

POLITECNICO DI MILANO

V Facoltà di Ingegneria

Corso di Laurea Specialistica in Ingegneria delle Telecomunicazioni

Dipartimento di Elettronica e Informazione



# Valutazione delle prestazioni di tecniche di inoltro collaborativo basata su modelli di ottimizzazione

Relatore: professor Antonio Capone

Correlatore: Stefano Gualandi

Tesi di Laurea Specialistica di:

Luca Cerri, matricola 734596

Matteo Tognoli, matricola 739745

Anno Accademico 2009-2010

# Ringraziamenti

Un primo ringraziamento va sicuramente al professor Antonio Capone per gli insegnamenti ricevuti, per averci proposto questo argomento di grande attualità e interesse su cui svolgere la nostra tesi e per aver trovato tempo da dedicarci nonostante i molti impegni.

Grazie infinite a Stefano Gualandi che ci ha sempre seguito con pazienza in tutto il nostro percorso sia nella parte progettuale che in quella di stesura dell'elaborato. Con grande disponibilità ci ha fornito consigli e aiutato nelle difficoltà incontrate.

Un enorme grazie va ai nostri genitori e familiari che ci hanno sopportato e supportato con grande costanza in questi anni di università e soprattutto durante questi mesi finali in cui abbiamo sfidato la loro infinita pazienza. Sempre pronti ad aiutarci e incoraggiarci nei momenti difficili ed orgogliosi dei nostri passi, hanno saputo infonderci grande fiducia e dimostrarci il loro incondizionato affetto.

Rivolgiamo dovuti ringraziamenti ai nostri compagni di sventura che hanno condiviso con noi questi cinque anni densi di stress ed esami, hanno combattuto al nostro fianco nelle battaglie più dure e nonostante la nostra instabilità durante lo studio non ci hanno mai abbandonato. In questo poco tempo siete diventati dei veri e cari amici. Per quelli di voi che ancora sono alle prese con gli esami: Tenete duro!

Esprimiamo due diversi tipi di gratitudine ad Alessia: da Luca per l'affetto, la comprensione e la fiducia dimostratemi in questi quattro anni passati insieme, e da Matteo per aver sopportato gli 'scleri' di Luca anche quando erano causati da me.

Infine non può mancare un grazie sincero agli amici di una vita, sempre presenti nei momenti belli e in quelli grigi, a festeggiare con noi e a tirarci su il morale se necessario, dimostrando che su di loro si può sempre contare.

Grazie a tutti, Luca & Matteo.

# Indice

<b>Introduzione</b>	<b>9</b>
<b>1 La collaborazione nelle reti wireless</b>	<b>12</b>
1.1 Reti wireless mesh . . . . .	13
1.2 Reti ad hoc . . . . .	15
1.3 Meccanismi collaborativi . . . . .	18
1.3.1 Il canale radio e la diversità spaziale . . . . .	18
1.3.2 La collaborazione a livello fisico . . . . .	22
1.4 Accesso multiplo e scheduling . . . . .	25
<b>2 Ottimizzazione delle risorse radio</b>	<b>28</b>
2.1 Stato dell'arte . . . . .	30
2.2 Generazione di colonne . . . . .	34
2.3 Algoritmi euristici . . . . .	37
2.4 Strumenti utilizzati . . . . .	38
<b>3 Modelli ed algoritmi per l'ottimizzazione della collaborazione</b>	<b>40</b>
3.1 La rete . . . . .	40
3.2 Applicazione della generazione di colonne . . . . .	41
3.3 Il pricing . . . . .	42
3.3.1 Pricing con SINR . . . . .	43
3.3.2 Pricing protocollare . . . . .	44
3.4 Pricing con collaborazione . . . . .	46
3.4.1 Pricing SINR con collaborazione . . . . .	48

3.4.2	Pricing protocollare con collaborazione . . . . .	51
3.4.3	Accorgimenti per il pricing protocollare . . . . .	53
3.4.4	Euristica con cammini minimi . . . . .	54
3.5	Esempio di collaborazione . . . . .	56
3.6	Pricing con potenza variabile . . . . .	58
3.6.1	Pricing SINR con collaborazione e potenza variabile .	58
3.6.2	Pricing protocollare con collaborazione e potenza variabile . . . . .	61
3.6.3	Euristica con cammini minimi a potenza variabile . .	62
3.7	Esempio di controllo di potenza . . . . .	62
<b>4</b>	<b>Risultati numerici e confronti</b>	<b>64</b>
4.1	Condizioni di test . . . . .	64
4.2	Modelli SINR a potenza variabile . . . . .	65
4.3	Modelli protocollari a potenza fissa . . . . .	67
4.4	Settaggio degli algoritmi . . . . .	70
4.5	Confronti fra gli algoritmi . . . . .	72
4.5.1	Potenza fissa . . . . .	72
4.5.2	Potenza variabile . . . . .	78
4.6	Cooperazione e controllo di potenza . . . . .	83
4.7	Esempio di applicazione . . . . .	90
4.8	Scalabilità . . . . .	93
<b>5</b>	<b>Conclusioni</b>	<b>96</b>

# Elenco delle figure

1.1	Esempio di rete mesh . . . . .	14
1.2	Esempio di rete ad hoc . . . . .	16
1.3	Probabilità d'errore in diversità e in presenza di fading . . . . .	20
1.4	Schema di funzionamento della diversità spaziale . . . . .	21
1.5	Esempio di collaborazione fra i nodi . . . . .	23
1.6	Trama TDMA suddivisa in slots di trasmissione . . . . .	26
2.1	Schema di funzionamento della generazione di colonne . . . . .	36
2.2	Schema del processo di ottimizzazione . . . . .	38
3.1	Esempio di grafo e relativo supergrafo . . . . .	47
3.2	Tipi di links nel supergrafo . . . . .	48
3.3	Esempio di adiacenza semplificata . . . . .	53
3.4	Esempio di grafo ridotto con i primi 10 cammini minimi . . . . .	55
3.5	Esempio di utilizzo della collaborazione . . . . .	57
3.6	Esempio di utilizzo del controllo di potenza . . . . .	63
4.1	Tempi per iterazione per gli algoritmi SINR a potenza variabile . . . . .	67
4.2	Grafico dei tempi di risoluzione per i modelli protocollari . . . . .	69
4.3	Tempi con diverse opzioni nei problemi di pricing . . . . .	71
4.4	Progressione del bound negli algoritmi a potenza fissa . . . . .	77
4.5	Progressione del bound negli algoritmi a potenza variabile . . . . .	82
4.6	Grafici riassuntivi per collaborazione e controllo di potenza . . . . .	89
4.7	Grafo di scenario cellulare e domande analizzate . . . . .	91
4.8	Inoltro di due domande su grafo cellulare . . . . .	92

# Elenco delle tabelle

2.1	Tabella riassuntiva dello stato dell'arte . . . . .	33
3.1	Esempio di riduzione archi nell'euristica utilizzando $k = 20$ .	56
4.1	Tempi e iterazioni per gli algoritmi SINR a potenza variabile	66
4.2	Confronto fra gli algoritmi basati sul modello protocollare .	68
4.3	Risultati ottenuti con i vari algoritmi a potenza fissa . . . .	73
4.4	Tempi di esecuzione degli algoritmi a potenza fissa . . . . .	74
4.5	Risultati ottenuti con i vari algoritmi a potenza variabile . .	79
4.6	Tempi di esecuzione degli algoritmi a potenza variabile . . .	80
4.7	Soluzioni e tempi degli algoritmi con massimo $P=0,2mW$ . .	85
4.8	Risultati della collaborazione con diversi livelli di potenza . .	87
4.9	Risultati e tempi ottenuti con l'euristica SP . . . . .	93
4.10	Risultati e tempi ottenuti con l'euristica SP con $k=10$ . . . .	94
4.11	Risultati e tempi ottenuti con l'euristica protocollare . . . .	94

# Sommario

Nelle reti completamente wireless la trasmissione di informazioni tra nodi che non possono comunicare direttamente tra loro viene garantita attraverso meccanismi collaborativi che sfruttano i nodi intermedi per realizzare un inoltro hop-by-hop a livello rete. L'inoltro cooperativo invece porta la collaborazione a livello fisico introducendo la trasmissione contemporanea dello stesso messaggio da più nodi per ottenere un guadagno di diversità in ricezione. I vantaggi ottenibili con questo approccio sono stati studiati ampiamente dal punto di vista trasmissivo soprattutto considerando comunicazioni isolate. Al contrario il suo impatto a livello di rete con molteplici flussi di traffico ha ricevuto fino ad ora attenzione limitata: è stato mostrato che il problema è computazionalmente complesso, che la collaborazione introduce dell'interferenza aggiuntiva che può ridurre il riuso spaziale e che alcuni parametri, quali densità della rete e volume di traffico, hanno un ruolo centrale nel determinarne l'utilità. Il nostro lavoro considera l'ottimizzazione congiunta della collaborazione, dell'instradamento delle domande di traffico e dello scheduling delle trasmissioni all'interno della trama con l'obiettivo di minimizzarne il numero di slot. Abbiamo prodotto degli algoritmi approssimati per risolvere reti di dimensioni reali e dalle prove riportate mostreremo che in tutti i casi considerati questi possono ottenere soluzioni entro pochi punti percentuali dall'ottimo in tempi ridotti. Inoltre abbiamo aggiunto il controllo di potenza tra le variabili di progetto ed esteso a questo scenario gli algoritmi proposti. I risultati numerici mostrano che il controllo di potenza, oltre ad assicurare un risparmio energetico, riduce l'interferenza aggiuntiva introdotta dalla collaborazione rendendo possibile un riuso più efficiente e ulteriori riduzioni della lunghezza di trama. Dato che i parametri della rete in analisi influenzano molto i risultati gli strumenti da noi proposti consentono una valutazione veloce dei benefici ottenibili con l'introduzione dell'inoltro collaborativo nei singoli casi.

# Abstract

In infrastructure-less wireless networks the exchange of information between nodes that cannot communicate directly can be achieved using intermediate nodes as relays to forward the traffic. The most widely adopted approach is based on hop-by-hop forwarding on a path to destination. Cooperative relaying brings cooperation to the physical layer using simultaneous transmissions from multiple sources to obtain a diversity gain at the receiver. Benefits of this technique have been quite well-analyzed from a transmission point of view, mostly considering a single source and destination. However the effects at the network layer of multiple concurrent flows have received little attention so far. It has been proved that the problem is computationally complex, that collaboration adds interference that limits resource reuse. Besides other parameters, such as network density and traffic volume, influence heavily its usefulness. We consider the problem of joint routing, scheduling and cooperative relaying with the objective of minimizing the time slots used in a frame. We provide two heuristic algorithms to solve realistic size topologies and show in our tests that they produce near-optimal solutions in a small run time. Then we add power control to the optimization variables and extend the proposed algorithms to this case. Numerical results prove that, in addition to energy savings, power control reduces interference caused by cooperation making a more efficient spatial reuse and an additional reduction in frame size possible. Considering that network parameters can heavily influence results, our algorithms allow a quick evaluation of the improvements introduced by cooperative relaying in specific network scenarios.



# Introduzione

Negli ultimi anni le reti radio hanno riscosso un notevole successo commerciale in svariati campi. Il mercato delle reti radiomobili è in continua crescita e offre ormai servizi avanzati a banda larga e in completa mobilità. Il WiFi ha preso sempre più piede in ambito locale e l'introduzione di WiMAX ha portato le soluzioni senza fili in diretta competizione con quelle cablate nelle reti d'accesso. Questo enorme successo è dovuto a diversi fattori fra cui la comodità dal lato utente, costi e prestazioni competitivi dal lato operatore.

Vi sono inoltre una serie di settori emergenti nel campo delle reti completamente wireless, fra cui le reti mesh e ad hoc. Le prime sono reti distribuite, costituite da router indipendenti che comunicano fra loro attraverso il mezzo radio, hanno come principale campo di impiego la sostituzione o ampliamento della rete d'accesso e possono reagire dinamicamente ed eventuali guasti e variazioni delle condizioni del canale. Le reti ad hoc sono utilizzate per mettere in comunicazione gli utenti senza un'infrastruttura preesistente, sono dotate di ampia capacità di auto-organizzazione e quindi molto adatte a scenari con elevata mobilità o dove sia richiesto un dispiegamento rapido come in situazioni di emergenza. Un ulteriore sfruttamento della comunicazione senza fili è quello delle reti di sensori con gli ambiti di applicabilità più disparati, che vanno dal monitoraggio dei processi industriali al controllo a distanza dei vari dispositivi elettronici in ambiente domestico.

Vista la crescente domanda e la più diffusa consapevolezza di questi servizi la ricerca è molto attiva in questa direzione per ottenere prodotti sempre più efficienti, a capacità elevate e onnipresenti nella vita quotidiana. Per avere questi miglioramenti nelle prestazioni sono costantemente al vaglio nuove soluzioni tecnologiche. Prima di procedere a un'implementazione pratica si

effettua spesso un'analisi teorica tramite la costruzione di modelli in modo da definire i margini di guadagno ottenibili e quindi decidere se vale la pena procedere in base ai costi di sviluppo previsti.

In questo contesto il nostro lavoro si pone l'obiettivo di valutare l'utilità dell'impiego di meccanismi di collaborazione per l'inoltro del traffico nelle reti multi-hop completamente wireless. In questo tipo di reti, quando non è possibile la comunicazione diretta fra sorgente e destinazione, è necessario l'aiuto di nodi intermedi per l'inoltro dell'informazione: questo è un primo esempio di cooperazione. Tuttavia essa è applicabile anche a livello fisico, dove si possono sfruttare trasmissioni contemporanee della stessa informazione da nodi differenti in modo da rendere la ricezione più affidabile o coprire distanze maggiori. Il nostro intento è quello di costruire un modello di quest'ultimo problema, offrire degli approcci risolutivi efficienti e fornire una valutazione del guadagno effettivo ottenibile. Per fare ciò non prenderemo in considerazione le problematiche implementative di questa tecnica, in quanto è logico introdurle solo dopo aver avuto conferma della sua utilità in condizioni ideali.

Gli aspetti di cui invece ci occuperemo sono l'instradamento di domande di traffico fra coppie di nodi, la loro gestione contemporanea (scheduling), la collaborazione a livello fisico e il controllo di potenza, considerati congiuntamente in un approccio multi-livello. Partendo da un modello, presentato in letteratura, di rete wireless multi-hop, collaborativa e a potenza fissa, abbiamo elaborato, data la sua complessità, due algoritmi semplificati. Uno mira, attraverso una nuova formulazione, a diminuire la difficoltà intrinseca del problema, mentre l'altro a ridurre le dimensioni, ovvero il numero delle variabili in gioco. Abbiamo ottenuto, nei casi analizzati, buone soluzioni, per la maggior parte coincidenti con l'ottimo o comunque con un errore piccolo, abbinate a una notevole riduzione dei tempi d'esecuzione. In seguito abbiamo esteso il modello al caso di potenza variabile, che più si avvicina alle applicazioni reali, e, analizzando i risultati ottenuti, abbiamo valutato l'utilità della cooperazione di livello fisico e quanto l'introduzione del controllo di potenza permetta un ulteriore miglioramento delle prestazioni ed un maggiore risparmio energetico.

Nel *capitolo 1* introduciamo lo scenario delle reti wireless, i loro ambiti di applicazione e le soluzioni più comunemente adottate. Di queste proponiamo le principali caratteristiche strutturali, una visione generale dei protocolli utilizzati e i benefici apportati dal loro impiego. Inoltre descriviamo le basi su cui si fonda il meccanismo della collaborazione e le sue principali caratteristiche.

Il *capitolo 2* espone il contributo dell'ottimizzazione in ambito tecnologico, utile alla rappresentazione modellistica di situazioni progettuali. In seguito analizziamo le soluzioni proposte per questo problema in letteratura, descriviamo l'algoritmo di generazione di colonne adottato nei modelli risolutivi e infine forniamo una breve presentazione degli strumenti utilizzati.

Nel *capitolo 3* abbiamo stilato i vari modelli a partire dal caso semplice, introducendo poi la collaborazione e infine il controllo di potenza accompagnati da esempi esplicativi. Parallelamente vengono presentati gli algoritmi semplificati con le relative ipotesi.

Il *capitolo 4* è dedicato ai test condotti, partendo dalle condizioni, dalla scelta della configurazione ottima di ciascun algoritmo per poi confrontarli fra di loro ed esporre i risultati ottenuti; infine riportiamo delle valutazioni di carattere tecnologico sulle possibilità d'impiego di collaborazione e controllo di potenza.

Infine nel *capitolo 5* riprendiamo i risultati conseguiti nel capitolo precedente con delle conclusioni riassuntive e delle proposte per un possibile proseguimento del lavoro.

# Capitolo 1

## La collaborazione nelle reti wireless

Le reti wireless costituiscono una forma di comunicazione importante per molti scopi, soprattutto in ambito aziendale ma ormai anche in quello privato. Le tecnologie wireless stanno avendo una continua crescita e sviluppo ed il loro impiego si è ormai diffuso e affermato in numerosi settori.

Tra queste tecnologie le più note e diffuse sono certamente WiFi e WiMAX. WiFi è la soluzione utilizzata comunemente in ambito locale per estendere o sostituire le reti LAN cablate in uffici, abitazioni e aree pubbliche; WiMAX è invece impiegato da operatori per offrire una copertura internet su aree più estese, ad esempio a sostituire connessioni via doppino telefonico come l'ADSL.

Nelle reti d'accesso tradizionali il collegamento senza fili è solo l'ultimo tra il terminale d'utente e una stazione di accesso, denominata stazione radio base o access point in base alla tecnologia. Il resto della rete dietro la stazione d'accesso è costituita da collegamenti cablati che rappresentano uno dei costi più rilevanti dell'intera infrastruttura di rete e il limite principale allo sviluppo della sua capacità di traffico verso sistemi a larghissima banda.

Le tecnologie wireless sono in grado di portare molti vantaggi anche alla dorsale della rete. In primo luogo, le soluzioni cablate sono costose: richiedono pesanti interventi sul territorio e interazione con gli ambiti di ingegneria

civile ed edile. Al di là dei costi, non sempre sono poi praticabili: particolari conformazioni del territorio, edifici storici o monumenti possono influenzare se non compromettere il progetto e l'impianto della rete cablata.

Due tra i principali elementi che influenzano le prestazioni reali di una rete di accesso senza fili sono la qualità del segnale a disposizione del terminale e l'affollamento della rete stessa. La prima si traduce nella vicinanza tra terminale e stazione base, la seconda nella popolazione che la singola cella deve servire. In entrambi i casi, per poter garantire delle buone prestazioni è necessario predisporre una rete con celle sempre più piccole (perlomeno in ambiti ad alta densità di popolazione). Questo causa un problema di prestazioni, legato proprio alla natura wireless delle reti di accesso, che comporta una backbone quanto più capillare e costosa.

## 1.1 Reti wireless mesh

Esistono diversi approcci alla creazione di una rete totalmente wireless: la soluzione più immediata consiste nel sostituire i collegamenti cablati punto-punto sulla dorsale con dei ponti radio. In questo caso si risolvono in parte i problemi legati ai costi e alla posa, ma la topologia della rete rimane di fatto invariata rispetto alle implementazioni tradizionali basate sul cavo.

Un'alternativa più radicale è rappresentata dalle reti ad hoc, all'interno delle quali i terminali sono collegati direttamente l'un l'altro, senza la necessità di stazioni base e quindi eliminando la distinzione tra reti di accesso e backbone. Questa soluzione è particolarmente indicata per la creazione rapida di reti provvisorie senza alcun tipo di infrastruttura organizzata, ma non può gestire strutture particolarmente estese e complesse.

La terza via è rappresentata dalla reti wireless mesh. In questo caso viene mantenuta la distinzione tra rete di accesso e dorsale, ma quest'ultima è costituita da una serie di apparati in grado di auto-configurarsi ed adattarsi dinamicamente alle condizioni al contorno. Ciascun nodo di una rete mesh è di fatto un router che può instradare il traffico in base alle esigenze della struttura complessiva introducendo una flessibilità che non si può ottenere in una rete cablata. Il termine mesh (in italiano, letteralmente, magliate) indi-

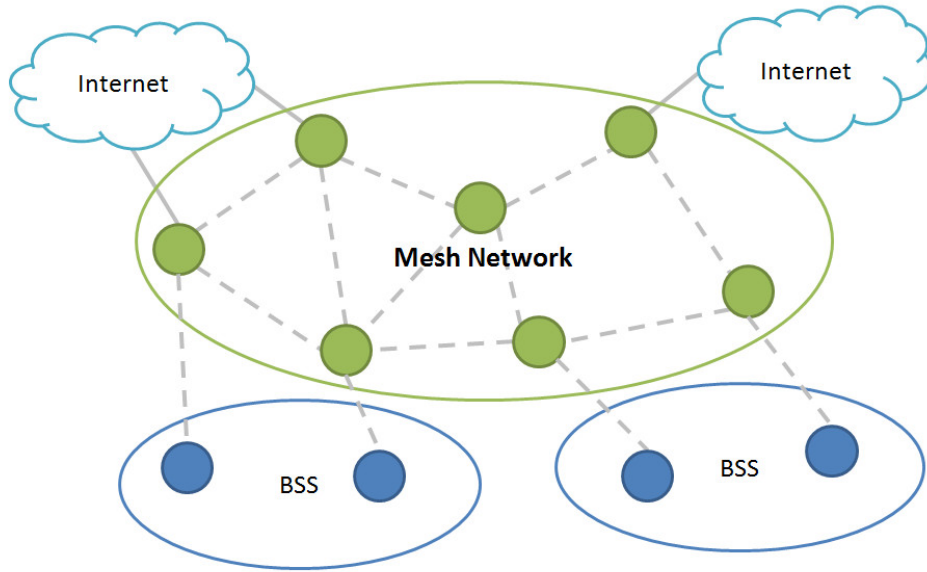


Figura 1.1: Esempio di rete mesh

ca come la topologia di backbone non sia di tipo punto-punto ma preveda il collegamento di ciascun elemento con più nodi, che sono in grado di interconnettersi grazie a link wireless. Solo alcuni apparati (i gateway o mesh portal) sono poi collegati alla rete cablata dell'organizzazione oppure del service provider mediante tratte cablate a banda larga; essi devono supportare funzioni di routing, sicurezza e gestione degli utenti mobili.

