fuori sequenza sarà il protocollo di strato superiore a ristabilire l'ordine delle sue PDU. Lo "acknowledged mode" presenta le seguenti caratteristiche:

- consegna senza errori: l'entità RLC ricevente consegna allo strato superiore le SDU prive di errori utilizzando la funzione di ritrasmissione.

- consegna unica: l'entità RLC consegnerà allo strato superiore ogni SDU soltanto una volta, utilizzando la funzione di rivelazione di duplicati
- consegna in sequenza: lo strato RLC consegna all'entità di strato superiore ricevente le SDU nello stesso ordine in cui l'entità di strato superiore che trasmette le consegna allo strato RLC.
- consegna fuori sequenza: alternativamente alla consegna in sequenza, lo strato RLC consegna all'entità di strato superiore ricevente le SDU in ordine differente da quello in cui l'entità di strato superiore che trasmette le consegna allo strato RLC.

## II.7.2 Le funzioni del RLC

Le funzioni eseguite dallo strato RLC sono le seguenti:

- Controllo della connessione: questa funzione provvede alla creazione, al mantenimento e al rilascio di una connessione RLC.
  - Segmentazione e riassetaggio: questa funzione provvede alla segmentazione (riassetaggio) delle PDU di strato superiore di lunghezza variabile in (a partire dalle) RLC PDU più piccole di dimensione fissa.
  - Concatenazione: se il contenuto di una RLC SDU non riempie un numero intero di RLC PDU, il primo segmento della successiva RLC SDU può essere messo nella stessa RLC PDU ove è presente l'ultimo segmento della RLC SDU considerata.
  - Riempimento: quando la concatenazione non è possibile, la RLC PDU che contiene l'ultimo segmento può essere completata con bit di riempimento.
  - Correzione d'errore: questa funzione provvede alla correzione di errore facendo uso di ritrasmissioni (nel "acknowledged mode").
  - Rivelazione di duplicati: questa funzione provvede alla rivelazione delle RLC PDU ricevute più di una volta e assicura che vengano consegnate allo strato superiore un'unica volta.
  - Controllo di flusso: questa funzione permette all'entità RLC ricevente di controllare la
-

velocità a cui l'entità RLC che trasmette deve inviare i dati.

- Controllo del numero di sequenza (unacknowledged mode): questa funzione garantisce l'integrità delle RLC PDU riassembleate. Inoltre provvede a scartare quelle RLC SDU che risultino corrotte dal controllo del numero di sequenza delle RLC PDU che le compongono.

### II.7.3 Le Protocol Data Unit (PDU)

Le PDU dello strato RLC sono divise in due gruppi:

#### 1. Data PDU

- *AMD PDU* (Acknowledged Mode Data PDU)

E' utilizzata per trasportare PDU numerate in sequenza contenenti dati di una RLC SDU. Viene utilizzata dall'entità RLC che opera nel "acknowledged mode".

- *UMD PDU* (Unacknowledged Mode Data PDU)

E' utilizzata per trasportare PDU numerate in sequenza contenenti dati di una RLC SDU. Viene utilizzata dall'entità RLC che opera nel "unacknowledged mode".

#### 2. Control PDU

- *BGN PDU* (Begin)

E' utilizzata da un'entità RLC per stabilire una connessione con un'entità RLC remota.

- *BGACK PDU* (Begin Acknowledge)

E' un riscontro positivo alla BGN PDU.

- *BGREJ PDU* (Begin Reject)

E' utilizzata per rifiutare una richiesta di connessione da parte di un'entità RLC remota.

- *END PDU* (End)

E' utilizzata per rilasciare una connessione con un'entità RLC remota.

- *ENDACK PDU* (End Acknowledge)

E' un riscontro positivo alla END PDU.

---

- *STATUS PDU* (Status Response)

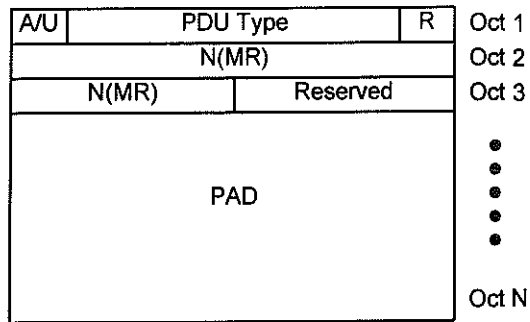
E' utilizzata dall'entità RLC ricevente per informare l'entità RLC che trasmette che alcune PDU sono andate perse nella trasmissione. Il suo utilizzo può essere causato dalla rivelazione di PDU mancanti da parte dell'entità ricevente, oppure in conseguenza ad una richiesta di riscontro da parte dell'entità che trasmette.

Nella pagina successiva sono riportati i formati delle varie PDU. Viene poi presentata una descrizione dei parametri che in esse compaiono.

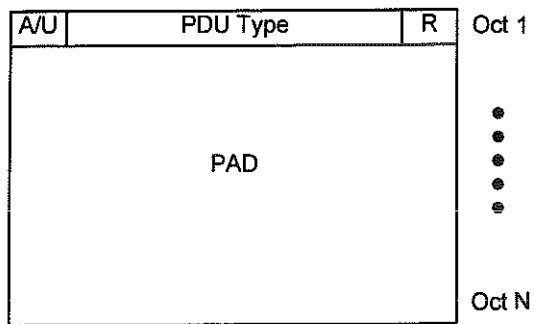
---



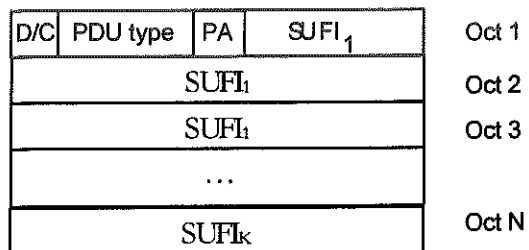
### BGAK PDU



### BGREJ, END, ENDACK PDU



### STATUS PDU



I parametri che compaiono nelle PDU hanno i seguenti significati:

- **A/U bit: 1 bit**

Questo bit indica se si tratta di una AMD PDU oppure di una UMD PDU/Control PDU.

Bit	Description
0	Unacknowledged mode data PDU/ Control PDU
1	Acknowledged mode data PDU

- **D/C bit: 1 bit**

Questo bit indica il tipo di un "acknowledged mode" PDU. Può essere una Data PDU o una Control PDU.

Bit	Description
0	Control PDU
1	Acknowledged mode data PDU

- **PDU Type: 6 bit (3 bit per le STATUS PDU)**

Indica il tipo di Control PDU

Bit	PDU Type	Bit	PDU Type
111111	BGN	111010	STATUS
111110	BGAK		
111101	BGREJ	111001 – 110000	Reserved
111100	END		
111011	ENDAK		

- **Sequence Number (SN)**

Indica il numero di sequenza dei segmenti RLC.

PDU type	Length	Notes
AMD PDU	12 bit	Used for retransmission and reassembly
UMD PDU	6 bit	Used for reassembly <i>Especially "110000" – "111111" are reserved for PDU Type (Control PDU)</i>

- **Polling bit (P): 1 bit**

E' utilizzato dall'entità RLC che trasmette per la richiesta di una STATUS PDU da parte dell'entità RLC ricevente.

---

Bit	Description
0	-
1	Request a status report

- **Exstention bit (E):** 1 bit

Indica se il byte successivo fa parte del campo intestazione o del campo dati

Bit	Description
0	The next octet is data
1	The next octet is header information (LI)

- **Reserved (R):**

Il suo utilizzo è in fase di studio

- **Lenght Indicator (LI):** 7 bit

E' opzionale e viene utilizzato se sono applicate le funzioni di concatenazione o di riempimento. Indica la fine dell'ultimo segmento di una SDU.

- **Poll Answer (PA):** 1 bit

Indica se la STATUS PDU è la risposta ad una richiesta o no.

Bit	Description
0	The status report is not the answer to a polling request
1	The status report is the answer to a polling request

- **SUFI (SUper-Field):** 1 bit

Il SUFI comprende tre campi:

- *Type* (2 bit): indica il tipo di super-field, che può essere list, bitmap o acknowledgment.

Bit	Description
00	List ( <b>LIST</b> )
01	Bitmap ( <b>BITMAP</b> )
10	Acknowledgement ( <b>ACK</b> )
11	No More Data ( <b>NO_MORE</b> ) [H1]

---

- *Length* (dipende dal tipo di super-field): indica la lunghezza, variabile, del campo Value.
- *Value*: numero di bit variabile che dipende dai campi Type e Length.

#### Il SUFI di tipo List:

In tal caso il campo Type assume il valore LIST, il campo Length un valore N ed il campo Value è costituito da una lista di N coppie del tipo (SN, L).

Type = LIST
Length
SN <sub>1</sub>
L <sub>1</sub>
SN <sub>2</sub>
L <sub>2</sub>
...
SN <sub>LENGTH</sub>
L <sub>LENGTH</sub>

Length (4 bit): il numero di coppie (SN, L) presenti nel campo Value.

SN<sub>i</sub> (12 bit): il numero di sequenza della PDU che non è stata ricevuta correttamente.

L<sub>i</sub> (4 bit): il numero di PDU consecutive non ricevute correttamente a partire da quella di numero di sequenza SN<sub>i</sub>.

#### Il SUFI di tipo Bitmap:

Il campo Type assume il valore BITMAP, il campo Length un valore N, ed il campo Value è costituito da un numero di sequenza (FSN) ed una bitmap.

Type = BITMAP
LENGTH
FSN
Bitmap

Length (4 bit): la dimensione della bitmap in byte ( max dimensione = 128 bit).

FSN (12 bit): il numero di sequenza del primo bit nella bitmap.

Bitmap: stato di ricezione delle PDU aventi numero di sequenza compreso nell'intervallo

[FSN, FSN + N\*8 - 1]: se il bit di posizione  $i$  nella bitmap ha valore 0 la PDU con numero di sequenza pari a (FSN +  $i$ ) non è stata ricevuta correttamente, altrimenti (se il bit ha valore 1) la PDU è stata ricevuta correttamente.

**Il SUFI di tipo Acknowledgment:**

Il campo Type assume il valore ACK, il campo Length è assente, ed il campo Value è costituito da un numero di sequenza (LSN).

Type = ACK
LSN

LSN (12 bit): riscontra la corretta ricezione di tutte le PDU con numero di sequenza minore di LSN (Last Sequence Number) e che non sono state indicate come ricevute in maniera errata in una parte precedente della STATUS PDU.

**Il SUFI di tipo No More Data:**

Tale super-field indica la fine della parte dati di una STATUS PDU.

- **N(SQ):** 1 bit

Questo campo indica il numero di sequenza della connessione. Quando una nuova BGN PDU viene trasmessa, tale valore viene incrementato. Permette al ricevitore di identificare le BGN PDU ritrasmesse.

- **N(MR):** 12 bit

Numero di sequenza della prima PDU che l'entità ricevente non è in grado di accettare.

- **Data:**

In questo campo vengono inseriti i dati provenienti dallo strato superiore.

---

## II.7.4 La tecnica di ritrasmissione

Lo schema di ritrasmissione utilizzato è del tipo a ritrasmissione selettiva (Selective Retransmission). In tale schema intervengono due tipi di riscontri da parte dell'entità RLC ricevente:

- l'entità ricevente deve inviare una STATUS PDU all'entità che trasmette, in risposta ad una richiesta esplicita da parte di quest'ultima. La richiesta viene effettuata settando il bit di Polling (P) nella PDU trasmessa.
- l'entità ricevente può inviare di sua iniziativa una STATUS PDU all'entità che trasmette qualora si accorga dell'errata ricezione di una o più PDU (si parla in questo caso di Unsolicited STATUS PDU).

L'entità ricevente dovrebbe richiedere l'invio di una STATUS PDU solo quando trasmette l'ultimo segmento di una RLC SDU, riducendo in tal modo l'overhead e migliorando il throughput. Tuttavia, se le RLC SDU sono molto piccole, si avrà ugualmente un overhead piuttosto elevato. Per risolvere il problema viene introdotto un timer che non permette un eccessivo numero di richieste.

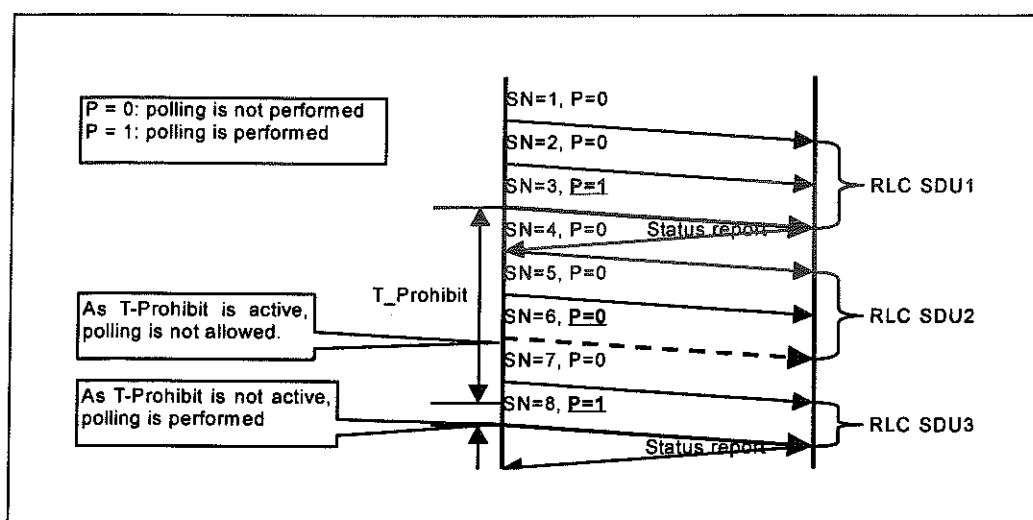


Figura 16. Riduzione di eccessive richieste

In particolare i timer che intervengono nello schema di ritrasmissione sono due:

- *Timer\_STAT*: Viene attivato quando una AMD PDU con il bit di Polling settato viene trasmessa. Viene fermato quando l'entità che trasmette riceve un Ack o un Nack per l'AMD PDU trasmessa. Se viene effettuata un'altra richiesta (polling) quando questo timer è attivo, esso viene fermato e riattivato.
- *Timer\_Prohibit*: E' il timer citato in precedenza. Viene attivato quando una AMD PDU con il bit di Polling settato viene trasmessa. Mentre è attivo non è possibile inviare ulteriori richieste di riscontri di PDU contenenti l'ultimo segmento di una SDU. Se viene effettuata un'altra richiesta (polling) quando questo timer è attivo (dovuta però esclusivamente a riscontri di PDU ritrasmesse), esso viene fermato e riattivato. Non viene fermato alla ricezione di Ack o Nack. Non viene eseguita alcuna azione al momento della sua scadenza.

La ritrasmissione delle PDU avviene dunque in tre casi:

- 1) Viene ricevuta una STATUS PDU (precedentemente richiesta) che richiede la ritrasmissione di alcune PDU. In tal caso, nella ritrasmissione dell'ultima PDU richiesta viene settato di nuovo il bit di Polling.

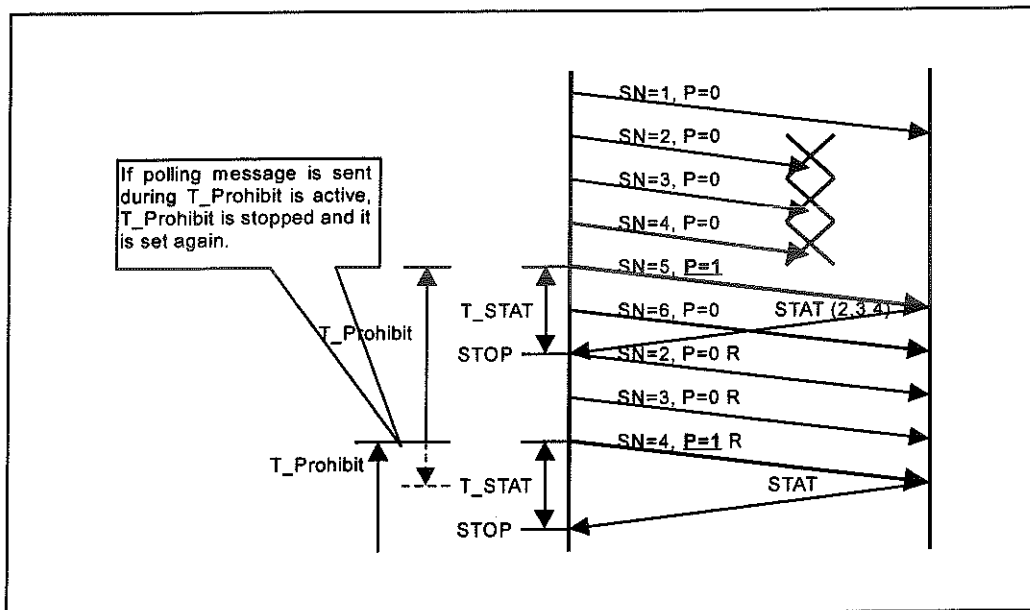


Figura 17. Ritrasmissione causata dalla ricezione di una STATUS PDU richiesta

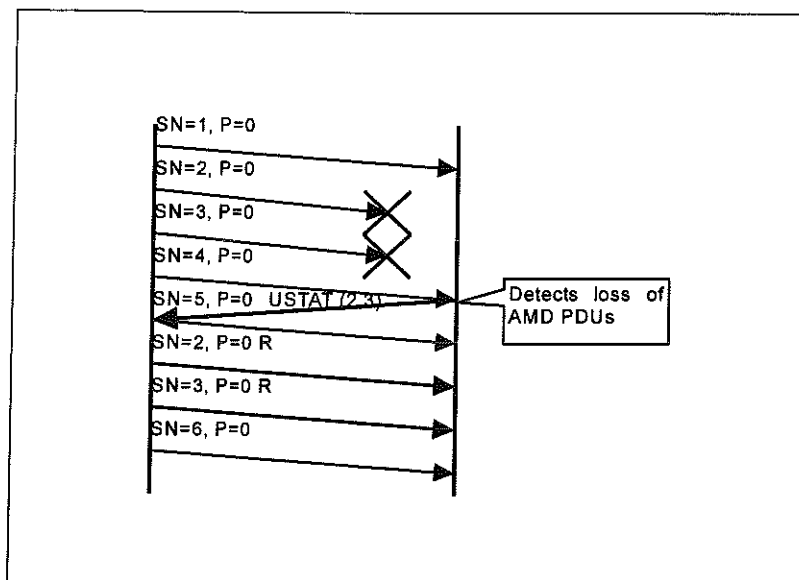


figura 18. Ritrasmissione causata dalla ricezione di una U-STATUS PDU

- 2) Viene ricevuta una Unsolicited STATUS PDU. In tal caso vengono ritrasmesse le PDU richieste ma non viene settato di nuovo il bit di Polling.
- 3) Il Timer\_STAT scade (a causa della perdita, durante la trasmissione, della AMD PDU che conteneva il bit di Polling settato oppure della conseguente STATUS PDU). In tal caso viene ritrasmessa l'ultima AMD PDU trasmessa in cui viene settato il bit di Polling. Il Timer\_STAT viene riavviato.

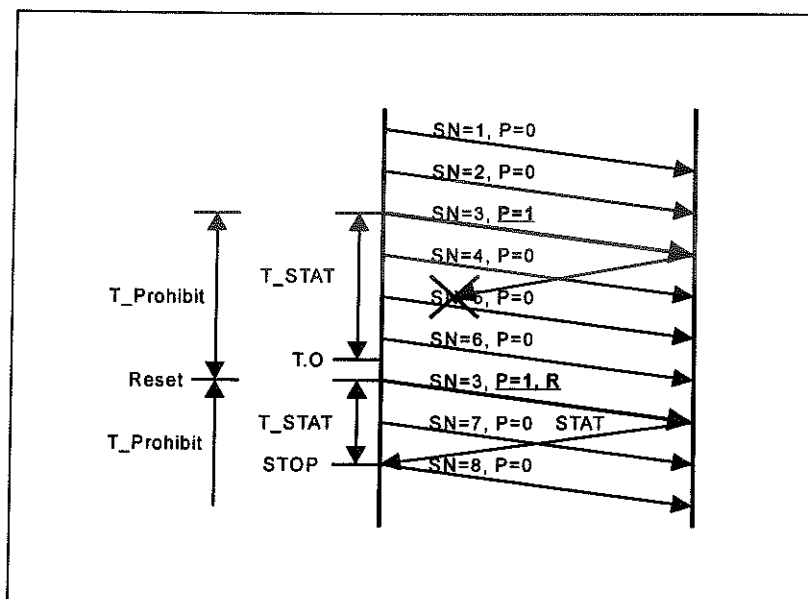


Figura 19. Ritrasmissione causata dalla scadenza del Timer\_STAT



---

## CAPITOLO

# 3

---

## ALGORITMI PROPOSTI PER LA GESTIONE DELLE RISORSE RADIO

*Una volta presentato lo stato dell'arte del sistema UMTS, con particolare riferimento all'interfaccia radio ed ai primi due livelli della pila dei protocolli OSI, verranno di seguito introdotte le problematiche affrontate in questo lavoro.*

*Si tenga conto che, data l'originalità del contesto studiato, ci si è trovati di fronte ad una vastissima gamma di possibili argomenti di ricerca, tutti di notevole interesse. Individuare delle priorità non è stato affatto semplice, anche perché la maggior parte dei lavori di ricerca sull'argomento sono quasi tutti in corso d'opera.*

*Sulle scelte fatte ha influito anche l'attuale stato di avanzamento delle specifiche. Si è scelto infatti di trattare delle problematiche concernenti parti del sistema le cui specifiche possono considerarsi sufficientemente consolidate.*

### **III.1 Traffico Internet su UMTS**

Si è già discusso dell'inarrestabile crescita che negli ultimi anni ha interessato il settore della telefonia mobile. I sistemi radiomobili hanno ormai raggiunto livelli di utenza che fino a pochi anni fa nessuno aveva il coraggio di pensare. Le più recenti previsioni indicano che entro il 2003, a livello mondiale, ci saranno più di un miliardo di terminali in esercizio, e questo significa un superamento del numero di linee fisse telefoniche previste per quella data (cosa peraltro già avvenuta in alcune realtà nazionali come l'Italia).

A questa sbalorditiva crescita si accompagna l'esplosione sul mercato di un altro settore in rapida espansione: Internet. Su tale fronte ogni mese si registrano circa 18 milioni di nuovi utenti, con un traffico dati che raddoppia ogni 6 mesi circa, al punto che si può ormai tranquillamente asserire che Internet sta diventando il mezzo più importante per raccogliere e distribuire informazione a livello mondiale.

La telefonia mobile e la trasmissione dati tramite Internet sono quindi, senza ombra di dubbio, i due più grandi successi di fine secolo nel campo delle telecomunicazioni, e, da quanto si può osservare, questo potrebbe essere solo il punto di partenza per il nuovo millennio.

In quest'ottica il nuovo settore di comunicazioni mobili multimediali di terza generazione permetterà di far confluire in un'unica realtà i passi evolutivi di cui sopra, esaltando, per la telefonia mobile e per Internet, le pur promettenti prospettive separate di crescita.

Il sistema UMTS nasce pertanto in una fase di convergenza tra il mondo delle telecomunicazioni e quello dell'*Information Technology*. Questo nuovo sistema si presenta come un'ambiziosa sintesi di queste due storie evolutive, soprattutto sul piano dei servizi.

Ecco spiegata la nostra scelta di occuparci dell'introduzione del traffico dati Internet su piattaforma UMTS. In particolare come primo approccio si è scelto di analizzare un traffico dati senza garanzie sulla qualità del servizio (QoS), come il traffico Internet attualmente presente su rete fissa, che sarà presumibilmente il primo nuovo servizio ad essere offerto agli utenti dei sistemi di terza generazione.

---

Per quanto riguarda il traffico voce, servizio centrale dei sistemi di seconda generazione, si ipotizza che, almeno in una prima fase evolutiva di UMTS, essa venga mantenuto prevalentemente sulla rete GSM. Le stime prevedono che inizialmente soltanto una piccola parte di tale traffico (circa il 10% del traffico complessivo del sistema) venga dirottata sulla rete UMTS. Questa politica è resa possibile dal fatto che tutti i terminali UMTS devono essere di tipo *dual mode*, cioè poter commutare da un funzionamento GSM-GPRS ad uno UMTS.

In questo scenario la maggior parte della capacità del sistema, cioè il restante 90%, viene riservata ai nuovi servizi che presumibilmente, almeno in una prima fase, saranno incentrati sulla trasmissione dati Internet. Lo scenario evolutivo prevede solo successivamente l'introduzione di nuovi servizi multimediali che, richiedendo stringenti garanzie sulla QoS, introdurranno nuove problematiche di studio.

### **III.2 Ripartizione della capacità del sistema tra up e down link**

Una delle caratteristiche principali del traffico Internet è quella di presentare una forte asimmetria tra le due direzioni di traffico.

In virtù di questa considerazione si è scelto nel seguente lavoro di analizzare la rete di accesso UMTS con interfaccia radio cosiddetta TD-CDMA, che è quella che meglio si adatta ad un traffico di tipo asimmetrico tra le tratte di up e down link.

Nella soluzione alternativa W-CDMA le due direzioni di traffico vengono separate su due bande distinte e di pari capacità (bande appaiate) secondo la tecnica di Frequency Division Duplex (FDD). In questo modo la ripartizione delle risorse radio disponibili (frequenze e codici) è fissata a priori in modo simmetrico. Con tale scelta si possono gestire idoneamente servizi con traffico bilanciato, quale è tipicamente quello voce, ma non è in alcun modo possibile adattare la capacità del sistema a traffici asimmetrici.

Infatti, se in una delle due direzioni il traffico generato è maggiore della capacità del sistema e nell'altra minore, inevitabilmente le risorse frequenziali assegnate a quest'ultima vengono sprecate a scapito della tratta maggiormente caricata in cui gli utenti

---

sperimenteranno un degrado della qualità del servizio in termini di aumento del ritardo di trasmissione o di rifiuto della connessione.