Rispetto alle reti wireless comuni le reti mesh hanno un'architettura magliata in cui i mesh point possono dialogare tra loro e hanno tutti funzioni di routing, cioè sono in grado di inoltrare i pacchetti verso un altro nodo: è sufficiente che uno di questi sia connesso alla rete cablata per estendere la copertura. Abbiamo quindi una rete capace di coprire grandi distanze, specialmente su terreni 'difficili'.

Tutti i mesh point della rete parlano con tutti gli altri e possono essere posizionati ovunque ci sia una presa di corrente. Laddove non arriva l'elettricità, si possono anche usare access point alimentati a batteria o dotati di piccoli pannelli solari. Ulteriori mesh point collegati fisicamente alla rete possono essere aggiunti per aumentare l'affidabilità complessiva del sistema o per migliorare il throughput, riducendo il numero di salti (hop) necessari

per raggiungere il punto di interconnessione.

Le reti a maglie possono includere dispositivi sia fissi che mobili. L'area operativa del singolo access-point viene generalmente definita hot spot. Questi apparati hanno un doppio accesso radio, in due bande di frequenza diverse: uno per comunicare con i client, basato sul diffuso 802.11b/g, che funziona nella banda a 2,4 GHz, e l'altro per dialogare fra di loro, ovvero per il backhauling, nella banda a 5 GHz con l'802.11a, al fine di limitare l'interferenza con i dispositivi degli utenti. Con il termine backhauling si intende il servizio di smistamento del traffico dal nodo di origine al punto di accesso fisso da cui può essere distribuito su una rete cablata.

La topologia a maglia è particolarmente affidabile e consente notevoli vantaggi sia nella gestione dei guasti, che nell'ampliamento della rete. Ad esempio le interruzioni di servizio, dovute alla caduta di un collegamento o di un nodo, sono eliminate attraverso l'uso di algoritmi adattativi di routing e funzionalità di auto-configurazione con cui il sistema è in grado di riconfigurarsi e trasmettere il segnale utilizzando un percorso alternativo; la soluzione mesh consente di aggregare gli hot-spot ampliando così le aree di copertura, tramite le funzionalità di auto-discovery in caso di aggiunta dei nodi; infatti la rete è in grado di organizzarsi, di configurarsi dinamicamente, riconoscendo i nodi che vengono aggiunti con un intervento manuale limitato. Devono poi essere previste nei mesh access point le funzionalità di sicurezza per autenticare l'utente e proteggerne il traffico.

La soluzione mesh, grazie queste funzionalità, minimizza i costi d'installazione e dell'infrastruttura di rete e consente di renderla rapidamente operativa e meno dispendiosa nella gestione.

## 1.2 Reti ad hoc

Negli ultimi anni si è andata delineando una nuova generazione di sistemi di comunicazione wireless. Questi sistemi sono caratterizzati da un'elevata mobilità e nascono dall'esigenza di un inserimento in rete rapido e autonomo di terminali indipendenti e dalla necessità di rendere la rete libera da qual-

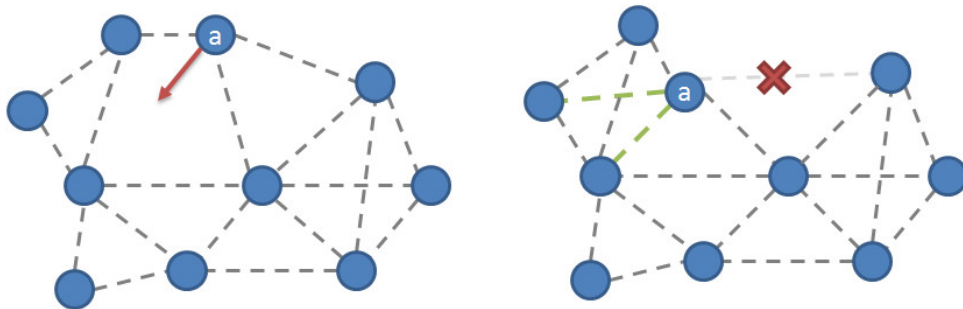


Figura 1.2: Esempio di rete ad hoc

siasi infrastruttura preesistente. Questa nuova tipologia di rete viene definita MANET (Mobile Ad Hoc NETWORK).

Una MANET é un sistema autonomo di dispositivi che fungono sia da host che router, connessi con collegamenti di tipo wireless formando un grafo di forma arbitraria. E' un sistema broadcast non centrale multi-hop in cui ogni nodo quindi ricopre entrambe le funzionalità che sono invece ben distinte in una rete cablata. Questa prerogativa delle reti MANET fa si che un nodo, per poter partecipare alla rete, debba occuparsi non solo dell'invio e ricezione dei propri pacchetti ma anche dell'inoltro dei pacchetti provenienti da altri nodi. Per fare questo, ogni nodo deve destinare parte delle sue risorse alla collaborazione con gli altri. Inoltre tali router sono liberi di muoversi casualmente e di auto-organizzarsi, sebbene la topologia wireless vari rapidamente ed in modo imprevedibile. Questa rete può operare da sola oppure essere connessa alla rete Internet.

Lo spazio di applicazioni delle MANET varia da reti di piccole dimensioni quasi statiche in cui pochi nodi entrano ed escono dalla rete, a reti su larga scala caratterizzate da un'alta mobilità dei nodi. Gli scenari di utilizzo sono quelli in cui sia sconveniente, impossibile o scomodo creare un'infrastruttura di rete fissa.

Esempi significativi di utilizzo di queste strutture di rete possono essere l'instaurazione di una rete robusta, efficiente e dinamica sui luoghi di catastrofi, in scenari di emergenza o di operazioni militari in campo ostile.



Altri scenari possibili sono quelli di conferenze, eventi o riunioni nelle quali i partecipanti vogliano instaurare in modo semplice e spontaneo una rete per condividere informazioni o scambiare documenti.

La topologia della rete non è determinabile a priori, rendendo impossibili assunzioni sulla disponibilità delle diverse entità interagenti. A causa della mobilità l'insieme dei nodi vicini a ciascun host varia nel tempo. Disconnessioni, partizioni e modifiche della rete sono eventi comuni che causano cambiamenti nella collaborazione fra entità nuove e precedentemente sconosciute. Un ulteriore elemento di complessità deriva dalla possibilità di partecipazione tramite dispositivi eterogenei sia per natura sia per prestazioni, quali laptop, PDA (palmari) o telefoni cellulari. Tali dispositivi sono caratterizzati dalla scarsità delle risorse a disposizione, in particolare banda di trasmissione e batteria.

In una rete ad-hoc tutti i nodi hanno un rapporto cooperativo, dato che non esistono punti di riferimento né fissi (access point) né mobili che coordinino sottoinsiemi di nodi. Di conseguenza tutti i dispositivi devono prendere decisioni in maniera collettiva. A causa di tutte queste caratteristiche la definizione e la scelta di protocolli di rete che si occupino del routing dell'informazione risultano problematiche e a questo proposito è stato istituito un working group, il MANET WG, all'interno dell'IETF.

In generale un qualsiasi movimento di nodi all'interno della rete ne modifica la topologia e quindi richiede la modifica, la creazione o la cancellazione di cammini di routing. Quindi nelle MANET occorre utilizzare protocolli nuovi che facciano fronte alle esigenze di queste topologie.

E' possibile suddividere i protocolli di routing ad-hoc in due grandi famiglie: proattivi e reattivi.

In un protocollo proattivo i nodi si scambiano continuamente informazioni sul routing. Un nodo conosce in ogni momento la topologia della rete e le rotte verso tutti gli altri nodi che la compongono. Questa tipologia di protocolli si adatta molto bene a situazioni in cui vi siano molte comunicazioni tra molte coppie diverse di nodi. Infatti il continuo scambio di informazioni porta ad un cospicuo consumo di risorse di rete che non sarebbe giustificato in caso di poche comunicazioni.

Per questo motivo si sono studiati anche protocolli reattivi. Questi protocolli ricavano le informazioni di routing on-demand, cioè cercano una rotta tra due nodi solo nel momento in cui vi sia una richiesta di comunicazione da parte del nodo sorgente. Nei protocolli reattivi un nodo ha unicamente una conoscenza parziale della topologia di rete e le informazioni di routing ottenute con la discovery non vengono mantenute costantemente aggiornate ma, una volta terminata la comunicazione, vengono lasciate scadere, con un minore dispendio in termini di traffico di segnalazione.

La crescente proliferazione di dispositivi portatili che possono disporre di connettività wireless, i recenti sviluppi di queste tecnologie e l'emergere delle MANET aprono quindi uno scenario innovativo nel quale gli utenti non richiedono solo la possibilità di fruire dei tradizionali servizi Internet, ma anche la possibilità di beneficiare di servizi collaborativi avanzati. In particolare gli utenti necessitano della possibilità di interagire ovunque essi siano, in qualunque momento e possibilmente anche quando sono in movimento.

## **1.3 Meccanismi collaborativi**

Le modalità di inoltro del traffico hop-by-hop viste finora per le reti mesh e ad hoc sono già considerate meccanismi di collaborazione, in quanto per andare da sorgente a destinazione ci si può avvalere dell'aiuto dei nodi intermedi che mettono a disposizione le proprie risorse per inoltrare l'informazione. Oltre a questo tipo di collaborazione a livello di rete se ne può pensare un altro a livello fisico che si basa sullo sfruttamento delle peculiarità del canale radio e delle potenzialità offerte dalla diversità.

### **1.3.1 Il canale radio e la diversità spaziale**

Il canale radio è un mezzo intrinsecamente broadcast le cui caratteristiche variano notevolmente a seconda dell'ambiente considerato. Perciò l'informazione trasmessa è soggetta a vari fenomeni di riflessione, diffusione e diffrazione che dipendono dalla frequenza e dagli ostacoli presenti sul territorio;

questi aspetti influenzano spesso negativamente la qualità della trasmissione e necessitano di una doverosa attenzione nello studio del sistema.

L'effetto principale causato da questi fenomeni è l'attenuazione che, se nel caso di spazio libero assume un andamento proporzionale alla distanza al quadrato, può essere ulteriormente aggravata da due cause principali quali sono ombreggiamento e propagazione multipercorso. La prima è dovuta alla presenza di grossi ostacoli che causano una pesante ostruzione al segnale e introducono un'incertezza sul valore dell'attenuazione che viene comunemente rappresentata con una distribuzione log-normale. La seconda è causata dalla ricezione di più copie del segnale che hanno seguito cammini diversi e hanno subito sfasamenti e attenuazioni differenti; quando è presente una componente dominante (cammino diretto) rispetto alle altre si parla di fading di Rice, mentre se non si ha un segnale prevalente si definisce fading di Rayleigh. I nomi derivano dalle due distribuzioni statistiche che sono comunemente utilizzate per rappresentarli. Gli effetti dell'attenuazione si ripercuotono sulla probabilità di errore del canale che, se nel caso di spazio libero dipende dall'SNR attraverso una funzione gaussiana, aggiungendo il fading diventa una relazione di proporzionalità inversa.

Dato che l'analisi precisa della propagazione è complessa e non attuabile si predilige spesso derivare modelli empirici, basati sull'osservazione che, a seconda dell'ambiente di propagazione, usano un'attenuazione media proporzionale a  $d^\gamma$ , con  $d$  distanza e  $\gamma \in [2, 5]$ . Con  $\gamma = 2$  ci si riconduce alla propagazione in spazio libero mentre si usano coefficienti elevati per ambienti difficili come quello urbano.

Uno degli strumenti principali per migliorare la comunicazione sul mezzo radio è la trasmissione su più canali indipendenti che consentano di ottenere misure incorrelate e migliorare la stima del segnale. L'uso della diversità permette di incrementare l'affidabilità della comunicazione, ridurre il consumo energetico e diminuire la latenza. In modo più specifico si può aumentare la resistenza al fading e agli errori 1.3 dei singoli nodi grazie ad una maggiore disponibilità di cammini sorgente-destinazione, oppure implementare un risparmio energetico più efficiente. Esistono vari tipi di diversità che sfruttano le principali grandezze fisiche del segnale radio come spazio, tempo,

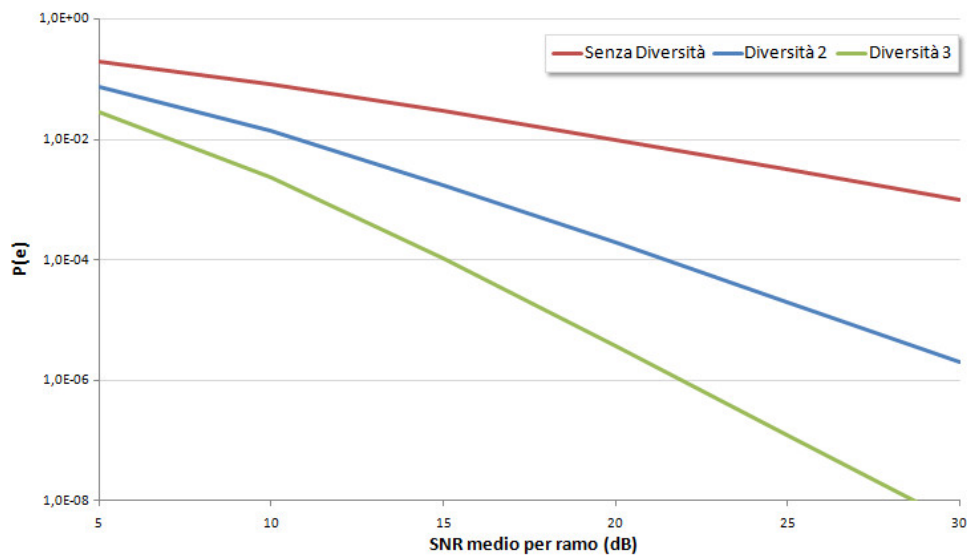


Figura 1.3: Esempio dell'andamento della probabilità d'errore in base alla diversità in un canale con fading

frequenza e polarizzazione.

Per i nostri scopi la più importante è la diversità spaziale, che consiste nella trasmissione e/o la ricezione di uno stesso simbolo via antenne multiple. La stessa informazione viene codificata e suddivisa su antenne diverse in trasmissione e inviata contemporaneamente. L'onda radio può effettivamente propagarsi in diverse direzioni ed eventualmente essere riflessa o rifratta da alcuni oggetti dell'ambiente fisico, così un'antenna può ricevere, traslate nel tempo, più informazioni identiche che hanno avuto percorsi diversi con riflessi successivi.

Se le antenne in ricezione sono fisicamente separate da una distanza tale da rendere i segnali ricevuti indipendenti fra loro, è probabile che, quando un'antenna sperimenta un basso rapporto segnale/rumore, la trasmissione possa comunque essere correttamente decodificata a partire dal segnale delle antenne adiacenti.

Per una corretta decodifica dell'informazione in ricezione sono state elaborate diverse tecniche di ricombinazione per sistemi in diversità. Dati più segnali in ricezione si può scegliere di considerare quello con SNR maggiore oppure effettuare una ricombinazione. La tecnica ottima è quella MRC (ma-

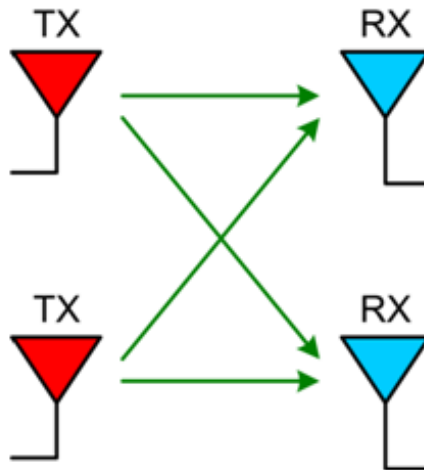


Figura 1.4: Schema di funzionamento della diversità spaziale

ximum ratio combining) che somma i segnali rifasati pesandoli in base ai guadagni stimati ottenendo di fatto un SNR finale pari alla somma di quelli di ciascun ramo. Sono state inoltre create svariate tecniche di codifica per sfruttare appieno i vantaggi offerti da un canale con diversità tra cui la più semplice e nota è il codice di Alamouti.

Si possono distinguere poi i casi di ricezione in diversità, che fanno uso di molteplici antenne riceventi (Single Input Multiple Output ovvero canali SIMO), e di trasmissione in diversità, che fanno uso di molteplici antenne trasmettenti (Multiple Input Single Output ovvero canali MISO). Canali con antenne multiple sia in ricezione che in trasmissione (Multiple Input Multiple Output ovvero canali MIMO) sviluppano una potenzialità ed efficienza ancora maggiore.

La ridondanza spaziale può essere, infatti, utilizzata per aumentare la velocità di trasmissione dei segnali o l'affidabilità. Solitamente si ottiene un livello maggiore di SNR in ricezione che cresce linearmente con il numero delle antenne e che produce una crescita logaritmica della capacità di canale come ricavabile dal teorema di Shannon. In realtà per i sistemi MIMO, se è presente uno scattering significativo, questo concetto può essere esteso per ottenere un aumento della capacità lineare con il numero di antenne. Per avvicinarsi a questo limite sono necessarie opportune codifiche spazio-temporali

sulle antenne e una stima precisa delle condizioni del canale sotto forma della sua matrice trasmissiva; queste permettono, sfruttando lo scattering esistente, di creare un insieme di canali virtuali in parallelo ed indipendenti fra i due array di antenne in trasmissione e in ricezione.

### 1.3.2 La collaborazione a livello fisico

Il concetto di diversità considerato può essere ulteriormente esteso introducendo la cooperazione fra nodi per le reti multi-hop, in cui nodi separati spazialmente agiscono di fatto come un sistema di antenne MIMO distribuito.

L'intento della cooperazione è quello di sfruttare al meglio le risorse di rete e l'approccio più comunemente adottato è quello che sfrutta la diversità spaziale impiegando più nodi per la trasmissione dello stesso messaggio.

L'idea della collaborazione risale a Van Der Meulen [26] e Cover [9] dove viene proposto l'uso di trasmissioni di più relays combinate in ricezione. Gli studi sono stati poi estesi all'analisi della capacità di canale anche attraverso l'utilizzo di codici distribuiti, la ritrasmissione, l'allocazione di potenza e lo sfruttamento dei guadagni di diversità.

Solitamente le reti collaborative hanno un'infrastruttura decentralizzata, in cui ogni nodo ha un'antenna singola omnidirezionale e può dinamicamente aggiustare la sua potenza, ovviamente in un range di valori discreti, per controllare il suo raggio di trasmissione. Generalmente i vari nodi possono sia trasmettere che ricevere oppure essere suddivisi in base al loro compito: alcuni fungono da ricevitori, altri da trasmettitori e altri ancora da ripetitori.

Sono stati infatti considerati scenari con più sorgenti e destinazioni che richiedono protocolli per l'accesso multiplo il routing e la segnalazione [22], [5], [21].

La collaborazione inizia con la trasmissione di un nodo singolo verso un gruppo di nodi che, avendo ricevuto la stessa informazione, può poi cooperare per trasmettere la stessa verso un altro gruppo di nodi, agendo come un array virtuale di antenne [13] [12]. Quindi ogni trasmissione può essere sia broadcast, quando l'informazione trasmessa da un nodo singolo viene

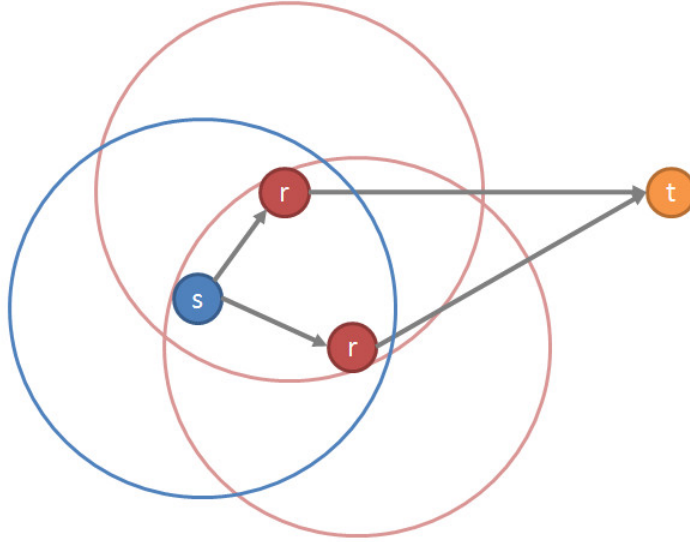


Figura 1.5: Esempio di collaborazione fra i nodi

ricevuta da più nodi, sia cooperativa, quando più nodi inviano simultaneamente l'informazione a un singolo ricevitore. Nel primo caso si parla di Modo Broadcast, mentre nel secondo di Modo Cooperativo.

In figura 1.5 è mostrata una semplice rete wireless formata da 4 nodi, una sorgente ( $s$ ), una destinazione ( $t$ ) e due nodi relay ( $r$ ), in cui viene messo in evidenza un caso di routing collaborativo fra di essi. Visto che entrambi i ripetitori si trovano all'interno del raggio di trasmissione del nodo sorgente entrambi ricevono l'informazione da lui trasmessa senza costi addizionali. La cooperazione risiede nell'hop successivo: infatti la potenza dei due ripetitori viene combinata per poter raggiungere la destinazione, altrimenti irraggiungibile singolarmente.

Nella trasmissione con collaborazione, inoltre, possiamo distinguere due modalità: Amplify-and-Forward, detto anche relaying non rigenerativo, e Decode-and-Forward, detto invece rigenerativo [17], [15]. Nella prima il segnale è semplicemente amplificato e ritrasmesso dai nodi relay mentre nel secondo caso si effettua la decodifica e ricodifica del segnale a ogni nodo. Ne consegue che per l'Amplify-and-Forward l'SNR delle trasmissioni andrà a degradare a causa dell'accumulo di rumore sulle tratte radio, mentre nel caso Decode-and-Forward le tratte sono considerabili indipendenti per quanto ri-

guarda l'SNR; tuttavia questa tecnica prevede operazioni aggiuntive in ogni nodo, quindi dispositivi più complessi, e la possibilità di propagare errori di decodifica. Esiste inoltre una variante del Decode-and-Forward, definita come Decode-and-Reencode, in cui il nodo relay decodifica il segnale ricevuto e costruisce delle nuove parole di codice per ricodificarlo, creando un aumento di ridondanza al ricevitore che ottiene entrambi i segnali codificati in maniera differente. Permane la possibilità di propagare errori di decodifica.

I protocolli di relaying possono essere implementati in modo statico, in cui il segnale viene sempre inoltrato, o adattativo, dove il nodo decide se effettuare o meno l'inoltro dell'informazione nell'ottica di bilanciare i benefici ottenuti dal relaying con la possibile propagazione di errori. Questi ultimi richiedono un feedback di canale per prendere la decisione corretta.

Il problema di trovare il cammino ottimo richiede decisioni in più passi, dove ad ogni passo un insieme di nodi può collaborare per trasmettere i pacchetti ad un determinato nodo successivo. In questo senso il cammino tra sorgente e destinazione può essere visto come una sequenza di trasmissioni collaborative fra gruppi di nodi, con una appropriata allocazione di potenza per ognuno di essi.

Per superare diversi limiti del forwarding hop-by-hop la collaborazione a livello trasmissivo nelle reti wireless è stata valutata per diverse tecnologie emergenti e l'estensione alle reti collaborative delle tecniche di progetto è un argomento molto attuale e non ancora completamente esplorato. Introducendo la collaborazione si può quindi permettere: trasmissioni a raggio più ampio, la copertura di nodi altrimenti irraggiungibili e riduzione del numero di hop tra nodi distanti, grazie alle quali si possono avere anche importanti ripercussioni sulla capacità del sistema, la sua affidabilità e applicabilità in casi particolari.

L'analisi del guadagno di performance a livello di sistema nel relaying cooperativo in reti wireless multi-hop richiede la valutazione dell'efficienza nel riutilizzo delle risorse radio quando si servono differenti flussi di traffico fra varie coppie di nodi e ha ricevuto fino ad ora scarsa attenzione. La valutazione della capacità di rete infatti non è semplice neanche nel caso di multi-hop forwarding non collaborativo per cui tuttavia sono stati stimati dei bounds



[14].

Gli studi in [24] [23] dedicano la loro attenzione all'analisi di semplici schemi collaborativi dove i nodi sono divisi in gruppi distinti di sources, destinations e relays. In entrambi sono sviluppati algoritmi per l'assegnazione del relay ottimo a ciascuna comunicazione (minimizzando la potenza o massimizzando il throughput) senza considerazioni sull'utilizzo di trasmissioni contemporanee e quindi sull'interferenza. In [24] viene mostrato che l'uso della collaborazione consente un risparmio energetico e soprattutto che la scelta stessa del relay da utilizzare può influenzare molto l'entità di questo risparmio; in [23] invece si mostra come la scelta ottima del relay contribuisca nell'aumentare la capacità di canale. Bisogna tener tuttavia ben presente che i risultati sono validi nei casi di trasmissioni isolate considerati e non è certo se continuino a valere in presenza di interferenza.

Un tentativo di stimare il guadagno apportato dal relaying cooperativo si trova in [27] in cui viene dimostrato che la collaborazione è efficiente in scenari isolati o sparsi ma che al crescere della dimensione della rete e della densità la performance è penalizzata dall'accresciuto livello di interferenza. In particolare viene mostrato che esiste una soglia critica sul guadagno, passando da trasmissione diretta a trasmissione con relaying in uno scenario isolato, oltre la quale la collaborazione diventa conveniente in reti di grandi dimensioni.

In questo senso solitamente la collaborazione risulta essere molto sensibile alla topologia di rete e al tipo di di traffico. Solitamente in reti sparse e con pochi nodi il vantaggio della cooperazione è molto rilevante in quanto permette di estendere il raggio di trasmissione dei nodi e di ridurre il numero di hop senza impattare eccessivamente sulle interferenze percepite dagli altri nodi. Al contrario in reti dense prevale l'aumento dell'interferenza e si riduce drasticamente il livello di riuso.

## 1.4 Accesso multiplo e scheduling

Come già detto in precedenza il mezzo radio è condiviso e broadcast, si necessita quindi di tecniche per il coordinamento dell'accesso al canale da

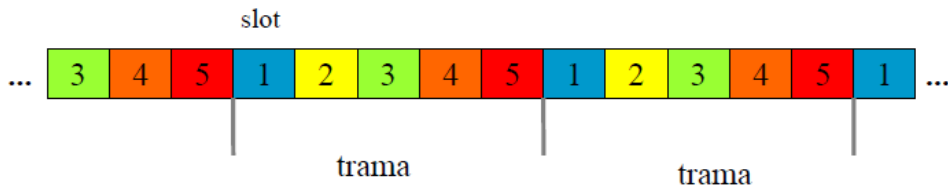


Figura 1.6: Trama TDMA suddivisa in slots di trasmissione

parte dei dispositivi che devono trasmettere. Queste tecniche si basano sulla distinzione tra i vari flussi da trasmettere, operata utilizzando i parametri del livello fisico come frequenza, tempo, codice e lunghezza d'onda.

Nel nostro caso analizziamo un sistema gestito mediante un accesso multiplo a divisione di tempo (TDMA). Questa è una tecnica in cui il canale viene condiviso mediante ripartizione del tempo di accesso quindi ogni utente usa l'intera banda disponibile ma per un tempo limitato. Il tempo è organizzato in trame, composte da un certo numero di slots tutti della stessa durata. È possibile implementare canali a diversa velocità assegnando più time slots allo stesso utente.

In scenari wireless, data la particolare natura del mezzo, è possibile sfruttare una suddivisione spaziale delle risorse congiuntamente a quella temporale (Spatial TDMA, STDMA) ovvero lo stesso slot viene utilizzato contemporaneamente da nodi sufficientemente distanti da poter considerare l'interferenza mutua trascurabile.

Nei sistemi TDMA è necessario un sincronismo di trama, ossia un coordinamento per trovare una base temporale comune ai vari utenti, in modo che ciascuna trasmissione occupi esattamente il time slot ad essa assegnato senza sovrapporsi alle altre. Questo viene gestito con un tempo di guardia fra un canale e l'altro. Per fare ciò si possono utilizzare diverse tecniche in base alla natura del canale: in canali broadcast centrali solitamente si utilizza quella conosciuta come timing advance, mentre in quelli broadcast non centrali il tutto risulta molto più complesso.

Inoltre se la condivisione delle risorse deve essere dinamica l'accesso multiplo non può essere gestito solamente a livello fisico, ma necessita di un

protocollo a livello superiore per evitare o controllare eventi di collisione e di conseguente ritrasmissione. Fra questi vi sono algoritmi che si limitano a gestire le collisioni attraverso ritrasmissioni, altri che cercano di evitarle monitorando il canale, e altri ancora che implementano un sistema di permessi cercando di evitare il problema a priori.

L'assegnamento degli slot alle comunicazioni in atto è gestita da una funzione di scheduling che cerca di massimizzare il throughput, ossia il numero di pacchetti inviati nell'unità di tempo, minimizzare il tempo di permanenza in coda di un pacchetto e mantenere una buona equità nella spartizione del mezzo. Eventualmente alle comunicazioni si può associare una certa QoS (quality of service) per gestire le priorità dei diversi flussi, solitamente considerando i parametri di banda, ritardo e probabilità di perdita di un pacchetto.

## Capitolo 2

# Ottimizzazione delle risorse radio

Nell'ambito delle reti completamente wireless multi-hop sono state considerate ed analizzate in letteratura diverse problematiche di ottimizzazione. Infatti sia il progetto che l'amministrazione di una rete estesa prevede un numero elevato di decisioni che hanno un impatto immediato sul costo della rete, sulla sua funzionalità e sulla possibilità di interventi di aggiornamento ed espansione. Quindi gli strumenti forniti dalla ricerca operativa permettono di valutare l'utilità e l'applicabilità di una soluzione tecnologica, analizzando i vantaggi ottenibili al limite con un impiego ottimo delle risorse. Questo è fondamentale per decidere se sviluppare il progetto considerato e, successivamente, controllare l'andamento dell'implementazione. Se l'algoritmo di risoluzione del problema è particolarmente efficiente e sottostà ai criteri di progetto è possibile impiegarlo direttamente nel sistema.

Il primo problema da considerare è normalmente quello di allocazione di risorse che consiste nella scelta dei siti dove impiantare le antenne e gli altri apparati di rete in modo da garantire la copertura necessaria e l'eventuale allocazione delle frequenze. Successivamente si possono considerare i problemi di routing, scheduling, traffic engineering e controllo di potenza, fondamentali per la messa in opera e l'amministrazione efficiente del sistema.

In letteratura esistono anche approcci trasversali che considerano con-

giuntamente più problemi tra quelli riportati. Questi hanno il vantaggio di fornire potenzialmente soluzioni migliori, in quanto analizzano il problema nel suo insieme, tuttavia implicano un numero di variabili e vincoli molto elevato rendendo il problema di difficile risoluzione.

In generale per affrontare un problema con strumenti di ottimizzazione sono fondamentali la formulazione di un modello matematico e la complessità ad esso associata. Esistono infatti problemi risolvibili con algoritmi semplici ed efficienti che arrivano alla soluzione in tempo polinomiale rispetto alle dimensioni dell'istanza in analisi. Altri invece non possono essere risolti con algoritmi specializzati e quindi si deve ricorrere a tecniche generali come il Branch & Bound o il Branch & Cut, utilizzato da molti solver commerciali, che però hanno tempi di convergenza esponenziali. Questi problemi sono definiti come NP-completi. Aumentando ulteriormente la complessità esiste un'altra categoria, quella degli NP-difficili, che hanno come sottoproblema un problema NP-completo.

I problemi che analizzeremo prevedono l'ottimizzazione congiunta di scheduling, routing e controllo di potenza che appartengono tutti alla categoria degli NP-difficili in quanto anche il solo problema di scheduling si può ridurre a quello di colorazione del grafo [4] che è NP-difficile.

Un primo meccanismo analizzato si basa sul inoltro hop-by-hop dell'informazione attraverso un cammino composto da una sequenza di nodi interposti tra sorgente e destinazione con un protocollo di instradamento che sceglie dinamicamente il cammino da seguire [1] ed è stato adottato in varie implementazioni pratiche in reti ad hoc, di sensori e mesh.

Per quanto riguarda il caso non collaborativo si trovano in letteratura numerosissimi studi riguardanti algoritmi per l'ottimizzazione del routing e scheduling in reti wireless STDMA che considerano svariati scenari e ipotesi. Dato che il problema è NP-difficile oltre ad approcci esatti esistono anche molte euristiche utili a trattare reti di dimensioni elevate. Al contrario gli studi che considerano l'aggiunta della collaborazione al problema sono ancora pochi.

## 2.1 Stato dell'arte

In [11] viene proposto un algoritmo per risolvere il problema che considera anche il controllo di potenza con l'obiettivo di minimizzare la potenza media trasmessa con vincoli di traffico da smaltire, di potenza massima trasmissibile da ciascuna antenna e di SINR. L'algoritmo utilizzato usa un metodo iterativo sul problema duale, ha complessità nel peggior caso esponenziale e converge in tempi accettabili con grafi fino a circa 15 link. Viene tuttavia indicata la possibilità di adottare tecniche di clustering come proposte in [10] per ottenere una maggiore scalabilità in reti di grandi dimensioni e sparse. Tra i risultati è da notare che la soluzione ottima non sempre richiede l'instradamento su path a potenza minima.

Un algoritmo di approssimazione in tempo polinomiale per il problema con controllo di potenza di un flusso in una rete TDMA/CDMA è proposto in [3] dove si considera anche come estenderlo al caso con più flussi distinti. L'assunzione di utilizzare trasmissioni a spettro espanso consente di semplificare il problema di interferenza tra nodi trasmettenti in contemporanea.

Quello in [4] è un algoritmo esatto con l'obiettivo di trovare la trama di lunghezza minima consentita considerando i vincoli di interferenza a livello fisico. L'approccio utilizzato si basa sulla generazione di colonne e introduce alcuni accorgimenti per rendere la risoluzione del problema maggiormente scalabile. Le istanze risolvibili riportate arrivano a 60 nodi e 400 link.

In [6] è presentato un algoritmo di generazione di colonne, derivato da [4], con l'aggiunta del controllo di potenza e adattamento della velocità di trasmissione in reti mesh. Non viene invece considerato il routing e il modello di interferenza usata è quello fisico. Nell'articolo viene anche discussa la possibilità di utilizzare algoritmi euristici per risolvere il problema di pricing tuttavia viene concluso che il miglior trade-off tra risultati e tempo computazionale si ottiene con la risoluzione esatta. Le istanze considerate arrivano a 30 nodi nel caso senza controllo di potenza e adattamento della velocità. In [7] si riprende l'algoritmo di generazione di colonne basato su formulazioni MIP presentato in [6] con aggiunta dello scheduling e se ne confronta le prestazioni con un approccio basato sul constraint programming (CP).

L'approccio CP ha prestazioni migliori nel caso a potenza fissa ma il metodo classico prevale nei casi più complessi. Viene inoltre investigata la possibilità di risolvere il problema di pricing alla prima soluzione intera invece che all'ottimo, che comporta un maggior numero di iterazioni ma minor tempo per iterazione, e l'uso di un euristica per eliminare colonne poco utili dal master quando il loro numero cresce troppo.

Una euristica per il problema con controllo di potenza si trova in [18] applicato a reti ad-hoc con domande di traffico da soddisfare tra ogni coppia di nodi e l'obiettivo di ottenere la trama a potenza minima. L'euristica consiste in un algoritmo greedy per trovare configurazioni con il numero massimo di link attivi e non interferenti tra loro in base a condizioni di SINR; nell'algoritmo il controllo di potenza viene affinato a ogni iterazione quando un link viene aggiunto alla configurazione.

L'articolo [16] propone un metodo di generazione di colonne non lineare per il problema con controllo di potenza e adattamento della velocità trasmissiva che ha come obiettivo la massimizzazione del throughput e considera la possibilità di introdurre equità nella ripartizione di risorse tra i flussi. L'analisi valuta inoltre i benefici dell'introduzione di controllo di potenza, adattamento della velocità di trasmissione e routing nel problema: dagli esperimenti condotti i primi due portano a un incremento del throughput in media del 22% mentre il routing libero (rispetto a quello prefissato) porta a un incremento in media del 50% e un aumento del fattore di riutilizzo di circa 1.4. Le topologie risolvibili sono al di sotto dei 20 nodi.

Anche in [25] si cerca la massimizzazione del throughput per il problema con controllo di potenza in reti mesh, considerando l'interferenza al livello fisico e parametri di equità nella ripartizione delle risorse; il routing non è considerato ma viene discussa la sua possibile introduzione. L'algoritmo presentato è una euristica basata sul rounding: si fanno rilassamenti successivi del problema ILP e man mano si fissano delle variabili finché non si ha una soluzione intera.

In [2] si introduce un modello con controllo di potenza e interferenza a livello fisico per la ricerca dello schedule di lunghezza ottima e una euristica per trovare upper e lower bounds per il problema. L'euristica si basa su

un grafo di interferenza generalizzato per livelli di potenza variabile su cui vengono cercati attraverso un algoritmo greedy insiemi massimi indipendenti da cui, attraverso un passo di pruning e uno di massimizzazione, si ricavano insiemi ammissibili di cardinalità vicina alla massima.

Il lavoro in [19] affronta il problema considerando traffici che convergono verso dei gateways. Ciò permette di trasformare il problema, utilizzando un grafo associato, in uno di flusso massimo che ha come duale il taglio a capacità minima. Tale problema può essere risolto con un algoritmo di cross-line column generation che partendo da un set limitato di variabili e vincoli aggiunge nuove variabili (colonne) e vincoli (tagli). L'algoritmo risolve all'ottimo istanze fino ai 200 nodi in tempi ridotti, ma va considerato che nelle simulazioni riportate si considera un modello di interferenza protocollare molto semplice.

[20] propone un algoritmo di programmazione dinamica approssimata per il problema di scheduling in reti mesh, le domande di traffico sono sui link quindi non c'è routing, la potenza per nodo e il numero di time slot sono fissati e l'obiettivo è il throughput massimo. Il problema è ridotto a problemi più piccoli sui singoli slot poi collegati attraverso programmazione dinamica o una sua versione approssimata con metodi euristici e a complessità ridotta.

In [8] si estende l'approccio a generazione di colonne visto in [7] al caso collaborativo attraverso la creazione di un grafo di collaborazione contenente oltre a nodi e archi anche supernodi (coppie di nodi) e superarchi tra di essi. Grazie al grafo creato l'approccio risolutivo con generazione di colonne diventa utilizzabile anche in questo caso, tuttavia la complessità cresce in quanto la dimensione del grafo di collaborazione è esponenziale rispetto alla dimensione del grafo iniziale.

Per avere un'idea generale dei risultati fino ad ora ottenuti negli articoli in letteratura, riportiamo la tabella riassuntiva 2.1 contenente il tipo di problema affrontato, l'approccio risolutivo e la dimensione delle istanze raggiunta.



Articolo	Problema trattato	Interferenza	Approccio	Dimensione
[11]	R S P <sub>v</sub>	SINR	OPT	15 L
[3]	R S P <sub>v</sub>	No	EUR	-
[4]	R S	SINR	OPT	60 N, 396 L
[6]	S P <sub>v</sub> Ra	SINR	OPT	30 N
[7]	R S P <sub>v</sub> Ra	SINR	OPT	15 N, 170 L
[18]	R S P <sub>v</sub>	SINR	EUR	-
[16]	R S P <sub>v</sub> Ra	SINR	OPT	20 N
[25]	S P <sub>v</sub>	SINR	EUR	30 N
[2]	S P <sub>v</sub>	SINR	EUR	100 L
[19]	R S	PROT	OPT	200 N
[20]	S	SINR	EUR	36 N, 60 L
[24]	R P <sub>v</sub> Co	No	EUR	-
[23]	R P <sub>v</sub> Co	No	EUR	-
[8]	R S Co	SINR	OPT	10 N, 90 L*

(a) Tabella degli algoritmi

Sigla	Significato	Sigla	Significato
R	Routing	PROT	Interferenza protocollare
S	Scheduling	OPT	Ottimo
P <sub>v</sub>	Potenza variabile	EUR	Euristica
Ra	Rate adaptation	N	N° nodi
Co	Cooperazione	L	N° links
SINR	Interferenza a livello fisico	No	Nessuna interferenza

(b) Legenda della tabella 2.1a

Tabella 2.1: Tabella riassuntiva dello stato dell'arte

## 2.2 Generazione di colonne

Tra i problemi caratterizzati da un elevato numero di variabili e/o vincoli ne esistono alcuni che hanno una struttura tale per cui è possibile risolverli decomponendoli in sottoproblemi di più facile soluzione. In generale l'obiettivo è quello di risolvere tali problemi senza considerare esplicitamente tutte le variabili e/o vincoli. La generazione di colonne è un metodo adottato quando il numero delle variabili è troppo elevato e prevede di partire con un problema contenente solo alcune di esse aggiungendo progressivamente solo quelle che risultano effettivamente necessarie a risolvere il problema.

Supponiamo di dover risolvere un problema di programmazione lineare, che chiameremo problema master,

$$\max\{c^T x : Ax \leq b, x \geq 0\} \quad (2.1)$$

dove il numero di variabili del sistema è talmente elevato da escludere che lo si possa manipolare nella sua interezza in tempi di calcolo ragionevoli. In tal caso si devono trattare in modo implicito le variabili del sistema. In generale supponiamo che la matrice  $A$  abbia  $m$  righe ed  $n$  colonne con  $n$  molto grande, escludendo però il caso di problema illimitato.

Definiamo ora un problema parziale (master ridotto)

$$\max\{\gamma^T z : Dz \leq b, z \geq 0\} \quad (2.2)$$

con  $D$  sottomatrice di  $A$  con  $m$  righe e  $q \ll n$  colonne e  $\gamma$  sottovettore di  $c$  associato alle colonne di  $D$ .

Una volta selezionato opportunamente un problema parziale iniziale, il metodo prosegue risolvendo una sequenza di problemi parziali, ognuno ottenuto aggiungendo al precedente una colonna ricavata, appunto, con un opportuno algoritmo di generazione di colonne. In questo algoritmo assumiamo che ogni problema parziale sia ammissibile e limitato e chiamiamo con  $z^o$  e  $\lambda^o$  la coppia di soluzioni ottime primale e duale, dove il problema duale, detto anche di pricing, è ottenuto dal primale utilizzando i costi ridotti negativi di un gruppo di vincoli.

Osserviamo che la soluzione  $x^o = z^o, z \geq 0$  è una soluzione ammissibile per il problema primale completo. Se la soluzione  $\lambda^o$  fosse ammissibile per il problema duale completo, allora per il teorema della dualità, le soluzioni  $x^o$  e  $\lambda^o$  sarebbero ottime rispettivamente per i problemi primale e duale completi.

Al contrario se  $\lambda^o$  non è ammissibile per il problema duale completo, allora esiste almeno un vincolo duale  $a_j^T \lambda \leq c_j$  violato da  $\lambda^o$ , ovvero tale che  $a_j^T \lambda^o > c_j$ ; ovviamente il vincolo violato non può appartenere al duale parziale perché  $\lambda^o$  è per ipotesi ammissibile per tale problema. Si decide quindi secondo il simplesso dinamico di aggiungere al problema duale parziale il vincolo violato. Ciò corrisponde nel problema primale parziale ad aggiungere la colonna corrispondente alla variabile  $x_j$  alla matrice  $D$ . Si passa poi a risolvere di nuovo con il simplesso il problema primale così aumentato.

Il vincolo duale  $j$ -esimo viene violato se e solo se il costo ridotto  $c_j - a_j^T \lambda^o$  è negativo. Lo scopo dell'algoritmo di generazione di colonne è quindi quello di individuare, se esiste, una colonna del problema primale completo con costo ridotto negativo rispetto alla soluzione duale  $\lambda^o$ , ovvero un vincolo del problema duale completo violato da tale soluzione. Ciò deve essere fatto in maniera implicita, senza calcolare tutti i costi ridotti.