Questo problema può essere superato attraverso l'adozione di una tecnica di duplexing a divisione di tempo (Time Division Duplex), come quella presente nella soluzione ibrida TD-CDMA. In questo caso la ripartizione della capacità del sistema tra le due tratte non viene effettuata in frequenza, ma attraverso i time slot della trama radio. Assegnando un numero diverso di time slot alle due tratte è possibile adattare la capacità del sistema anche a traffici asimmetrici.

Dal punto di vista della ripartizione dei time slot della trama radio le specifiche UMTS non sono affatto vincolanti. L'unica imposizione è quella di assegnare ogni time slot ad un'unica direzione di traffico e garantirne almeno uno per entrambe le tratte in ogni frame. Rispettate tali richieste sono possibili le 4 soluzioni prese in esame nel capitolo 2 (par. II.5.2).

Tralascieremo il problema del punto di swicthing singolo o multiplo, in quanto esso non si inserisce nel contesto in esame. La soluzione con punti di swicthing multipli presenta più passaggi all'interno della stessa trama tra time slot in up e down link. Ciò vuol dire che i time slot in up e down link non sono tutti compattati all'inizio o alla fine della trama, ma vi si alternano. Ciò giustifica tra l'altro la scelta di inserire alla fine di ciascun burst un periodo di guardia che consenta la commutazione dei dispositivi dallo stato di ricezione a quello di trasmissione.

Una soluzione con swicthing multiplo acquista significato in un contesto di elevata velocità di spostamento del mobile e quindi in presenza di *fading veloce*, in cui si suppone che le caratteristiche del canale radiomobile possano variare in un intervallo di tempo paragonabile alla durata del frame (10 ms). In questo scenario è chiaro che ciascun utente, vedendo un canale propagativo proprio, è caratterizzato da un certo andamento della probabilità di errore al variare del time slot di una stessa trama. Conseguentemente può essere più o meno vantaggioso, in termini di BER, allocare in un certo time slot un utente

---

piuttosto che un altro. Ecco il perché dell'utilizzo di punti di swicthing multipli all'interno della stessa trama.

Alcuni studi hanno tentato di approcciare tecniche di "allocazione veloce" degli utenti, ma la complessità di tale problema ha fatto sì che a tutt'oggi non siano presenti in letteratura lavori validi sull'argomento.

Nel nostro caso quindi si è deciso di analizzare un caso di bassa velocità di spostamento degli utenti e quindi di *fading lento* in cui è possibile considerare le caratteristiche del canale di propagazione visto da ciascun utente costanti per un periodo di tempo pari ad almeno qualche decina di trame (tempi di variazione dell'ordine del secondo o del decimo di secondo). Conseguentemente la probabilità di errore sperimentata da un utente è costante all'interno di un intero frame. Per un dato utente tutti i time slot sono equivalenti dal punto di vista del BER ed è quindi possibile considerare compattati quelli assegnati all'up e al down link (rispetto alla soluzione con punti di swicthing multipli c'è soltanto una differenza di ritardo di propagazione dovuto alla disposizione dei time slot che comunque, essendo contenuta al di sotto della durata del frame, è considerata trascurabile).

Fatte queste necessarie premesse resta da analizzare il problema di *quanti* (e non *quali*) time slot in ogni frame devono essere assegnati rispettivamente all'up e al down link.

Si presentano due soluzioni alternative, una che abbiamo chiamato *statica*, che ripartisce una volta per tutte il numero dei time slot in base a considerazioni statistiche sul bilanciamento del traffico e mantiene fisso il bound tra up e down link all'interno della matrice tempo-codice ed una *dinamica* che varia nel tempo la posizione di tale bound in funzione del traffico.

E' interessante studiare le prestazioni di queste due scelte in quanto, come già osservato, la soluzione TD-CDMA trova il suo punto di forza proprio nella sua "adattabilità" a servizi di tipo asimmetrico.

E' da considerare infatti che l'alto grado di asimmetria che caratterizza i servizi dati è presente sia a livello di "sessione" (trasmissione terminale-rete dell'ordine dei minuti), sia su archi temporali più lunghi (la giornata). Si pensi che il livello di asimmetria (definito come il

---

rapporto tra i bit trasmessi sul down link e quelli trasmessi sull'up link) può raggiungere valori di svariate decine o centinaia nell'ambito della stessa sessione. Fissando in modo statico il numero di time slot per le due tratte è possibile quindi adattarsi alla asimmetria del traffico soltanto in media, precludendosi la possibilità di seguire le caratteristiche oscillazioni nel breve periodo.

Alla luce di queste considerazioni si è pensato di proporre un algoritmo di ripartizione dinamica dei time slot. Al momento dell'allocazione delle risorse e quindi ad ogni frame, lo scheduler calcola il numero di utenti attivi per quella trama rispettivamente nelle due tratte e ripartisce proporzionalmente i time slot. Per nostra scelta eventuali resti nella ripartizione vengono assegnati al down link.

Questo metodo di ripartizione ha ovviamente senso in un contesto di scheduling in cui, supponendo che tutti gli utenti generino mediamente un traffico omogeneo, si mira ad ottenere la massima *fairness* nella ripartizione delle risorse.

E' intuitivo supporre che questo algoritmo di scheduling *dinamico* delle risorse presenta inequivocabili vantaggi rispetto ad uno che considera una ripartizione statica dei time slot basata su considerazioni statistiche del bilanciamento del traffico su un lungo periodo temporale.

Nel capitolo 5 analizzeremo gli scostamenti in termini di throughput e di ritardi di trasmissione tra le due tecniche, ma per ben interpretare i vantaggi evidenziati a favore della tecnica *dinamica* va chiarito un problema che non è stato possibile inserire nelle nostre valutazioni simulate.

Si tratta di un problema di *interferenza intercella*, che, essendo il nostro modello a singola base station, non è stato possibile introdurre nelle simulazioni.

La tecnica di ripartizione dinamica dei time slot comporta difatti un aumento dell'*interferenza intercella* rispetto alla soluzione statica.

Vediamo a cosa è dovuto questo fenomeno.

---

Il problema dell'interferenza intercella risiede nel fatto che un sistema cellulare a divisione di codice quale quello UMTS è tipicamente caratterizzato da un fattore di riuso frequenziale pari ad 1. Ciò equivale a dire che in ogni cella si utilizzano tutte le risorse radio a disposizione e quindi tutti i codici e tutta la banda. Così facendo è evidente che se da un lato si aumenta la capacità del sistema e si evitano problemi di pianificazione delle frequenze (come la clusterizzazione del GSM), dall'altro si generano inevitabili problemi di interferenza intercella. Questi problemi sono risolti soltanto parzialmente dal processo di *scrambling* (vedi par II.5.3.2). Infatti, nonostante l'utilizzo dei codici di scrambling diversi, che tra l'altro sono in numero finito (128), non è possibile rendere perfettamente ortogonali segnali generati in celle diverse.

In up link, se due utenti mobili appartenenti a celle adiacenti trasmettono simultaneamente con lo stesso codice di spreading, i due segnali trasmessi giungono alle due stazioni base sovrapposti e, nonostante l'utilizzo di due codici di scrambling differenti, generano interferenza reciproca. Analogamente in down link si genera interferenza intercella quando due stazioni base trasmettono allo stesso tempo verso due utenti delle rispettive celle utilizzando lo stesso codice di spreading.

Questo è un primo tipo di interferenza intercella ed è il meno grave. Infatti se la base station è situata, come è ragionevole che sia, al centro della cella, nei due casi precedenti la potenza del segnale interferente è sempre minore di quella del segnale utile, in quanto necessariamente il primo deve aver seguito un cammino di propagazione più lungo (si ipotizza una condizione ideale di controllo di potenza). Il caso limite si ha quando l'utente è situato al margine della cella ed è quindi potenzialmente equidistante dalle due base station.

Il secondo tipo di interferenza si genera quando si verifica la possibilità che due utenti di celle adiacenti siano contemporaneamente in stati diversi, cioè uno in trasmissione e l'altro in ricezione. In questo caso infatti se entrambi utilizzano lo stesso codice, il segnale trasmesso dal primo utente è ricevuto dal secondo sovrapposto a quello utile generato dalla base station. In questo caso la condizione precedente viene a cadere, in quanto gli utenti

---

possono essere entrambi al confine delle rispettive celle e quindi molto vicini. Teoricamente il segnale interferente potrebbe avere una potenza molto maggiore di quello utile.

Questo secondo tipo di interferenza però è presente soltanto nella soluzione con ripartizione dinamica dei time slot.

Adottando una soluzione di ripartizione con singolo punto di switching, infatti, è possibile considerare due gruppi di time slot consecutivi, il primo dedicato al down link e il secondo all'up link (è consuetudine anteporre il down link all'up link).

E' ragionevole supporre che le considerazioni statistiche sul bilanciamento del traffico sul lungo periodo siano uguali per tutte le celle. Conseguentemente, adottando una soluzione di ripartizione statica, il bound tra up e down link nella trama è fissato nella medesima posizione per tutte le celle. Ipotizzando una sincronizzazione delle trame di diverse base station, comunque necessaria per altri tipi di problemi, risulta che i periodi di up e down link sono temporalmente coincidenti per tutte le celle. Non si pone quindi in questo caso la possibilità che due celle adiacenti siano allo stesso tempo rispettivamente in stato di up e down link.

Al contrario la soluzione dinamica sposta il bound tra up e down link ad ogni frame in base alla valutazione del numero di utenti attivi nelle due tratte. Variando quest'ultimo da cella a cella in modo indipendente, è inevitabile che tale bound potrà trovarsi in posizione diversa in due celle adiacenti con conseguente generazione di interferenza intercella del secondo tipo.

La scelta della soluzione dinamica rispetto a quella statica non appare quindi scontata. Alla luce di questa considerazione acquistano quindi valore i risultati presentati nel capitolo 5. Si rende necessario studiare i vantaggi conseguibili con la soluzione dinamica in termini di throughput e ritardi di trasmissione, in modo da valutare se essi giustifichino il degrado delle prestazioni del sistema a causa del relativo aumento di interferenza.

---

### III.3 Algoritmi di scheduling adattivi al canale di propagazione

Una delle principali innovazioni introdotte dai sistemi radiomobili di terza generazione è la tecnica di accesso a divisione di codice.

Si ricorda che la tecnica CDMA, in cui i diversi segnali sono contraddistinti da codici diversi, idealmente fa sì che l'interferenza reciproca si annulli completamente. Nella realtà, però, le condizioni di propagazione riducono le proprietà di ortogonalità delle sequenze usate. Di conseguenza la capacità del sistema risulta *limitata dall'interferenza reciproca*.

Contrariamente ai sistemi di prima e seconda generazione non esiste una limitazione prestabilita delle risorse (numero delle portanti o numero dei time slot): all'aumentare del numero di codici utilizzato e quindi della capacità del sistema, la qualità percepita dagli utenti presenti è degradata leggermente. Il numero di codici utilizzato può essere aumentato fino a quando il livello di interferenza sarà tale da non garantire il requisito di qualità desiderato. Per questo motivo i sistemi CDMA sono caratterizzati dall'aver una degradazione della qualità graduale (*soft degradation*) all'aumentare della capacità del sistema.

La capacità del sistema è quindi *guidata* dall'interferenza.

L'interferenza complessiva del sistema risulta essere la somma dell'interferenza *intercella* e di quella *intracella*. Per ragioni di semplicità implementativa non è stato possibile tenere in conto il primo tipo di interferenza in quanto, come già osservato, il nostro modello è a singola base station.

Vediamo dunque come il problema dell'interferenza intracella impatta sull'algoritmo di scheduling di trasmissione dati tipo Internet.

Come analizzato nel capitolo 2, l'interfaccia radio TD-CDMA è caratterizzata da una matrice tempo-codice, i cui singoli elementi sono detti *Risorse Unitarie (RU)* e rappresentano un codice OVSF con SF=16 trasmesso in un particolare time slot (per chiarire meglio questi concetti si vedano i paragrafi 3 e 4 del capitolo successivo). All'interno di un singolo time slot possono essere allocate fino ad un massimo di 16 RU. All'aumentare del

---

numero di RU e quindi di codici allocati in un time slot (TS) aumenta l'interferenza intracella tra gli utenti di quel TS.

Il contesto da noi analizzato è quello di trasmissione di dati di tipo Internet, che nella terminologia UMTS sono detti UDD (*Unconstrained Data Delay*). Per la trasmissione di tali dati non sono richiesti specifici requisiti di qualità del servizio, in quanto, come dice il nome stesso, non ci sono vincoli temporali di consegna. Non esiste nemmeno un vincolo sulla BER, come invece avviene sul traffico voce.

Per tale tipo di traffico quindi non essendo fissati requisiti di QoS, non esiste un criterio dettato dalle specifiche per fissare una soglia minima di rapporto segnale-interferenza e conseguentemente scegliere il numero di RU per TS in modo da mantenersi al di sopra di tale soglia.

Per calcolare il numero di RU da allocare in un singolo TS, l'ipotesi fortemente semplificativa, adottata in alcuni lavori analizzati e, in primo approccio, anche in questa sede, è quella di considerare un andamento a gradino dell'interferenza in funzione delle RU allocate. Si suppone che vi sia una soglia (solitamente posta a 8 RU) al di sotto della quale l'interferenza intracella è trascurabile e al di sopra della quale si sperimenta un'interferenza intollerabile, per cui la trasmissione è ritenuta inefficiente. Questa semplificazione porta banalmente a fissare ad 8 il numero di RU per TS. Ai nostri occhi questa semplificazione è risultata troppo drastica, non fosse altro per il fatto che dimezza a priori la capacità del canale.

Si è quindi cercato di migliorare le prestazioni del sistema introducendo un algoritmo di scheduling con un numero di RU per TS non fissato a priori, ma variabile da frame a frame in funzione delle condizioni di interferenza sperimentate

Il nostro problema è quindi trovare un criterio per fissare il numero di RU per TS. Tale criterio è stato scelto nella *massimizzazione del throughput*.

All'aumentare del numero di RU per TS aumenta l'interferenza intracella e conseguentemente il BER per RU. Lavori sull'andamento dell'interferenza intracella all'aumentare del numero di codici utilizzati sono per la maggior parte ancora in corso

---

d'opera. L'analisi di questa funzione è realmente complessa e gli studi in merito non sono ancora consolidati, per cui, per il momento, è stato impossibile tenerne conto nel nostro lavoro.

Per dare un'idea della complessità del problema basti accennare che esistono lavori che dimostrano la non equivalenza dei codici di spreading dal punto di vista interferenziale: presi due codici a caso l'interferenza reciproca non è costante, ma varia al variare dei codici. Le implicazioni di questa considerazione sull'algoritmo di scheduling sono evidenti, ma l'instabilità e l'incompletezza di tali conclusioni non ne rendono ancora possibile la loro utilizzazione in algoritmi di scheduling.

In assenza di lavori utili, l'unica informazione certa, peraltro lapalissiana, che abbiamo sulla funzione di probabilità di errore dovuta ad interferenza al variare del numero delle RU per TS è che si tratta di una funzione monotona non decrescente.

Definiamo a questo punto la funzione da massimizzare. Si tratta del throughput medio per time slot, definito come numero di RU trasmesse correttamente in un TS. Tale valore è chiaramente funzione della probabilità di errore e quindi, attraverso quest'ultima, del numero di RU per TS (d'ora in poi  $N$ ):

$$T(N) = E\{\text{RU trasmesse correttamente in un TS}\} = [1 - p_e(N)] * N$$

Esiste quindi un valore di  $N$  ottimo ( $N_{\text{opt}}$ ) in grado di massimizzare il valore del throughput  $T(N)$ :

$$N_{\text{opt}} = \arg \max T(N) = \arg \max [1 - p_e(N)] * N$$

Di seguito è riportato l'andamento del throughput corrispondente ad un particolare andamento della probabilità di errore (funzione a sigmoide).

---

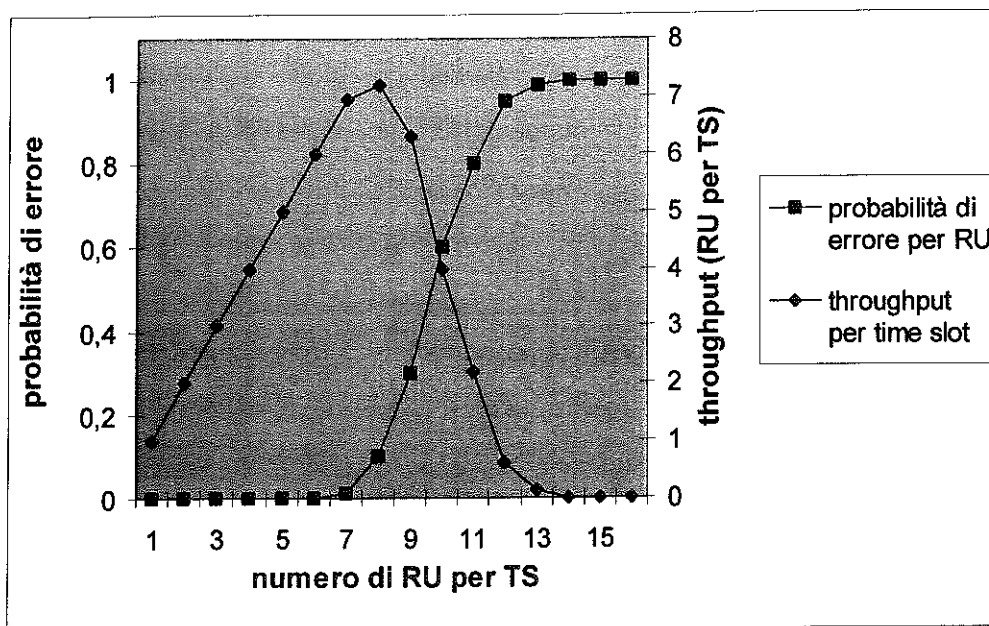


Figura 1. Esempio di andamento del throughput in funzione del numero di RU per TS

Si osserva che all'aumentare di  $N$  aumenta il numero di RU trasmesse, ma parallelamente aumenta anche la probabilità di errore. Il numero ottimo di RU per TS ( $N_{opt}$ ) deriva quindi da un compromesso tra queste due tendenze.

L'algoritmo di scheduling per risultare efficiente deve allocare un numero di RU per TS pari ad  $N_{opt}$ , che come visto dipende strettamente dalla funzione di interferenza. L'andamento della probabilità di errore in funzione del numero di RU allocate non è prevedibile né teoricamente, né empiricamente, perché dipende da un numero di parametri troppo elevato strettamente legati alle condizioni del canale di propagazione. Inoltre quest'ultime cambiano da utente ad utente e, a causa della mobilità di quest'ultimi, variano nel tempo rendendo dinamica la suddetta funzione e facendo conseguentemente variare il valore di  $N_{opt}$ .

Da quanto esposto finora dovrebbe risultare evidente che l'unica soluzione plausibile in questo contesto è quella di implementare un *algoritmo adattivo* che sia in grado di "seguire" l'andamento dell'interferenza nel tempo variando opportunamente il numero di RU allocate.

Prima di passare a considerare l'algoritmo da noi proposto facciamo alcune considerazioni preliminari.

Innanzitutto bisogna chiarire che quella finora tenuta in conto è solo una delle componenti della probabilità di errore. Tale componente è dovuta all'interferenza tra codici allocati nello stesso TS e può quindi considerarsi a buon diritto una probabilità di errore costante per tutti gli utenti, in quanto si suppone di allocare lo stesso numero di RU per ogni TS. Ipotizzando che i codici siano tutti equivalenti tra loro, ogni codice associato ad una RU sperimenta lo stesso rapporto segnale-interferenza di tutti gli altri.

In realtà vi è una complicazione di cui non si è tenuto conto. L'ipotesi di equivalenza dei codici dal punto di vista dell'interferenza consiste nel presupporre che allocando un qualsiasi codice insieme ad altri  $N$  codici in un TS l'interferenza intracella sperimentata sia sempre la stessa e dipenda esclusivamente da  $N$ . Questa ipotesi però è sicuramente contraddetta nel caso in cui più RU di uno stesso TS appartengano allo stesso utente. In tal caso infatti, supponendo una trasmissione in up link, i codici trasmessi giungono alla stazione base sicuramente sincronizzati per cui per loro il rapporto segnale-interferenza è sicuramente più alto. È chiaro che l'algoritmo di scheduling in up link deve cercare di raggruppare i codici di uno stesso utente all'interno di uno stesso slot, in quanto in questo modo, grazie alla sincronizzazione, è possibile ridurre l'effetto dell'interferenza intracella. In down link il problema è più semplice in quanto i codici giungono all'utente mobile dalla BS attraverso un unico cammino e quindi tutti sincronizzati. In questo caso il problema di distinguere TS con RU appartenenti allo stesso utente non si pone.

La componente di errore di cui finora non si è tenuto conto è quella relativa al singolo utente. Tale componente dipende esclusivamente dalle condizioni del canale di propagazione visto dal singolo utente e non dall'algoritmo di scheduling, almeno in un contesto di fading lento. (In un contesto di fading veloce, con utenti mobili che si spostano a velocità molto elevate, in cui le condizioni di propagazione del canale variano in un tempo paragonabile a quello di un TS (10 ms) ha senso anche introdurre un algoritmo che tenga conto di questa seconda componente della probabilità di errore, in quanto tale probabilità di

---

errore varia al variare del TS assegnato nella trama con un andamento variabile da utente a utente). Tale componente di errore, dipendendo solo dal rapporto segnale-rumore, non è influenzata direttamente dall'algoritmo di scheduling, ma influenza l'algoritmo stesso. Infatti le due cause di errore (interferenza e propagazione) sono separabili solo da un punto di vista teorico. La probabilità di errore misurata dalla stazione base è la somma delle due componenti prima esposte ed è evidentemente impossibile risalire all'errore dovuto alla sola interferenza tra codici. Un algoritmo veramente equo dovrebbe cercare di equilibrare la probabilità di errore tra gli utenti agendo sull'unica componente influenzabile, cioè l'interferenza. Per questo motivo utenti in condizioni critiche di propagazione dovrebbero essere allocati in slot con poche RU, in modo da compensare il basso rapporto segnale-rumore sul canale con un elevato rapporto segnale-interferenza.

In questo lavoro, la probabilità di errore specifica di ogni singolo utente non è stata tenuta in considerazione, o, per meglio dire, si è considerata una probabilità di errore di propagazione anch'essa uniforme su tutti gli utenti. Questa assunzione è in parte giustificabile dal fatto che il sistema UMTS prevede di per se una suddivisione in classi di utenti con le stesse caratteristiche di propagazione (perlomeno la stessa velocità di spostamento) attraverso l'interfrequency handover dell'HCS (vedasi a tal proposito il paragrafo I.5.5). Inoltre l'algoritmo proposto è stato pensato per essere applicato in un caso molto più generale e realistico. Esso resta valido, infatti, anche se si considera la presenza di classi di utenti con caratteristiche propagative differenti e quindi probabilità di errore dovute al canale diverse da utente a utente. Una preventiva divisione in classi degli utenti con la stessa probabilità di errore rende applicabile l'algoritmo di scheduling proposto ad ogni singola classe. In questo modo si può conseguire non solo l'obiettivo di massimizzare il throughput per ogni classe, ma anche quello precedentemente esposto di allocare utenti in condizioni critiche di propagazione in time slot con un basso numero di RU, in modo da bilanciare il basso rapporto segnale-rumore con un più alto rapporto segnale-interferenza.

Andiamo ora ad illustrare l'algoritmo proposto.

---

Osserviamo subito che lo scheduler, accentrato nella base station, effettua l'allocazione ad ogni frame ed è quindi in grado di cambiare il numero di RU per TS ogni 10 ms. Esso non conosce con esattezza la curva della probabilità di errore al variare di  $N$ . Ad ogni trama l'unica informazione utile di cui lo scheduler dispone per calcolare il valore di  $N$  da utilizzare è la probabilità di errore misurata nei frame precedenti. La stazione base calcola ad ogni frame la probabilità di errore sperimentata dagli utenti, come rapporto tra le RU ricevute correttamente e quelle complessivamente trasmesse. Questo è possibile sia per l'up link che per il down link.

Per l'up link il calcolo è immediato, in quanto le RU trasmesse giungono direttamente alla stazione base che ne verifica la correttezza. In down link si sfrutta invece il meccanismo di feedback della ritrasmissione. In ipotesi di modalità di trasmissione con *acknowledgment* l'utente mobile che riceve dei pacchetti errati lo comunica alla stazione base tramite delle richieste di ritrasmissione.

Il calcolo della probabilità di errore attraverso la misura delle RU errate, effettuato in maniera puntuale e senza mediazioni temporali potrebbe destare delle perplessità. Questa stima infatti potrebbe essere poco significativa in virtù delle fluttuazioni statistiche dovute al fatto che le RU errate sono realizzazioni di una decisione aleatoria. E' il caso quindi di osservare esplicitamente che l'algoritmo assume significato per alti carichi della rete, per cui il numero di RU trasmesse è sempre molto alto (attorno al centinaio), conseguentemente il numero di realizzazioni della decisione aleatoria che si va a considerare è elevato e, in virtù del teorema del limite centrale, la stima può essere ritenuta affidabile.

Intuitivamente lo scheduler deve aumentare  $N$  quando il numero di RU errate è basso e diminuirlo quando questo diventa troppo alto.

Premettiamo che l'algoritmo implementato è identico per le due tratte, ma viene applicato separatamente per up e down link in quanto la probabilità di errore può essere diversa nei due casi.