Per garantire che il problema parziale sia sempre ammissibile basta scegliere opportunamente il problema parziale con cui partire, cioè la corrispondente matrice iniziale  $D_o$ . Si inizia quindi con l'individuare (euristicamente) una soluzione ammissibile del problema primale completo  $x^h$  e si prende come  $D_o$  la sottomatrice di  $A$  le cui colonne corrispondono alle componenti di tale soluzione. Siccome si procede aggiungendo colonne alla matrice  $D$ , essa conterrà sempre  $D_o$  come sottomatrice così che il vettore  $x^h$  continuerà ad essere ammissibile e quindi a garantire l'ammissibilità del problema parziale, come richiesto.

Se procedendo in questo senso si arriva alla situazione in cui tutte le colonne del problema completo hanno costo ridotto positivo o nullo il vettore  $x^o = z^o$  è una soluzione ottima del problema completo originario.

In figura 2.1 è riportato schematicamente il funzionamento dell'algoritmo di generazione di colonne. Dal problema master completo si ricava un master ridotto dal cui rilassamento si estraggono i coefficienti  $\pi$  che vengono

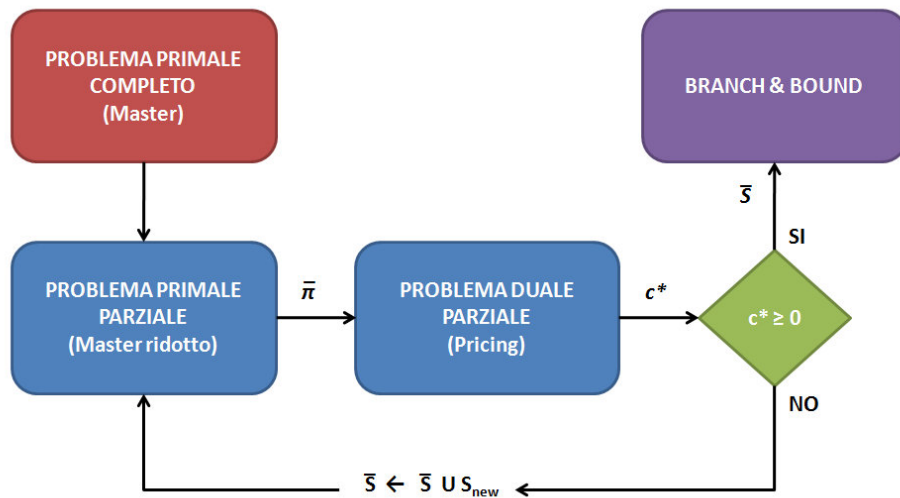


Figura 2.1: Schema di funzionamento della generazione di colonne ( $\bar{\pi}$  indica il vettore dei coefficienti duali,  $c^*$  i costi ridotti)

utilizzati per ottimizzare la funzione obiettivo del pricing. Quindi si controlla il valore ritornato dal pricing per scoprire se esistono vincoli duali violati. In questo caso si procede con l'aggiunta della nuova colonna, altrimenti si risolve il problema master ridotto intero.

L'algoritmo di generazione di colonne è specifico per ciascun problema ed è proprio nella sua costruzione che risiede la difficoltà di questo metodo.

Nel caso di problemi di Programmazione Lineare, l'algoritmo di generazione di colonne determina la soluzione ottima del problema. Nel caso di problemi di Programmazione Lineare Intera, l'algoritmo applicato al rilassamento lineare del problema può terminare con una soluzione frazionaria che sarà un upper bound o un lower bound in base al tipo di problema. Incorporando la generazione di colonne in un algoritmo di Branch & Bound si può arrivare a determinare la soluzione ottima intera. Tale metodo prende il nome di Branch & Price. Il fatto che in alcuni casi la convergenza risulti lenta, può dipendere dall'esistenza di molte soluzioni di base degeneri. Può infatti verificarsi che l'aggiunta di una colonna all'insieme del problema primale parziale non determini alcun miglioramento nel valore ottimo del problema.

Questo è il prezzo che si paga nella risoluzione di problemi complessi e di

dimensioni elevate, che senza questo metodo non sarebbe nemmeno pensabile affrontare.

## 2.3 Algoritmi euristici

Spesso la determinazione della soluzione ottima di un problema complesso può risultare troppo onerosa in termini di tempo di calcolo. Quindi, la necessità di risolvere il problema velocemente impedisce l'utilizzo di approcci esatti; nasce in questo caso l'esigenza di sviluppare delle tecniche in grado comunque di fornire delle buone soluzioni ammissibili in tempi ridotti, sacrificando l'ottimalità della soluzione in favore dell'efficienza con cui questa viene calcolata.

Tipicamente, la determinazione di buone soluzioni è sufficiente nelle applicazioni reali (in particolare per problemi di grandi dimensioni). Ciò è dovuto ad una serie di fattori:

- molti dei parametri in gioco nelle applicazioni reali sono delle stime che possono essere soggette ad errore, per cui non vale la pena di aspettare troppo tempo per avere una soluzione il cui valore (o la cui ammissibilità) è di valutazione incerta;
- spesso si è interessati ad avere una possibile soluzione per il problema in esame al fine di valutare velocemente degli scenari di lavoro (fase operativa);
- spesso si lavora in tempo reale, per cui si vuole avere una buona soluzione ammissibile in tempi molto ridotti (minuti o secondi);
- a volte le applicazioni reali presentano molti vincoli complessi, cioè difficilmente rappresentabili con modelli di programmazione lineare intera (PLI), per cui è impossibile pensare ad algoritmi esatti basati su tali formulazioni.

Per ottenere delle buone soluzioni approssimate si utilizzano algoritmi euristici, ossia un qualunque metodo che fornisca una soluzione per un problema.

Generalmente si vuole che un algoritmo euristico richieda un tempo di calcolo polinomiale nella dimensione dell'istanza che deve risolvere, ma per problemi complessi in realtà si rilassa questo vincolo, richiedendo solo che l'algoritmo sia sufficientemente veloce. Si deve tener presente che un algoritmo euristico può essere anche in grado di determinare una soluzione ottima di un problema, ma non può in generale dimostrarne l'ottimalità. E' tuttavia importante, se possibile, fare una stima di quanto ci si discosta dall'ottimo.

## 2.4 Strumenti utilizzati

Un solver (o risolutore) è un software che riceve in input una descrizione di un problema di ottimizzazione e produce in output la soluzione ottima del modello e informazioni ad essa collegate. Esistono numerosissimi solver commerciali sia per applicazioni specifiche che per un utilizzo ad ampio spettro. Questi ultimi utilizzano di solito algoritmi di simplesso più o meno raffinati per la programmazione lineare e Branch & Bound o suoi derivati per i problemi interi. CPLEX, uno dei solver più noti, e che abbiamo utilizzato nel nostro lavoro utilizza un Branch & Cut con l'aggiunta di varie soluzioni che ne ottimizzano la performance.

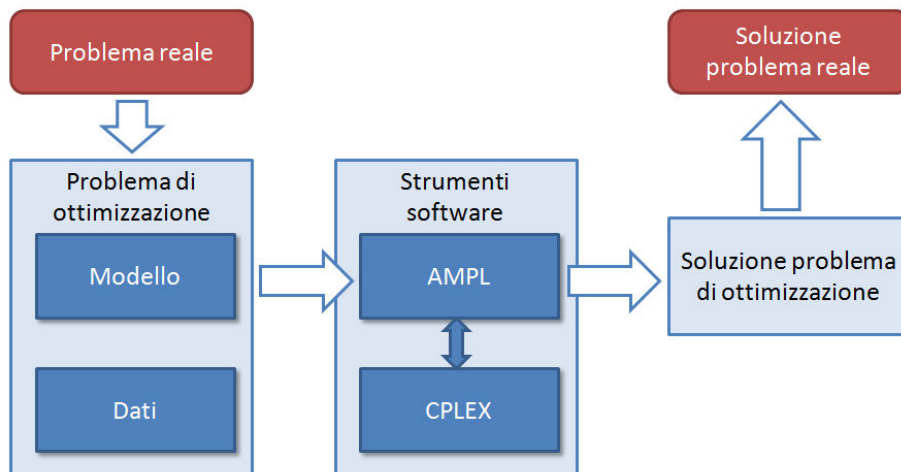


Figura 2.2: Schema del processo di ottimizzazione

I generatori algebrici di modelli, fra cui AMPL, costituiscono un'interfaccia che semplifica l'interazione tra l'utente e il solver. Le caratteristiche principali di questi software sono:

- fornire un linguaggio semplice per descrivere modelli complessi, che sia contemporaneamente
  - ad alto livello, cioè comprensibile all'utente;
  - formalmente strutturato, cioè accessibile a un risolutore;
- permettere all'utente di comunicare con il risolutore attraverso file di testo anziché attraverso strutture dati, in modo da non richiedergli conoscenze informatiche approfondite e da poter formulare il modello con un semplice editor, qualunque sia la piattaforma su cui viene scritto e quella su cui viene risolto;
- permettere all'utente di comunicare con diversi risolutori, in modo da poter sfruttare quelli disponibili;
- tenere distinti la struttura logica del modello (variabili, funzioni obiettivo, vincoli e le loro relazioni) dai dati numerici.

Quest'ultimo punto è molto rilevante in quanto evita che piccole modifiche nella struttura del modello o nei dati comportino di riscrivere tutto: diventa infatti semplice usare lo stesso modello su dati differenti o applicare modelli diversi allo stesso problema.

## Capitolo 3

# Modelli ed algoritmi per l'ottimizzazione della collaborazione

La base su cui si fonda l'ottimizzazione è la stesura di un modello del problema. Esso deve ricalcare il più possibile la situazione reale introducendo tuttavia le semplificazioni matematiche necessarie alla sua trattazione analitica. La qualità del modello è quindi fondamentale per ottenere dei buoni risultati.

### 3.1 La rete

Consideriamo una rete radio multihop con nodi in posizioni note e fissi, dotati di antenne omnidirezionali e che accedono al mezzo radio in modalità TDMA. In tale scenario l'orizzonte temporale è diviso in trame ripetute ciclicamente, ciascuna suddivisa in slot a cui sono assegnati i link di comunicazione. Link diversi possono utilizzare lo stesso time slot nel caso in cui le due o più trasmissioni non siano tra di loro interferenti, ovvero siano sufficientemente distanti nello spazio. Per questo motivo spesso si parla di STDMA (spatial TDMA). Sono assegnate inoltre delle domande di traffico  $d \in D$  definite da un nodo sorgente e da uno di destinazione che devono essere soddisfatte.



L'obiettivo è quindi quello di minimizzare la lunghezza totale della trama cercando di sfruttare il più possibile le trasmissioni contemporanee.

La rete si può modellizzare con un grafo  $G = (N, L)$  costituito da  $N$  nodi e da  $L$  archi direzionati  $(i, j)$  tra di essi. Un arco tra due nodi esiste se la trasmissione è ottenibile con un livello di SNR accettabile  $\gamma$ . Si determinano quindi i guadagni  $g_{ij}$  tra le coppie di nodi e vengono tracciati solo gli archi tali che:

$$SNR_{ij} = \frac{P_v g_{ij}}{\eta} \geq \gamma$$

## 3.2 Applicazione della generazione di colonne

La risoluzione del modello dovrà considerare le possibili combinazioni di archi che possono coesistere in una configurazione. Si può dimostrare facilmente che il numero di configurazioni cresce esponenzialmente con il numero di nodi e la densità del grafo considerato. Un approccio risolutivo efficiente è quello di utilizzare la generazione di colonne in modo da considerare solo le combinazioni effettivamente necessarie tralasciando tutte le altre. L'applicazione di tale metodo per le reti radio multihop utilizzando formulazioni basate sul SINR è stato proposto in [4] e sviluppato in [7].

La generazione di colonne consiste in un problema master che sceglie la combinazione migliore da un pool di colonne generate attraverso un problema detto di pricing. Il risultato ottenuto è un rilassamento del problema iniziale che dovrebbe essere incorporato in una procedura di Branch & Price per il calcolo dell'ottimo intero. Tuttavia nei casi da noi considerati, terminata la generazione di colonne e calcolato l'upper bound risolvendo il master all'ottimo intero, vedremo che i due bound coincidono. In questa trattazione analizzeremo diverse strategie per formulare il problema di pricing; possiamo tuttavia stilare un unico modello master con valenza generale come proposto in [7].

Introduciamo le variabili intere positive  $\lambda_s$  per ogni configurazione  $s \in S$  che rappresentano il numero di slot occupati dalla configurazione stessa. Dato che solo una configurazione può essere attiva in un time slot la quantità da minimizzare è la somma delle variabili  $\lambda_s$ . Definiamo inoltre le variabili binarie di flusso  $f_{ij}^d$  che indicano se l'arco  $(i, j)$  è utilizzato per instradare i pacchetti della demand  $d$  e i relativi parametri  $b_i^d$  che assumono valori  $\{1, 0, -1\}$  a seconda che il nodo  $i$  sia sorgente, destinazione o nodo di transito.

Il master è quindi formulato in programmazione lineare intera come segue.

$$\min \sum_{s \in S} \lambda_s \quad (3.1)$$

$$s.t. \sum_{(i,j) \in L} f_{ij}^d - \sum_{(j,i) \in L} f_{ji}^d = b_i^d, \quad \forall i \in V, d \in D \quad (3.2)$$

$$\sum_{s \in S_{ij}} \lambda_s \geq \sum_{d \in D} f_{ij}^d, \quad \forall (i, j) \in L \quad (3.3)$$

$$f_{ij}^d \in \{0, 1\}, \quad \forall (i, j) \in L, d \in D \quad (3.4)$$

$$\lambda_s \in Z^+, \quad \forall s \in S \quad (3.5)$$

La funzione obiettivo (3.1) minimizza il numero di time slots. I vincoli (3.2) bilanciano il flusso in ogni nodo per ciascuna demand  $d$  e i vincoli (3.3) assicurano che ci siano configurazioni sufficienti a soddisfare le domande di traffico su ogni arco.

### 3.3 Il pricing

Nello sviluppo del problema di pricing abbiamo valutato tre vie alternative: una basata sulla valutazione del SINR, come presentata in [7], che ci consente di ottenere soluzioni ottime, una semplificata che chiameremo protocollare e un'euristica che utilizza il modello SINR. In tutte le formulazioni per ora considereremo fissa la potenza erogata da ogni nodo.

### 3.3.1 Pricing con SINR

Il pricing con SINR è quello che rispecchia maggiormente la realtà fisica della trasmissione in quanto si basa sull'interferenza mutua tra i canali e sul rumore termico che vengono esplicitati attraverso dei vincoli nel problema di pricing. Il SINR sul link  $(i, j)$  è definito come:

$$SINR_{ij} = \frac{P_i g_{ij}}{\eta + \sum_{l \in N \setminus \{i, j\}} P_l g_{lj}} \geq \gamma$$

dove  $P_i$  è la potenza trasmessa dal nodo  $i$ .

Definiamo le variabili  $z_{ij}$  binarie per indicare se l'arco  $(i, j)$  appartiene o meno alla configurazione ottima e i parametri  $\sigma_{ij}$  che sono il duale dei vincoli (3.3).

$$\max \sum_{(i,j) \in L} \sigma_{ij} z_{ij} \quad (3.6)$$

$$s.t. \sum_{(i,j) \in L} z_{ij} + \sum_{(j,i) \in L} z_{ji} \leq 1, \quad \forall i \in N \quad (3.7)$$

$$P_i g_{ij} \geq \gamma \left( \eta + \sum_{\substack{h \in N \setminus \{i, j\} \\ l \in N \setminus \{i, j\}}} P_h g_{hj} z_{hl} \right) z_{ij}, \quad \forall (i, j) \in L \quad (3.8)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in L \quad (3.9)$$

La funzione obiettivo (3.6) massimizza il numero di configurazioni prese pesandole sui parametri duali  $\sigma_{ij}$ . I vincoli (3.7) implicano che per ciascun nodo si può attivare al massimo un arco o entrante o uscente da esso, mentre i vincoli (3.8) forzano il rispetto della condizione di SINR. Questo ultimo gruppo di vincoli non è lineare e di conseguenza dovrà essere linearizzato attraverso un parametro  $M$  come segue:

$$P_i g_{ij} \geq \gamma \left( \eta + \sum_{\substack{h \in N \setminus \{i,j\} \\ l \in N \setminus \{i,j\}}} P_h g_{hj} z_{hl} \right) - M_{ij}(1 - z_{ij}), \quad \forall (i, j) \in L \quad (3.10)$$

L'efficacia di questi vincoli dipende dalla scelta delle  $M_{ij}$  che devono essere le più stringenti possibili in modo da facilitare la risoluzione del problema ed evitare di incorrere in problemi di precisione numerica, infatti i numeri in gioco sono di molti ordini di grandezza diversi tra loro e alcuni sono vicini alla precisione numerica del solver.

### 3.3.2 Pricing protocollare

L'utilizzo dei vincoli di SINR risulta complesso poiché questi sono numerosi, devono essere linearizzati e per la precisione numerica sopra citata. Per ovviare a questi problemi abbiamo elaborato una formulazione che non considera direttamente l'interferenza nei vincoli e che definiremo protocollare.

Questa si basa sulla definizione di una lista di adiacenza  $\Delta_{ij}$ , calcolata a priori per ciascun arco  $(i, j)$  del grafo, che contiene tutti i nodi che, se attivati singolarmente, causerebbero interferenza alla comunicazione. Ovviamente un modello così fatto è un'approssimazione di (3.3.1) quindi i risultati saranno euristici e tanto più fedeli quanto più la definizione di adiacenza sarà coerente con la formulazione SINR.

$$\max \sum_{(i,j) \in L} \sigma_{ij} z_{ij} \quad (3.11)$$

$$s.t. \sum_{(i,j) \in L} z_{ij} + \sum_{(j,i) \in L} z_{ji} \leq 1, \quad \forall i \in N \quad (3.12)$$

$$\sum_{(l,k) \in L: l \in \Delta_{ij}} z_{lk} \leq M_{ij}(1 - z_{ij}), \quad \forall (i, j) \in L : |\Delta_{ij}| > 0 \quad (3.13)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in L \quad (3.14)$$

La funzione obiettivo (3.11) e il primo gruppo di vincoli (3.12) sono i medesimi della versione SINR e i vincoli (3.13) forzano la disattivazione dei nodi contenuti in  $\Delta_{i,j}$  quando l'arco  $(i, j)$  è attivo. Anche essi sono linearizzati come in (3.10), tuttavia la scelta della  $M_{ij}$  non è ugualmente critica poiché si lavora con numeri interi e nello stesso ordine di grandezza.

Il problema principale che si può riscontrare nei modelli protocollari proposti in letteratura è la difficoltà nel tener conto di interferenze multiple per la determinazione delle adiacenze. La definizione più semplice possibile dell'adiacenza, presente spesso in letteratura, è:

$$\Delta_{ij} = \{l \in N : \exists (l, j) \in L, l \neq \{i, j\}\},$$

che è l'insieme dei nodi direttamente connessi al ricevitore.

Tuttavia si può verificare con delle prove che tale definizione da risultati spesso non ammissibili secondo il SINR in quanto possono verificarsi casi in cui nodi anche non direttamente connessi a  $j$  sono abbastanza vicini da causare interferenza elevata al ricevitore.

Per ottenere un'adiacenza più coerente con la realtà si è utilizzato il modello SINR per valutare l'interferenza media  $\Phi$  nelle configurazioni ottime e da questo è emerso che, su istanze delle medesime dimensioni,  $\Phi$  si mantiene abbastanza costante da essere ridotta a un unico parametro.

Semplifichiamo la definizione di SINR in modo da considerare un unico interferente per volta scaricando su  $\Phi$  le interferenze aggiuntive viste dal ricevitore. Otteniamo:

$$\frac{Pg_{ij}}{\Phi + \eta + Pg_{lj}} \geq \gamma,$$

e invertendola:

$$g_{lj} \leq g_{ij} - \tau$$

dove

$$\tau = \frac{\gamma(\Phi + \eta)}{P}$$

$\tau$  è un margine sul SINR, determinato dall'interferenza media, che permette di definire un'area intorno al nodo  $j$  nella quale l'attivazione di un altro nodo trasmittente non è accettabile. La nuova definizione di adiacenza è quindi:

$$\Delta_{ij} = \{l \in N : l \neq \{i, j\}, (\exists(l, k) \in L : k \neq \{i, j\}), g_{kj} > g_{ij} - \tau\}$$

Utilizzando tale adiacenza si ottengono un maggior numero di colonne valide e soluzioni più vicine a quelle con la formulazione SINR. Per avere sempre e solo soluzioni ammissibili è necessario un settaggio fine della  $\Phi$  sulla specifica istanza e di altri accorgimenti di cui parleremo in seguito.

### 3.4 Pricing con collaborazione

I modelli di pricing visti possono essere applicati a uno scenario di collaborazione con modifiche ridotte introducendo un grafo di collaborazione come mostrato in [8]. Il grafo di collaborazione  $G'_k = (V, A)$  è un grafo contenente tutti i nodi del grafo originale più i supernodi fino a cardinalità massima  $k$ , con archi e superarchi a collegarli. Per  $k = 1$  ci riconduciamo al grafo originale, per  $k = 2$  consideriamo supernodi costituiti da coppie di nodi semplici ed è il caso che prenderemo in analisi, poichè  $k > 2$  comporta la creazione di grafi talmente grandi da renderne difficile l'utilizzo. Nonostante il modello rimanga simile al caso non collaborativo la complessità aumenta parallelamente alle dimensioni del grafo. Per questo motivo oltre a una modalità risolutiva affine a quella vista sul caso non collaborativo, proporremo due soluzioni euristiche semplificate: una che punta a ridurre la complessità del problema di pricing e una che invece semplifichi il grafo di collaborazione.

D'ora in poi indicheremo per semplicità i nodi con gli indici  $v$  e  $w$ , i supernodi con  $i$  e  $j$ , gli archi con  $(v, w)$  e i superarchi con  $(i, j)$ . L'insieme dei nodi appartenenti a un supernodo  $i$  sarà definito come  $\Gamma(i)$  e gli archi

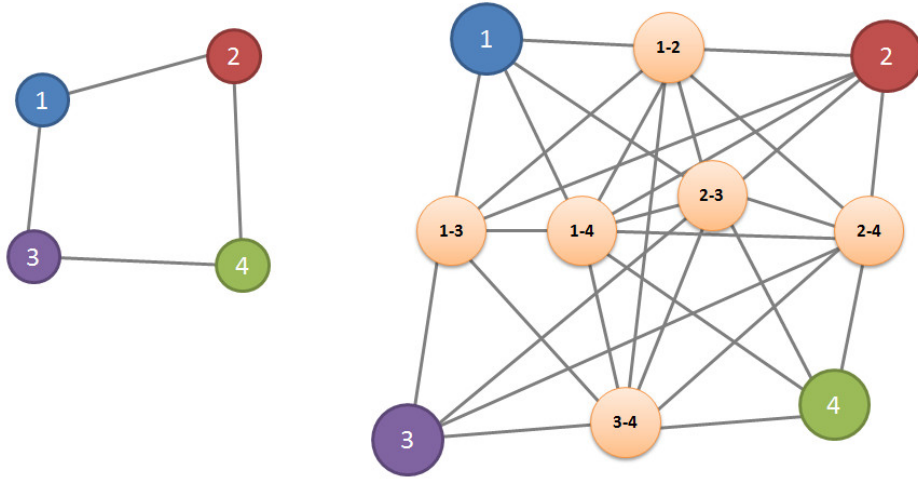


Figura 3.1: Esempio di grafo(sinistra) e relativo supergrafo(destra)

appartenenti a un superarco  $(i, j)$  come  $\Lambda(i, j)$ . L'insieme  $A$  comprende ora sia archi semplici che superarchi.

Per l'esistenza di un superarco  $(i, j)$  deve valere la seguente condizione di SNR per tutti i nodi appartenenti al supernodo ricevente,  $w \in \Gamma(j)$ :

$$SNR_{iw} = \frac{\sum_{v \in \Gamma(i)} P_v g_{vw}}{\eta} \geq \gamma.$$

In base al tipo di supernodi e a questa condizione si possono definire diversi tipi di superarchi:

1. *One-to-one*: sono i link originali che collegano nodi semplici, ovvero  $\Gamma(i) = \{v\}$ ,  $\Gamma(j) = \{w\}$  e  $(v, w) \in L$ .
2. *One-to-many*: collega un nodo singolo a un supernodo, viene chiamato anche *broadcast link* ed è definito come  $\Gamma(i) = \{v\}$ ,  $|\Gamma(j)| > 1$  e  $(v, w) \in L \forall w \in \Gamma(j)$ . A questa categoria appartengono anche i cosiddetti *buffering link*, in cui nel supernodo ricevente è compreso lo stesso nodo che trasmette.
3. *Many-to-one*: va da un supernodo a un nodo semplice per cui  $|\Gamma(i)| > 1$ ,  $\Gamma(j) = w$  con  $w \in N \setminus \Gamma(i)$ . In questo caso si deve notare che le

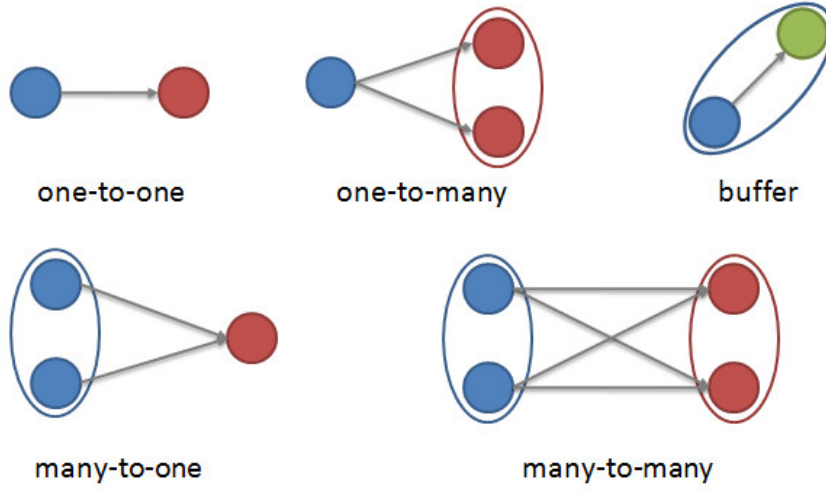


Figura 3.2: Tipi di links nel supergrafo

coppie  $(v, w)$  non corrispondono necessariamente ad archi della topologia originale in quanto l'SNR al nodo ricevitore è determinato da una somma di potenza. Questo tipo link è definito come *cooperating link*.

4. *Many-to-many*: indica un collegamento tra supernodi e un arco appartiene a questa classe se  $|\Gamma(i)| > 1$  e  $|\Gamma(j)| > 1$ . Anche in questo caso possono esserci coppie di nodi non direttamente collegate da archi della topologia originaria. Questo tipo di arco è definito anche come *broad-cooperating link* o *multicasting link*.

### 3.4.1 Pricing SINR con collaborazione

Il modello senza collaborazione si estende al caso collaborativo considerando gli stessi vincoli ma adattati al supergrafo:

$$\max \sum_{(i,j) \in A} \sigma_{ij} z_{ij} \quad (3.15)$$

$$s.t. \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} + \sum_{\substack{(j,i) \in A: \\ v \in \Gamma(i)}} z_{ji} \leq 1, \quad \forall v \in N \quad (3.16)$$



$$\sum_{v \in \Gamma(i)} P_v g_{vw} \geq \gamma \left( \eta + \sum_{\substack{(k,l) \in A, u \in \Gamma(k): \\ k \neq i, l \neq j}} P_u g_{uw} z_{lk} \right) - M_{ij}^w (1 - z_{ij}),$$

$$\forall (i, j) \in A, w \in \Gamma(j) \quad (3.17)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A \quad (3.18)$$

La funzione obiettivo (3.15) rimane esattamente la stessa, il primo gruppo di vincoli (3.16) implica che un supernodo può o trasmettere o ricevere e che ogni nodo semplice può essere contenuto in un solo supernodo attivo nella configurazione. Per i vincoli di SINR (3.17), ad ogni nodo appartenente a un supernodo ricevente, la somma delle potenze ricevute dai nodi contenuti nel supernodo trasmittente deve superare di  $\gamma$  volte quella ricevuta dai nodi interferenti. Anche in questo caso i vincoli di SINR non sono lineari e sono stati linearizzati con una  $M_{ijw}$  abbastanza stringente per non incorrere nei problemi di precisione già citati.

In questa formulazione i vincoli di SINR sono molto numerosi, infatti c'è un vincolo per ogni superarco che corrisponde nel caso peggiore a  $O(|N|^4)$ . Lo stesso vale anche per le variabili  $z_{ij}$  anch'esse definite sui superarchi. Quindi è utile introdurre una formulazione alternativa, proposta sempre in [8], che utilizza nuove variabili riducendo però il numero di vincoli. Le nuove variabili sono:

$$q_{vw} = \begin{cases} 1 & \text{se il nodo } v \text{ trasmette a } w \\ 0 & \text{altrimenti} \end{cases}$$

$$q'_{vw} = \begin{cases} 1 & \text{se il nodo } v \text{ causa interferenza a } w \\ 0 & \text{altrimenti} \end{cases}$$

$$x_v = \begin{cases} 1 & \text{se il nodo } v \text{ trasmette} \\ 0 & \text{altrimenti} \end{cases}$$

$$y_w = \begin{cases} 1 & \text{se il nodo } w \text{ riceve} \\ 0 & \text{altrimenti} \end{cases}$$

Il modello diventa:

$$\max \sum_{(i,j) \in A} \sigma_{ij} z_{ij} \quad (3.19)$$

$$s.t. \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} + \sum_{\substack{(j,i) \in A: \\ v \in \Gamma(i)}} z_{ji} \leq 1, \quad \forall v \in N \quad (3.20)$$

$$\sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i), w \in \Gamma(j)}} z_{ij} = q_{vw}, \quad \forall v \in N, w \in N, v \neq w \quad (3.21)$$

$$x_v = q_{vw} + q'_{vw}, \quad \forall v \in N, w \in N, v \neq w \quad (3.22)$$

$$q_{vw} \leq y_w, \quad \forall v \in N, w \in N, v \neq w \quad (3.23)$$

$$\sum_{v \in N} P_v g_{vw} q_{vw} \geq \gamma \left( \eta + \sum_{u \in N} P_u g_{uw} q'_{uw} \right) - M_w (1 - y_w), \quad \forall w \in N \quad (3.24)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A \quad (3.25)$$

$$q_{vw}, q'_{vw} \in \{0, 1\}, \quad \forall v \in N, w \in N, v \neq w \quad (3.26)$$

$$x_v, y_v \in \{0, 1\}, \quad \forall v \in N \quad (3.27)$$

I vincoli (3.21) implicano che  $v$  può trasmettere a  $w$  solo se  $(v, w) \in \Lambda(i, j)$ , i vincoli (3.22) che un nodo  $v$  attivo o trasmette o interferisce con un altro nodo  $w$ , i vincoli (3.23) che se è attivo l'arco  $(v, w)$  il nodo  $w$  è ricevitore, e infine i vincoli (3.24) sono quelli di SINR linearizzati come in (3.10) e con le medesime considerazioni fatte nei precedenti modelli di SINR per il calcolo del parametro  $M$ .

Si può constatare che in questo caso il numero dei vincoli è circa  $O(|N|^2)$ , nettamente inferiore rispetto al modello iniziale. Inoltre il numero di variabili passa da  $O(|N|^4)$  a  $O(|N|^4 + 2|N|^2 + 2|N|) \approx O(|N|^4)$ , non introducendo quindi variazioni sensibili.

### 3.4.2 Pricing protocollare con collaborazione

Il modello protocollare con collaborazione rimane lo stesso visto per il caso non collaborativo, tranne gli adattamenti per considerare i supernodi.

$$\max \sum_{(i,j) \in A} \sigma_{ij} z_{ij} \quad (3.28)$$

$$s.t. \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} + \sum_{\substack{(j,i) \in A: \\ v \in \Gamma(i)}} z_{ji} \leq 1, \quad \forall v \in N \quad (3.29)$$

$$\sum_{(l,k) \in A: l \in \Delta_{ij}} z_{lk} \leq M_{ij}(1 - z_{ij}), \quad \forall (i,j) \in A : |\Delta_{ij}| > 0 \quad (3.30)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i,j) \in A \quad (3.31)$$

Tuttavia deve essere adattata la definizione di adiacenza  $\Delta_{ij}$ , poiché in questa lista si deve tener conto dei nodi semplici interferenti e dei supernodi che contengono almeno un nodo interferente con  $j$ .

$$\begin{aligned} \Delta_{ij} = \{ & l \in V : l \neq \{i, j\}, \exists (l, k) \in A : \\ & (k \neq \{i, j\}, \exists w \in \Gamma(j) : (\sum_{u \in \Gamma(l)} g_{uw} \gamma > \sum_{v \in \Gamma(i)} g_{vw} - \tau)) \} \end{aligned}$$

Nel caso collaborativo, nonostante l'eliminazione del problema di precisione numerica, la performance è comunque penalizzata dal fatto che i vincoli (3.30) crescono proporzionalmente al numero di superarchi, quindi come  $O(|N|^4)$ . Perciò, considerando istanze di dimensioni crescenti, si nota come il gap di prestazione fra questo modello e quello di SINR va ad assottigliarsi. Un'ulteriore complicazione è costituita dalle numerose liste di adiacenza (una per arco quindi nel caso peggiore  $|N|^4$ ), ciascuna delle quali può contenere molti supernodi (nel caso peggiore il numero totale di supernodi quindi  $|N|^2$ ).

Si rende quindi necessaria un'ulteriore semplificazione del problema ottenuta utilizzando un metodo simile a quello visto in (3.19), ovvero cercando di

fondare la formulazione sulle coppie di nodi singoli invece che sui superarchi. Le variabili utilizzate sono quelle viste in precedenza e il modello risultante è:

$$\max \sum_{(i,j) \in A} \sigma_{ij} z_{ij} \quad (3.32)$$

$$s.t. \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} + \sum_{\substack{(j,i) \in A: \\ v \in \Gamma(i)}} z_{ji} \leq 1, \quad \forall v \in N \quad (3.33)$$

$$\sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i), w \in \Gamma(j)}} z_{ij} = q_{vw}, \quad \forall v \in N, w \in N, v \neq w \quad (3.34)$$

$$x_v = q_{vw} + q'_{vw}, \quad \forall v \in N, w \in N, v \neq w \quad (3.35)$$

$$q_{vw} \leq y_w, \quad \forall v \in N, w \in N, v \neq w \quad (3.36)$$

$$\sum_{u \in N: u \in \Delta_{vw}} q'_{uw} \leq (1 - q_{vw} M_{vw}), \quad \forall v \in N, w \in N, v \neq w \quad (3.37)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A \quad (3.38)$$

$$q_{vw}, q'_{vw} \in \{0, 1\}, \quad \forall v \in N, w \in N, v \neq w \quad (3.39)$$

$$x_v, y_v \in \{0, 1\}, \quad \forall v \in N \quad (3.40)$$

I vincoli hanno le stesse funzioni spiegate prima e a quelli di SINR sono stati sostituiti i vincoli (3.37) che quando un arco è attivo ( $q_{vw} = 1$ ) spengono tutti i nodi che gli fanno interferenza. Anche in questo caso il numero dei vincoli va come  $O(|N|^2)$ .

L'adiacenza è stata adattata a questo nuovo problema e risulta:

$$\Delta_{vw} = \{u \in N : u \neq \{v, w\}, g_{uw}\gamma > g_{vw} - \tau\}$$

I risultati che si ottengono con questo modello non sono perfettamente coincidenti con quelli ottenibili con il modello protocollare visto in (3.28), poichè invece di considerare le adiacenze sui superarchi consideriamo le adiacenze sulle coppie di nodi, ovvero:

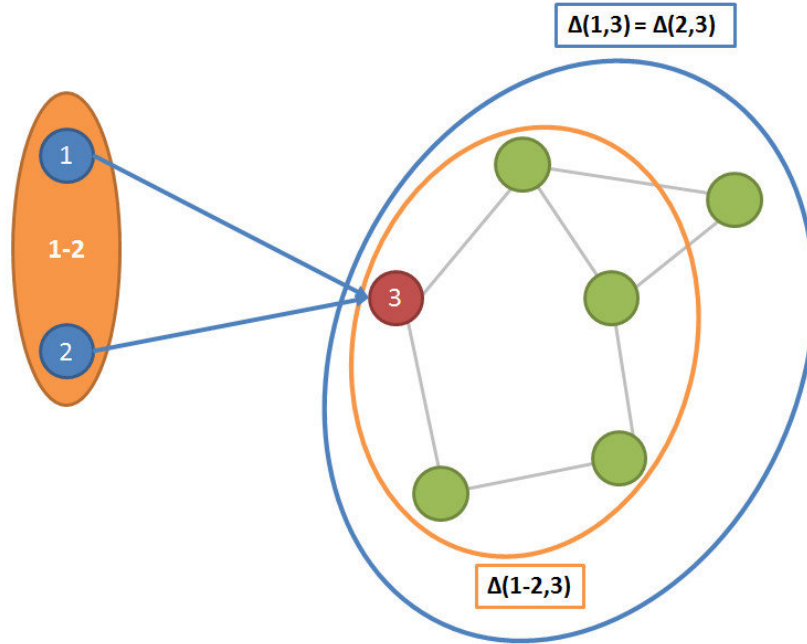


Figura 3.3: Esempio di adiacenza semplificata

$$\Delta_{ij} \neq \bigcup_{\substack{v,w \in N, v \neq w: \\ v \in \Gamma(i), w \in \Gamma(j)}} \Delta_{vw},$$

come si può dedurre dalla figura 3.3, dove si nota che in questo modo l'adiacenza può risultare sovradimensionata. Tuttavia abbiamo verificato sperimentalmente e mostreremo in seguito che nonostante l'ulteriore approssimazione la qualità delle soluzioni non peggiora in modo rilevante, ma i tempi di risoluzione si riducono drasticamente.

### 3.4.3 Accorgimenti per il pricing protocollare

Per migliorare ulteriormente la performance del pricing protocollare con collaborazione abbiamo introdotto delle modifiche aggiuntive:

- *Phi adattiva*: per rendere il calcolo delle adiacenze ancora più flessibile abbiamo deciso di utilizzare, invece di un unico parametro  $\Phi$ , dei parametri  $\Phi_{vw}$  in modo da poter considerare differenti condizioni di in-

terferenza media, determinate dalla posizione fisica del nodo. I valori di  $\Phi_{vw}$  vengono inizializzati tutti a un valore molto basso e quando viene generata una colonna si controlla se sottostà ai vincoli di SINR; in caso negativo le  $\Phi_{vw}$  degli archi che violano il vincolo vengono aumentate di un fattore  $k$ . In questo modo si parte con delle adiacenze piccole per ogni nodo e vengono espanse solo quelle per cui si sperimenta un'interferenza troppo elevata.

- *Ricavo colonna ammissibile*: quando si ottiene una colonna non ammissibile con il pricing protocollare, invece che non aggiungere nulla e aumentare solamente le  $\Phi_{vw}$ , si può cercare di derivare una colonna ammissibile da quella generata. Per fare ciò togliamo sequenzialmente gli archi che violano i vincoli di SINR fino ad ottenere una soluzione; se essa ha price maggiore di uno viene aggiunta.
- *Aggiunta taglio*: per evitare che la generazione di colonne perda troppe iterazioni cercando di prendere le stesse configurazioni prima che l'aumento delle  $\Phi_{vw}$  le escluda, inseriamo nel problema di pricing un taglio che impedirà di prendere nuovamente la stessa colonna.

Utilizzando questi tre accorgimenti otteniamo un aggiustamento delle adiacenze più fine rendendo il modello più aderente a quello SINR e quindi con un'approssimazione più buona. D'altro canto ci permette di eliminare iterazioni inutili riducendo i tempi di risoluzione.

### 3.4.4 Euristiche con cammini minimi

Oltre al pricing protocollare abbiamo elaborato una seconda strategia euristica per ottenere buone soluzioni in tempi ragionevoli. La difficoltà intrinseca che si riscontra nel problema con collaborazione è dovuta alla dimensione del supergrafo il cui numero di archi cresce circa come  $|N|^4$ . In tal senso questa euristica mira a ridurre la dimensione dello stesso in modo da poter continuare ad utilizzare il modello SINR anche su istanze di dimensioni maggiori. Infatti il pricing continua ad essere quello presentato in 3.4.1, ma applicato

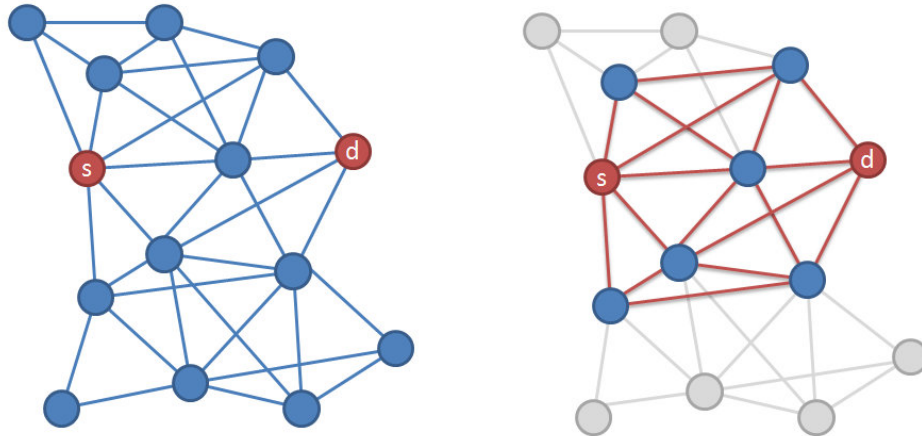


Figura 3.4: Esempio di grafo ridotto con i primi 10 cammini minimi

a un grafo molto ridotto e quindi con performances molto migliori in termini di tempo.

Per ciascuna delle domande di traffico del problema calcoliamo i primi  $k$  cammini minimi tra la sorgente e la destinazione risolvendo man mano problemi di cammino minimo e aggiungendo ad ogni iterazione un taglio che elimini la soluzione appena ottenuta. Tutti gli archi delle cammini trovati vengono aggiunti a un insieme  $A'$  che costituirà il supergrafo ridotto di cui riportiamo un esempio in figura 3.4.

L'utilizzo di questa euristica è giustificato dal fatto che l'obiettivo del problema è quello di minimizzare la lunghezza della trama da trasmettere e, quindi, le rotte più probabilmente utilizzate saranno quelle più corte. L'utilizzo di cammini di lunghezza maggiore, infatti, si ha solo nel tentativo di ridurre l'interferenza; tuttavia se eccessivamente lunghi portano, di contro, a un aumento della dimensione della trama.

L'individuazione del parametro  $k$  che offre il miglior compromesso tra qualità della soluzione e tempo non è semplice, tuttavia si può verificare sperimentalmente che già con poche rotte (5 o 10) la soluzione è vicina o coincide a quella ottima.

Nella figura 3.1 si può vedere che un altro parametro che influenza il comportamento di questa euristica è il numero delle domande di traffico, in

N° Istanza	Nodi	Domande	A	A'
1	10	5	314	104
2	10	20	501	292
3	15	5	635	92
4	15	20	610	190
5	20	5	1378	107
6	20	20	1358	250
7	25	5	6784	163
8	25	20	8528	670

Tabella 3.1: Esempio di riduzione archi nell'euristica utilizzando  $k = 20$

quanto la riduzione del grafo sarà maggiore quando queste sono poche. Si nota inoltre che all'aumento delle dimensioni del grafo la riduzione percentuale sul numero dei superarchi è maggiore, una caratteristica auspicabile per scalare verso istanze di dimensioni elevate.