A prescindere dalla direzione di traffico, l'algoritmo per il calcolo di  $N$  al frame  $n+1$ -esimo è il seguente:

---

---

```

If  $RU\_ERR_{n+1}(N_{n+1}=N_n) - RU\_ERR_{n+1}(N_{n+1}=N_n-1) > TS_{n+1}$ 
then  $N_{n+1}=N_n-1$ 
else if  $\sum_{i=m}^{n+1} TS_i > RU\_ERR_{n+1}(N_{n+1}=N_n+1) - RU\_ERR_{n+1}(N_{n+1}=N_n)$ 
then  $N_{n+1}=N_n+1$ 
else  $N_{n+1}=N_n$ 

```

Dove il pedice indica il frame che stiamo considerando,  $RU\_ERR_{n+1}(N_{n+1}=N_n)$ ,  $RU\_ERR_{n+1}(N_{n+1}=N_n-1)$  e  $RU\_ERR_{n+1}(N_{n+1}=N_n+1)$  rappresentano il numero previsto di RU ricevute erratamente nel frame  $n+1$ , considerando di utilizzare per tale frame un numero di RU per TS ( $N_{n+1}$ ) rispettivamente pari al numero utilizzato nel frame precedente e a tale numero diminuito o aumentato di una unità. L'indice di partenza della sommatoria  $m$  rappresenta l'ultimo frame in cui si è verificato un cambiamento di  $N$ .

Ad ogni frame si presentano tre possibilità: mantenere costante, aumentare o diminuire di una unità il numero  $N$  rispetto al frame precedente. L'algoritmo si basa sull'ipotesi di poter stimare il numero di RU ricevute erratamente nei tre casi.

Diminuendo di una unità  $N$  si ha una perdita certa di capacità in termini di RU pari al numero di TS. Se il conseguente miglioramento stimato in termini di probabilità di errore è tale da compensare la perdita certa (cioè si risparmiano un numero di RU errate maggiore del numero di TS), allora si effettua il decremento. Viceversa, se il *guadagno stimato* non supera la *perdita certa* non ha senso effettuare un tentativo in discesa e si mantiene costante  $N$  rispetto al frame precedente.

Per ogni frame in cui viene mantenuto costante  $N$  si perde un potenziale aumento del numero di RU pari al numero di TS, ottenibile con un aumento di  $N$ . Ogni qual volta viene mantenuto costante il numero  $N$  si incrementa la *perdita certa accumulata* del numero di TS utilizzati per quel frame. Quando tale perdita certa accumulata supera l'aumento di RU errate stimato per l'incremento di  $N$  di una unità, allora si effettua un tentativo in salita.

---

Vediamo ora come si ottengono le stime delle RU errate utilizzate nelle condizioni di salita e di discesa.

In un primo momento si è deciso di implementare un algoritmo semplificato *senza memoria*. L'unica informazione utilizzata è il numero di RU errate trasmesse nel frame precedente. Supponendo che la curva della probabilità di errore non cambi è possibile stimare il numero di RU errate qualora si mantenga costante N nella maniera seguente:

$$RU\_ERR_{n+1} (N_{n+1}=N_n) = RU\_ERR_n * \frac{TS_{n+1}}{TS_n}$$

Dove  $RU\_ERR_n$  rappresenta il numero di RU errate nel frame precedente.

Essendo l'algoritmo senza memoria, non si dispone di altre misure oltre  $RU\_ERR_n$ . Per calcolare  $RU\_ERR_{n+1} (N_{n+1}=N_n-1)$  e  $RU\_ERR_{n+1} (N_{n+1}=N_n+1)$  si considera un *caso limite*.

Supponiamo di trovarci in presenza di un andamento della probabilità di errore in funzione di N perfettamente a gradino. La probabilità di errore è nulla fino alla discontinuità del gradino e unitaria al di sopra.

Per la condizione di discesa il caso limite è quello di trovarsi esattamente al di sopra della discontinuità del gradino. In tal caso una diminuzione di N comporterebbe un recupero di tutte le RU errate, per cui consideriamo:

$$RU\_ERR_{n+1} (N_{n+1}=N_n-1) = 0.$$

In questo *best case* ci si trova a considerare la stima del massimo guadagno conseguibile in termini di recupero delle RU errate.

Viceversa nella condizione di salita il caso limite è trovarsi esattamente al di sotto della discontinuità del gradino. Un conseguente aumento di N fa passare da una situazione di errore nullo ad una perdita di tutte le RU trasmesse, per cui la stima della perdita diviene:

---

$$RU\_ERR_{n+1}(N_{n+1}=N_n+1) - RU\_ERR_{n+1}(N_{n+1}=N_n) = TS_{n+1} * N_n$$

Questa volta si considera quindi un *worst case* e si sostituisce la stima con la massima perdita conseguibile.

Riassumendo le assunzioni fatte l'algoritmo si semplifica nel seguente modo:

```

If  $RU\_ERR_n > TS_n$ 
then  $N_{n+1}=N_n-1$ 
else if  $\sum_{i=m}^{n+1} TS_i > N_n * TS_{n+1}$ 
    then  $N_{n+1}=N_n+1$ 
    else  $N_{n+1}=N_n$ 

```

Si è successivamente affinato questo primo algoritmo semplificato introducendo uno stimatore dell'andamento della probabilità di errore in funzione di N che tenesse in memoria le misure della probabilità di errore effettuate in più frame precedenti.

Consideriamo un vettore di 16 elementi, pari ai possibili valori di N. In ciascuno dei campi di questo vettore è memorizzato il valore della probabilità di errore relativo all'utilizzo di un numero N di RU per TS. Questo vettore viene aggiornato ad ogni frame secondo la seguente legge:

$$p_e^{n+1}[i] = \begin{cases} \alpha * p_e^n[i] + (1-\alpha) * 0 & \text{if } i < N_n \\ \frac{RU\_ERR_n}{RU\_TRAS_n} & \text{if } i = N_n \\ \alpha * p_e^n[i] + (1-\alpha) * 1 & \text{if } i > N_n \end{cases}$$

Il campo corrispondente al valore di N utilizzato nell'ultimo frame viene aggiornato con la misura esatta, mentre tutti gli altri campi conservano la memoria dell'ultima misura effettuata per il rispettivo valore di N. La memoria di tale misura decade nel tempo

esponenzialmente attraverso il valore del *fattore di smoothing*  $\alpha$ . All'aumentare di  $\alpha$  aumenta il tempo di memorizzazione dell'ultima misura. L'andamento temporale delle stime è costituito da una distribuzione geometrica di ragione  $\alpha$ , il cui valor medio è  $\frac{1}{1-\alpha}$ .

Poiché l'aggiornamento viene fatto ad ogni frame e poiché la distribuzione geometrica decade rapidamente dopo la sua media, un numero di  $\frac{1}{1-\alpha}$  frame può essere considerato con buona approssimazione un indicatore del tempo di memoria dello stimatore.

Si noti come la legge di aggiornamento del vettore sia differente a seconda che ci si trovi al di sopra o al di sotto del campo relativo all'ultimo valore misurato. Questa scelta è giustificata dalla seguente considerazione. Supponiamo di mantenerci sullo stesso valore di  $N$  per un tempo superiore a  $\frac{1}{1-\alpha}$  frame. Dopo tale tempo la memoria del vettore è decaduta e non si hanno più informazioni utili sulla probabilità di errore per valori di  $N$  diversi da quello considerato. In questo caso si ricade nella situazione dell'algoritmo semplificato senza memoria. Per fare in modo che i due algoritmi vengano a coincidere abbiamo differenziato le leggi di aggiornamento in modo che, con la perdita di memoria, i campi del vettore al di sotto dell'ultimo valore di  $N$  tendano a 0, mentre quelli al di sopra tendano ad 1. Così facendo, nel caso limite di totale perdita di memoria i due algoritmi vengono a coincidere.

Vediamo come si trasforma l'algoritmo attraverso l'utilizzo del vettore di memorizzazione della probabilità di errore. L'unico cambiamento è la sostituzione delle tre stime del numero di RU errate nei tre casi di aumento, diminuzione e mantenimento del valore di  $N$ . La legge con cui si calcolano le tre stime è semplicemente la seguente:

$$RU\_ERR_{n+1}(N_{n+1}=N_n \pm 1) = p_e^{n+1} [N_n \pm 1] * (N_n \pm 1) * TS_{n+1}$$


---

Normalizzando le condizioni di salita e discesa di  $N$  rispetto al numero di TS al frame  $n+1$ -esimo si ottiene:

```

if  $p_e^{n+1}[N_n] * N_n - p_e^{n+1}[N_n - 1] * (N_n - 1) > 1$ 
then  $N_{n+1} = N_n - 1$ 
else if  $\frac{\sum_{i=m}^{n+1} TS_i}{TS_{n+1}} > p_e^{n+1}[N_n + 1] * (N_n + 1) - p_e^{n+1}[N_n] * N_n$ 
then  $N_{n+1} = N_n + 1$ 
else  $N_{n+1} = N_n$ 

```

Lo scopo dell'introduzione del fattore di smoothing è quello di rendere efficiente l'algoritmo anche nel caso di variazione nel tempo della curva della probabilità di errore. Introducendo tale parametro si ha un grado di libertà in più che ci permette di gestire l'algoritmo in presenza di diverse velocità di cambiamento della probabilità di errore.

Per problemi che saranno meglio chiariti in seguito, si è infine deciso di adottare un terzo algoritmo di scelta del numero di RU per TS. Per questo terzo algoritmo si è deciso di cambiare la filosofia euristica adottata nella scelta della *condizione in salita*. Abbiamo visto che nell'algoritmo appena presentato la condizione di salita è effettuata attraverso una variabile di accumulo della perdita certa. Tale scelta deriva da considerazioni euristiche sulle stime della probabilità di errore. Se disponessimo dei valori esatti di tutti i campi del vettore stima della probabilità di errore non sarebbe più necessario utilizzare la variabile di accumulo, in quanto si utilizzerebbe una condizione analoga a quella per la discesa. In quest'ottica si muove il terzo algoritmo considerato. Esso presenta la seguente struttura:

```

If  $RU\_ERR_{n+1}(N_{n+1}=N_n) - RU\_ERR_{n+1}(N_{n+1}=N_n-1) > TS_{n+1}$ 
then  $N_{n+1}=N_n-1$ 
else if  $TS_{n+1} > RU\_ERR_{n+1}(N_{n+1}=N_n+1) - RU\_ERR_{n+1}(N_{n+1}=N_n)$ 
    then  $N_{n+1}=N_n+1$ 
    else  $N_{n+1}=N_n$ 

```

Sostituendo i valori delle stime espressi tramite il vettore delle probabilità di errore e normalizzando rispetto al numero di TS al frame n+1-esimo si ottiene:

```

If  $p_e^{n+1}[N_n] * N_n - p_e^{n+1}[N_n - 1] * (N_n-1) > 1$ 
then  $N_{n+1}=N_n-1$ 
else if  $p_e^{n+1}[N_n + 1] * (N_n+1) - p_e^{n+1}[N_n] * N_n < 1$ 
    then  $N_{n+1}=N_n+1$ 
    else  $N_{n+1}=N_n$ 

```

Anche la regola di aggiornamento del vettore delle stime della probabilità di errore è stata modificata. Per ottenere delle stime quanto più vicine al valore misurato si è eliminato il *fattore di correzione*  $1-\alpha$ . Si è però dovuto tenere in conto che la probabilità di errore in funzione delle RU per TS ha un andamento monotono non decrescente, per cui i campi del vettore stima della probabilità superiori a quello contenente l'ultima misura sono aggiornati con un valore non inferiore a quest'ultima.

---

$$p_e^{n+1}[i] = \begin{cases} \alpha * p_e^n[i] & \text{if } i < N_n \\ \frac{RU\_ERR_n}{RU\_TRAS_n} & \text{if } i = N_n \\ \max(\alpha * p_e^n[i], \frac{RU\_ERR_n}{RU\_TRASM_n}) & \text{if } i > N_n \end{cases}$$

Si noti come quest'ultimo algoritmo sia perfettamente equivalente al precedente per quanto riguarda la *condizione di discesa*, mentre adotti un criterio differente per quella di salita che prescinde dalla logica dell'accumulo della perdita certa del precedente algoritmo.

---

---

# CAPITOLO

# 4

---

## ARCHITETTURA DEL SIMULATORE

*Di seguito viene presentata l'architettura del simulatore della rete di accesso UTRAN realizzato per valutare le prestazioni degli algoritmi di scheduling proposti nel capitolo precedente per la trasmissione dati Internet nel sistema UMTS.*

### **IV.1 Finalità del simulatore**

Lo studio svolto si è occupato della verifica delle prestazioni ottenibili, principalmente in termini di throughput e di ritardo nella trasmissione di un pacchetto IP, utilizzando l'interfaccia radio UMTS nella versione TDD per la fornitura di accesso a servizi dati in modalità *best-effort*. Nella terminologia ETSI tali servizi vengono detti di tipo UDD (*Unconstrained Delay Data*).

Il punto di partenza sono i documenti prodotti dal 3GPP, grazie ai quali è stato fatto un

quadro dell'attuale stato di avanzamento nella produzione delle specifiche tecniche riguardanti l'interfaccia radio UMTS. Da questi si è partiti per la definizione di un modello di simulazione della rete di accesso radio UMTS (UTRAN). I numerosi punti ancora non specificati nelle normative (soprattutto nel caso del MAC) e quelli lasciati liberi all'implementazione sono stati definiti introducendo ipotesi di funzionamento il più semplici possibili, senza però compromettere l'efficienza degli algoritmi proposti. Tutte le ipotesi comunque sono state introdotte nel completo rispetto di quanto riportato nei documenti del 3GPP.

L'obiettivo principale è stato quello di individuare un particolare algoritmo di scheduling per la trasmissione dati di tipo best-effort che si adattasse il più efficientemente possibile al contesto studiato.

Attualmente non si trova nei documenti del 3GPP una descrizione univoca della capacità o della tipologia di canali assegnati a questo tipo di trasmissione, né tanto meno dell'algoritmo eseguito per l'assegnazione degli stessi. Sono state pertanto avanzate delle ipotesi su tali argomenti, ipotesi che, come è facile intuire, influenzano non poco il comportamento e le prestazioni dell'interfaccia radio.

## **IV.2 Ambiente di simulazione**

Per implementare il modello di simulazione è stato utilizzato il *RESearch Queueing package* (RESQ). RESQ è un linguaggio di simulazione sviluppato dalla *IBM Research* per definire e risolvere modelli di reti di code estese.

Un modello RESQ è costituito da un insieme di code, una popolazione di *job*, un insieme di nodi e regole per descrivere come i *job* circolano tra i nodi e le code.

Sono previsti due tipi di coda: la coda attiva e la coda passiva. La coda attiva ha un insieme di server che forniscono servizio ai *job*. Un *job* che arriva ad una coda attiva attende di essere selezionato, in base alla disciplina prevista, per ricevere il servizio. Dopo essere stato servito il *job* lascia la coda. Una coda attiva può essere usata, ad esempio, per modellare un canale di comunicazione o più in generale tutte le situazioni in cui c'è

---

competizione di risorse. Una coda passiva considera una nozione più generale di utilizzo di una risorsa e permette di modellare fenomeni come la sincronizzazione tra eventi ed il possesso simultaneo di una risorsa. Una coda passiva è costituita da un *pool* di *token* da allocare, ciascuno dei quali rappresenta una singola unità di risorsa. Inoltre una coda passiva è costituita da un insieme di nodi attraverso i quali un *job* può acquisire, creare, rilasciare o distruggere i *token*.

### IV.3 Canali per la trasmissione dati

Nell'era della multimedialità, è presumibile che i servizi a pacchetto ricoprano un ruolo sempre più importante. Di conseguenza sono state specificate numerose modalità per la trasmissione di tali servizi. A seconda del tipo di traffico da veicolare e cioè delle dimensioni e della frequenza dei pacchetti da trasmettere sono state definite tre modalità di trasmissione.

Nel caso di trasmissione di pacchetti brevi ed infrequenti, è previsto di trasmettere i dati su *canali di segnalazione comuni* (PCH, FACH, e RACH), in modo analogo a quanto avviene con il GSM per il servizio SMS.

Nel caso di pacchetti infrequenti le cui dimensioni possano richiedere una velocità di trasmissione elevata, si ricorre al *canale condiviso in down link* (DSCH) e al *canale a contesa in up link* (USCH).

La terza modalità prevista per la trasmissione di servizi a pacchetto consiste nell'instaurare un *canale dedicato* (DCH).

Nel nostro lavoro è stata esclusa a priori la possibilità di trasmissione di dati a pacchetto sul RACH. Infatti per come è stata definita ad oggi la tipologia del burst di accesso sul RACH, essa non si presta alla trasmissione dati del tipo analizzato essendo troppo esiguo lo spazio del burst in confronto alle dimensioni di un pacchetto IP ed in considerazione anche del fatto che il RACH presenta rischio di collisione. Per quanto riguarda la trasmissione su canali condivisi, la sua definizione ancora in corso di elaborazione non ne ha reso possibile l'analisi.

---

La trasmissione dati avviene dunque esclusivamente sui canali di trasporto dedicati, DCH (*Dedicated Channel*). Si è supposto poi di utilizzare per i DCH una tipologia di burst di tipo 1 (quella con il midambolo più esteso), esclusivamente per ragioni di semplicità, visto che esso può essere usato indifferentemente sia in up link che in down link.

Per semplificare la trattazione introduciamo il concetto di *Risorsa Unitaria* (d'ora in poi RU), definita come la minima quantità di dati che è possibile trasmettere in un burst. Abbiamo già visto che l'informazione trasportata in un burst dipende strettamente dal fattore di spreading utilizzato (aumenta al diminuire dello SF), per cui ad una RU è associato il più alto SF disponibile, cioè 16.

Abbiamo visto che nella modalità TDD il chip rate è fissato e si hanno 2560 chip per ogni burst. In un burst di tipo 1, eliminati il midambolo e il periodo di guardia, restano 1952 chip per la trasmissione dei dati a cui equivalgono, fissato lo SF a 16, 122 simboli utili.

Ciascuna risorsa unitaria può trasportare quindi 122 simboli utili, che corrispondono ad un numero di bit utili pari a 244, dato che la modulazione è di tipo QPSK. È chiaro che questi 244 bit utili non sono tutti bit di utente in quanto vanno tenuti in conto sia il codice di protezione sul canale, sia gli overhead introdotti dagli strati MAC e RLC.

In questa sede non si è ritenuto opportuno approfondire le tecniche di codifica proposte negli standard e si è assunto l'utilizzo di un codice di protezione di tipo FEC (*Forward Error Correction*), come indicato nelle specifiche per i servizi Non Real Time, con una ridondanza di tipo "2" ad "1", che prevede quindi che per ogni bit di informazione se ne trasmettano due sul canale.

Poiché nel nostro studio la trasmissione dati avviene solo su canali dedicati DCH ed ogni canale è assegnato solo per la trasmissione di una RLC SDU, non si presenta il problema della moltiplicazione. Pertanto, come riportato dalle specifiche del 3GPP, non è necessaria alcuna intestazione di protocollo MAC.

Per quanto riguarda invece lo strato RLC, delle tre modalità di trasmissione previste per tale protocollo, in questo lavoro è stato preso in considerazione esclusivamente l'

---

“*acknowledged mode*”. Esso, come indicato nel paragrafo II.7.1, prevede la ritrasmissione delle RLC PDU che giungono in ricezione affette da errori.

Delle sei tipologie di RLC PDU descritte nella prima parte risulta di nostro interesse soltanto la AMD PDU, che determina il formato per la trasmissione dati (è stata infatti trascurata la fase di instaurazione e rilascio della connessione tra due entità RLC remote).

Come indicato nel secondo capitolo, per l’instestazione di una AMD PDU sono previsti 24 bit. Da ciò ne segue che, dato che il numero di bit che viaggiano in una RU è pari a 244, la metà di essi è costituito da codice di protezione e data l’assenza di intestazione del protocollo MAC, il campo dati è costituito da 98 bit (in effetti tale valore è stato poi scelto pari a 96 per fornire così un numero intero di 12 byte).

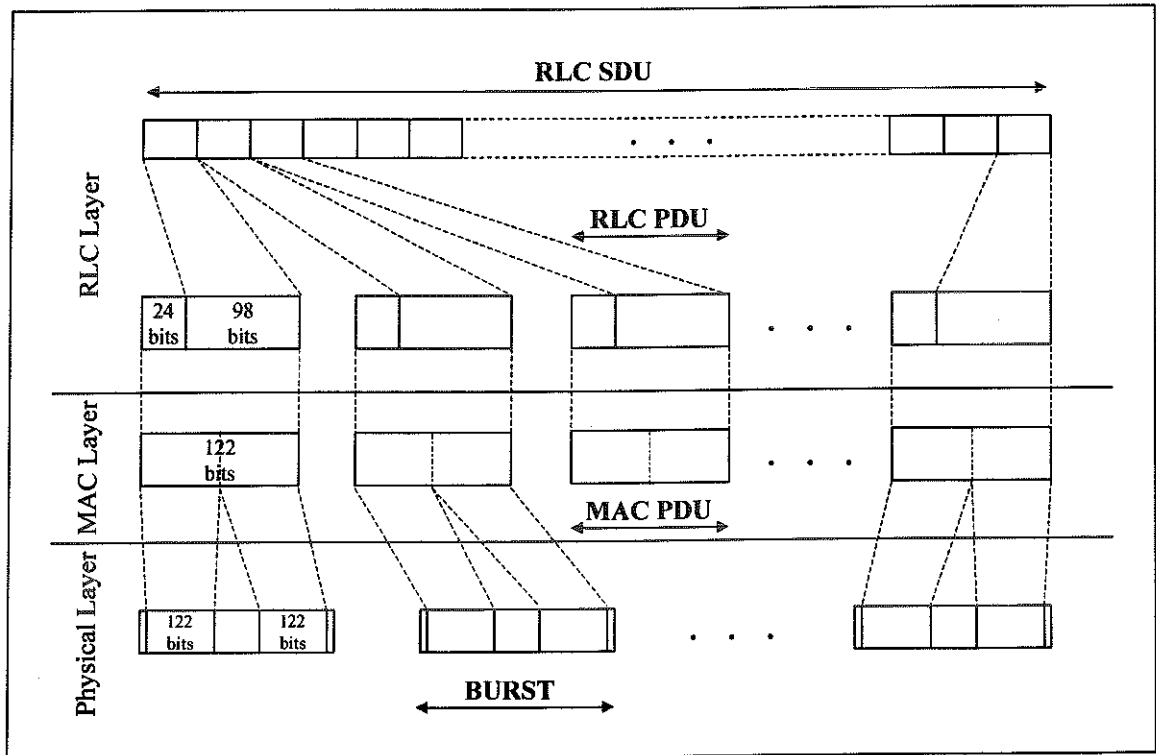


Figura 1. Passaggio dei dati tra gli strati dell'interfaccia radio

La nostra entità elementare di trasmissione è dunque la RU, equivalente a 12 byte di utente. Abbiamo però visto nel capitolo precedente che lo Spreading Factor nei sistemi UMTS è variabile, per cui bisogna giustificare la nostra assunzione di fissarlo a 16.

#### **IV.4 Trasmissione multicodice**

Per i canali DCH sono previste due modalità di trasmissione: quella “*multicode*” con SF fisso, dove una maggiore capacità è ottenuta utilizzando più codici in uno stesso time slot, ma mantenendo fisso lo SF e quella “*singlecode*” con SF variabile, dove un mobile all’interno di un time slot utilizza sempre un singolo codice, ma varia lo SF in funzione del data rate. In realtà le due soluzioni sono del tutto equivalenti in quanto un utente che vuole aumentare il suo data rate può alternativamente o diminuire lo SF o aumentare il numero di codici di trasmissione; per cui trasmettere all’interno dello stesso time slot 1, 2, 4, 8 o 16 RU (burst con SF=16) è equivalente a trasmettere un unico burst con SF uguale rispettivamente a 16, 8, 4, 2 o 1.

Utilizzando la soluzione “*multicode*” con SF=16 non si ha nessun aumento degli overhead, né rilevante aumento dell’interferenza all’interno del time slot. Dal punto di vista dell’interferenza apparentemente sembra svantaggioso per un utente utilizzare X codici diversi con SF=16, piuttosto che un unico codice con SF=16/X (X potenza di 2), perché intuitivamente all’aumentare dei codici si suppone che aumenti l’interferenza. In realtà non è così perché l’interferenza tra utenti che condividono lo stesso time slot nasce principalmente dal fatto che codici di utenti diversi non possono essere sincronizzati. Conseguentemente se un unico utente utilizza per aumentare il suo data rate più codici con SF fissato a 16, piuttosto che un unico codice con SF più basso, non altera in modo rilevante l’interferenza sul sistema, in quanto si suppone che i suoi codici siano sincronizzati. È proprio ciò che accade in down link, dove, essendo tutti i burst trasmessi in modo sincrono dalla base station, lo SF è fissato a priori dallo standard a 16. In up link codici di utenti diversi arrivano alla base station inevitabilmente privi di sincronizzazione, mentre più codici

---

trasmessi dallo stesso utente giungono attraverso lo stesso cammino e quindi sicuramente ancora sincronizzati.

Giustificata la scelta del multicode è necessaria però una precisazione. Abbiamo visto nel capitolo 2 che lo standard prevede che in up link un utente possa effettuare trasmissioni multicode utilizzando però un massimo di due canali fisici per time slot contemporaneamente, mentre nella tratta inversa è possibile trasmettere fino a 16 codici diversi. Mentre per la base station infatti è facile trasmettere su 16 codici contemporaneamente, la stessa cosa non è possibile per l'apparato mobile per evidenti problemi tecnologici. È chiaro quindi che se l'utente mobile vuole trasmettere, all'interno dello stesso time slot, una quantità di dati equivalente a più di due RU deve ineluttabilmente variare lo SF e abbassarlo a meno di 16. In questa circostanza il nostro modello con SF fissato a 16 non è più aderente alle specifiche, ma data l'equivalenza precedentemente spiegata, la nostra assunzione del multicode può ritenersi una valida semplificazione che comunque non invalida i risultati ottenuti. Si tenga presente soltanto che quando nel modello si assegna ad un utente mobile in up link più di due RU, nella realtà questo corrisponde ad assegnare un massimo di due codici con SF più bassi di 16.

## **IV.5 Struttura della trama**

Spiegata l'assunzione della trasmissione multicode, il nostro problema di assegnazione dinamica delle risorse si riduce alla suddivisione tra gli utenti attivi di una matrice tempocodice i cui elementi fondamentali sono proprio le RU. Seguendo la struttura della trama analizzata nel capitolo 2, la suddetta matrice risulta organizzata in 15 colonne, rappresentate dai time slot. Di questi 15 time slot 3 sono assegnati staticamente in ogni frame uno al RACH e 2 ai canali di controllo e segnalazione (BCH e SCH), mentre i restanti 12 restano disponibili per la trasmissione dati di utente.

---

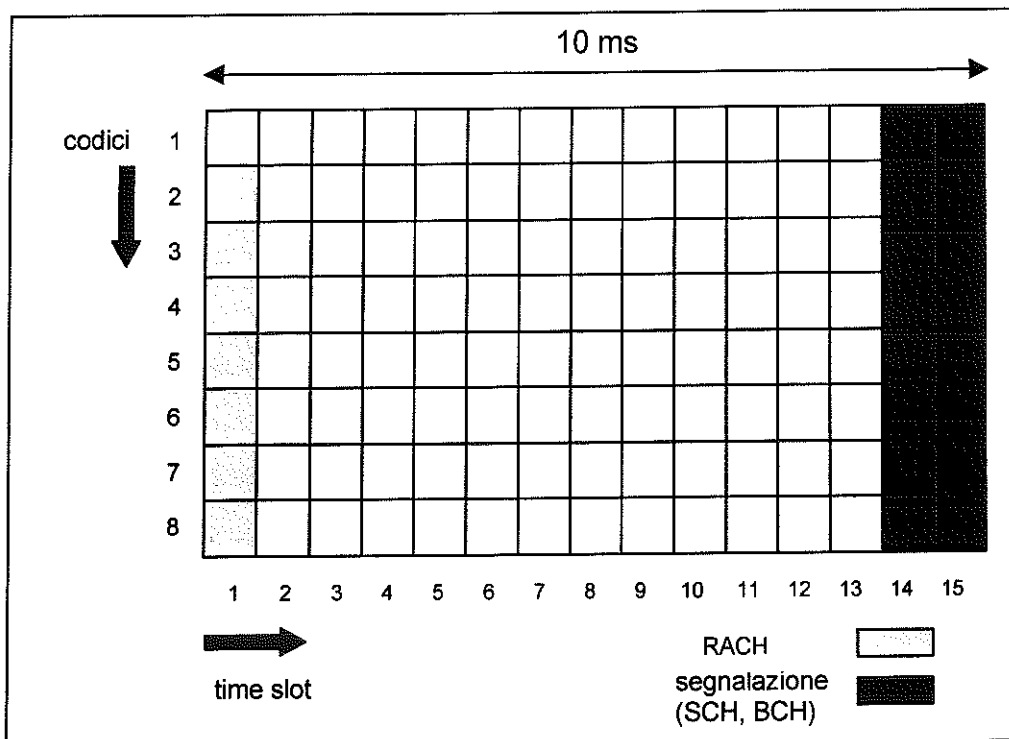


Figura 2. Modello di una trama UMTS

## IV.6 Capacità del canale

Abbiamo già visto nel capitolo precedente che, nella nostra analisi il numero di RU allocate in ogni TS non è fissato a priori. In linea teorica è possibile allocare fino ad un massimo di 16 RU per slot, il che equivale ad una capacità del canale pari a  $16 \times 12 = 192$  RU/frame, cioè  $192 \times 12 / 0.01 = 230400$  byte/s (circa 2 Mbit/s), al netto dei bit di codifica. In realtà la capacità del canale è *limitata dall'interferenza*. L'interferenza intracella sperimentata da ciascun utente in un time slot aumenta all'aumentare del numero di RU allocate nel TS stesso. Prevedere questo andamento a priori è impossibile, né si trova attualmente in letteratura uno studio accurato della BER in funzione del numero di utenti che condividono il canale. Le vicissitudini trasmissive che creano l'interferenza intracella sono molto complesse e i parametri in gioco innumerevoli. Il loro studio esula dagli scopi di questo lavoro.

È chiaro però che il numero di RU allocate in ogni TS è un elemento essenziale nell'analisi delle prestazioni dell'algoritmo di scheduling, in quanto influenza in modo

decisivo la capacità del canale. Un approccio brutale utilizzato in prima istanza è quello di pensare la funzione BER vs RU allocate con un andamento a gradino. Questo equivale a ipotizzare interferenza nulla fino ad un certo limite di RU allocate in un TS e intollerabile al di sopra. Tipicamente lavori precedenti indicano in 8 RU per TS tale limite; anche se bisogna tenere in considerazione che se più RU sono assegnate ad un unico utente o se il time slot è per traffico in down link, allora i codici possono ritenersi a buon diritto sincronizzati e le condizioni di interferenza intracella migliorano fino a poter spostare la soglia a 10 RU per time slot. In base a tali considerazioni si è deciso di fissare la *capacità nominale* del sistema in corrispondenza di 8 RU per TS, anche se, applicando il nostro algoritmo di scheduling adattivo alle condizioni di interferenza, viene modificato dinamicamente il numero di RU per TS e quindi la capacità complessiva del sistema.

#### **IV.7 Utenti e traffico generato**

Il modello di simulazione prevede una singola base station con un certo numero di utenti in up link e down link, inizialmente fissato a 5 per entrambi. In questo lavoro si è preso in considerazione il caso ideale di cella isolata perfettamente con controllo di potenza ideale (tutti i segnali sono ricevuti a pari potenza). Non sono quindi stati presi in analisi i problemi di interferenza intercella e di handover considerati nel primo capitolo. La popolazione di utenti della cella è costante anche se il numero di utenti attivi varia nel tempo in virtù del meccanismo di attivazione e rilascio del canale implementato. Ogni utente è rappresentato da un sottomodulo per cui l'aumento del numero di utenti implica semplicemente la replica dei sottomoduli. C'è da osservare che gli utenti in down link sono evidentemente fittizi, in quanto il traffico in tale direzione giunge alla base station dalla rete fissa (Core Network) ed è tutto memorizzato nei suoi buffer di trasmissione.

Ogni utente è caratterizzato da una sorgente di traffico che genera pacchetti IP con lunghezza uniformemente distribuita con probabilità  $\frac{1}{2}$  tra 40 e 60 byte e  $\frac{1}{2}$  tra 500 e 1500 byte. I pacchetti piccoli, tra 40 e 60 byte, descrivono un traffico interattivo di tipo *Telnet* oppure segnalazione di controllo a livello TCP (messaggi di conferma o richieste di

---

ritrasmissione), mentre i pacchetti più lunghi, tra 500 e 1500, rappresentano un traffico dati *bulk*, tipo *File Transfert*. Il tempo di interarrivo dei pacchetti è espresso da una distribuzione standard con valor medio funzione dell'*Offered Load* (OL) e un certo *Coefficiente di Variazione* (CV). Sia l'OL che il CV sono assegnati in ingresso all'inizio della simulazione.

### **IV.8 Offered load**

L'OL è definito come il rapporto percentuale tra il traffico complessivamente generato dagli utenti e la massima capacità del canale. Nel calcolo dell'OL il numero di RU per TS è stato considerato pari a 8, per cui la capacità nominale complessiva assunta per il canale risulta essere  $12 \times 8$  RU/frame, cioè 9.6 RU/ms. Considerato che la lunghezza media dei pacchetti è di  $0.5 \times 50 + 0.5 \times 1000 = 525$  byte, cioè circa 44 RU, uguagliando il traffico generato al prodotto tra OL e la capacità complessiva del canale si ha la relazione che lega il tempo di interarrivo dei pacchetti all'OL, cioè  $t_{int} = 42.3 / OL$  pacchetti/ms.

### **IV.9 Procedura di attivazione e rilascio del canale**

Una volta generati, i pacchetti, che per noi rappresentano pacchetti IP, vengono frammentati in blocchi da 12 byte, quantità pari, come visto, alla capacità di una RU. Nello stack dei protocolli questa funzione di frammentazione dei pacchetti è assegnata al RLC, che vi aggiunge la sua intestazione completando il burst da trasmettere. Inoltre se l'ultimo segmento ha dimensioni minori di 12 byte si suppone che sia compito del RLC completarlo con bit di riempimento.

I segmenti generati in questo modo vengono raccolti nei buffer di trasmissione. A questo punto i due sottomodelli di utente in up link e in down link si differenziano, in quanto l'utente mobile deve effettuare una richiesta di accesso al sistema prima che gli vengano allocate le risorse.

L'algoritmo di assegnazione delle risorse è infatti centralizzato nella base station. Quest'ultima conosce lo situazione del traffico in down link, in quanto tale traffico è in realtà memorizzato nei suoi buffer di trasmissione, mentre è del tutto allo scuro della

---

situazione degli utenti in up link. È per tale motivo che l'utente mobile che attiva una trasmissione deve preventivamente effettuare una richiesta di accesso. Non disponendo di alcuna risorsa dedicata, esso effettua una richiesta a contesa sul RACH.

In realtà, ogni volta che si attiva un nuovo utente, esso deve eseguire tutta una serie di operazioni. Innanzitutto deve leggere nella trama ricevuta dalla stazione base il canale di sincronizzazione. Una volta sincronizzato ricava dal canale SCH la posizione nella trama del canale broadcast (BCH). Leggendo quest'ultimo canale l'utente viene a conoscenza della struttura della trama e del set di time slot e codici assegnati al RACH. A questo punto il mobile sceglie casualmente uno di questi codici per effettuare la sua richiesta.

#### IV.9.1 Implementazione del RACH

Nel nostro simulatore tutto questo viene implementato nel sottomodulo di utente in up link nella maniera seguente. Quando un utente si attiva il primo segmento che giunge nel buffer di trasmissione effettua la richiesta di attivazione. Si sincronizza con il time slot dedicato al RACH, per ipotesi il primo di ogni trama, ed effettua una scelta casuale di uno tra i 16 codici possibili. Alla fine del RACH viene effettuata la verifica: se in quel frame il codice scelto non è stato utilizzato da nessun altro utente e se non sono stati scelti più di 8 codici si attiva il canale, altrimenti si dà il via ad un opportuno algoritmo di backoff. Essendo il RACH mappato in un solo time slot, se per la stessa trama due utenti scelgono casualmente lo stesso codice si ha collisione ed entrambi devono riprovare l'accesso. Allo stesso modo se più di 8 codici sono scelti da 8 utenti diversi, nel time slot del RACH si verifica un'interferenza che, per quanto visto, produce un BER talmente alto da far fallire tutte le richieste. In questo caso tutti gli utenti ritentano la trasmissione.

L'algoritmo di backoff implementato nel simulatore è il seguente. Prima di effettuare la ritrasmissione l'utente attende un numero di frame scelto casualmente e con probabilità uniforme in una finestra larga  $2^N$ , dove con N si indica il numero di fallimenti consecutivi sperimentati dal singolo utente. La finestra non si allarga indefinitamente all'aumentare del numero di fallimenti, ma si arresta ad un limite massimo di 10 frame.

---

L'algoritmo di gestione della multiutenza utilizzato è dunque di tipo *Slotted Aloha*.

Come già osservato, l'accesso a contesa è valido solo per gli utenti in up link, mentre per quelli in down link, appena generato il primo pacchetto, esso giunge direttamente nei buffer della base station che può direttamente rilevarne la presenza ed attivare il canale relativo.

I documenti del 3GPP specificano soltanto che le procedure di attivazione e rilascio del canale sono a carico del protocollo RLC, ma non ne individuano le modalità. Come appena mostrato, la procedura di attivazione del canale non presenta particolari problemi, mentre di maggiore interesse risulta essere la definizione della tecnica di rilascio.

#### IV.9.2 Procedura di rilascio del canale

Alcuni studi propongono il rilascio del canale ad avvenuta trasmissione di ogni pacchetto IP. Questa soluzione è poco efficiente dal punto di vista dell'up link, in quanto rilasciare troppo spesso il canale significa aumentare le richieste di riattivazione e quindi sovraccaricare il RACH con conseguente aumento della probabilità di collisione. Tale soluzione può aver senso in un contesto tipo GPRS, in cui a causa dei problemi di banda e delle difficoltà di rappresentazione dell'informazione sul terminale mobile, la quantità di dati trasferita in una sessione è sempre relativamente piccola (le specifiche limitano la dimensione massima di una pagina WAP a circa 1400 byte). Questa assunzione cade invece in un ambiente più evoluto quale quello UMTS, in virtù della varietà e della complessità dei servizi previsti negli standard. Nei sistemi di terza generazione si prevede infatti l'offerta di servizi dati con una banda considerevolmente aumentata e l'introduzione di nuovi terminali mobili in grado di rappresentare un livello di informazione più complessa dei sistemi GPRS.

In base a queste considerazioni, si è pensato nel nostro simulatore di effettuare il rilascio del canale non ad avvenuta trasmissione di ogni pacchetto IP, ma solo al momento in cui i buffer di trasmissione dell'utente si sono svuotati.

Consideriamo un canale attivo. L'utente che ha attivato tale canale possiede delle risorse (RU) assegnategli in ogni frame dalla stazione base per la trasmissione dei suoi segmenti. Se durante la trasmissione di un singolo pacchetto IP (frammentato in singole RU) la sorgente

---

genera un nuovo pacchetto, l'utente sa già di dover chiedere nuove risorse ed è quindi inutile rilasciare il canale e ripassare dal RACH, in quanto il canale può più efficientemente essere lasciato attivo. È come se l'utente comunicasse alla stazione base la sua nuova richiesta di trasmissione del secondo pacchetto non passando nuovamente dal RACH, ma semplicemente utilizzando i canali dedicati assegnatigli dalla base station per la trasmissione dell'informazione.

In particolare, per la trasmissione dati sui canali dedicati lo standard prevede che, terminata la trasmissione del pacchetto, invece di abbattere la comunicazione, venga mantenuto attivo il canale (in modo da assicurare la sincronizzazione tra trasmettitore e ricevitore), ma riducendo il livello di interferenza generata e, quindi, l'occupazione delle risorse. Nel momento in cui un nuovo pacchetto è trasmesso, la risorsa trasmissiva è nuovamente impegnata. Se, invece, dopo un intervallo di tempo predefinito non giunge alcun pacchetto, la connessione è abbattuta.

In questo modo si ottiene non solo il vantaggio di non congestionare il RACH con continue richieste di attivazione che si avrebbero nella trasmissione di piccoli pacchetti IP, ma anche di velocizzare la procedura di trasferimento. Infatti ogni volta che si vuole effettuare una richiesta di risorse a contesa in up link, il mobile deve aspettare lo slot del RACH introducendo un conseguente aumento nel ritardo di trasmissione.

Sarebbe interessante confrontare le prestazioni in termini di collisioni sul RACH delle due tecniche viste. In futuri studi sul simulatore si può prevedere di aumentare gli utenti, in modo da rendere significativo il numero delle collisioni sul RACH e studiare il differente comportamento delle due soluzioni. Ci si aspetta che all'aumentare del numero di utenti e della quantità di traffico generato da ciascun utente, le prestazioni in termini di congestione migliorino a favore della tecnica di rilascio da noi adottata. Uno svantaggio è che prevedendo un numero molto elevato di utenti, alcune nuove richieste di servizio possano trovare la rete satura. Se ogni utente genera traffico con rate molto elevato, è prevedibile che esso rimanga attivo per un lungo periodo di tempo senza rilasciare il canale, per cui in presenza di un elevato numero di utenti con queste caratteristiche di traffico, la rete di

---

accesso può risultare congestionata ed eventuali nuove richieste di accesso devono essere rifiutate. A causa dell'algoritmo di gestione delle risorse di tipo  $1/N$  frame per frame da noi adottato e illustrato di seguito, questa probabilità di congestione della rete di accesso si verifica solo quando il numero delle RU complessivamente disponibili supera il numero degli utenti attivi, evento che, dato l'elevato numero di RU ( $12 \times (RU \times TS)$ ), risulta essere sufficientemente raro. Per questo motivo tra l'altro non si è reputato necessario implementare nel simulatore una tecnica di rifiuto della connessione in caso di congestione della rete (tecniche di Admission Controll).

#### **IV.10 Dimensionamento dei buffer di trasmissione**

Tutti i segmenti generati sia in up link che in down link aspettano nei buffer di trasmissione (per il RESQ code passive), finché la stazione base non assegna loro le risorse necessarie alla trasmissione secondo le regole previste dall'algoritmo di scheduling. La dimensione dei buffer di trasmissione è limitata a 10 volte la dimensione massima prevista per un pacchetto IP (1500 byte). Ciascun buffer può quindi contenere fino ad un massimo di 1250 RLC SDU ( $((1500/12) \times 10)$ ), successivamente mappate in RU. Se al sopraggiungere di un pacchetto IP il buffer di trasmissione è pieno tale pacchetto viene scartato integralmente. È stato implementato un meccanismo per calcolare la probabilità di scarto che diventa significativa all'aumentare dell'offered load (per  $OL > 1$ ). La limitazione dei buffer di trasmissione è necessaria per rendere significativa la misura dei ritardi di trasmissione nei casi in cui il carico della rete supera la sua capacità. In questo caso infatti, supponendo illimitata la capacità dei buffer, tale misura non ha senso perché le code nei buffer crescono indefinitamente insieme ai ritardi che vengono a dipendere strettamente dalla durata simulazioni.

#### **IV.11 Algoritmo di scheduling**

Il nostro algoritmo di scheduling può essere concettualmente diviso in due passi distinti.

---

In un primo momento lo scheduler calcola ad ogni frame il numero di time slot da assegnare ad up e down link ed il numero di Risorse Unitarie per time slot attraverso i due algoritmi presentati nel capitolo 3, adattivi rispettivamente al traffico e alle condizioni di interferenza. In questo modo si ottiene il numero di Risorse Unitarie complessivamente disponibili per quel frame per le due direzioni di traffico.

Il secondo passo dell'algoritmo si propone di ripartire frame per frame in maniera equa tra gli utenti attivi le risorse disponibili. È chiaro che l'allocazione delle risorse non può essere statica, cioè non è possibile assegnare ad un utente un certo numero di RU per ogni frame per tutto il tempo in cui esso rimane attivo. Supponiamo infatti che un utente si attivi quando la rete è completamente scarica, cioè non vi sono altri utenti attivi nella cella. Ad esso viene assegnata tutta la capacità del sistema, cioè tutte le RU possibili. Supponiamo altresì che tale utente debba trasmettere un volume di traffico molto elevato, per cui, nonostante l'elevata velocità di trasmissione assegnatagli, esso impiega più di un frame per smaltire il suo traffico. Se l'algoritmo di assegnazione delle risorse è statico, all'utente in questione vengono riservate tutte le RU per tutta la durata della connessione; in questo modo l'intero accesso risulta congestionato ed un secondo utente che richiede il servizio prima che il primo abbia terminato viene ovviamente respinto.

Questo approccio è quindi privo di significato. Nel nostro algoritmo tutte le risorse del sistema vengono ripartite equamente tra gli utenti attivi, intesi come somma dei vecchi utenti che non hanno ancora svuotato i loro buffer e i nuovi utenti appena attivati. In questo modo è pressoché impossibile che un utente venga respinto per congestione della rete, a meno che il numero di utenti attivi non superi il numero di RU disponibili, evento che nelle nostre simulazioni non si verifica mai.

L'algoritmo di ripartizione delle risorse è quindi di tipo *round robin 1/N*, tenuto conto del fatto che la risorsa minima assegnabile non è infinitesima, ma la sua granularità è pari a quella di una RU (12 byte di informazione).

Riassumiamo il nostro procedimento di assegnazione delle risorse. Lo scheduler calcola il numero complessivo di utenti attivi, come somma degli utenti attivi in up link e in down

---

link. (Esistono due vettori degli utenti uno in up link e uno in down link. Ciascun elemento di un vettore corrisponde ad un utente e vale 1 o 0 a seconda che l'utente sia attivo o meno).

Ricordando il vincolo delle specifiche che impone di assegnare ciascun time slot ad un'unica direzione di traffico, lo scheduler assegna, in modo proporzionale al rispettivo numero di utenti, i 12 time slot alle due tratte. Successivamente calcola il numero di RU per time slot adattandolo alle condizioni di interferenza tra gli utenti misurate nei frame precedenti. Moltiplicando il numero di time slot assegnati alla tratta considerata per il numero di RU per time slot calcolato si ottiene il numero di RU complessive per quella tratta, che viene diviso per il numero di utenti in quella direzione, ottenendo il numero di RU per utente. Dell'ultima divisione viene ovviamente considerata la parte intera inferiore essendo le RU indivisibili. Vengono quindi calcolati i resti della divisione. Le RU ottenute come resto della divisione vengono assegnate una alla volta ad utenti diversi fino ad esaurimento.

Ciascun utente può trasmettere in ogni frame un numero di segmenti RLC pari al numero di RU assegnategli. Tale algoritmo garantisce la massima *fairness* tra gli utenti attivi, ma non consente alcun controllo sui ritardi di trasmissione. Il numero di RU assegnate ad un utente varia da frame a frame in base al numero di utenti attivi, quindi la banda allocata all'utente varia durante la sessione di collegamento. Essendo il numero di utenti attivi nei vari frame imprevedibile è quindi impossibile effettuare previsioni o fornire garanzie sui tempi di trasmissione.

## **IV.12 Primi risultati delle simulazioni**

Il simulatore è in grado di misurare oltre alla probabilità di perdita di pacchetti dovuta al riempimento dei buffer di trasmissione anche le stime di throughput aggregato e ritardi medi di trasmissione di un pacchetto IP separatamente in up link e in down link.

Di seguito sono riportati i grafici che mostrano i risultati delle prime simulazioni effettuate. Gli andamenti sono ovvi e facilmente giustificabili, ma servono a validare il funzionamento del simulatore ed a dare una rappresentazione del tipo di statistiche

---

considerate. Il corretto funzionamento del simulatore viene verificato considerando un caso ideale di assenza di errore di trasmissione sul canale. Con questa semplificazione è possibile prevedere analiticamente il comportamento dell' algoritmo di scheduling implementato. Si osserva che i risultati delle simulazioni presentati di seguito sono perfettamente in accordo con quelli ottenibili analiticamente.

La durata delle simulazione è pari a quella di 10.000 trame. Per garantire l'affidabilità statistica delle simulazioni i risultati proposti sono stati ottenuti con il *Metodo delle Prove Ripetute* [23], in cui si sono generate 10 prove statisticamente indipendenti dello stesso scenario. I risultati presentano un intervallo di confidenza del 90% e l'affidabilità ottenuta è sempre al di sotto del 4-5%.

Questi primi tre grafici mostrano al variare dell' offered load le prestazioni del sistema in termini di throughput, ritardo di trasmissione di un pacchetto IP e sua probabilità di perdita a causa del riempimento dei buffer di trasmissione. I risultati sono stati ottenuti considerando il caso ideale di assenza di errore, con traffico simmetrico con coefficiente di variazione pari

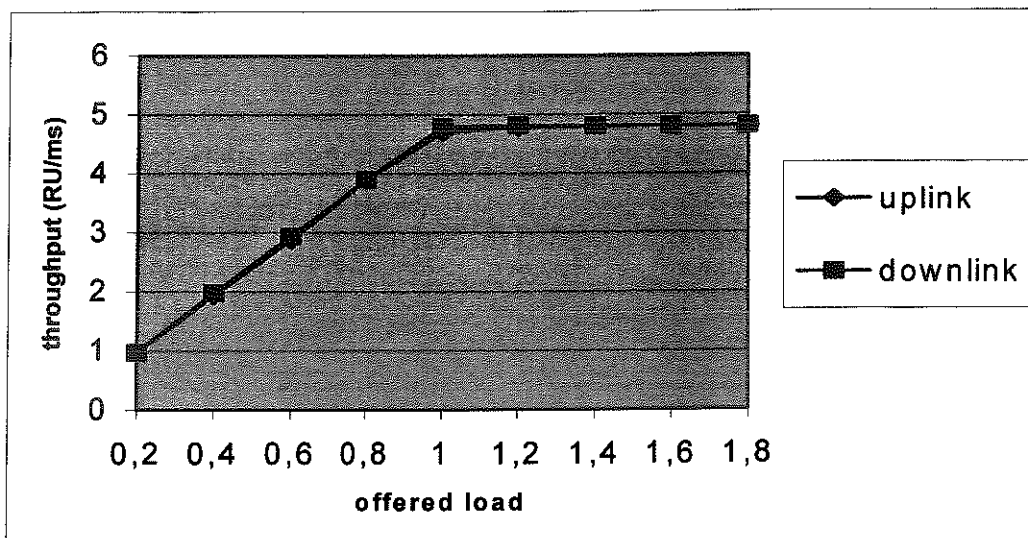


Figura 3. Andamento del throughput in funzione dell' offered load

ad 1 ed un numero di Risorse Unitarie per time slot staticamente fissato a 8. Il primo grafico mostra come il throughput è praticamente coincidente per le due direzioni di traffico in quanto, a causa dello scarso numero di utenti implementato, probabilità di collisione sul

RACH è trascurabile e quindi la tratta in up link non viene affatto penalizzata. Come è logico aspettarsi, il throughput aumenta linearmente con l'offered load fino alla situazione di saturazione della rete di accesso radio ( $OL=1$ ). Da questo punto in poi per carichi di traffico superiori a quello gestibile ( $OL>1$ ), il throughput si assesta su di un valore costante e pari per ciascuna tratta alla metà della massima capacità della rete, che, come già osservato, è di 9,8 RU/ms. Resta da osservare che le misure di capacità vengono presentate in termini di RU/ms, in quanto una RU rappresenta una quantità di dati costante pari a 12 byte di informazione utile.