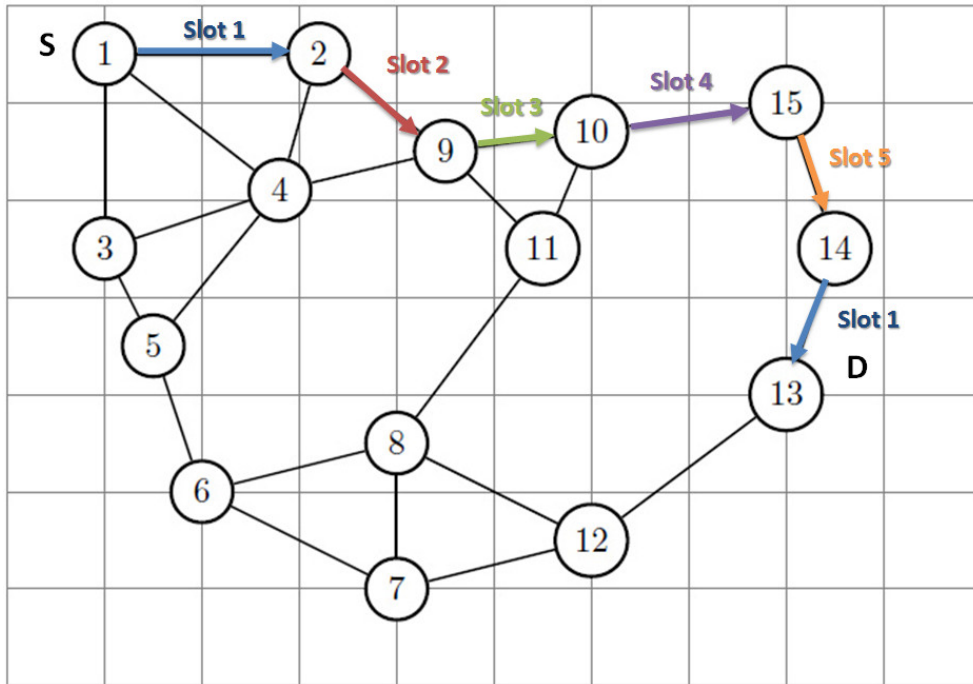
### 3.5 Esempio di collaborazione

In questa sezione presentiamo un esempio su un'istanza creata ad hoc per mettere in luce il meccanismo della collaborazione e i suoi potenziali vantaggi.

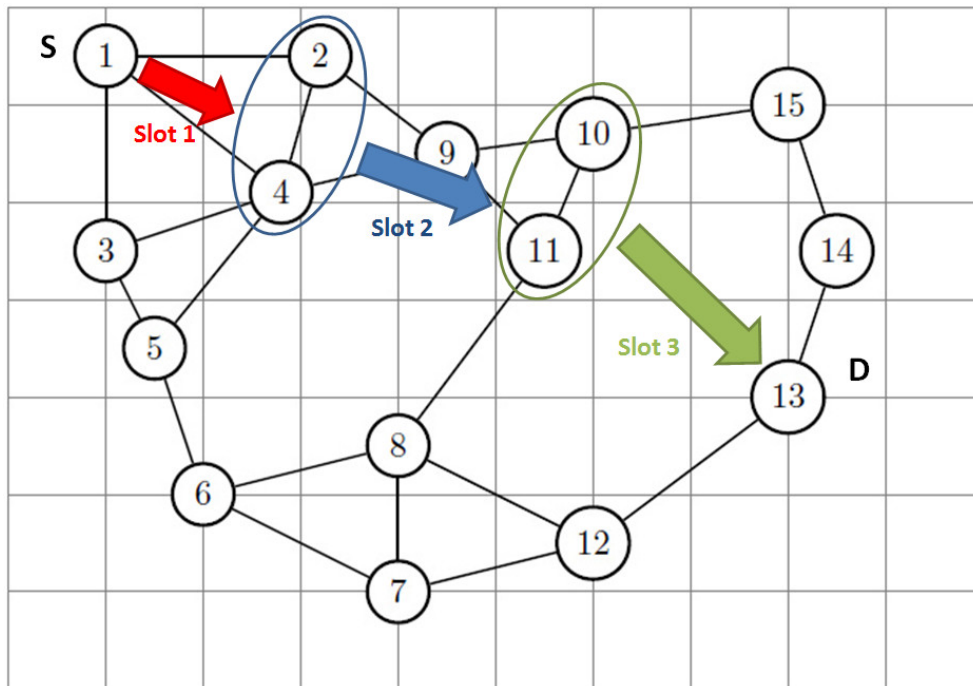
In un primo caso, rappresentato in figura 3.5, si può vedere come il relaying cooperativo riduca il numero di slot della trama, rispetto al caso non cooperativo, necessari ad inoltrare la domanda di traffico dalla sorgente (nodo 1) alla destinazione (nodo 13). Lo sfruttamento della collaborazione per il superarco (2/4,10/11) permette di ridurre gli hop saltando un nodo intermedio, mentre per il superarco (10/11,13) consente di superare una zona vuota del grafo che ben rispecchia uno spazio dove è impossibile collocare antenne (configurazione del territorio o concessioni edilizie).

In casi come questo il vantaggio è molto rilevante e in questo particolare esempio abbiamo una riduzione del 40% delle risorse di trama utilizzate. La soluzione ottima senza collaborazione riesce a sfruttare poco il riuso spaziale degli slot quindi solo una configurazione ha due archi e la possibilità di usare trasmissioni con range maggiore fa la differenza.





(a) Senza collaborazione



(b) Con collaborazione

Figura 3.5: Esempio di utilizzo della collaborazione

## 3.6 Pricing con potenza variabile

Nei modelli finora presentati abbiamo considerato la potenza emessa prefissata e uguale per tutti i nodi. Se utilizzassimo invece la potenza come variabile otterremmo un risparmio energetico e una riduzione dell'interferenza che ci consentirebbe di ricavare potenzialmente soluzioni migliori, attraverso un riuso spaziale più efficiente. Definiamo quindi un insieme dei livelli di potenza disponibili  $\Pi$  per le antenne trasmettenti.

Questo modello è molto più vicino al caso reale, infatti i trasmettitori utilizzati nelle reti wireless sono in molti casi effettivamente in grado di variare la potenza trasmessa sia per aumentare la copertura, sia per ridurre consumi e interferenze.

### 3.6.1 Pricing SINR con collaborazione e potenza variabile

Partendo dal modello SINR con collaborazione abbiamo introdotto le nuove variabili  $p_v \in \Pi$ , definite per ogni nodo semplice  $v$ , e riadattato i vincoli:

$$\max \sum_{(i,j) \in A} \sigma_{ij} z_{ij} - \epsilon \sum_{v \in N} p_v \quad (3.41)$$

$$s.t. \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} + \sum_{\substack{(j,i) \in A: \\ v \in \Gamma(i)}} z_{ji} \leq 1, \quad \forall v \in N \quad (3.42)$$

$$\sum_{v \in \Gamma(i)} p_v g_{vw} \geq \gamma \left( \eta + \sum_{\substack{(k,l) \in A, u \in \Gamma(k): \\ k \neq i, l \neq j}} p_u g_{uw} \right) - M_{ij}^w (1 - z_{ij}), \quad \forall (i,j) \in A, w \in \Gamma(j) \quad (3.43)$$

$$p_v \leq \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} P, \quad \forall v \in N \quad (3.44)$$

$$p_v \geq \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} P_{min}, \quad \forall v \in N \quad (3.45)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A \quad (3.46)$$

$$p_v \in \Pi, \quad \forall v \in N \quad (3.47)$$

La funzione obiettivo (3.41) ha il nuovo termine di sommatoria delle potenze, pesato per un numero molto piccolo  $\epsilon$  che serve a far prendere la minore potenza possibile senza alterare il ruolo delle  $\sigma$ . I vincoli (3.42) sono quelli visti in (3.16), quelli di SINR (3.43) contengono la nuova variabile  $p_v$  e sono stati linearizzati come in (3.10), i vincoli (3.44) servono a legare le variabili  $z_{ij}$  alle  $p_v$ , e infine i vincoli (3.45) sono necessari per evitare che un nodo attivo emetta potenza nulla.

Come visto per il caso a potenza fissa (3.15) il numero di vincoli di SINR e di variabili cresce nel caso peggiore come  $O(|N|^4)$ , quindi abbiamo elaborato una formulazione alternativa affine a quella vista in (3.19) che riduce il numero di vincoli introducendo nuove variabili. Queste sono:

- $p_v$ : potenza erogata dal nodo  $v$ ;
- $p_{vw}^U$ : potenza utile, trasmessa fra la coppia di nodi  $v$  e  $w$  corrispondenti a un superarco attivo;
- $p_{vw}^I$ : potenza interferente, trasmessa fra la coppia di nodi  $v$  e  $w$  corrispondenti a un superarco non attivo.

Tutte e tre le potenze sono a valori discreti.

Il modello è:

$$\max \sum_{(i,j) \in A} \sigma_{ij} z_{ij} - \epsilon \sum_{v \in N} p_v, \quad (3.48)$$

$$s.t. \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} + \sum_{\substack{(j,i) \in A: \\ v \in \Gamma(i)}} z_{ji} \leq 1, \quad \forall v \in N \quad (3.49)$$

$$\sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i), w \in \Gamma(j)}} P z_{ij} \geq p_{vw}^U, \quad \forall v \in N, w \in N, v \neq w, \quad (3.50)$$

$$p_{vw}^U \geq \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i), w \in \Gamma(j)}} P_{min} z_{ij}, \quad \forall v \in N, w \in N, v \neq w, \quad (3.51)$$

$$\sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} P z_{ij} \geq p_v, \quad \forall v \in N, \quad (3.52)$$

$$p^v \geq \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} P_{\min} z_{ij}, \quad \forall v \in N, \quad (3.53)$$

$$p_{vw}^I \leq \left( \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i)}} z_{ij} - \sum_{\substack{(i,j) \in A: \\ v \in \Gamma(i), w \in \Gamma(j)}} z_{ij} \right), \quad \forall v \in N, w \in N, v \neq w, \quad (3.54)$$

$$p_{vw}^U + p_{vw}^I = p_v, \quad \forall v \in N, w \in N, v \neq w, \quad (3.55)$$

$$\sum_{v \in N} p_{vw}^U g_{vw} \geq \gamma \left( \eta + \sum_{u \in N} p_{uw}^I g_{uw} \right) - M_w \left( 1 - \sum_{\substack{(i,j) \in A: \\ w \in \Gamma(j)}} z_{ij} \right), \quad \forall w \in N, \quad (3.56)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A, \quad (3.57)$$

$$p_v \in \Pi, \quad \forall v \in N, \quad (3.58)$$

$$p_{vw}^U, p_{vw}^I \in [0, P], \quad \forall v \in N, w \in N, v \neq w. \quad (3.59)$$

La funzione obiettivo e il primo gruppo di vincoli sono già stati introdotti in (3.19). I vincoli (3.50) e i vincoli (3.51) regolano i limiti della potenza emessa su coppie di nodi nei superarchi attivi, mentre i vincoli (3.54) impediscono di mettere potenza interferente uscente da nodi non attivi. I vincoli (3.52) e (3.53) non sono strettamente necessari per definire il nostro problema (infatti i vincoli (3.50) e (3.51) sarebbero sufficienti per il corretto controllo di potenza), tuttavia abbiamo riscontrato sperimentalmente che la loro inclusione rafforza il rilassamento del problema migliorandone la performance. I vincoli (3.55) stabiliscono che la potenza trasmessa tra due nodi o è potenza utile o è potenza interferente. Infine i vincoli (3.56) sono quelli visti nel modello (3.19) a potenza fissa e adattati alle nuove variabili. Anche in questo caso la complessità in numero di vincoli si riduce a circa  $O(|N|^2)$ , mentre quella delle variabili resta approssimabile a  $O(|N|^4)$ .

### 3.6.2 Pricing protocollare con collaborazione e potenza variabile

Per quanto riguarda l'estensione del modello protocollare alla potenza variabile sarebbe possibile rivedere tutto il modello presentato in modo da includere diversi livelli di potenza per ogni nodo nelle variabili e nei vincoli. Tuttavia questo ci porterebbe a introdurre una notevole complessità aggiuntiva al problema. Dato che lo scopo principale del modello protocollare è quello di ottenere soluzioni euristiche di buona qualità e in tempi ridotti, abbiamo preferito mantenere il modello visto in precedenza introducendo piccole modifiche per adattarlo alla potenza variabile.

In tal senso il procedimento utilizzato per il pricing rimane lo stesso del caso di potenza fissa; tuttavia viene introdotto uno step aggiuntivo per determinare, una volta trovata una colonna, se esiste un assegnamento di potenze per i nodi trasmettenti tale da rispettare i vincoli di SINR e da minimizzare le potenze utilizzate. Nel caso tale configurazione non esista vengono applicati gli accorgimenti descritti in 3.4.3.

Un'altra variazione introdotta è un'ulteriore modifica al calcolo delle  $\Phi_{vw}$ ; infatti avendo le potenze variabili in media il livello di interferenza visto da un nodo sarà inferiore e di conseguenza le adiacenze saranno più ristrette.

$$\Delta_{vw} = \{u \in N : u \neq \{v, w\}, Cg_{uw}\gamma > g_{vw} - \tau\},$$

dove  $C$  è definita come:

$$C = \frac{P_{min} + P}{2P}$$

In questo modo assumiamo che la potenza dell'interferente sia la media tra il massimo e il minimo nel range di valori disponibili.

### 3.6.3 Euristicica con cammini minimi a potenza variabile

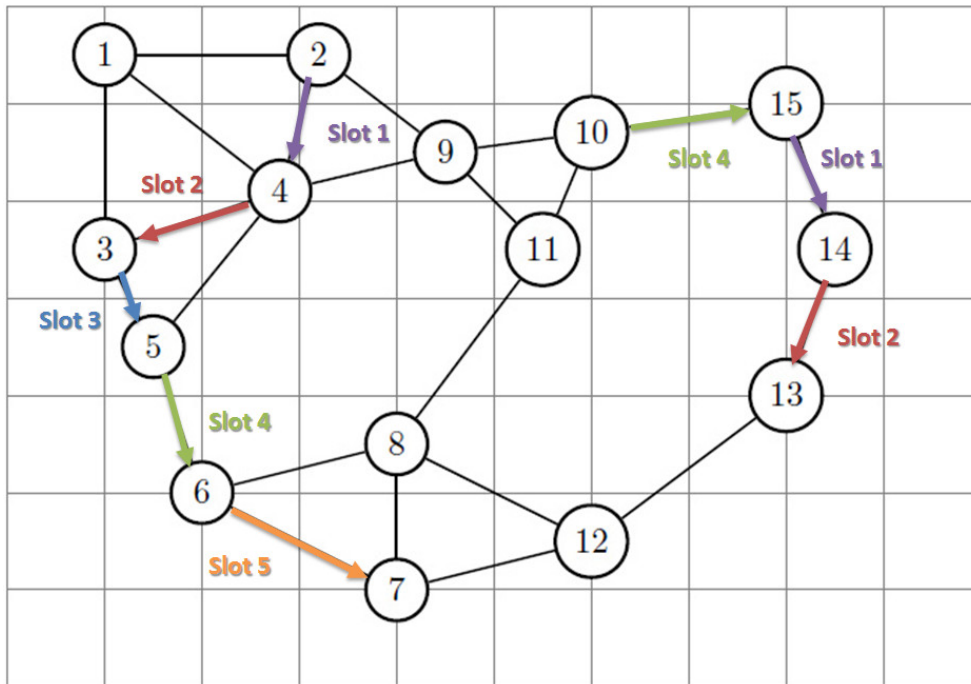
L'euristica presentata in 3.4.4 è facilmente estendibile al caso di potenza variabile, in quanto il procedimento agisce solo sulla creazione del grafo ed è indipendente dal problema di pricing utilizzato, sia esso a potenza fissa o variabile.

Dato che in questo scenario il controllo dell'interferenza assume un ruolo ancora più importante, sarà più probabile l'utilizzo di rotte più lunghe che consentano di creare configurazioni più numerose che occuperanno meno slot nella trama. Di conseguenza acquista maggiore importanza la scelta del parametro  $k$ .

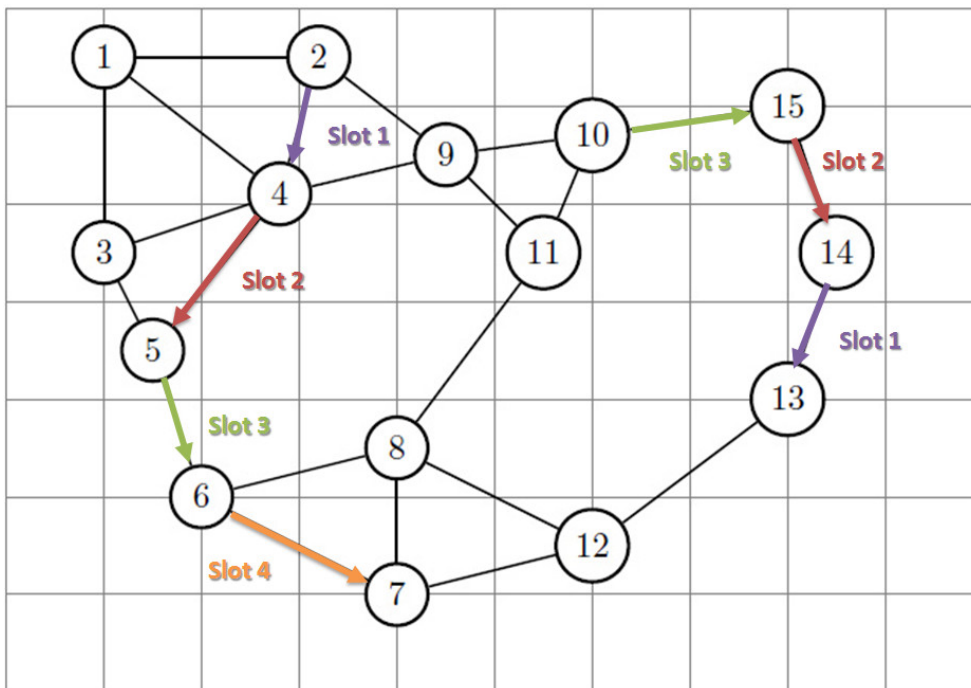
## 3.7 Esempio di controllo di potenza

Abbiamo analizzato un ulteriore esempio sullo stesso grafo presentato in 3.5 per mettere in evidenza il funzionamento del controllo di potenza e la sua utilità. Per semplificare la rappresentazione riportiamo un caso dove non entra in gioco la collaborazione.

Consideriamo le domande di traffico (2,7) e (10,13): nel caso a potenza fissa sono necessari in totale 5 slot della trama, in quanto l'uso di una potenza elevata non permette di utilizzare l'arco (4, 5) con alcun arco utile per le domande di traffico considerate: la soluzione ottima passa sugli archi (4, 3) e (3, 5) (utilizzare (4, 5) in uno slot separato avrebbe dato il medesimo risultato). Con l'introduzione del controllo di potenza invece vengono utilizzati contemporaneamente gli archi (4, 5) e (15, 14) guadagnando uno slot nella trama. Ciò è reso possibile dalla minore potenza emessa dai trasmettitori che limita l'interferenza.



(a) Potenza fissa



(b) Potenza variabile

Figura 3.6: Esempio di utilizzo del controllo di potenza

# Capitolo 4

## Risultati numerici e confronti

In questo capitolo riporteremo dapprima i test condotti sui modelli proposti, confrontando principalmente i tempi di esecuzione e la qualità delle soluzioni ottenute.

Per quanto riguarda sia i modelli SINR a potenza variabile sia i modelli protocollari a potenza fissa analizzeremo le variazioni e semplificazioni apportate per valutarne l'effettiva efficacia. Metteremo poi a confronto le tre strategie principali, ovvero modello basato sul SINR, modello protocollare ed euristica con riduzione del grafo, sia a potenza fissa che a potenza variabile.

Infine utilizzeremo i test effettuati per mettere in luce vari aspetti tecnologici come l'utilità della collaborazione e del controllo di potenza.

### 4.1 Condizioni di test

I modelli presentati sono stati scritti in linguaggio AMPL e testati attraverso uno script utilizzando come solver CPLEX 11. Le impostazioni utilizzate per CPLEX sono `integrality=1e-09` e `feasibility=1e-09`, introdotte a causa dell'instabilità numerica dei problemi di SINR, e `mipemphasis=1` che spinge il processo di Branch & Cut alla ricerca aggressiva di nuove soluzioni ammissibili piuttosto che alla prova della loro ottimalità. Questo è giustificato dal fatto che spesso sarà conveniente non risolvere il sottoproblema di pricing all'ottimo ma alla prima soluzione ammissibile 4.4.



Per i test utilizzeremo delle istanze casuali generate su un'area quadrata di  $1 \text{ Km}^2$  con diversi numeri di nodi e di domande di traffico tali da mostrare le differenze fra gli algoritmi e la loro diversa scalabilità senza dover prendere in considerazione tempi di risoluzione superiori a qualche ora.

A questo proposito il limite di tempo utilizzato, salva diversa indicazione, è di due ore di tempo macchina (`_total_solve_time=7200`) in modo da non far pesare sui confronti i tempi di overhead necessari ad AMPL per gestire modelli e dati. Riporteremo tuttavia quando questi overhead diventano tali da rendere inaccettabile la performance dell'algoritmo. Sempre salvo diversa indicazione i parametri del modello sono la potenza  $P = 0.1 \text{ mW}$ , l'SNR richiesto  $\gamma = 20$ , il rumore termico  $\eta = 3 \cdot 10^{-10}$  e il guadagno di propagazione sugli archi  $G_{ij} = d_{ij}^{-3}$ , che implica una trasmissione in ambiente non ideale.

Le colonne iniziali dell'algoritmo sono quelle generate dalla risoluzione del problema senza collaborazione più delle colonne euristiche. Queste sono ottenute prendendo i singoli superarchi di tutti i cammini disgiunti (in ordine crescente di lunghezza) tra le coppie di sorgenti e destinazioni.

## 4.2 Modelli SINR a potenza variabile

Per il problema di SINR con potenza variabile abbiamo presentato, nel capitolo precedente, due modelli: uno con formulazione basata sui superarchi 3.41, l'altro invece sulle coppie di nodi semplici 3.48 che chiameremo *DiSINR*.

Come spiegato in dettaglio il secondo modello ci consente di ridurre drasticamente il numero di vincoli presenti nel problema con l'aggiunta di un numero limitato di variabili.

Per confrontare i due algoritmi utilizzeremo istanze da 5, 10, 20 domande e 10, 15 e 20 nodi, che corrispondono a grafi di collaborazione con un numero di superarchi che va dai 200 ai 1500.