Il secondo grafico evidenzia come il ritardo medio di trasmissione di un pacchetto IP può ritenersi a buon diritto trascurabile fino ad offered load di 0.8. Al di sopra di tale valore il ritardo subisce un brusco aumento fino a superare un secondo. Per elevati OL il ritardo medio tende poi a stabilizzarsi asintoticamente attorno al valore di circa 1.3 secondi. Questo risultato è ampiamente prevedibile in virtù del fatto che il ritardo coincide con il tempo di

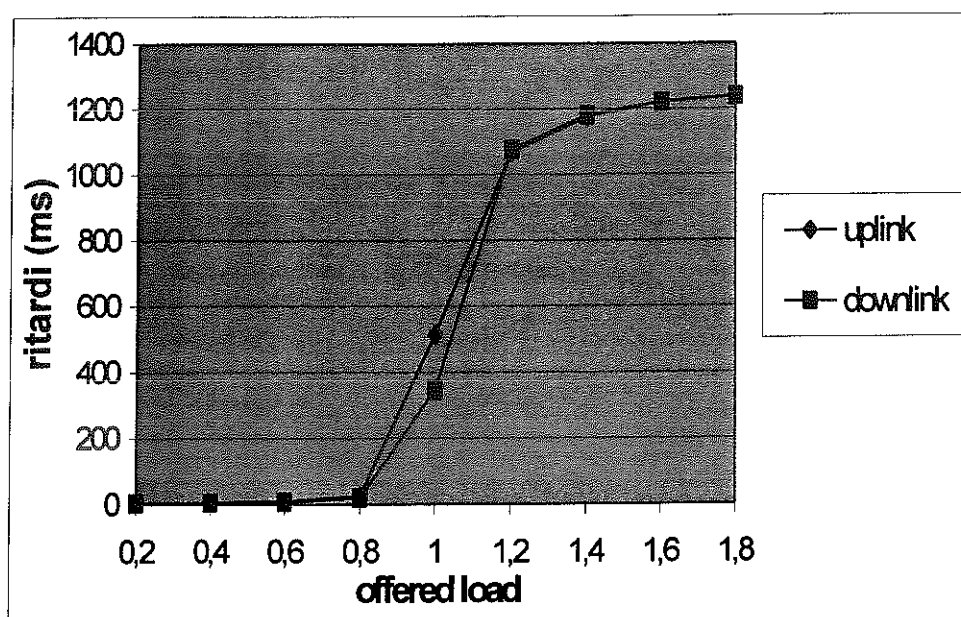


Figura 4. Andamento dei ritardi medi di trasmissione in funzione dell'offered load

svuotamento del buffer. Per carichi di traffico così alti, infatti, i buffer di trasmissione sono praticamente sempre pieni per cui il ritardo di trasmissione di un pacchetto è direttamente legato alla loro dimensione.

Come visto precedentemente i buffer sono stati dimensionati per contenere 10 pacchetti IP di massima lunghezza (1500 byte), il che equivale a dire che un buffer può contenere fino a 1250 RU. Poiché abbiamo un buffer per utente e 10 utenti, la quantità di dati complessivamente contenibile nei buffer è di 12500 RU. Dividendo per la capacità del canale si ottiene il tempo di svuotamento del buffer, cioè  $12500/9,8=1275,5$  ms, valore molto vicino a quello ottenuto come tempo di ritardo nelle simulazioni con alto OL.

Infine presentiamo il grafico della probabilità di scarto di un pacchetto IP dovuta alla saturazione del buffer. Fino al valore limite di offered load pari ad 1, tale probabilità si mantiene molto bassa per poi aumentare con un andamento sublineare tendente asintoticamente ad 1. Infatti, dato un offered load  $x > 1$ , la probabilità di scarto è  $(x-1)/x$ .

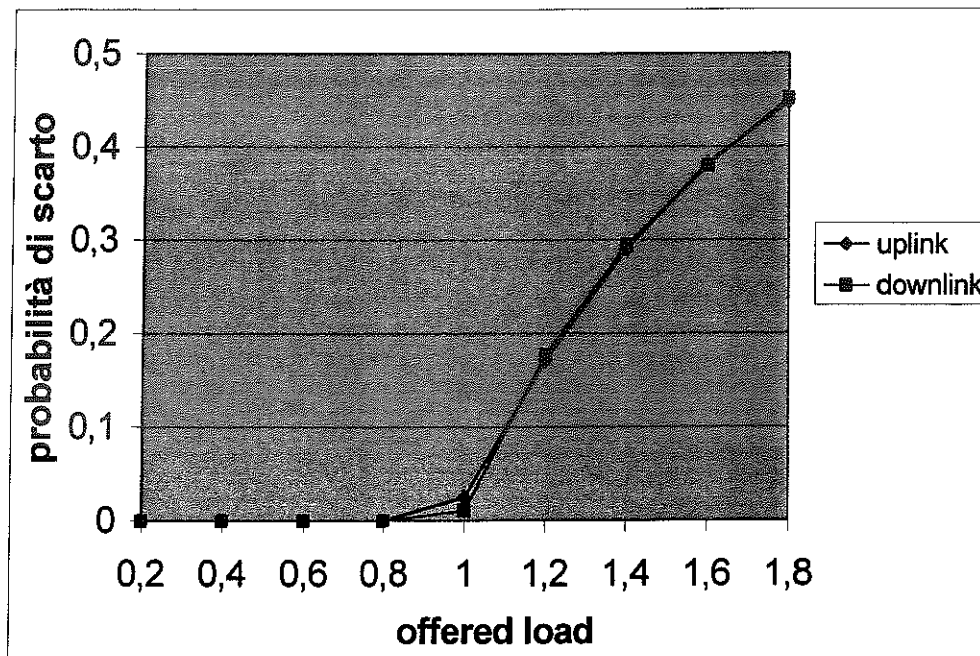


Figura 5. Andamento delle probabilità di scarto in funzione dell'offered load



---

# CAPITOLO

# 5

---

## RISULTATI DELLE SIMULAZIONI

*In questo capitolo vengono presentate le valutazioni delle performance degli algoritmi di scheduling proposti. Attraverso i risultati delle simulazioni si confrontano le tecniche statica e dinamica di ripartizione dei time slot tra up e down link e i tre algoritmi adattivi proposti per calcolare il numero di Risorse Unitarie da allocare in ogni Time Slot.*

### **V.1 Ripartizione up link/down link statica e dinamica**

Per valutare i vantaggi conseguibili attraverso un'assegnazione dei time slot effettuata frame per frame in base al numero di utenti attivi nelle due tratte, piuttosto che una volta per tutte in base a considerazioni statistiche sul bilanciamento del traffico, ci si è posti nella condizione più favorevole alla soluzione statica. Si è supposto infatti di conoscere perfettamente il bilanciamento del traffico generato e, nella soluzione statica, si sono

conseguentemente ripartiti i time slot.

Questa è una condizione ideale a favore della soluzione statica, in quanto il bilanciamento del traffico generato si mantiene costante per tutta la simulazione. Nella realtà, anche se sul lungo periodo (per esempio nell'arco di una giornata) il rapporto tra i bit trasmessi nelle due direzioni di traffico si mantiene costante, nel breve periodo esso presenta delle fluttuazioni. Tali fluttuazioni sono ingestibili attraverso la soluzione statica e ne degradano inevitabilmente le prestazioni.

Nelle simulazioni si è preferito considerare il caso ottimo per la soluzione statica in cui i time slot vengono ripartiti inizialmente in modo proporzionale al bilanciamento del traffico tra up e down link che rimane costante per tutta la simulazione.

Variando l'offered load di up e down link si è cambiato il valore medio del tempo di interarrivo dei pacchetti in modo da sbilanciare il traffico. Nelle simulazioni però tale tempo di interarrivo è una variabile statistica con una *distribuzione standard* che, oltre al valor medio, è caratterizzata dal **coefficiente di variazione** (rapporto tra la deviazione standard e il valor medio della distribuzione).

Si è quindi deciso di analizzare il comportamento dei due algoritmi di ripartizione dei time slot in funzione del coefficiente di variazione del tempo di interarrivo del traffico generato.

Tale parametro caratterizza la *burstness* del traffico generato. All'aumentare del coefficiente di variazione i pacchetti tendono sempre più ad essere generati a burst, cioè in alcuni periodi di tempo viene generato un gran numero di pacchetti mentre in altri la sorgente rimane inattiva. In tal modo i tempi di attività e inattività degli utenti variano in modo meno uniforme. Data la tecnica di attivazione e rilascio del canale da noi utilizzata che si basa sullo stato dei buffer di trasmissione di ciascun utente, si alterneranno per ogni utente periodi di attività e inattività sempre più lunghi dovuti rispettivamente alla generazione di molti pacchetti e all'inattività della sorgente. Un algoritmo dinamico che varia la ripartizione dei time slot in base al numero di utenti attivi nelle due tratte, può adattarsi meglio di un algoritmo statico ad un traffico tipo burst, com'è tipicamente il

---

traffico Internet. Se in un particolare frame infatti ci sono più utenti attivi in una direzione significa che il traffico complessivamente presente nei buffer di trasmissione è maggiore, per cui l'algoritmo tende ad aumentare proporzionalmente il numero di time slot.

Per le prime simulazioni si sono considerati tre tipi di bilanciamento. Essendo il traffico Internet tipicamente molto maggiore in down link che in up link, si è partiti da un caso simmetrico, per poi considerare traffici in down link 3 e 11 volte superiori a quelli in up link.

La ripartizione statica è stata effettuata in modo proporzionale al traffico. Avendo considerato 12 time slot dedicati al traffico dati, nel caso simmetrico la ripartizione statica è stata 6 – 6, mentre nei casi di rapporto 3 a 1 e 11 a 1 la suddivisione è stata rispettivamente 9 – 3 e 11 – 1. Nell'ultimo caso con rapporto 11 a 1 si sono differenziati anche i tipi di pacchetti sulle due tratte: si sono considerati pacchetti di dimensioni uniformemente distribuita tra 500 e 1500 byte per il down link e tra 40 e 60 per l'up link.

Le simulazioni sono state effettuate nel caso ideale di assenza di errore sul canale e si è considerato un numero di Risorse Unitarie per time slot fissato a 8.

I grafici di seguito riportati sono stati ottenuti considerando un offered load complessivo della rete pari a 0.8. Essendo la condizione di carico della rete non particolarmente critica, non si hanno significative probabilità di scarto per riempimento dei buffer di trasmissione all'aumento del coefficiente di variazione. Questo rende significativa l'analisi dei ritardi medi misurati nella trasmissione di un pacchetto IP, che è influenzata soltanto dal numero di time slot assegnati alle due tratte e non da eventuali perdite di pacchetti. Conseguentemente i grafici dei throughput appaiono poco significativi in quanto coincidono per i due algoritmi.

I throughput medi misurati per up e down link si mantengono esattamente nello stesso rapporto del traffico generato, come è ovvio che sia in assenza di perdite.

Nel grafico seguente sono mostrati i throughput ottenuti per l'up e il down link dei due algoritmi, nel caso di rapporto tra i traffici generati di 3 a 1.

---

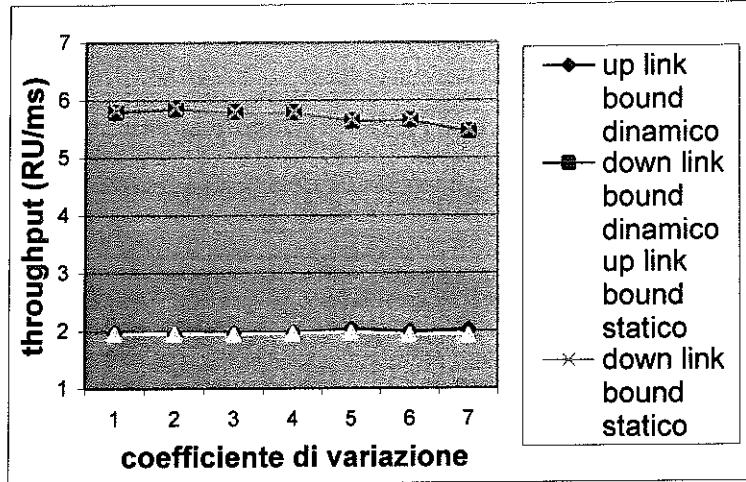


Figura 1. Confronto dei due metodi di ripartizione in termini di throughput medio in presenza di traffico asimmetrico con rapporto 3 a 1

Più interessanti appaiono invece i confronti dei due algoritmi in termini di ritardo medio di trasmissione di un pacchetto IP in up e down link.

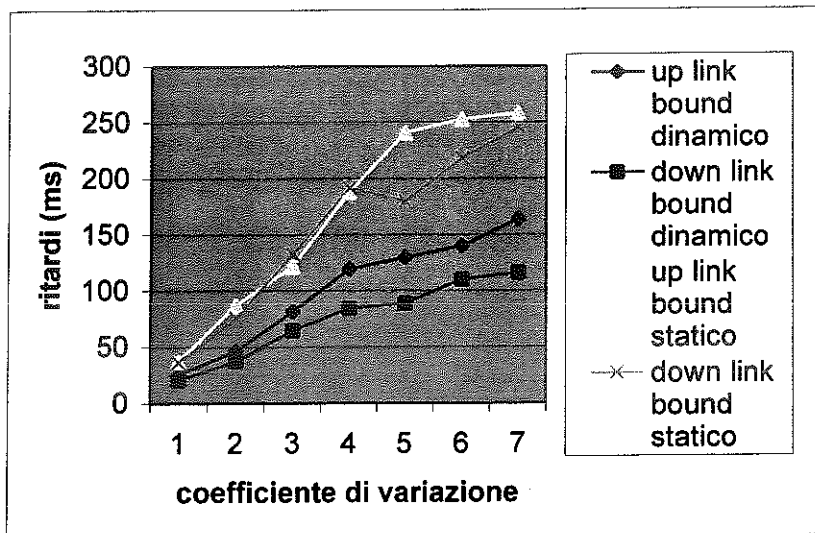


Figura 2. Confronto dei due metodi di ripartizione in termini di ritardo medio di trasmissione in presenza di traffico simmetrico

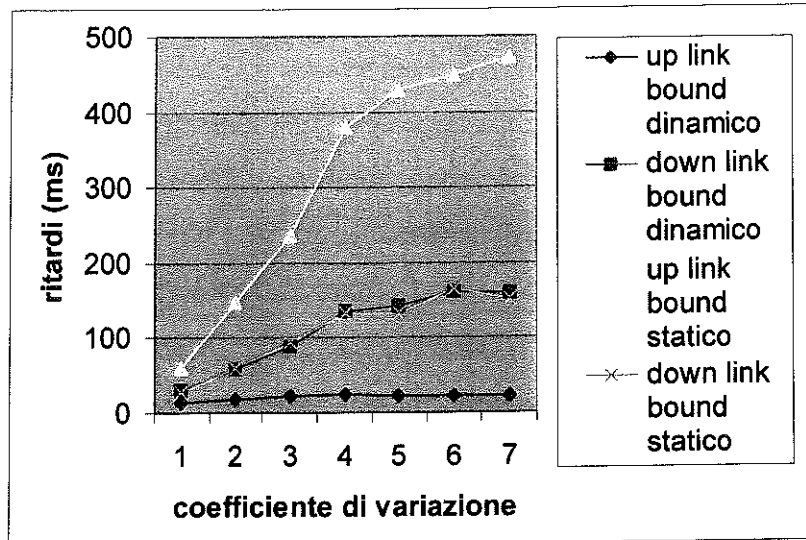


Figura 3. Confronto dei due metodi di ripartizione in termini di ritardo medio di trasmissione in presenza di traffico asimmetrico con rapporto 3 a 1

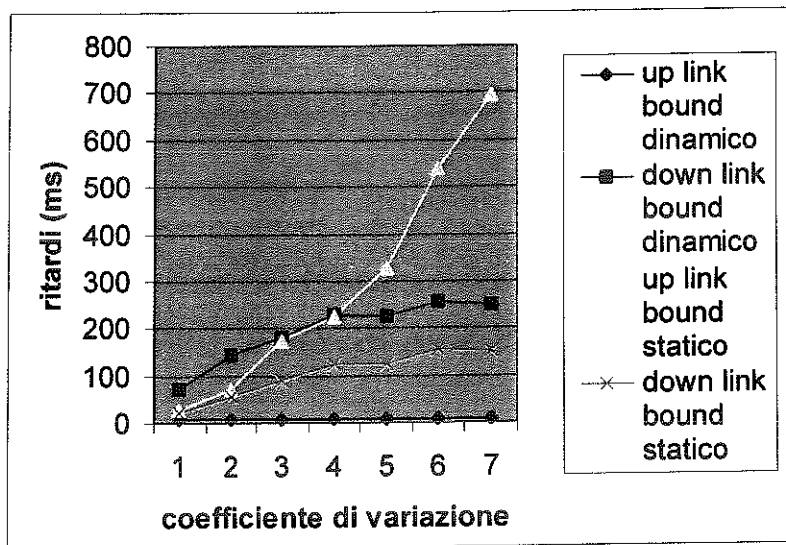


Figura 4. Confronto dei due metodi di ripartizione in termini di ritardo medio di trasmissione in presenza di traffico asimmetrico con rapporto 11 a 1

In tutti i tre grafici si nota chiaramente come l'aumentare del coefficiente di variazione i ritardi medi di trasmissione di un pacchetto IP tendono a crescere.

In figura 2 è mostrato il caso di traffico simmetrico. Gli andamenti misurati per le due

tecniche di ripartizione sono simili, ma si osserva che i ritardi misurati con la ripartizione dinamica dei time slot sono sempre minori sia per l'up link che per il down link.

Differente è il caso di figura 3 in cui è mostrato il caso di traffico asimmetrico con rapporto 3 a 1. In virtù di tale bilanciamento del traffico, la soluzione statica assegna per ogni frame 9 time slot al down link e 3 all'up link. I 9 time slot assegnati dalla soluzione statica al down link sono sempre sufficienti per smaltire efficientemente il traffico generato, infatti i ritardi medi di trasmissione nella tratta in salita sono uguali a quelli ottenuti con la soluzione dinamica. Altrettanto non può dirsi per l'up link. Infatti in tale direzione 3 time slot per frame non sono sufficienti a gestire un traffico a burst e le prestazioni della tecnica di ripartizione statica degradano sensibilmente all'aumentare del coefficiente di variazione. Al contrario nell'algoritmo di ripartizione dinamico i ritardi medi misurati per l'up link sono trascurabili e non variano all'aumentare del coefficiente di variazione. Ciò è dovuto al fatto che il traffico generato in up link è mediamente basso e l'algoritmo adattivo può facilmente gestirlo anche all'aumentare della burstness. Tale algoritmo in presenza di improvvisa generazione di un elevato numero di pacchetti da parte di utenti in up link non fa altro che aumentare momentaneamente il numero di time slot, per poi diminuirlo una volta trasmessi i pacchetti.

La figura 4 mostra infine il caso in cui il traffico in down link è 11 volte superiore a quello in up link. Le considerazioni sono analoghe al caso precedente. Si osserva un andamento ancor più critico dei tempi medi di ritardo in up link per la soluzione statica. Ciò è dovuto chiaramente al fatto che con un solo time slot per frame disponibile è impossibile gestire burst di pacchetti generati in up link.

E' interessante notare un'altra differenza tra i grafici di figura 3 e 4. Mentre per il caso di figura 3 gli andamenti dei ritardi in down link sono perfettamente sovrapposti per le due tecniche, in figura 4 si nota un leggero aumento dei ritardi a scapito della soluzione dinamica. Questa differenza è dovuta al fatto che, come già detto, nel secondo caso si sono differenziati i traffici nelle due tratte generando pacchetti lunghi in down link e corti in up link. In questo modo quando l'algoritmo dinamico sottrae banda in down link penalizza in

---

modo più grave gli utenti in quanto questi ultimi hanno da trasmettere pacchetti mediamente più lunghi.

Concludiamo il confronto mostrando un caso in cui il traffico generato è pari alla capacità del canale radio ed è simmetrico. In questa condizione critica di offered load pari ad uno si nota subito come la probabilità di scarto dei pacchetti per riempimento dei buffer di trasmissione non è più trascurabile.

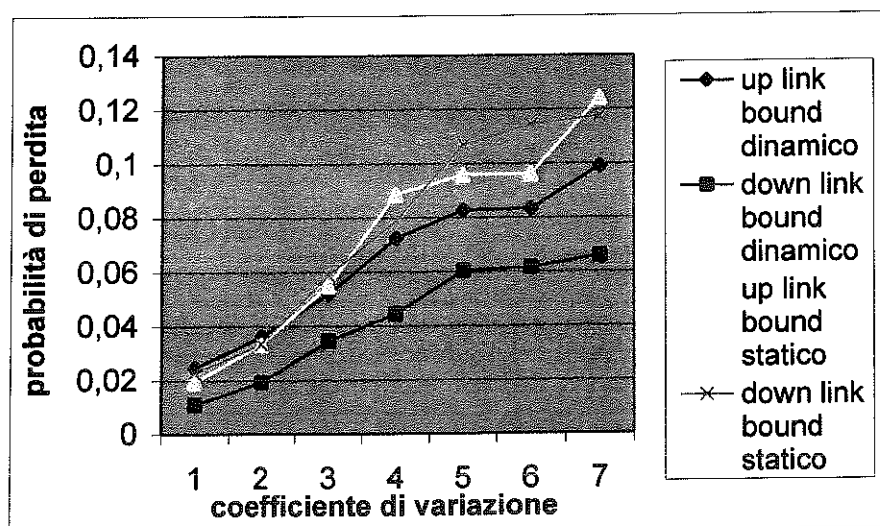


Figura 5. Confronto dei due metodi di ripartizione in termini di probabilità di perdita di un pacchetto in presenza di traffico simmetrico con offered load pari ad 1

Si osservi come la probabilità di perdita aumenta all'aumentare del coefficiente di variazione a causa del fatto che burst di pacchetti tendono a riempire i buffer di trasmissione. L'algoritmo dinamico presenta delle prestazioni leggermente migliori all'aumentare del coefficiente di variazione in virtù del fatto che esso tende ad aumentare le risorse per una delle due tratte in presenza di burst di pacchetti ed è quindi in grado di smaltire con più velocità tali burst evitando il riempimento dei buffer. E' il caso di osservare esplicitamente che la penalizzazione dell'up link rispetto al down link nella soluzione dinamica si deve al fatto che quando si presentano dei resti nella divisione proporzionale dei time slot, per nostra scelta, essi vengono assegnati al down link.

E' evidente che la probabilità di perdita non trascurabile impatta in maniera diretta sul throughput, come mostra la figura successiva.

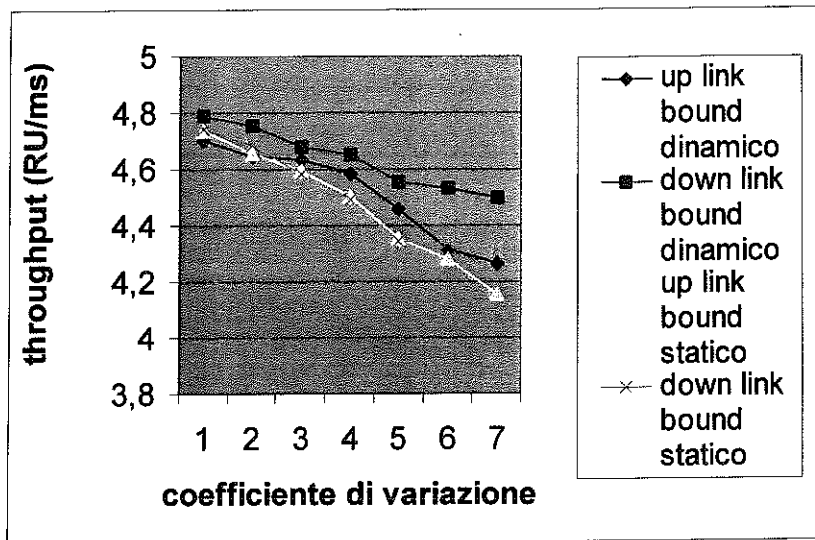


Figura 6. Confronto dei due metodi di ripartizione in termini di throughput in presenza di traffico simmetrico con offered load pari ad 1

Meno significativa è, in ultimo, l'analisi dei tempi di ritardo. In questo caso infatti non si nota nessun aumento del tempo medio di trasmissione all'aumentare della burstness del traffico. Ciò si spiega osservando che l'aumento del coefficiente di variazione del tempo di

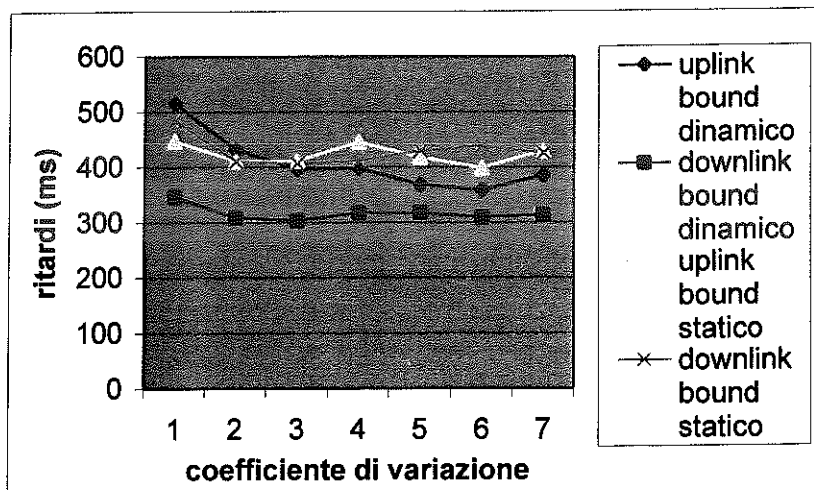


Figura 7. Confronto dei due metodi di ripartizione in termini di ritardo medio di trasmissione in presenza di traffico simmetrico con offered load pari ad 1

interarrivo dei pacchetti da un lato genera maggiori oscillazioni nel riempimento dei buffer di trasmissione, dall'altro diminuisce il valor medio di tali oscillazioni in conseguenza dell'abbassamento del throughput. In pratica questi due effetti si compensano determinando l'andamento costante dei ritardi al variare del coefficiente di variazione osservato nelle simulazioni. Si noti come per piccoli valori del coefficiente di variazione l'effetto della diminuzione del throughput predomina con conseguente leggero abbassamento del ritardo medio.