Considereremo due istanze per ogni coppia di numero di nodi - numero di domande per un totale di 18 istanze. I livelli di potenza considerati sono 3 (frazioni della potenza  $0.1 \text{ mW}$  considerata nel caso a potenza fissa). Il

$ N $	$ D $	Ist	$ A $	Tempi (s)		Iterazioni	
				SINR	DiSINR	SINR	DiSINR
10	5	I1	314	3,3	4,8	14	19
		I2	351	9,9	11,2	19	23
	10	I3	271	17,3	26,5	69	69
		I4	429	32,4	26,0	23	22
	20	I5	501	781,4	316,2	278	251
		I6	444	1006,0	588,5	434	385
15	5	I7	608	32,6	20,3	32	24
		I8	634	46,2	25,6	25	21
	10	I9	433	8,8	14,4	39	45
		I10	741	425,2	207,4	243	207
	20	I11	534	47,7	65,6	101	105
		I12	610	403,2	324,0	131	130
20	5	I13	1378	679,6	363,7	115	101
		I14	1106	1033,24	407,9	174	154
	10	I15	1713	*7200,0	*7200,0	519	738
		I16	1458	*7200,0	3065,2	692	489
	20	I17	1438	4035,7	1398,6	619	478
		I18	1092	809,0	991,4	327	352

Tabella 4.1: Tempi e iterazioni per gli algoritmi basati sul modello SINR a potenza variabile (\*: superato il limite di tempo)

pricing non verrà risolto all'ottimo a ogni iterazione ma si fermerà alla prima soluzione ammissibile.

La tabella 4.1 conferma che il tempo di risoluzione in generale è proporzionale alla dimensione della istanza (che è dovuta al numero di superarchi) e, anche se in proporzione minore, al numero di domande da soddisfare. Tuttavia anche la particolare topologia del grafo riveste una certa importanza come si può notare dalle differenze di tempi tra le coppie di istanze di uguale dimensione; questo aspetto è di valenza generale e sarà riscontrabile anche nei test successivi.

I risultati, ovvero le lunghezze di trama, ottenuti non sono riportati in tabella in quanto questi sono sempre i medesimi dato che i modelli sono perfettamente equivalenti.

Il modello SINR ha tempi migliori su alcune delle istanze piccole e con

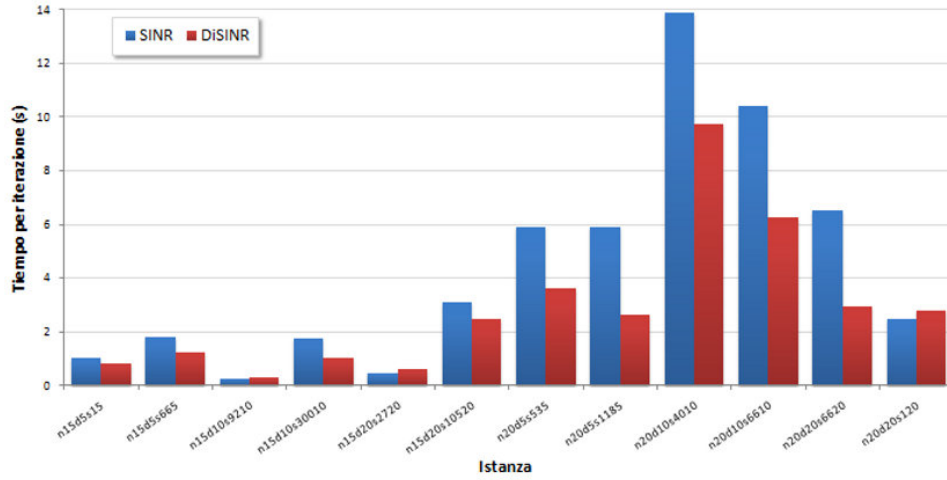


Figura 4.1: Grafico dei tempi per iterazione per le istanze da 15 e 20 nodi per gli algoritmi basati sul modello SINR a potenza variabile

poche domande tuttavia quando una delle due cresce i tempi aumentano velocemente. Al contrario utilizzando il modello *DiSINR* l'aumento dei tempi risulta più lento grazie alla semplificazione del pricing e si ottengono performance migliori di quelle del primo. Il fatto che il miglioramento di prestazioni nelle istanze con più nodi e domande sia imputabile effettivamente al modello semplificato è confermato dai tempi per iterazione nei due casi che riportiamo in 4.1 per le istanze da 15 e 20 nodi dove le differenze sono più marcate.

### 4.3 Modelli protocollari a potenza fissa

Per la risoluzione euristica basata sul modello protocollare abbiamo elaborato tre diverse varianti:

- *Prot*: pricing con modello che utilizza le adiacenze  $\Delta_{ij}$  sui superarchi 3.28;
- *Red*: pricing con modello ridotto basato su adiacenze  $\Delta_{vw}$  fra coppie di nodi semplici 3.32;

N	D	Ist	Soluzioni (UB)				Tempi (s)		
			Opt	Prot	Red	Adcut	Prot	Red	Adcut
10	5	I1	13	13	14	14	2,4	1,1	1,3
		I2	12	12	12	12	0,8	0,7	0,8
	10	I3	26	29	29	27	1,1	0,9	1,0
		I4	26	26	26	26	15,8	2,0	2,3
	20	I5	32	32	32	32	21,9	6,7	7,3
		I6	43	43	(43)45	44	15,4	10,2	7,9
15	5	I7	22	23	24	23	4,1	1,0	3,5
		I8	23	23	23	23	13,4	2,6	2,6
	10	I9	39	39	40	40	4,3	1,7	2,2
		I10	40	40	(40)41	40	338,9	22,6	21,4
	20	I11	84	(82)87	(82)86	85	43,0	7,7	10,0
		I12	72	74	75	72	23,1	6,7	10,1
20	5	I13	18	18	(18)19	18	166,0	16,2	35,3
		I14	22	22	22	22	250,6	13,5	16,2
	10	I15	36	(36)37	(36)37	36	2454,1	208,5	239,6
		I16	39	40	40	40	3971,3	86,2	78,5
	20	I17	104	(104)105	(104)105	104	2305,4	154,4	133,7
		I18	85	(85)88	(86)88	85	348,2	39,5	40,1

Tabella 4.2: Confronto fra gli algoritmi basati sul modello protocollare (tra parentesi le soluzioni con colonne non ammissibili per il SINR)

- *Adcut*: pricing con modello ridotto con l'aggiunta dei tagli, della  $\Phi$  adattiva e della ricerca di colonne ammissibili 3.4.3.

Il passaggio dal primo modello a quello ridotto ricalca quello effettuato tra i modelli SINR e consente, quindi, di ridurre il numero dei vincoli con un modesto aumento delle variabili. I due modelli però non sono perfettamente equivalenti come accadeva nel caso di quelli SINR, così nel confronto dovremo valutare anche eventuali variazioni dei risultati dovute alla semplificazione.

Per i test consideriamo ancora 18 istanze con potenza fissa e le combinazioni di numero nodi e domande già citate (10, 15, 20 nodi e 5, 10, 20 domande). Anche in questo caso la risoluzione del pricing è alla prima soluzione ammissibile.

Riportiamo alcuni test eseguiti sugli algoritmi basati sui modelli protocollari presentati. Dalla tabella 4.2 emerge come i modelli *Prot* e *Red* inseriscono spesso delle colonne non ammissibili per i vincoli di SINR ottenendo

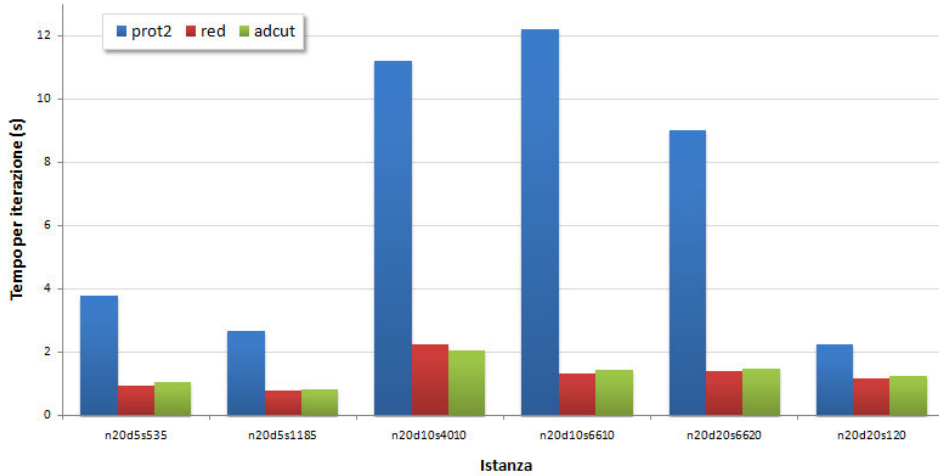


Figura 4.2: Grafico dei tempi di risoluzione in scala logaritmica per le istanze da 20 nodi per i modelli protocollari

soluzioni buone; tuttavia non considerando queste colonne esse peggiorano. Data la semplificazione apportata da *Prot* a *Red* le soluzioni ottenute con il primo sono leggermente migliori. *Adcut* ottiene sempre soluzioni ammissibili grazie all'uso dei tagli e, con le  $\Phi$  adattive, ricava risultati prossimi, se non coincidenti, all'ottimo (scarto massimo 7%).

Per quanto riguarda i tempi di risoluzione l'utilizzo delle adiacenze sui superarchi si rivela poco adatto in quanto la performance peggiora molto velocemente con la dimensione delle istanze. Al contrario gli altri due algoritmi sono più veloci e scalano in modo migliore. Questo è particolarmente evidente sui grafi da 20 nodi dei quali riportiamo nel grafico 4.2 i tempi per iterazione di pricing. La differenza supera un ordine di grandezza per le istanze più critiche, mentre è modesta e poco apprezzabile sulle istanze più piccole.

Va inoltre specificato che il modello *Prot* incorre in pesanti limitazioni in termini di utilizzo di memoria e di tempo di overhead necessario ad AMPL per gestire dati e modelli in quanto il problema oltre ad avere dimensioni molto elevate deve gestire un numero notevole di liste di adiacenza. Questa limitazione rende problematico spingere *Prot* su istanze con più di 20 nodi, mentre per gli altri due algoritmi il problema persiste ma è meno rilevante.

## 4.4 Settaggio degli algoritmi

Nella generazione di colonne non è necessario risolvere il pricing all'ottimo ma basta ottenere a ogni iterazione una soluzione a costo ridotto negativo. In generale risolvere all'ottimo può comportare un numero minore di iterazioni ma maggior tempo per ciascuna rispetto a una risoluzione parziale. Per ottenere una buona performance da un algoritmo è necessario valutare il compromesso migliore tra i due. Ad esempio per il modello (3.19) in [8] viene indicato che tale strategia è l'utilizzo della prima soluzione ammissibile a costo ridotto negativo. Ciò permane anche per la versione a potenza variabile, dato che il modello rimane molto complesso e la sua esecuzione all'ottimo comporta tempi lunghi.

Valuteremo, invece, per ciascuno degli algoritmi proposti la configurazione più opportuna considerando la risoluzione all'ottimo (*opt*), alla prima soluzione ammissibile (*first*) e l'utilizzo di un gap del 50% e del 25% tra lower bound e upper bound (*gap 0.5* e *gap 0.25*).

Nella figura 4.3 riportiamo dei grafici sintetici dei tempi impiegati da ciascun algoritmo al variare delle opzioni descritte. Gli andamenti sono ottenuti come media normalizzata sul tempo di *first* di varie istanze raggruppate per numero di nodi.

Per quanto riguarda i modelli protocollari le performance su istanze da 10 nodi è circa equivalente per tutti i settaggi tenendo conto del fatto che i tempi sono ridotti e quindi più suscettibili a variazioni delle misurazioni. Nelle istanze da 15 nodi si comincia a notare un peggioramento di *opt* che diventa drastico sui grafi da 20 nodi.

Per quanto riguarda le euristiche con shortest path il vantaggio di *first* risulta più apprezzabile (soprattutto rispetto ad *opt*) già dalle istanze piccole e va ad aumentare col crescere dei nodi; questo è giustificabile col fatto che queste euristiche usano il pricing SINR, intrinsecamente più complesso di quello protocollare.

Dati questi risultati, nei confronti successivi considereremo per tutti gli algoritmi i problemi di pricing risolti alla prima soluzione ammissibile.

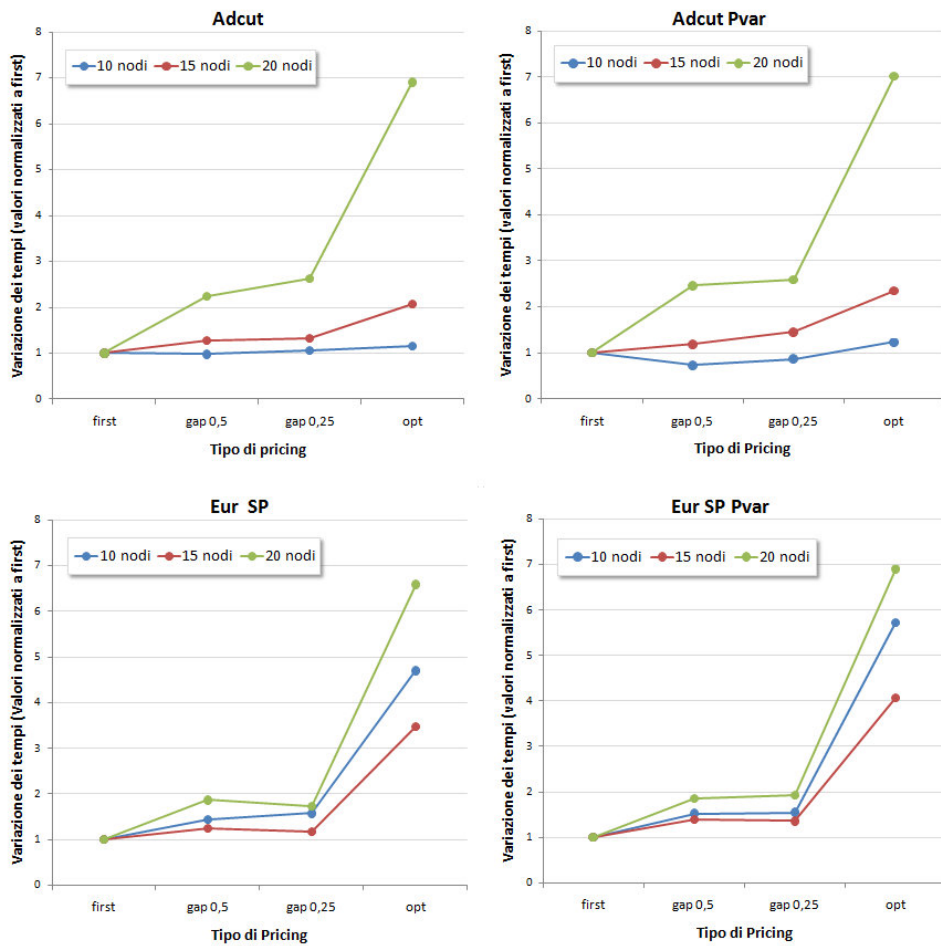


Figura 4.3: Andamenti dei tempi considerando diverse opzioni per la risoluzione del pricing

## 4.5 Confronti fra gli algoritmi

Di seguito riporteremo i test condotti sugli algoritmi proposti nelle loro versioni finali, considerando dapprima il caso a potenza fissa e poi l'introduzione del controllo di potenza. Per fornire una caratterizzazione completa dell'andamento dei modelli utilizzeremo 4 istanze per ogni coppia numero nodi - numero domande, confrontando i tempi di esecuzione, il numero di iterazioni e i risultati ottenuti.

Il termine di confronto per valutare il rapporto tra la qualità delle soluzioni e i tempi sarà sempre il modello SINR, quello che meglio rappresenta il sistema di trasmissione reale.

### 4.5.1 Potenza fissa

Gli algoritmi che prenderemo in analisi sono:

- *DiSINR*: è il modello di partenza presentato in [8] che considera esplicitamente i vincoli di SINR fra coppie di nodi semplici;
- *Adcut*: è l'euristica basata sul modello protocollare introdotto in 3.4.2, anch'esso sulle coppie di nodi semplici, con gli accorgimenti discussi in 3.4.3;
- *EurSP*: è l'euristica di riduzione del supergrafo presentata in 3.4.4 che utilizza una risoluzione basata sul *DiSINR*; il numero di cammini minimi per domanda considerati sarà pari a 20.

In tabella 4.3 sono riportati i risultati ottenuti con i tre algoritmi in termini di lower bound e upper bound. Notiamo subito che, in generale, i due coincidono quindi l'upper bound è l'ottimo del problema intero. Fanno eccezione i casi in cui l'esecuzione è terminata a causa del limite di tempo, dove il lower bound viene calcolato a posteriori ed è solitamente di scarsa qualità dato che la generazione di colonne non è completa.

Confrontando le soluzioni emerge che, quando siamo in possesso della soluzione ottima (*DiSINR*), sia *Adcut* sia *EurSP* ottengono risultati molto



$ N $	$ D $	Ist	$ A $	DiSINR		Acut		EurSP	
				LB	UB	LB	UB	LB	UB
10	5	I1	314	13	13	14	14	13	13
		I2	351	12	12	12	12	12	12
		I19	302	10	10	10	10	10	10
		I20	211	12	12	12	12	12	12
	10	I3	271	26	26	27	27	26	26
		I4	429	26	26	26	26	26	26
		I21	294	19	19	19	19	19	19
		I22	301	21	21	23	23	21	21
	20	I6	444	43	43	44	44	43	43
		I5	501	32	32	32	32	32	32
		I23	279	37	37	37	37	37	37
		I24	337	40	40	41	41	40	40
15	5	I7	608	22	22	23	23	22	22
		I8	634	23	23	23	23	23	23
		I25	426	22	22	22	22	22	22
		I26	605	17	17	17	17	17	17
	10	I9	433	39	39	40	40	39	39
		I10	741	40	40	40	40	40	40
		I27	617	45	45	45	45	45	45
		I28	605	33	33	33	33	33	33
	20	I11	534	84	84	85	85	84	84
		I12	610	72	72	72	72	72	72
		I29	617	81	81	82	82	81	81
		I30	593	81	81	81	81	81	81
20	5	I13	1378	18	18	18	18	18	18
		I14	1106	22	22	22	22	23	23
		I31	1713	18	18	18	18	18	18
		I32	1483	20	20	20	20	20	20
	10	I15	1713	*26	36	36	36	36	36
		I16	1458	39	39	40	40	40	40
		I33	1092	47	47	47	47	47	47
		I34	1902	*29	38	40	40	40	40
	20	I17	1438	104	104	104	104	104	104
		I18	1092	85	85	85	85	85	85
		I35	1520	*43	64	67	67	65	65
		I36	1506	*58	81	82	82	81	81

Tabella 4.3: Risultati ottenuti con i vari algoritmi a potenza fissa(\*: superato il limite di tempo)

N	D	Ist	A	Tempi			Iterazioni		
				DiSINR	Adcut	EurSP	DiSINR	Adcut	EurSP
10	5	I1	314	5,0	1,3	3,4	36	9	7
		I2	351	3,8	0,8	4,9	13	2	2
		I19	302	1,5	1,1	3,3	9	9	8
		I20	211	1,0	0,7	2,8	12	6	4
	10	I3	271	6,7	1,0	9,2	21	2	21
		I4	429	21,1	2,3	7,4	20	2	3
		I21	294	3,2	2,1	6,4	23	15	16
		I22	301	7,5	1,6	8,7	32	5	25
	20	I6	444	113,6	7,9	62,6	153	18	85
		I5	501	30,3	7,3	39,2	36	8	40
		I23	279	10,6	3,5	15,1	44	12	18
		I24	337	29,0	6,7	33,6	100	25	92
15	5	I7	608	22,8	3,5	3,7	50	10	8
		I8	634	21,6	2,6	3,3	19	7	3
		I25	426	0,8	0,8	3,5	5	5	3
		I26	605	13,4	14,9	6,3	71	59	20
	10	I9	433	6,1	2,2	6,9	30	10	7
		I10	741	178,1	21,4	11,3	210	47	16
		I27	617	94,0	26,5	15,2	156	87	41
		I28	605	50,3	19,2	13,6	120	80	38
	20	I11	534	51,3	10,0	23,3	87	24	47
		I12	610	172,9	10,1	36,4	79	19	40
		I29	617	88,9	28,2	25,2	136	61	34
		I30	593	18,2	11,2	19,0	44	30	24
20	5	I13	1378	253,9	35,3	6,4	66	34	3
		I14	1106	352,7	16,2	5,6	131	20	6
		I31	1713	1884,1	176,9	17,9	312	194	64
		I32	1483	6,1	10,2	8,2	19	21	16
	10	I15	1713	*7200,0	239,6	37,4	647	117	41
		I16	1458	2654,1	78,5	34,2	356	54	92
		I33	1092	507,4	57,0	16,5	199	81	23
		I34	1902	*7200,0	177,8	45,5	917	159	87
	20	I17	1438	177,0	133,7	48,7	163	90	54
		I18	1092	532,5	40,1	37,6	181	32	39
		I35	1520	*7200,0	432,1	116,8	985	287	119
		I36	1506	*7200,0	469,7	166,3	1021	337	199

Tabella 4.4: Tempi di esecuzione degli algoritmi a potenza fissa(\*: superato il limite di tempo)

vicini se non coincidenti ad essa con uno scarto massimo del 9,5% per il primo e del 4,5% per il secondo.

Sempre considerando le istanze che non hanno superato il limite di tempo si può dedurre che *EurSP* ha un comportamento più affidabile in quanto non raggiunge l'ottimo in soli due casi, mentre *Adcut*, pur mantenendo errori molto bassi, non arriva all'ottimo in dieci istanze.

In tabella 4.4 riportiamo invece i tempi e il numero di iterazioni effettuate. Tra le istanze delle stesse dimensioni c'è una notevole variabilità sia in termini di tempo che di iterazioni quindi, anche se permane la tendenza di una difficoltà crescente col numero di archi e domande, la particolare configurazione del grafo è di importanza cruciale nella determinazione della complessità del problema.

Entrambi gli algoritmi approssimati comportano una riduzione del numero di iterazioni di pricing che si fa più sensibile al crescere del numero di nodi. Paragonando i due, *Adcut* tende ad effettuare meno iterazioni sulle istanze piccole mentre *EurSP* ha una progressione migliore su quelle più grandi.