## V.2 Algoritmi di scheduling adattivi

Nel paragrafo precedente ci siamo occupati di valutare l'efficienza del metodo di ripartizione *verticale* della matrice tempo-codice tra le due direzioni di traffico. Nel seguente invece ci occupiamo del problema del bound *orizzontale* della matrice, cioè degli algoritmi che fissano per ogni frame il numero di Risorse Unitarie (RU) per Time Slot (TS).

Di seguito si analizzano quindi le prestazioni della classe di algoritmi adattivi proposti nel capitolo 3. Lo studio è stato diviso in due parti: dapprima si analizza il caso di *canale statico*, con andamento tempo invariante della probabilità di errore al variare del numero di RU per TS, e si valuta l'algoritmo al variare della funzione considerata; successivamente si considera una funzione di interferenza variante nel tempo e si valuta il comportamento degli algoritmi in presenza di *canale dinamico*.

### V.2.1 Analisi statica

Poiché non conosciamo a priori l'andamento della *funzione di interferenza* (probabilità di errore al variare delle RU per TS ( $N$ )), è necessario mostrare che gli algoritmi proposti sono efficienti per la più ampia classe possibile di funzioni. Per tale ragione l'analisi statica dei 3 algoritmi è stata condotta per diversi andamenti della probabilità di errore, valutando le eventuali variazioni di comportamento al variare della funzione.

---

Nei pochi lavori reperiti in letteratura sull'argomento, viene data particolare enfasi alla classe delle funzioni "a gradino". Si suppone infatti che, grazie alle tecniche di cancellazione dell'interferenza adottate (vedi par I.5.2), l'interferenza intracella si mantenga molto bassa per pochi codici allocati in un time slot, per poi aumentare in modo drastico superata una certa soglia, solitamente posta a 8 RU per TS.

La semplificazione più brutale di questo comportamento porta a considerare una funzione di probabilità di errore a gradino con una soglia posta, per compatibilità con i lavori precedenti, a 8 RU per TS, in cui la probabilità di errore è nulla fino ad  $N$  uguale a 8 e unitaria al di sopra di tale valore. Nel nostro lavoro si è ritenuto opportuno completare l'analisi di questa classe di funzioni introducendo altri due andamenti che consideriamo più realistici. Oltre la funzione a *gradino* con discontinuità di prima specie, si sono utilizzate una funzione continua con andamento a *sigmoide* ed una con una discontinuità nella derivata con andamento *esponenziale troncato*. (E' chiaro che, trattandosi di funzioni discrete, parliamo impropriamente di continuità o discontinuità riferendoci ad una funzione di raccordo che interpola i punti ottenuti per i vari valori di  $N$ ).

Infine, in aggiunta alle tre funzioni precedenti, si è voluto analizzare il comportamento degli algoritmi per una funzione di probabilità di errore di una classe diversa. Per far ciò si è considerata una funzione con crescita *lineare* della probabilità di errore da 0 per  $N=1$  a 1 per  $N=16$ . Fin da subito è utile osservare che questo tipo di funzione è più critica per i nostri algoritmi, perché l'assenza di una brusca variazione presente nelle altre probabilità di errore considerate rende più difficile l'individuazione del punto ottimo. E' parso necessario quindi mostrare il funzionamento degli algoritmi proposti anche in presenza di un tale tipo di funzione.

Di seguito proponiamo le funzioni di interferenza utilizzate nell'analisi statica con i relativi andamenti del throughput per TS calcolati come definito nel capitolo 3. Si osservi come, per nostra scelta, le funzioni sono state impostate in maniera che il massimo del throughput cada sempre in corrispondenza di un numero di RU per TS pari a 8.

---

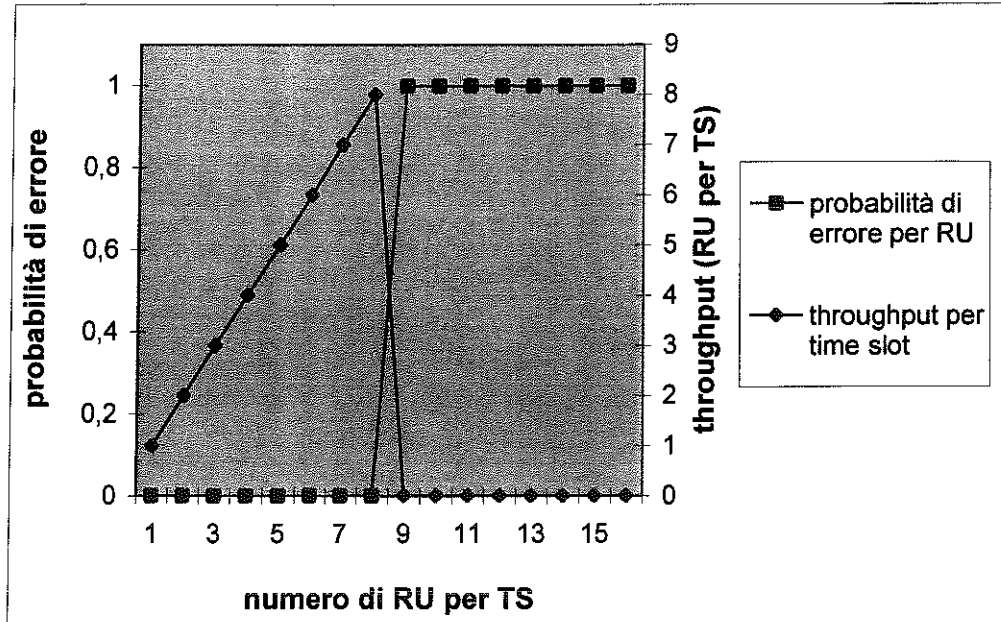


Figura 8. Probabilità di errore a gradino e relativo throughput

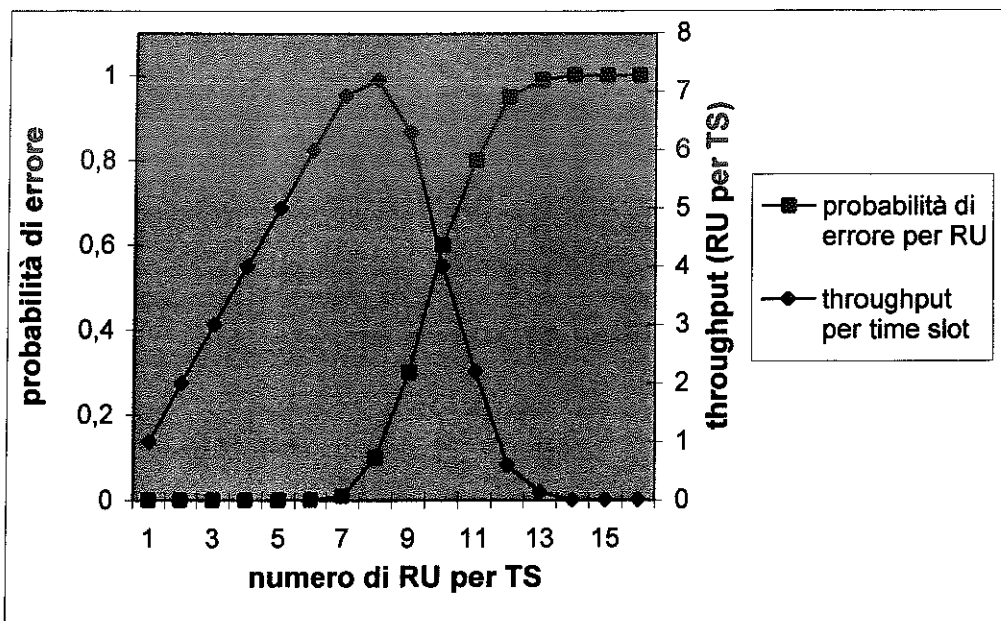


Figura 9. Probabilità di errore a sigmoide e relativo throughput

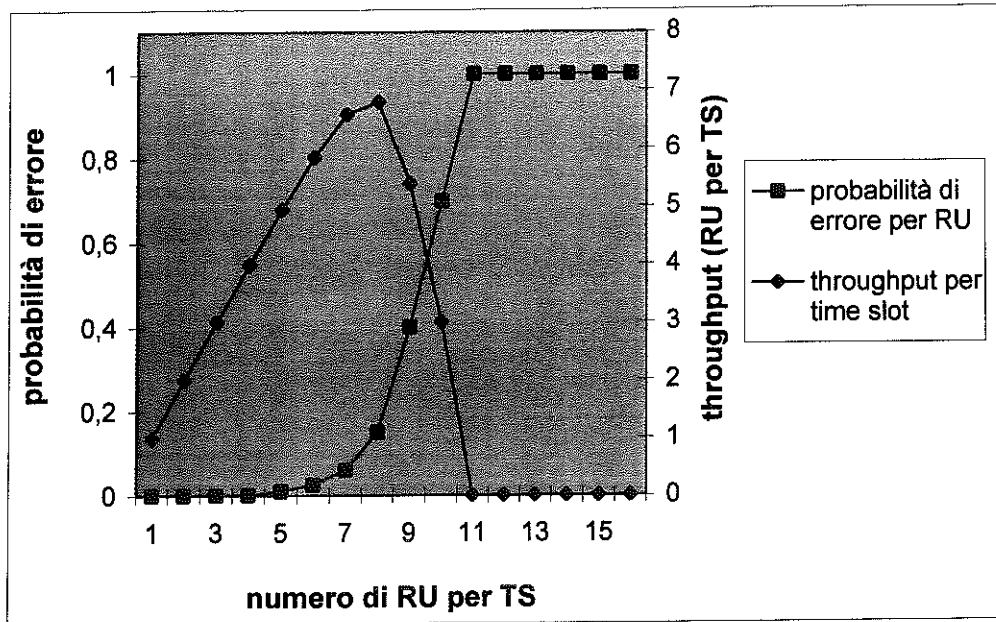


Figura 10. Probabilità di errore esponenziale troncato e relativo throughput

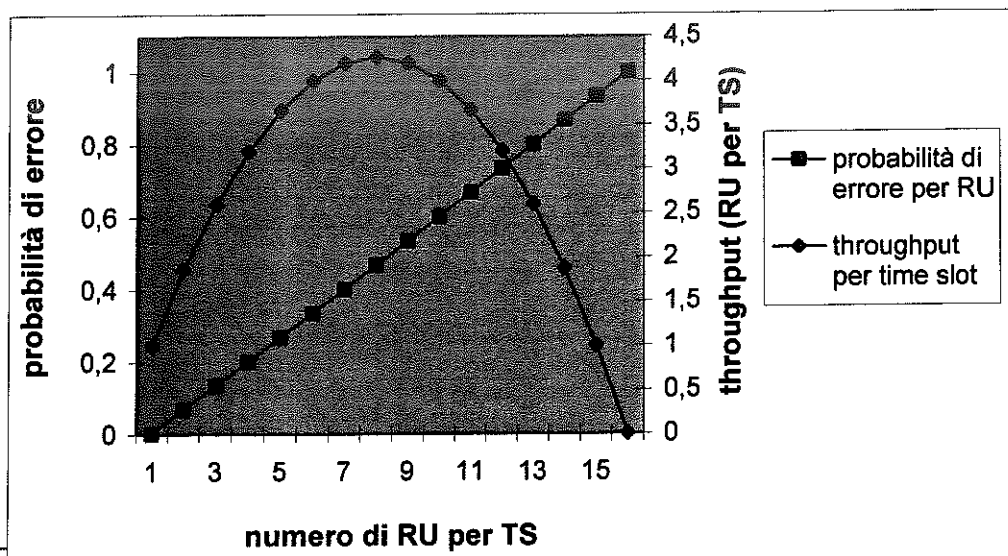


Figura 11. Probabilità di errore lineare e relativo throughput

Possiamo a questo punto presentare i risultati ottenuti con i tre algoritmi al variare della funzione di interferenza utilizzata nel simulatore.

Innanzitutto analizziamo i due algoritmi *con memoria* che presentano un comportamento dipendente dal *fattore di smoothing* ( $\alpha$ ) utilizzato. D'ora in poi chiameremo *algoritmo con memoria 1*, la diretta evoluzione dell'algoritmo senza memoria con utilizzo del fattore di correzione  $1-\alpha$  nell'aggiornamento delle stime, mentre con *algoritmo con memoria 2* si indicherà la tecnica che adotta una differente filosofia euristica, priva del fattore di correzione  $1-\alpha$  e del concetto di perdita accumulata (vedi III.3)

I risultati ottenuti in termini di throughput, nel caso di canale statico, per i due algoritmi con memoria al variare del fattore di smoothing nei 4 casi di probabilità di errore appena

$\alpha$	$\frac{1}{1-\alpha}$
0.7	3
0.9	10
0.97	33
0.99	100
0.997	333
0.999	1000
0.9997	3333
0.9999	10000

Tabella 1. Valori di  $\alpha$  utilizzati

visti sono presentati di seguito. I valori di  $\alpha$  utilizzati sono stati scelti in modo che i punti ottenuti fossero equidistanziati in una scala logaritmi. Sulle ascisse è rappresentato in scala logaritmica il valore  $\frac{1}{1-\alpha}$ , che, come già osservato nel capitolo 3, rappresenta un'indicazione, in termini di numero di aggiornamenti, del *tempo medio di memoria* di una stima del sistema. Infatti, dato che l'aggiornamento viene effettuato ad ogni frame, in pratica  $\frac{1}{1-\alpha}$  ci dice approssimativamente per quanti frame in media viene conservata una misura.

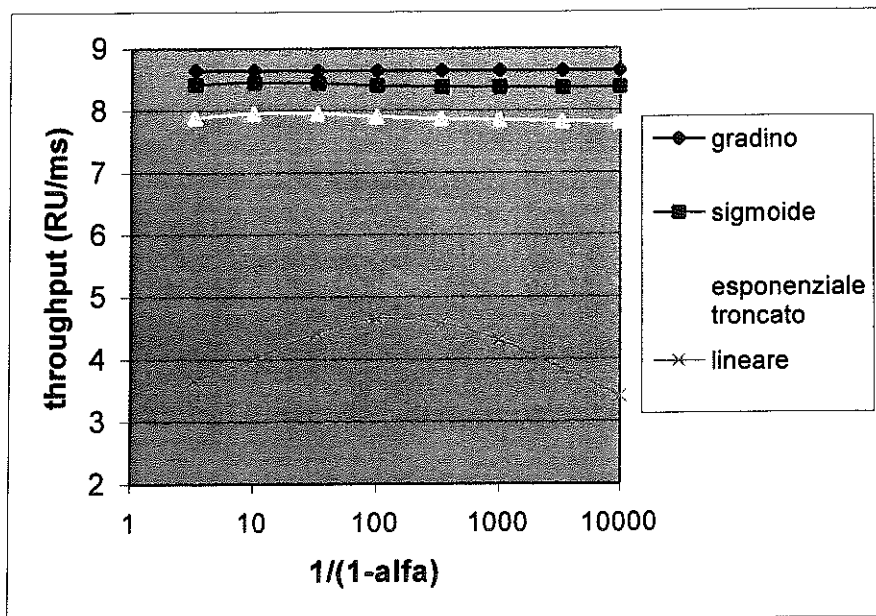


Figura 12. Comportamento dell'algorithm con memoria di tipo 1 nel caso statico al variare del fattore di smoothing per varie funzioni di interferenza

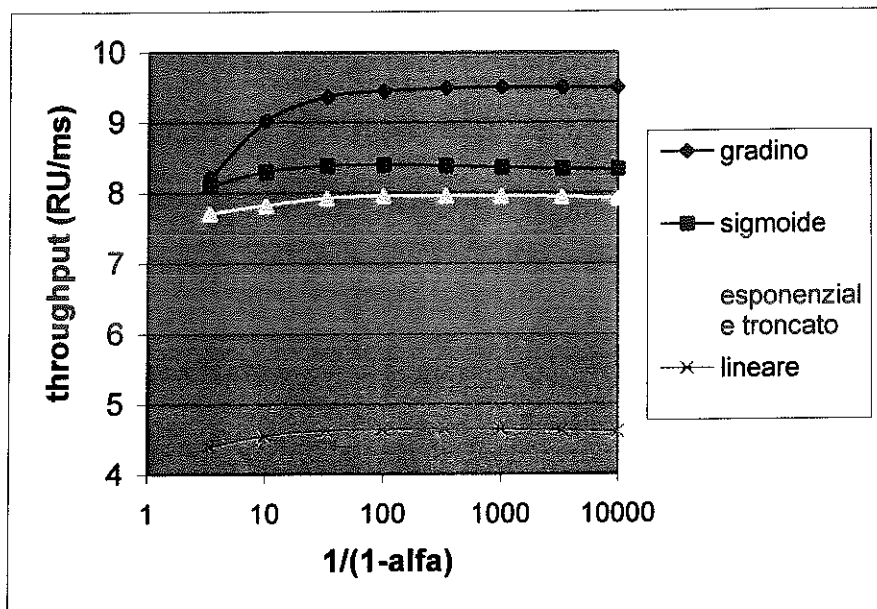


Figura 13. Comportamento dell'algorithm con memoria di tipo 2

---

nel caso statico al variare del fattore di smoothing per varie funzioni di interferenza

Essendo l'analisi statica, la funzione di interferenza non cambia per tutta la durata della simulazione, ragion per cui le misure di probabilità di errore effettuate non perdono di significato con il passare del tempo. In presenza di canale statico quindi il miglior stimatore è quello che conserva inalterate le misure effettuate. Questo equivale a dire che le stime sono tanto più accurate quanto più lento è il processo di decadimento, cioè quanto più alto è  $\alpha$ . Dai grafici appena presentati si osserva però che l'accuratezza delle stime impatta in maniera diversa sui due algoritmi con memoria.

L'algoritmo 2 presenta un comportamento auspicabile. Per tutte le funzioni di interferenza considerate, le prestazioni migliorano all'aumentare del fattore di smoothing tendendo al valore massimo di throughput ottenibile con  $\alpha=1$  (fig. 13). Viceversa l'algoritmo 1 disattende le previsioni, in quanto mostra un comportamento pressoché indipendente dal fattore di smoothing per le prime tre funzioni, mentre per la lineare si osserva un'imprevedibile andamento del throughput al variare di  $\alpha$  (fig. 14). Questo diverso comportamento è giustificabile in virtù della differente logica interna su cui poggiano i due algoritmi.

L'irregolarità dell'algoritmo con memoria 1 al variare del fattore di smoothing, che sarà messa meglio in evidenza nell'analisi dinamica, ci porta fin da ora ad escludere la sua possibile utilizzazione. Già nel caso statico, in presenza di funzione di interferenza lineare, non sarebbe in alcun modo possibile prevedere il valore di  $\alpha$  da utilizzare per massimizzare il throughput ed un errore nel dimensionamento di tale parametro porta a perdite non trascurabili. Per questo motivo si mostrano di seguito i confronti delle prestazioni soltanto tra l'algoritmo senza memoria e l'algoritmo con memoria 2. Per quest'ultimo algoritmo si fornisce il massimo valore di throughput conseguibile, cioè quello ottenuto in simulazioni con  $\alpha=1$ . I risultati dei due algoritmi sono a loro volta confrontati con il massimo throughput teoricamente ottenibile per valutare quanto nei due casi ci si avvicina al caso ideale. Quest'ultimo si otterrebbe quando, conoscendo a priori la funzione di interferenza, si fissa il

---

numero di RU per TS pari al numero ottimo per tale funzione. Il throughput ottenibile nel caso ideale è calcolato in maniera analitica moltiplicando il massimo throughput per TS dei grafici di figura 8, 9, 10 e 11 per il numero di TS (12) e dividendo per la durata del frame (10 ms).

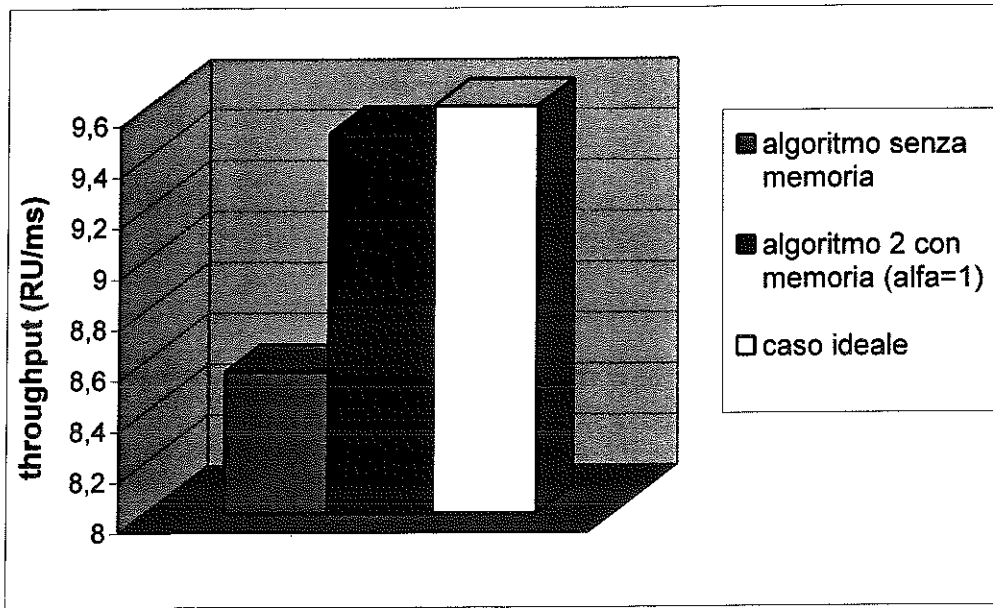


Figura 14. Confronto degli algoritmi nel caso statico con funzione di errore a gradino

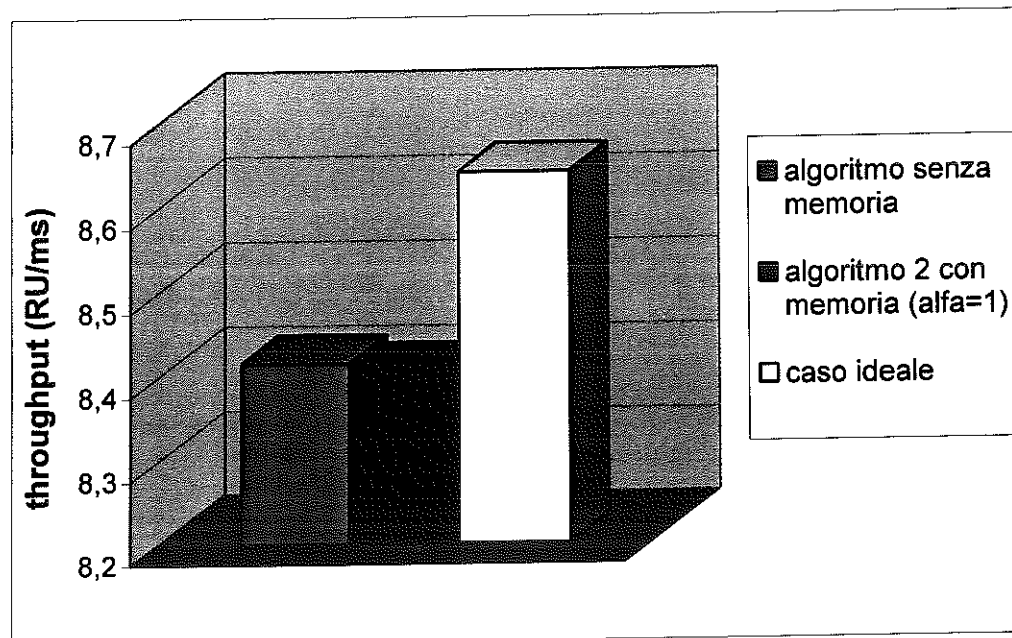


Figura 15 Confronto degli algoritmi nel caso statico con funzione di errore a sigmoide

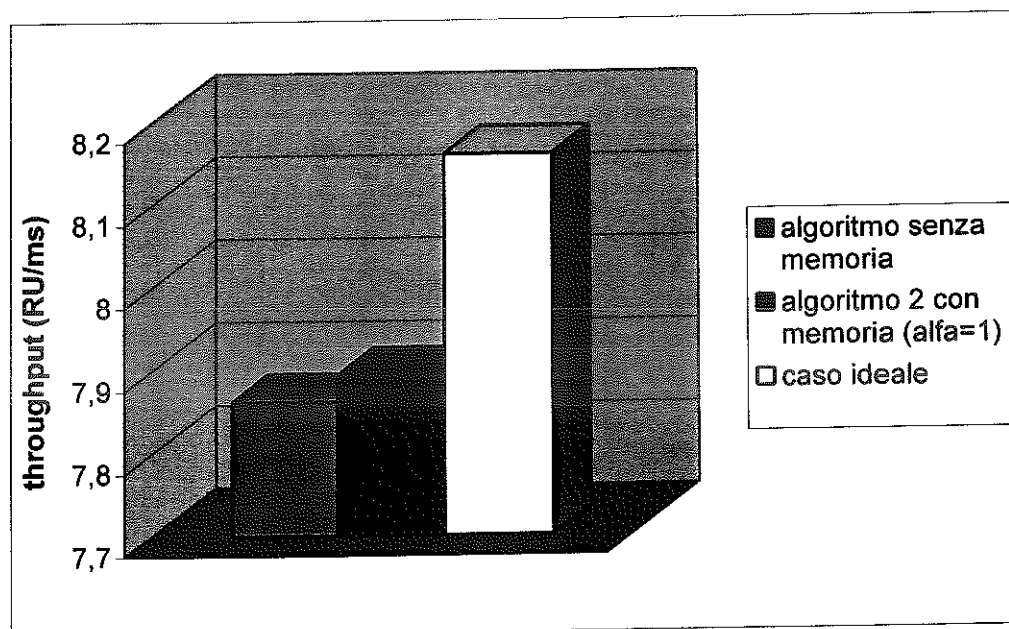


Figura 16. Confronto degli algoritmi nel caso statico con funzione di errore esponenziale troncata

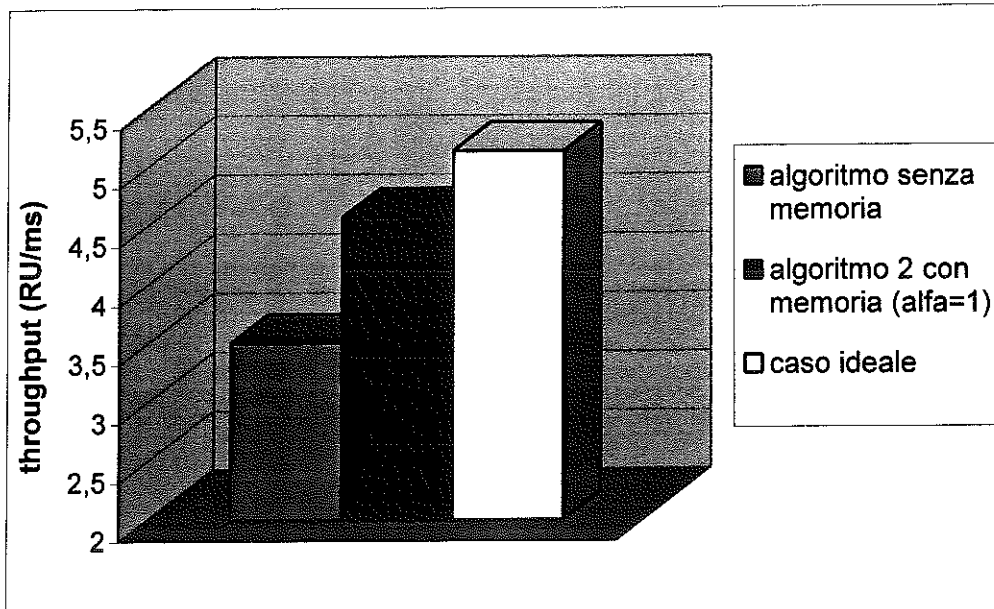


Figura 17. Confronto degli algoritmi nel caso statico con funzione di errore lineare

Dai grafici precedenti, ad eccezione del caso di funzione di errore con andamento a sigmoide, in cui le prestazioni dei due algoritmi sono quasi coincidenti, si nota sempre che il throughput conseguibile con l'algoritmo con memorizzazione delle misure è maggiore di quello privo di tale meccanismo.

## V.2.2 Analisi dinamica

Abbiamo appena mostrato come la classe di algoritmi proposti sia in grado funzionare, nel caso di canale statico, per diverse funzioni di interferenza. Vogliamo adesso considerare una situazione leggermente più realistica rispetto all'analisi statica, valutando le prestazioni dei tre algoritmi al variare nel tempo delle condizioni di interferenza del canale.

L'interferenza tra codici OVVSF allocati in uno stesso time slot deriva infatti dalle vicissitudini propagative del canale di ciascun utente. Data la mobilità di quest'ultimi, il canale radiomobile è tipicamente tempo variante, per cui le condizioni di interferenza cambiano nel tempo con un andamento imprevedibile. Si è quindi cercato di introdurre questo fenomeno nelle simulazioni per valutarne il relativo impatto sugli algoritmi.

La variazione del canale è stata modellata con una *catena di Markov* a due stati (*good* e *bad*). I tempi di permanenza nei due stati sono stati scelti con identica distribuzione esponenziale. Il tempo medio di permanenza di tale distribuzione è un parametro di ingresso del simulatore e, in quanto tale, viene fatto variare.

Nei due stati della catena di Markov vengono impostate due funzioni di interferenza con uguale andamento a sigmoide. Mentre la funzione dello stato *bad* presenta un massimo del throughput in corrispondenza di  $N=8$ , quella associata allo stato *good* è stata traslata in modo da spostare tale massimo in  $N=10$  ( $N$  rappresenta sempre il numero di RU per TS allocate). Di seguito sono mostrate le due funzioni di errore con il relativo andamento del throughput per time slot.

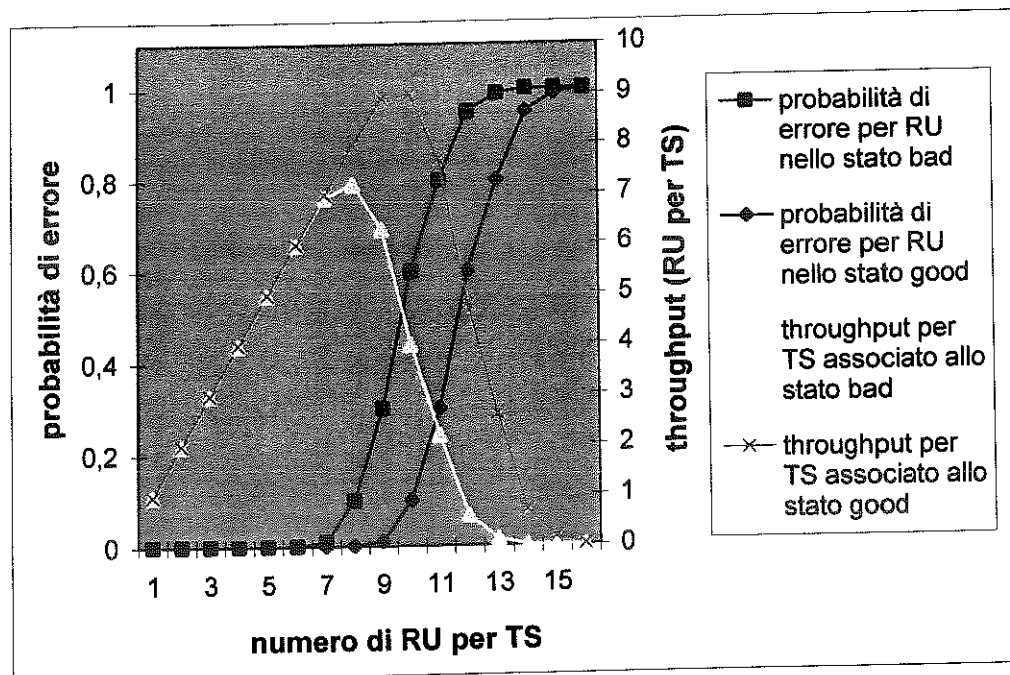


Figura 18. Funzioni di errore a sigmoide utilizzate nei due stati della catena di Markov e throughput per time slot associati

Analizziamo innanzitutto il comportamento dell'algoritmo semplificato senza memoria al variare della velocità di variazione del canale.

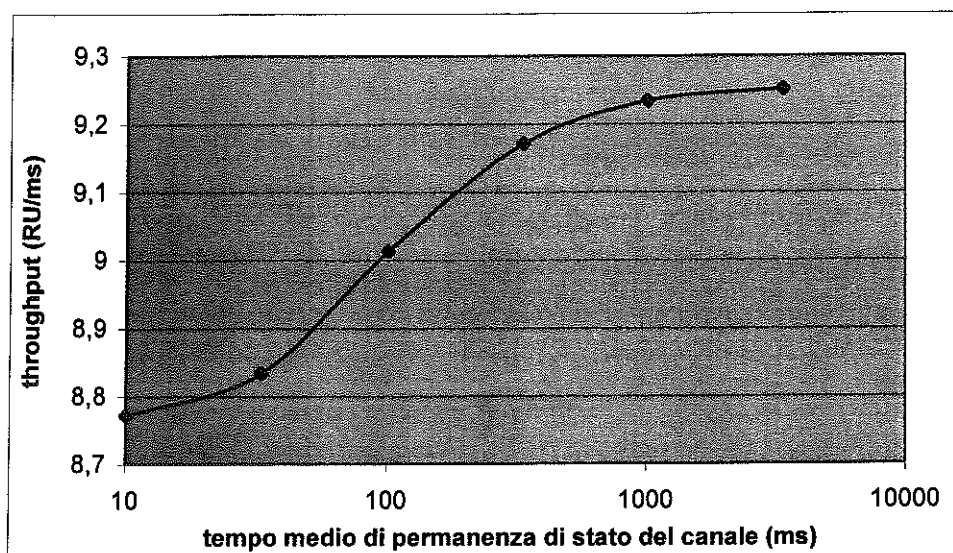


Figura 19 Comportamento dell'algoritmo senza memoria con funzioni di errore a sigmoide al variare della velocità di variazione della funzione di interferenza

Come è logico attendersi il throughput si abbassa all'aumentare della velocità di variazione del canale. Infatti tale algoritmo, non disponendo delle stime, non è stato pensato per adeguarsi a differenti velocità di variazione della funzione di interferenza. Una parte rilevante della perdita di throughput è dovuta ai transitori che si verificano al momento del salto tra stati. L'algoritmo senza memoria gestisce sempre allo stesso modo tali transitori a prescindere dalla velocità di variazione del canale e, conseguentemente, al diminuire del tempo medio di permanenza negli stati, aumentando il numero di transitori, si abbassa il throughput.

Come fatto in precedenza per l'analisi statica, passiamo ora a mostrare il comportamento dei due algoritmi con memoria al variare del fattore di smoothing. Questa volta l'andamento della funzione di interferenza rimane sempre lo stesso, ma vengono presentati differenti grafici al variare del tempo medio di permanenza nei due stati.

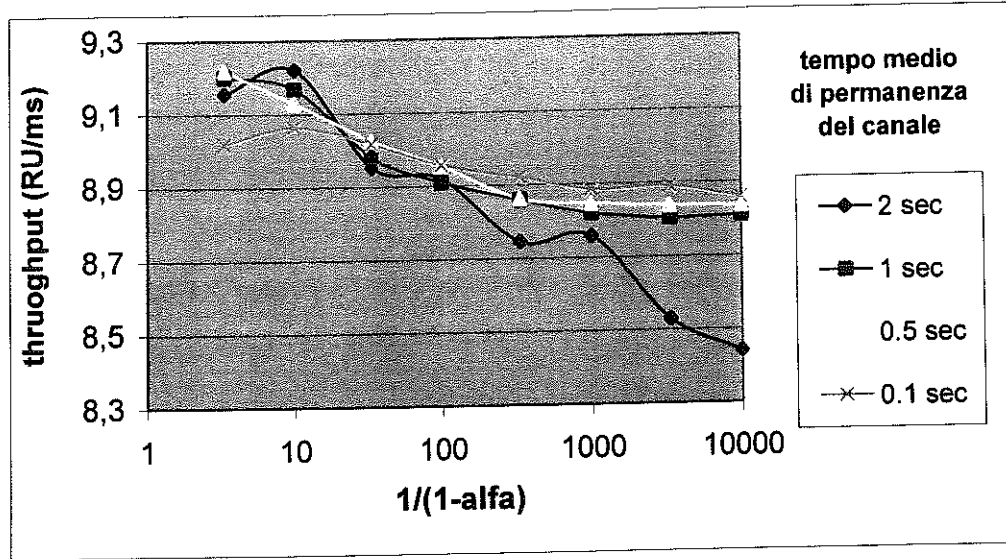


Figura 20. Comportamento dell'algoritmo con memoria di tipo 1 con funzioni di interferenza a sigmoide nel caso dinamico al variare del fattore di smoothing per varie velocità di variazione del canale

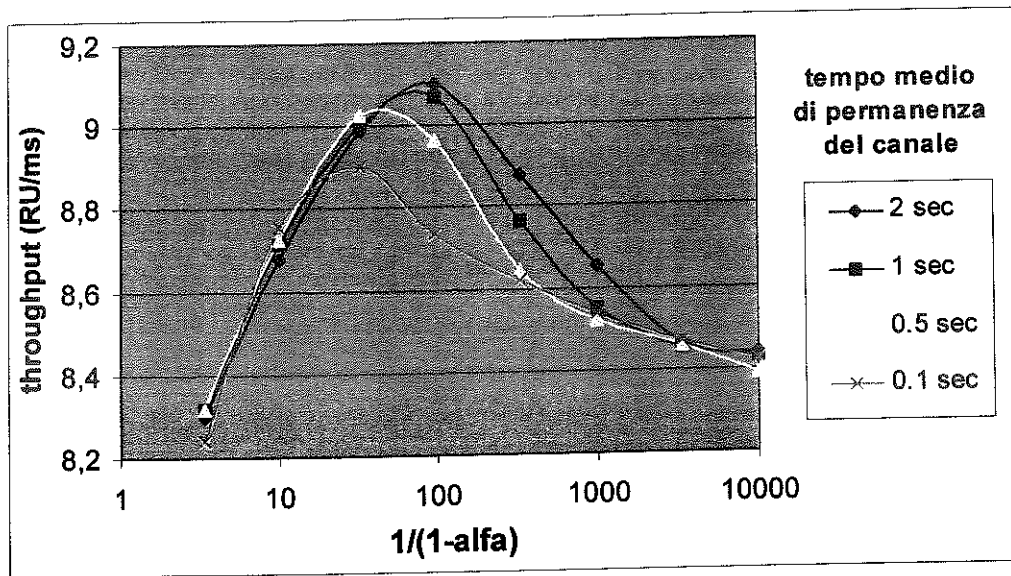


Figura 21. Comportamento dell'algoritmo con memoria di tipo 2 con funzione di interferenza a sigmoide nel caso dinamico al variare del fattore di smoothing per varie velocità di variazione del canale

Nel caso dinamico risulta ancor più evidente l'imprevedibilità di comportamento del primo algoritmo con memoria al variare del fattore di smoothing (fig. 20). Conoscendo la velocità di variazione della funzione di interferenza è pressoché impossibile trovare il valore ottimo di  $\alpha$ . Proprio la sua ingestibilità da questo punto di vista ne esclude quindi l'utilizzo a prescindere dalle prestazioni conseguibili.

Il comportamento del secondo algoritmo con memoria è invece del tutto prevedibile e rende possibile una gestione del fattore di smoothing al variare della velocità di cambiamento della funzione di interferenza.

Per un dato tempo medio di permanenza in uno stato esiste un valore di  $\alpha$  che massimizza il throughput (fig. 21). Tale valore consente di conservare in memoria la misura della probabilità di errore per un tempo che è strettamente legato al tempo medio di permanenza del canale. Se si scende sotto questo valore ottimo le prestazioni degradano, in quanto lo stimatore perde memoria troppo rapidamente, cioè le stime permangono in memoria per un tempo più breve del tempo di variazione del canale. Allo stesso modo, se si aumenta il fattore di smoothing al di sopra di tale valore, il throughput si abbassa, poiché lo stimatore conserva le misure in memoria per troppo tempo, il che equivale a dire che il tempo di decadimento delle misure è più lento della velocità di variazione del canale, per cui lo stimatore non riesce ad adattarsi con sufficiente velocità alle variazioni della funzione di interferenza.

Concludiamo le considerazioni sul grafico di figura 21 osservando come il massimo del throughput aumenti all'aumentare del tempo medio di permanenza negli stati. Anche questo risultato trova una facile giustificazione.

Il massimo di ciascuna funzione presentata si ottiene per un valore di  $\alpha$  ottimo. In corrispondenza di tale valore il tempo di memoria dello stimatore è perfettamente adattato alla velocità di variazione dell'interferenza, conseguentemente il transitorio tra due stati viene gestito in maniera ottimale a prescindere dalla velocità di variazione del canale. Poiché l'abbassamento del throughput è in gran parte dovuto alle perdite nel transitorio tra gli stati e poiché nel caso ottimo i transienti sono tutti pesati all'incirca allo stesso modo, ne

---

risulta che il throughput aumenta al diminuire del numero di transistori. A parità di durata della simulazione, all'aumentare del tempo di permanenza del canale diminuiscono i salti tra stati. Ecco spiegato l'aumento del throughput.

Per rendere più completa la nostra analisi dinamica si sono effettuate le stesse simulazioni, oltre che nel caso di funzioni a sigmoide, anche con funzioni di interferenza nei due stati entrambe a gradino, esponenziali troncate e lineari. Per uniformità si sono sempre considerate due funzioni per gli stati *bad* e *good* con i massimi del throughput rispettivamente a 8 e a 10.

Di seguito si presentano i risultati ottenuti per l'algoritmo con memoria 2 al variare del fattore di smoothing per diversi valori del tempo medio di permanenza del canale in uno stato. Le considerazioni sono analoghe a quelle svolte per il grafico di figura 21 nel caso di funzioni di interferenza a sigmoide. Dai grafici di figura 22, 23 e 24 sono stati poi ricavati i valori di  $\alpha$  ottimi e i relativi throughput che vengono infine utilizzati nelle tabelle conclusive per effettuare un confronto tra l'algoritmo senza memoria e quello con memoria di tipo 2.

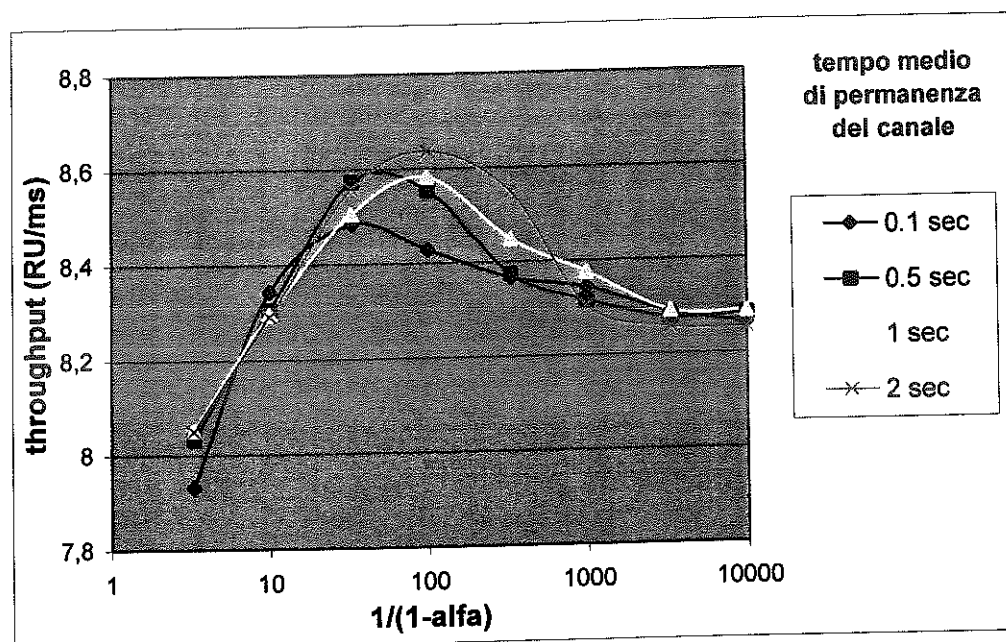


Figura 22. Comportamento dell'algoritmo con memoria di tipo 2 con funzioni di interferenza esponenziali troncate nel caso dinamico al variare del fattore di smoothing per varie velocità di variazione del canale

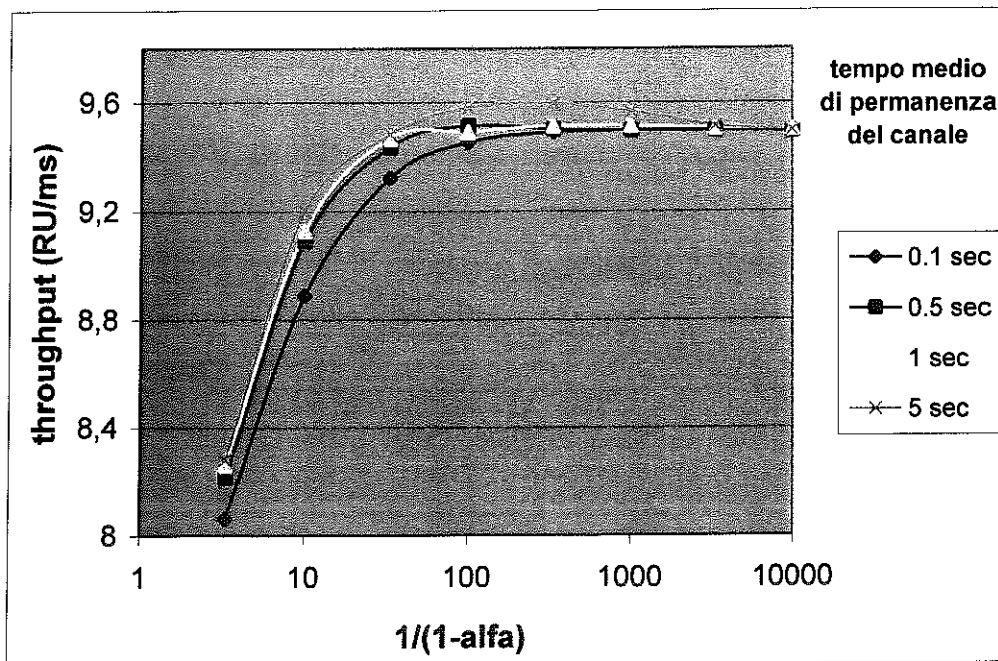


Figura 23. Comportamento dell'algoritmo con memoria di tipo 2 con funzioni di interferenza a gradino nel caso dinamico al variare del fattore di smoothing per varie velocità di variazione del canale

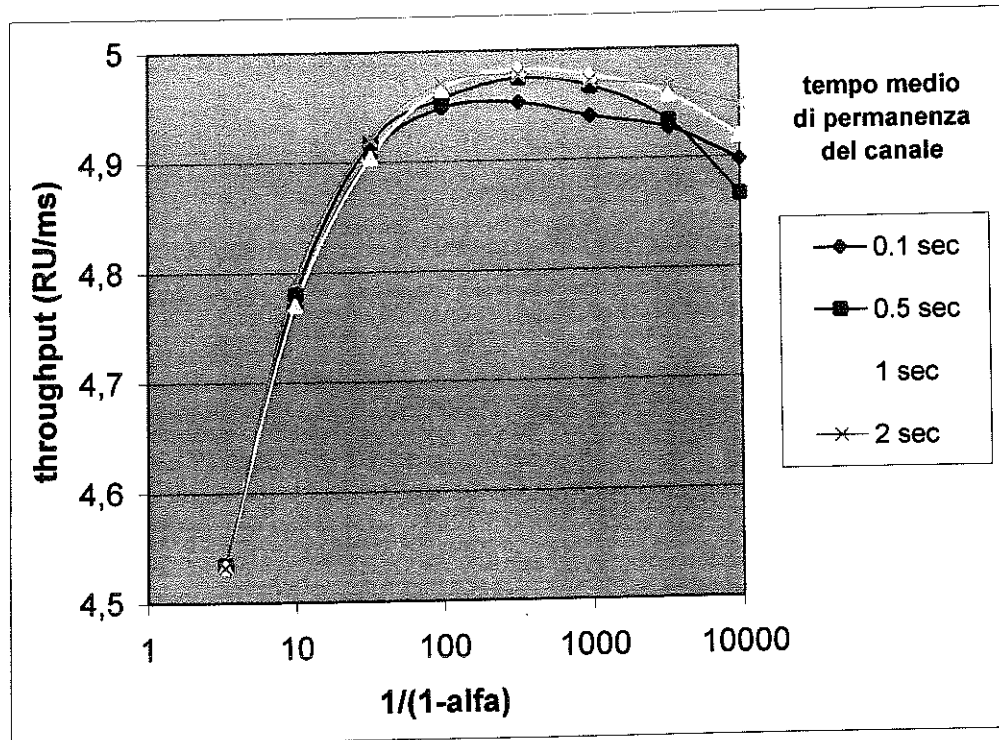


Figura 24. Comportamento dell'algoritmo con memoria di tipo 2 con funzioni di interferenza lineare nel caso dinamico al variare del fattore di smoothing per varie velocità di variazione del canale

Data l'irregolarità più volte osservata nel comportamento dell'algoritmo con memoria 1, di seguito è presentato soltanto il confronto dei throughput ottenuti con l'algoritmo senza memoria e quello con memoria 2, al variare del tempo medio di permanenza del canale negli stati. Per l'algoritmo con memoria si è supposto un corretto dimensionamento del valore di  $\alpha$ , per cui il throughput presentato è il massimo tra quelli ottenuti per i valori di  $\alpha$  provati (tale valore di  $\alpha_{\text{ottimo}}$  viene riportato a fianco del throughput).

Inoltre si è riportato anche il throughput ottenibile nel *caso ideale*. Quest'ultimo si verifica in presenza di un cambiamento *istantaneo* del numero di RU per TS ad ogni salto di stato. Il valore del throughput ideale è stato calcolato analiticamente considerando la media tra il throughput ottenibile separatamente per le due funzioni interferenza nel caso statico (osservando il tempo complessivo di permanenza nei due stati per ogni simulazione, si può ritenere sufficientemente accurata l'ipotesi di equa ripartizione).

tempo medio di permanenza in uno stato (sec)	algoritmo senza memoria	algoritmo con memoria		caso ideale
	throughput (RU/ms)	$\alpha_{ottimo}$	throughput (RU/ms)	throughput (RU/ms)
0,1	8,79847	0,99	9,09841	10,8
0,5	8,77299	0,99	9,51335	10,8
1	8,81181	0,99	9,51433	10,8
2	8,85818	0,997	9,59784	10,8

Tabella 2. Confronto dinamico degli algoritmi per funzioni di interferenza a gradino

tempo medio di permanenza in uno stato (sec)	algoritmo senza memoria	algoritmo con memoria		caso ideale
	throughput (RU/ms)	$\alpha_{ottimo}$	throughput (RU/ms)	throughput (RU/ms)
0,1	9,01332	0,97	8,90063	9,78
0,5	9,19033	0,97	9,02578	9,78
1	9,23363	0,99	9,06691	9,78
2	9,25673	0,99	9,09841	9,78

Tabella 3. Confronto dinamico degli algoritmi per funzioni di interferenza a sigmoide

tempo medio di permanenza in uno stato (sec)	algoritmo senza memoria	algoritmo con memoria		caso ideale
	throughput (RU/ms)	$\alpha_{ottimo}$	throughput (RU/ms)	throughput (RU/ms)
0,1	8,48187	0,97	8,4829	9,18
0,5	8,52139	0,97	8,57377	9,18
1	8,53585	0,99	8,58175	9,18
2	8,56928	0,99	8,636	9,18

Tabella 4. Confronto dinamico degli algoritmi per funzioni di interferenza esponenziali troncate

tempo medio di permanenza in uno stato (sec)	algoritmo senza memoria	algoritmo con memoria		caso ideale
	throughput (RU/ms)	$\alpha_{ottimo}$	throughput (RU/ms)	throughput (RU/ms)
0,1	3,49281	0,997	4,95242	5,556
0,5	3,49116	0,997	4,97459	5,556
1	3,49334	0,997	4,98246	5,556
2	3,49266	0,997	4,97778	5,556

Tabella 5. Confronto dinamico degli algoritmi per funzioni di interferenza lineare

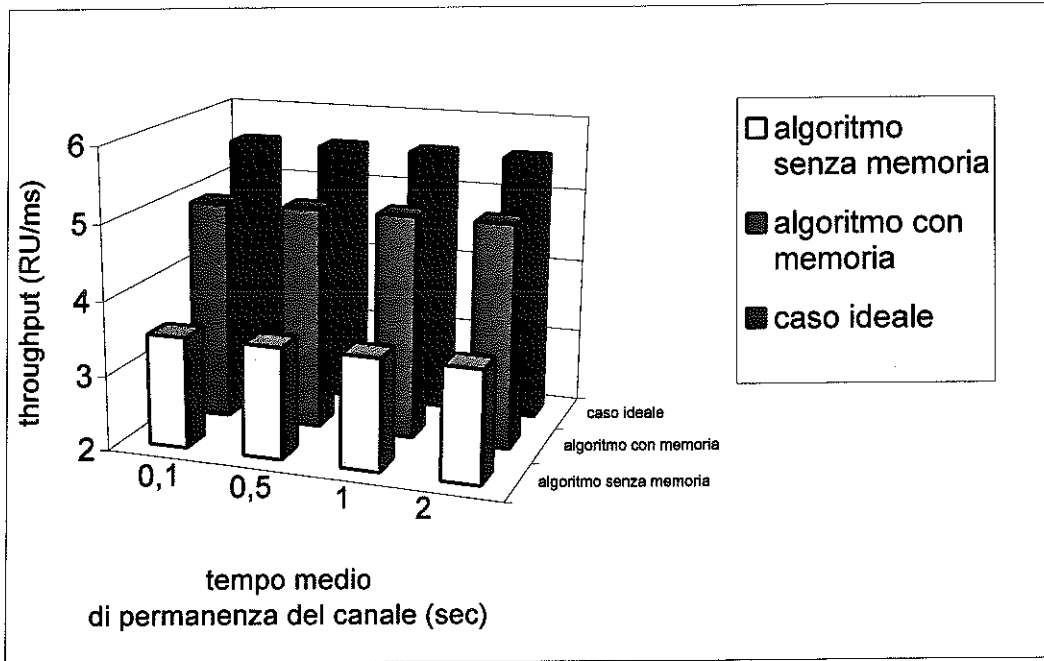


Figura 25. Confronto dei throughput ottenuti con gli algoritmi adattivi con e senza memoria al variare del tempo medio di permanenza in uno stato del canale con funzioni di interferenza lineare

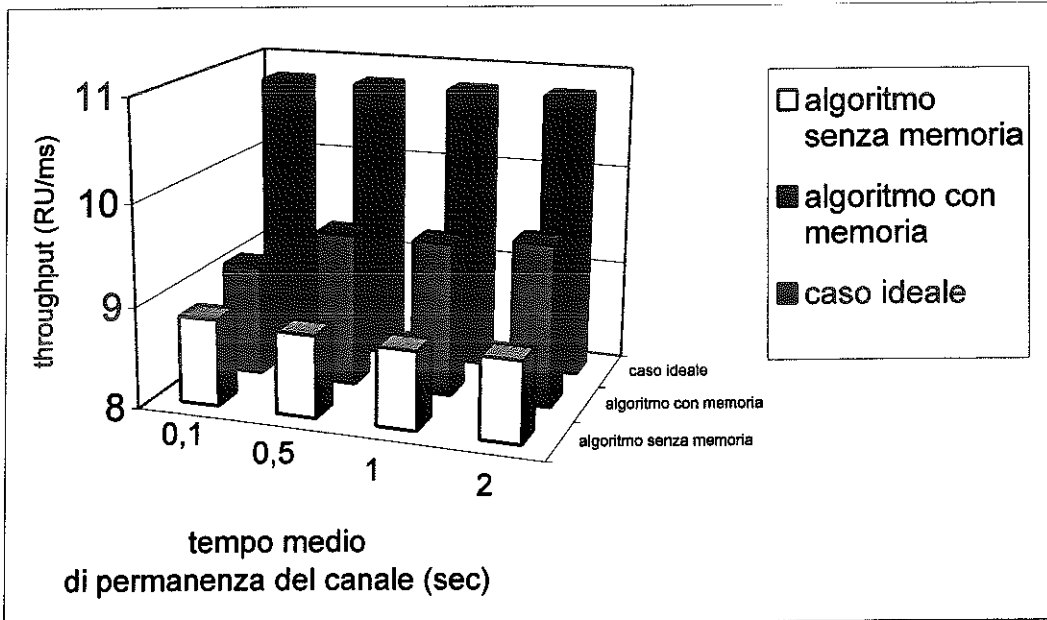


Figura 26 Confronto dei throughput ottenuti con gli algoritmi adattivi con e senza memoria al variare del tempo medio di permanenza del canale in uno stato con funzioni di interferenza a gradino

---

# CAPITOLO

# 6

---

## CONCLUSIONI

### VI.1 Proposta di algoritmi di scheduling

L'obiettivo di questo report è stato proporre una tecnica di *scheduling* per la trasmissione dati tramite *Internet* sulla rete di accesso *UMTS* in modalità *TDD*. Si è cercato, rispettando la massima *fairness* tra gli utenti, di fornire all'algoritmo presentato una doppia adattabilità. Da una lato si è tentato di adattare la ripartizione delle risorse radio a variazioni del bilanciamento del traffico nelle due direzioni di trasmissione, dall'altro di gestire la capacità trasmissiva adattandola a variazioni delle condizioni interferenziali del sistema.

Gli algoritmi proposti sono finalizzati a migliorare l'efficienza nell'utilizzazione delle risorse radio messe a disposizione per i sistemi radiomobili di terza generazione. Come più volte osservato, l'efficienza nello sfruttamento delle risorse radio è uno degli aspetti cruciali per l'introduzione di nuovi servizi ad elevati bit rate e deriva direttamente dall'efficienza della tecnica di accesso multiplo e degli algoritmi di scheduling adottati.

In quest'ottica si è studiata la tecnica di accesso *TD-CDMA*, proposta insieme alla *W-CDMA* per i sistemi wireless di terza generazione. Come visto nel capitolo 2, l'interfaccia

essere applicati anche in un contesto più realistico e cioè in presenza di diverse classi di utenti con differenti probabilità di errore. Se la probabilità di errore dovuta all'interferenza intracella può a buon diritto ritenersi uniforme su tutti gli utenti, altrettanto non può dirsi per la probabilità di errore dovuta al canale di propagazione. Infatti, caratteristica tipica del canale radiomobile è che ciascun utente vede un proprio canale di propagazione e il segnale trasmesso è ricevuto con un rapporto segnale rumore specifico di quel canale. La probabilità di errore per RU misurata dalla stazione base dipende quindi dalla somma di due effetti distinti, separabili solo teoricamente: l'interferenza intracella e le condizioni del canale di propagazione. Volendo tenere in conto questo effetto nei nostri algoritmi è sufficiente effettuare una preventiva divisione degli utenti in classi a seconda della loro probabilità di errore. Ripartito il numero dei time slot in maniera proporzionale al numero di utenti attivi per ogni classe, esattamente come effettuato per l'up e il down link, è sufficiente applicare l'algoritmo di determinazione del numero ottimo di RU per TS separatamente ad ogni classe. Si tenga presente che questa divisione è già stata implementata nel nostro simulatore nel caso particolare delle due classi di up e down link. Infatti prevedendo una diversa funzione di interferenza per le due direzioni di traffico, in virtù del fatto che i segnali in down link, al contrario di quelli in up link, sono sicuramente sincronizzati, una volta divisi i time slot per le due tratte, l'algoritmo di determinazione delle RU per TS viene applicato separatamente, esattamente come anticipato precedentemente.

Ipotizzando di riuscire in maniera perfetta a dividere gli utenti in classi con probabilità di errore uniforme si produce un effetto molto significativo anche in termini di *fairness*. Applicando infatti l'algoritmo adattivo di determinazione delle RU per TS separatamente ad ogni classe, si ottiene che utenti in condizioni propagative particolarmente critiche e quindi con probabilità di errore più alta vengono allocati in time slot con meno RU, viceversa per classi di utenza in buone condizioni di propagazione viene aumentato il numero di RU per TS. In questo modo oltre all'obiettivo di massimizzare il throughput per ciascuna classe, si realizza un trattamento più equo degli utenti in termini di probabilità di errore attraverso un bilanciamento tra rapporto segnale-rumore e rapporto segnale-interferenza.

---

---

Per inciso si osserva che la divisione degli utenti in classe è in parte già prevista nel sistema UMTS attraverso l'utilizzo di una struttura di celle sovrapposte di dimensioni diverse che utilizzano frequenze differenti. Questa organizzazione gerarchica, detta *Hyerarchical Cell Structure* e presentata nel paragrafo I.5.5, tende ad allocare gli utenti all'interno delle celle in base alla loro velocità di spostamento. Utenti veloci vengono serviti da celle più grandi in modo da diminuire la probabilità di *handover*. In questo modo si ha una prima classificazione degli utenti in base alle caratteristiche del loro canale trasmissivo (utenti particolarmente veloci sono caratterizzati da condizioni di propagazione più critiche).

Gli algoritmi proposti nel capitolo 3 per calcolare il numero ottimo di RU per TS sono tre. Il primo è un algoritmo senza memoria che utilizza soltanto la misura di probabilità di errore calcolata nell'ultimo frame. Per completare le condizioni di cambiamento del numero di RU per TS ci si mette in due casi limiti: un *best case* per la condizione di discesa ed un *worst case* per quella di salita.

Questa semplificazione viene superata nei due algoritmi con memoria nei quali si utilizza un vettore di memorizzazione delle misure di probabilità di errore per tutti i valori di RU per TS utilizzati nei frame precedenti. Il primo algoritmo con memoria è una naturale evoluzione di quello senza e ad esso si riconduce nel caso di completa perdita di memoria, mentre il secondo sottende una logica euristica leggermente diversa. In quest'ultimo infatti oltre l'eliminazione della perdita accumulata utilizzata nella condizione in salita, viene modificata anche la legge di aggiornamento del vettore delle stime della probabilità di errore. Come visto nel capitolo precedente queste differenze portano cambiamenti radicali nelle prestazioni degli algoritmi.

Come mostrato chiaramente nel capitolo precedente, il comportamento degli algoritmi con memoria dipende strettamente dal valore del fattore di *smoothing* utilizzato ( $\alpha$ ), parametro legato alla velocità di cancellazione nel tempo delle misure. Dalle analisi simulative effettuate, il primo algoritmo con memoria, sebbene presenti delle soddisfacenti prestazioni in termini assoluti, risulta essere non gestibile dal punto di vista del dimensionamento del fattore di *smoothing*. L'analisi statica e ancor più quella dinamica

---

mostrano un comportamento imprevedibile delle prestazioni dell'algoritmo al variare  $\alpha$ . Per tale ragione esso non può efficientemente essere implementato, in quanto risulta impossibile sviluppare un criterio per il dimensionamento del fattore di smoothing. Differente è il caso del secondo algoritmo con memoria che presenta un comportamento del tutto giustificabile e prevedibile. Nell'analisi statica, per tutte le funzioni di interferenza considerate (gradino, sigmoide, esponenziale troncato, lineare), le prestazioni migliorano al tendere di  $\alpha$  all'unità. In questa analisi infatti non viene fatta variare la funzione di interferenza nel corso della simulazione, per cui è facile intuire che le prestazioni migliorano all'aumentare del tempo di memorizzazione delle misure. Nell'analisi dinamica, la variazione della funzione di interferenza nel corso della simulazione rende più problematico il dimensionamento del fattore di smoothing. Il comportamento dell'algoritmo è comunque anche in questo caso regolare e giustificabile. Per tutte le funzioni di interferenza e le velocità di variazione del canale considerate le prestazioni migliorano all'aumentare di  $\alpha$  fino ad un valore ottimo ( $\alpha_{\text{ottimo}}$ ) per poi degradare. Il valore ottimo dipende dal tipo di funzione di interferenza e dalla velocità di variazione del canale.

A scapito di un leggero aumento della complessità implementativa, il secondo algoritmo con memoria presenta delle prestazioni in generale migliori dell'algoritmo senza memoria. La caratteristica di quest'ultimo, sia nel caso statico che dinamico, è quella di presentare delle prestazioni accettabili per alcuni tipi di funzioni di interferenza (in uno dei casi analizzati, la sigmoide, esso raggiunge dei throughput addirittura leggermente più alti dell'algoritmo con memoria), ma di entrare in crisi per funzioni di interferenza particolari come il gradino o la lineare. Viceversa l'algoritmo con memoria presenta delle prestazioni più stabili al variare del tipo di funzione di interferenza. Nelle tabelle e nei grafici conclusivi del capitolo precedente si mostra che l'algoritmo con memoria è quasi sempre migliore di quello senza e, sebbene venga leggermente superato dalle prestazioni di quest'ultimo in uno dai quattro casi considerati, presenta il fondamentale vantaggio di non entrare in crisi per particolari funzioni di interferenza.

---

Non potendo conoscere a priori il tipo di funzione di interferenza, l'analisi condotta ha cercato di esaminare una casistica più ampia possibile, in modo da avvalorare le conclusioni per la più grande classe di funzioni. Da tale analisi è emerso che il secondo algoritmo con memoria è quello a nostro avviso più efficiente, perché, al contrario dell'altro algoritmo con memoria, presenta un comportamento regolare e prevedibile al variare del fattore di smoothing e, al contrario dell'algoritmo senza memoria, mostra una buona efficienza indipendentemente dal tipo di funzione di interferenza considerata.

Si tenga conto in ultimo che il confronto tra l'algoritmo senza memoria e il secondo algoritmo con memoria presentato in conclusione del capitolo precedente, pende a favore dell'algoritmo con memoria ipotizzando un corretto dimensionamento del parametro  $\alpha$ . Resta da considerare quindi il problema della scelta del fattore di smoothing per l'algoritmo con memoria. Tale problema lascia spazio ad ulteriori studi e viene brevemente esaminato nel paragrafo successivo. Per quel che riguarda la nostra analisi è sufficiente in questa sede osservare che, sebbene il valore ottimo di  $\alpha$  dipenda, come è logico che sia, e dal tipo di funzioni considerate, e dalla velocità di variazione del canale, per tutta la casistica considerata i valori ottimi del fattore di smoothing si mantengono sempre all'interno di un range relativamente stretto (da 0.997 a 0.9997). Ciò rende non particolarmente critico il dimensionamento di tale parametro.

## VI.2 Ulteriori studi

La vastità e l'originalità del contesto affrontato in questa report lasciano amplissimo spazio ad ulteriori spazi di ricerca. Oltre ai numerosi problemi ancora aperti analizzati durante lo studio delle specifiche, sarebbe di notevole interesse proseguire il cammino intrapreso in questo lavoro.

Resta, per esempio, da analizzare il problema del dimensionamento del fattore di *smoothing* per l'algoritmo con memoria. Come già osservato la scelta del valore ottimo di  $\alpha$  va effettuata in base al tipo di funzione di interferenza e alla velocità di variazione del canale. Entrambi questi fattori non sono conoscibili a priori e variano dinamicamente.

---

Anche in questo caso quindi l'unica strada percorribile è quella di studiare un algoritmo adattivo che sia in grado di variare il valore di  $\alpha$  in funzione delle caratteristiche della funzione di interferenza. Tipicamente questo tipo di problemi viene affrontato in letteratura attraverso algoritmi cosiddetti di *exponential backoff*. Nel nostro caso si tratta di valutare dalla casistica studiata il valore ottimo minimo assunto da  $\alpha$  nel range di interesse della velocità di variazione del canale. Per ogni frame per cui si osserva una invarianza dell'interferenza si tende a far aumentare  $\alpha$ , per poi riportarlo al valore minimo iniziale qualora si osservi una variazione delle condizioni di interferenza. Così facendo si tende ad adattare il fattore di smoothing alla velocità di variazione del canale, in quanto, come si è già avuto modo di osservare, al diminuire della velocità del canale il valore di  $\alpha$  ottimo tende a spostarsi verso l'unità (valore ideale nel caso di funzione di interferenza statica).

Uno dei problemi considerati nell'implementazione del simulatore della rete di accesso e di cui però non si è avuto modo di valutare l'impatto sulle prestazioni del sistema è la tecnica di rilascio del canale. L'efficienza di tale tecnica è critica in up link, in quanto rilasciare troppo spesso il canale significa aumentare le richieste di riattivazione e quindi congestionare il canale di accesso a contesa (RACH), con conseguente aumento della probabilità di collisione. La nostra scelta è stata di effettuare il rilascio del canale non ad avvenuta trasmissione di ogni pacchetto IP (come spesso proposto da lavori precedenti), ma solo al momento in cui il buffer di trasmissione dell'utente si è svuotato. Tale possibilità è effettivamente realizzabile e prevista nelle specifiche. L'interessante confronto delle prestazioni delle due tecniche in termini di collisioni sul RACH, che richiede un preventivo aumento del numero di utenti in modo da rendere statisticamente significativo il numero di collisioni sul RACH, viene lasciata ad ulteriori studi.

---

## Lista degli acronimi

3GPP	3 <sup>rd</sup> Generation Partnership Project	MS	Mobile Station
ADC	American Digital Cellular	MSC	Mobile Switching Center
AMD	Acknowledged Mode Data	MUD	MultiUser Detection
AMPS	Advanced Mobile Phone Service	NMT	Nordic Mobile Telephone
ARIB	Association for Radio Industry and Business	NRT	Non Real Time service
AWGN	Additive White Gaussian Noise	OFDM	Orthogonal Frequency Division Multilexing
BCCH	Broadcast Control Channel	OL	Offered Load
BCH	Broadcast Channel	OSI	Open System Interconnection
BER	Bit Error Rate	OVSF	Orthogonal Variable Spreading Factor
BS	Base Station	PCCH	Paging Control Channel
CC	Call Control	P-CCPCH	Primary - Common Control Physical Channel
CCCH	Common Control Channel	PCH	Paging Channel
CDMA	Code Division Multiple Access	PDSCH	Physical Downlink Shared Channel
CN	Core Network	PDU	Protocol Data Unit
CV	Coefficiente di Variazione	PRACH	Physical Random Access Channel
DCA	Dynamic Channel Allocation	PSCH	Physical Synchronization Channel
DCCH	Dedicated Control Channel	PSTN	Public Switching Telephone network
DCH	Dedicated Channel	PUSCH	Physical Uplink Shared Channel
DPCH	Dedicated Physical Channel	QoS	Quality of Service
DS	Direct Sequence	QPSK	Quadrature Phase Shift Keying
DSCH	Downlink Shared Channel	RACH	Random Access Channel
DTCH	Dedicated Traffic Channel	RAN	Radio Access Network
ETSI	European Telecommunication Standard Institute	RLC	Radio Link Control
FACH	Forward Access Channel	RNC	Radio Network Controller
FDD	Frequency Division Duplex	RNS	Radio Network Subsystem
FDMA	Frequency Division Multiple Access	RRC	Radio Resource Control
FEC	Forward Error Correction	RT	Real Time service
FH	Frequency Hopping	RU	Risorsa Unitaria
FPLMTS	Future Public Land Mobile Telephone System	SAP	Service Access Point
GPRS	General Packet Radio Service	S-CCCH	Secondary - Common Control Physical Channel
GSM	Global System for Mobile Communication	SCCH	Synchronization Control Channel
GSN	GPRS Support Node	SCH	Synchronization Channel
HCS	Hyerarchical Cell Structure	SDU	Service Data unit
IMT-2000	International Mobile Telecommunication	TCP	Transmission Control Protocol
IP	Internet Protocol	TDD	Time Division Duplex
ISDN	Integrated Service Digital Network	TDM	Time Division Multiplexing
ITU	International Telecommunication Union	TIA	Telecommunications Industry Association
JDC	Japanese Digital Cellular	TPC	Transmission Power Control
LMMSE	Linear Minimum Mean Square Error	TS	Time Slot
MAC	Medium Access Control	TTA	Telecommunication Technology Association
MM	Mobility Management	UDD	Unconstrained Delay Data

UMD	Unacknowledged Mode Data
UMTS	Universal Mobile Telecommunication System
USCH	Uplink Shared Channel
UTRAN	UMTS Terrestrial Radio Access Network
WAP	Wireless Application Protocol
W-CDMA	Wideband – Code Division Multiple Access

---

## Bibliografia

---

- [1] Recommendation ITU-R M.1035, "*Framework for radio interfaces and radio sub-system functionality for international mobile telecommunication-2000 (IMT-2000)*", 1994
- [2] 3G TS 25.301 "*Radio Interface Protocol Architecture*", 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Radio Access Network, version 3.4.0 (2000-03)
- [3] 3G TS 25.221 "*Physical channels and mapping of transport channels onto physical channels (TDD)*", 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Radio Access Network, version 3.1.1 (1999-12)
- [4] 3G TS 25.222 "*Multiplexing and channel coding (TDD)*", 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Radio Acces Network, version 3.1.1 (1999-12)

- [5] 3G TS 25.223 “*Spreading and Modulation (TDD)*”, 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Radio Access Network, version 3.1.1 (1999-12)
- [6] 3G TS 25.224 “*Physical Layer Procedures (TDD)*”, 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Radio Access Network, version 3.1.1 (1999-12)
- [7] TS 25.321 “*MAC protocol specification*”, 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Radio Access Network, Working Group 2, version 2.0.0 (1999-04)
- [8] TS S2.22 “*Description of the RLC Protocol*”, 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Radio Access Network, Working Group 2, version 2.0.0 (1999-04)
- [9] 3G TR 25.922 “*Radio Resource Management Strategies*”, 3<sup>rd</sup> Generation Partnership Project, Technical Specification Group Radio Access Network, version 0.1.1 (1999-04)
- [10] a cura di F. Muratore, “*Le comunicazioni Mobili del futuro – UMTS: il nuovo sistema del 2001*”, CSELT, gennaio 2000
- [11] R. Reggianini, “*Appunti di Sistemi di Radiocomunicazione*”, SEU, marzo 1998
- [12] R. Prasad, T. Ojampera, “*An overview of CDMA evolution toward wideband CDMA*”, IEEE Communication Surveys, fourth quarter 1998
- [13] J. Yang, I. Koo, Y. Choi, K. Kim, “*A dynamic resource allocation scheme to maximize throughput in a multimedia CDMA system*”, IEEE Proc. VTC, pp. 348-351, 1999
- [14] B. Bae, D. Cho, “*An integrated scheduling algorithm in a wireless packet network with dynamic TDD*”, pp. 497-501, 1999
- [15] Anne-Gaele Acx, “*Channel allocation for the UTRA TDD mode*”, FRAMES Workshop, gennaio 1999

- [16] A. G. Acx, C. Mihailescu, L. Jorguleski, J. Punt, M. Berg, S. Petersson, F. Raji, P. Slanina, "*Radio resource management algorithms and layer 2 Protocols for the UTRA TDD mode*",
- [17] H. Zhang, "*Service disciplines for guaranteed performance in packet-switching networks*", Proceeding of the IEEE, vol. 83, no. 10, pp.1374-1396, ottobre 1995
- [18] Y. Lu, R. W. Brodersen, "*Integrated power control, error correction coding and scheduling for a CDMA downlink system*", IEEE journal on selected areas in communications, vol. 17, no. 5, pp. 978-990, giugno 1999
- [19] G. Ostermayer, P. Slanina, C. Holzl, C. Mecklenbrauker, F. Raji, T. Stadler, "*Scheduling Algorithm for UTRA TDD Mode*", proc. Eurocomm 2000, Monaco giugno 2000
- [20] P. Slanina, F. Raji, C. Mecklenbrauker, C. Holzl, G. Ostermayer, T. Stadler, t. Gruhn, "*Throughput of Hybrid ARQ Types for UTRA TDD Mode*", proc. Eurocomm 2000, Monaco giugno 2000
- [21] M. Haardt, A. Klein, R. Koehn, S. Oestreich, M. Purst, V. Sommer, T. Ulrich, "*The TD-CDMA Based UTRA TDD Mode*", IEEE Journal on Selected Areas in Communications, agosto 2000, Volume 18, Numero 8, pagg.1375-1385
- [22] H. Holma, S. Heikkinen, O.-A. Lehtinen, A. Toskala, "*Interference Considerations for the Time Division Duplex Mode of the UMTS Terrestrial Radio Access*", IEEE Journal on Selected Areas in Communications, agosto 2000, Volume 18, Numero 8, pagg.1386-1393
- [23] R. F. Gordon, P. G. Loewner, E. A. MacNair, "The Research Queueing Package Version 3. Language Reference Manual", IBM Thomas J. Watson Research Center
- [24] C. H. Sauer, "*Simulation of Computer Communication Systems*", Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1983
- [25] A. M. Law, W. David Kelton, "*Simulation Modeling & Analysis*", Mc Graw-Hill International Edition 1991