Anche per quanto riguarda i tempi, la riduzione rispetto a *DiSINR* cresce velocemente con l'aumento delle dimensioni dei grafi. Nelle istanze più grandi sono presenti anche alcuni timeout. Nonostante questo modello rappresenti già un'importante riduzione rispetto a quello per superarchi, la sua applicabilità su istanze superiori ai 20 nodi comincia a diventare problematica per i tempi e per il numero di iterazioni.

Nello specifico si può notare come *EurSP* riporta tempi peggiori anche di *DiSINR* sulle prime istanze: questo è dovuto al fatto che nel tempo impiegato è compreso anche quello necessario per il calcolo dei cammini minimi utilizzati per ridurre il grafo. A questi livelli tale tempo è preponderante, mentre nelle altre istanze diventa percentualmente piccolo ed emerge il guadagno sperimentato sul tempo per iterazione.

Se sulle istanze da 15 nodi nessuna delle due euristiche prevale sull'altra, analizzando quelle da 20 nodi si nota che *EurSP* si comporta meglio rispetto ad *Adcut*: i tempi sono infatti inferiori per tutti i casi e nella maggioranza sono sotto la metà. Per un'analisi più approfondita della scalabilità delle due

euristiche rimandiamo alla sezione 4.8.

Nel processo di generazione di colonne il miglioramento del bound non è lineare: questo migliora molto velocemente in una prima fase mentre successivamente una piccola riduzione del risultato corrisponde a un notevole dispendio di tempo e, raggiunto il bound finale, un ulteriore periodo è speso per provarne l'ottimalità. Dato che nel confronto consideriamo un metodo esatto e due euristiche, oltre a valutare i tempi finali è opportuno verificare che i due metodi approssimati raggiungano effettivamente lo stesso valore del bound del modello esatto ma in tempi inferiori. Se così non fosse l'uso delle euristiche non sarebbe giustificato perché si potrebbero ottenere gli stessi risultati negli stessi tempi fermando anticipatamente la generazione di colonne esatta.

Riportiamo in 4.4 l'evoluzione dell'upper bound per i tre algoritmi considerando tre istanze da 20 nodi in quanto su di esse è più semplice apprezzare le differenze a livello grafico; tuttavia le medesime conclusioni sono estensibili a livello più generale. Si nota subito che in tutti e tre i casi le euristiche offrono una progressione molto più veloce e in particolare *EurSP* ha la performance migliore coerentemente col fatto che è l'algoritmo con i tempi più bassi. Risulta ben evidente in questi casi che *DiSINR* ha una coda molto pronunciata e soprattutto che utilizza una parte di tempo considerevole (più della metà nell'istanza *I33*), dopo aver già raggiunto il bound finale, per provare che non è migliorabile. Anche le due euristiche presentano una coda allungata, ma essa costituisce una parte minore del tempo totale.

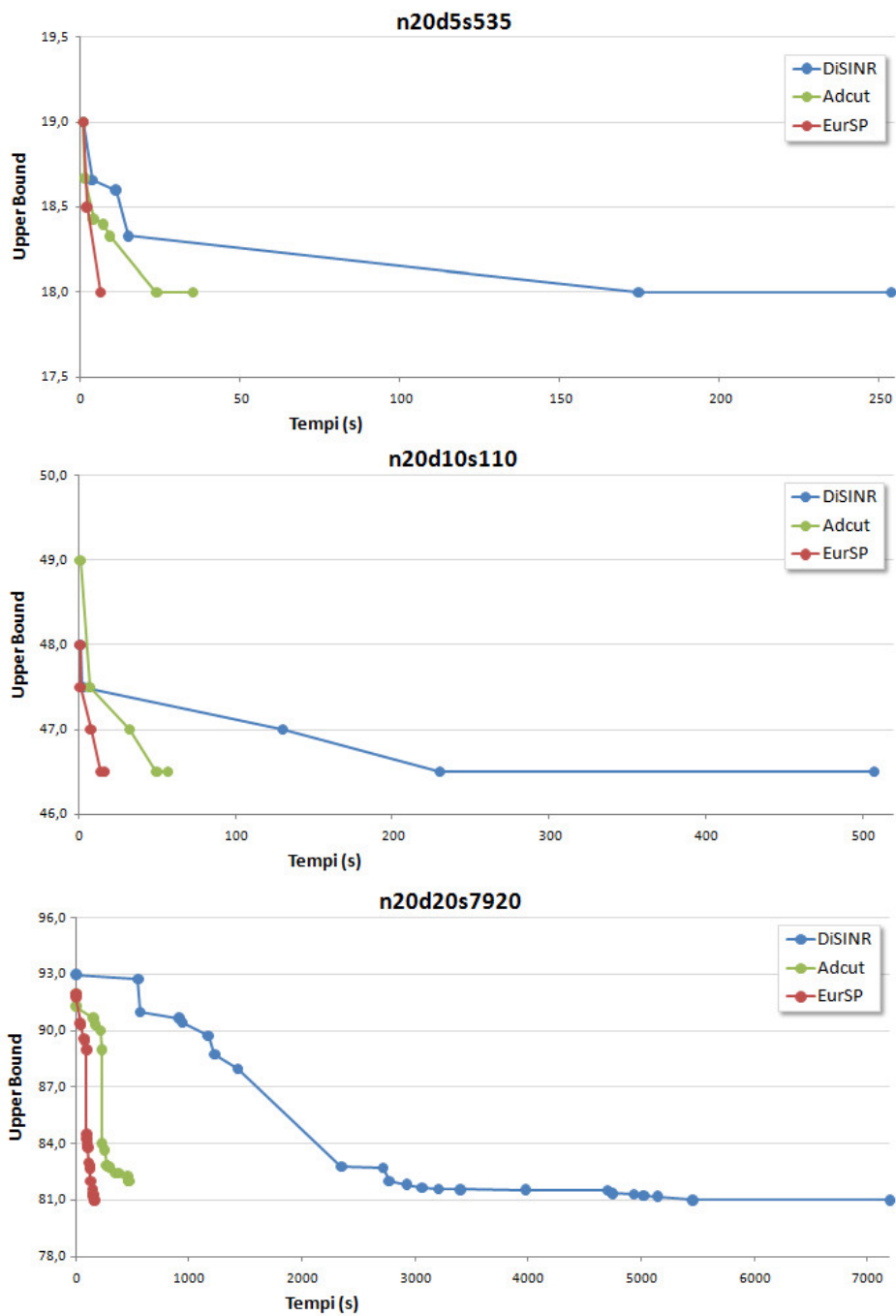


Figura 4.4: Progressione del bound negli algoritmi a potenza fissa

## 4.5.2 Potenza variabile

Nel confronto consideriamo che i trasmettitori possano usare tre livelli di potenza con massimo  $P = 0.1 \text{ mW}$ . Gli algoritmi analizzati sono:

- *DiSINR*: è il modello *DiSINR* con l'introduzione del controllo di potenza presentato in 3.6.1;
- *Adcut*: è lo stesso *Adcut* considerato per il caso a potenza fissa con una variazione per il settaggio dei parametri e un passaggio aggiuntivo per la ricerca delle potenze ottime come spiegato in 3.6.2;
- *EurSP*: è la stessa euristica di riduzione del supergrafo vista nella sezione precedente con l'utilizzo del modello *DiSINR* a potenza variabile per il pricing.

Come nel caso a potenza fissa riportiamo due tabelle con risultati, tempi di esecuzione e iterazioni impiegate. Possiamo subito notare in tabella 4.5 che il lower bound coincide con l'upper bound quindi l'upper bound è l'ottimo intero del problema tranne nei casi dove si verifica timeout.

Considerando le istanze di cui l'ottimo è determinato vediamo che gli algoritmi ottengono ancora buoni risultati: nello specifico *Adcut* ha nel caso peggiore uno scarto di 11,8% dall'ottimo mentre *EurSP* del 4,1%. Rispetto al caso a potenza fissa la performance dell'algoritmo basato sui cammini minimi rimane sostanzialmente invariata e non raggiunge l'ottimo in soli tre casi, mentre quella del modello protocollare peggiora di qualche punto: questo è il prezzo pagato per evitare di aumentarne la complessità.

Constatata la buona prova di *EurSP* possiamo anche concludere che i cammini privilegiati per l'inoltro sono quelli più brevi ovvero che l'utilizzo di rotte più lunghe non porta benefici nella riduzione dell'interferenza o nel riuso tali da ridurre la lunghezza della trama.

L'introduzione della potenza variabile porta a un vantaggio proporzionale al numero delle domande, infatti sulle istanze da 5 domande i risultati sono tutti invariati mentre si riscontrano miglioramenti su quelle da 10 e da 20 con un massimo del 18%. Questo è imputabile al fatto che avendo molte

N	D	Ist	A	DiSINR		Acut		EurSP	
				LB	UB	LB	UB	LB	UB
10	5	I1	314	13	13	13	13	13	13
		I2	351	12	12	12	12	12	12
		I19	302	10	10	10	10	10	10
		I20	211	12	12	12	12	12	12
	10	I3	271	25	25	27	27	25	25
		I4	429	25	25	25	25	25	25
		I21	294	19	19	19	19	19	19
		I22	301	20	20	21	21	20	20
	20	I6	444	41	41	43	43	41	41
		I5	501	27	27	29	29	27	27
		I23	279	37	37	37	37	37	37
		I24	337	39	39	39	39	39	39
15	5	I7	608	22	22	22	22	22	22
		I8	634	23	23	23	23	23	23
		I25	426	22	22	22	22	22	22
		I26	605	17	17	17	17	17	17
	10	I9	433	38	38	39	39	39	39
		I10	741	37	37	39	39	37	37
		I27	617	43	43	44	44	43	43
		I28	605	33	33	33	33	33	33
	20	I11	534	82	82	82	82	82	82
		I12	610	70	70	71	71	70	70
		I29	617	76	76	78	78	76	76
		I30	593	81	81	81	81	81	81
20	5	I13	1378	18	18	18	18	18	18
		I14	1106	22	22	22	22	22	22
		I31	1713	*11	17	18	18	17	17
		I32	1483	20	20	20	20	20	20
	10	I15	1713	*17	33	35	35	33	33
		I16	1458	38	38	39	39	39	39
		I33	1092	45	45	45	45	45	45
		I34	1902	36	36	38	38	36	36
	20	I17	1438	97	97	101	101	101	101
		I18	1092	81	81	82	82	81	81
		I35	1520	*33	59	65	65	61	61
		I36	1506	*49	78	80	80	78	78

Tabella 4.5: Risultati ottenuti con i vari algoritmi a potenza variabile(\*: superato il limite di tempo)

N	D	Ist	A	Tempi			Iterazioni		
				DiSINR	Adcut	EurSP	DiSINR	Adcut	EurSP
10	5	I1	314	4,8	4,2	4,0	19	34	9
		I2	351	11,2	3,2	5,5	23	20	10
		I19	302	4,7	1,6	3,5	23	11	9
		I20	211	1,3	1,8	3,1	8	16	4
10		I3	271	26,5	4,5	20,5	69	27	62
		I4	429	26,0	6,6	10,7	22	22	16
		I21	294	6,3	2,4	7,3	33	12	21
		I22	301	11,5	2,7	10,2	20	10	21
20		I6	444	588,5	9,1	242,7	385	24	255
		I5	501	316,2	9,5	139,4	251	18	162
		I23	279	22,1	5,7	23,0	68	25	43
		I24	337	38,2	9,4	33,0	94	39	71
15	5	I7	608	20,3	9,9	4,4	24	41	15
		I8	634	25,6	21	5,4	21	77	19
		I25	426	1,4	0,9	2,9	6	5	3
		I26	605	43,8	9,4	10,6	105	40	32
10		I9	433	14,4	5,7	7,7	45	32	17
		I10	741	207,4	65,9	23,5	207	158	56
		I27	617	50,2	31,5	21,5	103	87	47
		I28	605	46,5	13,6	14,3	62	58	32
20		I11	534	65,6	27,4	23,0	105	63	37
		I12	610	324,0	33,4	71,8	130	74	80
		I29	617	114,3	28,7	36,1	118	62	50
		I30	593	25,8	10,1	22,4	48	27	25
20	5	I13	1378	363,7	44,6	11,3	101	66	25
		I14	1106	407,9	86,5	16,1	154	143	36
		I31	1713	*7200,0	605	46,1	961	473	104
		I32	1483	12,0	13,6	7,9	22	26	12
10		I15	1713	*7200,0	680,6	132,5	738	427	147
		I16	1458	3065,2	293,8	57,5	489	321	104
		I33	1092	256,8	67,5	27,2	214	87	43
		I34	1902	6607,8	366,3	74,3	875	278	118
20		I17	1438	1398,6	496,9	151,4	478	333	166
		I18	1092	995,1	123,3	66,5	352	128	81
		I35	1520	*7200,0	843,6	348,7	915	417	204
		I36	1506	*7200,0	451,6	253,6	1011	299	225

Tabella 4.6: Tempi di esecuzione degli algoritmi a potenza variabile(\*: superato il limite di tempo)



trasmissioni da gestire l'uso di potenze differenziate può migliorare più efficacemente il riuso spaziale. Riporteremo analisi più approfondite sull'utilità dell'impiego del power control in 4.6.

Analizzando i tempi in 4.6 si verifica anche in questo caso una notevole variabilità per le istanze di uguali dimensioni. Entrambe le euristiche richiedono tempi e numero di iterazioni molto inferiori rispetto al modello *DiSINR* e la differenza va ad aumentare velocemente con la dimensione delle istanze. Permane inoltre la difficoltà di *DiSINR* a scalare verso istanze di dimensioni elevate.

Confrontando gli algoritmi euristici, *Adcut* riporta i tempi migliori sulle istanze da 10 in quanto permane per *EurSP* una penalità dovuta al tempo per il calcolo dei cammini minimi, vi è una sostanziale parità tra le due nelle istanze da 15, mentre *EurSP* è la migliore su quelle da 20. Tra i grafi da 10 nodi si possono riscontrare due casi particolari in cui *EurSP* ottiene tempi molto peggiori di *Adcut*; tuttavia questo è coerente col fatto che tali istanze riportano tempi più elevati della media anche per l'algoritmo *DiSINR* con cui l'euristica condivide il modello. E' quindi possibile che la riduzione del grafo in questi casi non sia sufficiente a superare la complessità dei due problemi.

Per quanto riguarda il numero delle iterazioni *EurSP* si comporta meglio di *Adcut* nei casi con 20 nodi mentre non vi è una prevalenza netta sulle istanze più piccole.

Sia i tempi e che il numero di iterazioni aumentano in generale rispetto al caso a potenza fissa, ma la perdita è molto variabile. Esistono comunque alcuni casi dove il controllo di potenza porta a un miglioramento della performance e si va quindi da un dimezzamento del tempo nel caso migliore a un aumento di 10 volte nel peggiore.

Riportiamo in figura 4.5 i grafici dell'evoluzione dell'upper bound per i tre algoritmi sulle medesime istanze utilizzate in precedenza. *EurSP* continua ad avere la progressione migliore fra le due euristiche, che mantengono un considerevole distacco dal modello *DiSINR*. Rispetto al caso a potenza fissa la discesa delle curve di *DiSINR* è meno ripida quindi la ricerca di una buona soluzione troncando la generazione di colonne sarebbe ancora meno

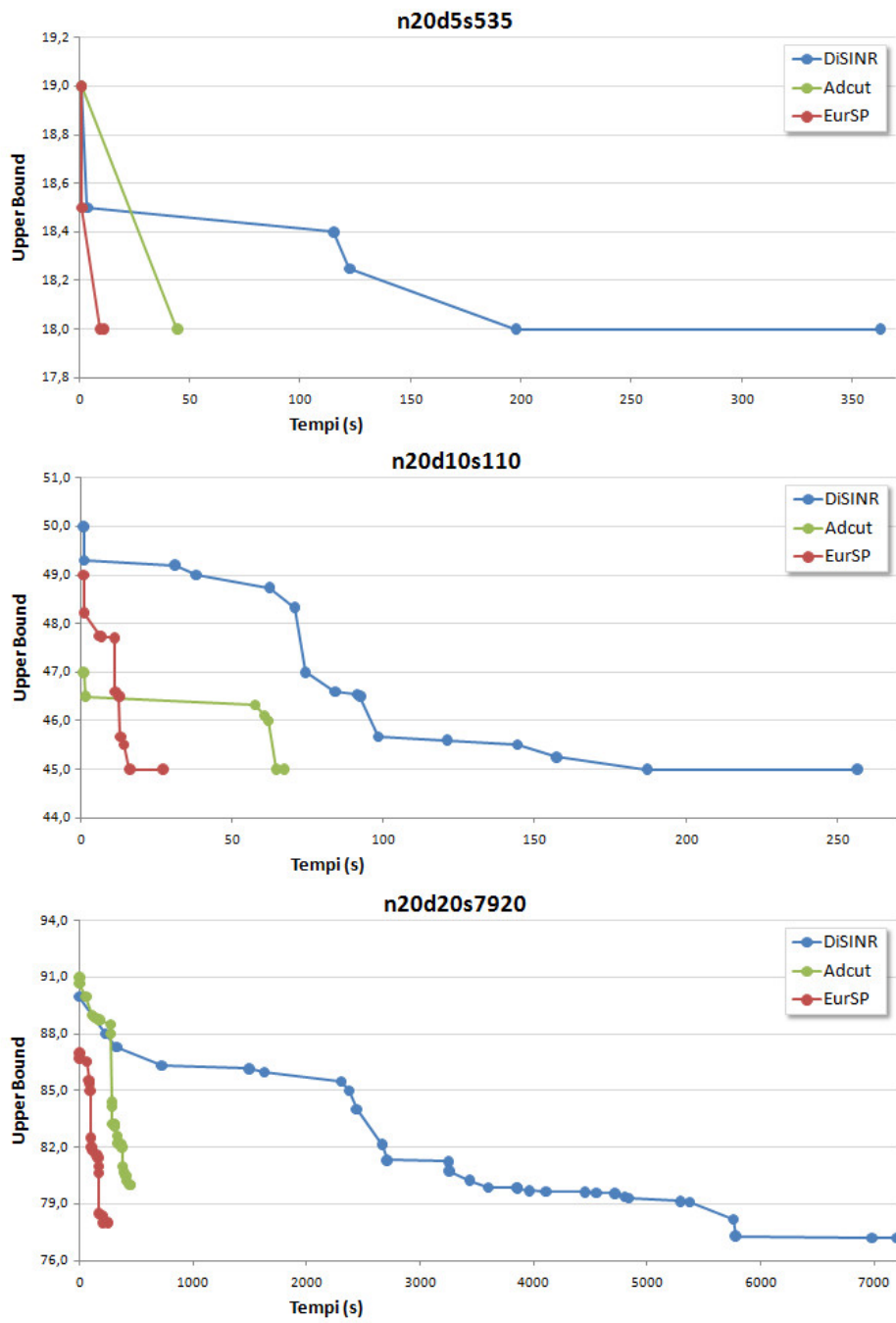


Figura 4.5: Progressione del bound negli algoritmi a potenza variabile

efficace. La variazione dell'andamento delle curve è riscontrabile anche per gli altri due algoritmi ma in modo meno pronunciato.

Un'altra osservazione è che variazioni moderate del bound di partenza (che può essere diverso fra i modelli per il modo in cui è calcolato) non influenzano sostanzialmente la performance e i rapporti di tempo tra gli algoritmi si ristabiliscono durante l'esecuzione. Ciò è ben visibile sull'istanza *I33* per l'andamento di *Adcut*, ma si può riscontrare, anche se in modo meno evidente in altri esempi, sia a potenza fissa che variabile.

## 4.6 Cooperazione e controllo di potenza

I confronti fra gli algoritmi sono stati utili per trovare dei metodi comodi e veloci per risolvere un problema di ottimizzazione, consentendoci così di fare delle valutazioni, supportate da adeguate prove, sull'utilizzo delle tecnologie disponibili.

Nelle reti wireless spesso si incontrano limitazioni di potenza, dovute alla scarsa disponibilità o ai limiti sulle emissioni delle antenne. E' quindi particolarmente importante valutare come la performance sia influenzata da differenti scelte sul controllo di potenza.

A questo proposito considereremo in dettaglio quattro diversi casi:

- Potenza fissa a  $0,1\text{ mW}$ ;
- Potenza fissa a  $0,2\text{ mW}$ ;
- Potenza variabile su 3 livelli con massimo  $0,1\text{ mW}$ ;
- Potenza variabile su 6 livelli con massimo  $0,2\text{ mW}$ .

Abbiamo condotto i test su un insieme di 18 istanze con i consueti abbinamenti di numero nodi - numero di domande, utilizzando le versioni finali dei tre algoritmi in modo da dare un'ulteriore prova delle performance ottenibili. Inoltre nell'analisi dell'utilità della collaborazione riporteremo i risultati ottenuti nei casi:

- Potenza fissa a  $0,4\text{ mW}$ ;

- Potenza variabile su 12 livelli con massimo  $0,4\text{ mW}$ .

I risultati per i casi con potenza fissa  $0,1\text{ mW}$  e variabile con massimo  $0,1\text{ mW}$  sono già presenti nelle quattro tabelle della sezione 4.5.

In tabella 4.7 riportiamo upper bound e tempi dei modelli a potenza fissa e variabile con valore massimo  $0,2\text{ mW}$ . Si può notare subito che il numero di superarchi aumenta con la potenza e di conseguenza crescono la complessità del problema e i tempi di risoluzione per tutti gli algoritmi. *DiSINR* subisce l'impatto maggiore, mentre *EurSP* è la meno influenzata in quanto il meccanismo di riduzione del grafo contrasta efficacemente l'aumento dei superarchi.

Per quanto riguarda le soluzioni, le considerazioni per le due euristiche restano le medesime viste per il caso a potenza  $0,1\text{ mW}$  e quindi le migliori sono ottenute da *EurSP*. In particolare possiamo notare che anche le istanze in cui non raggiungono l'ottimo sono le stesse, a dimostrazione del fatto che la difficoltà è data dalla struttura del grafo e permane anche variando i parametri in gioco.

L'utilizzo di una potenza maggiore consente trasmissioni più distanti e porta ad un riduzione notevole degli slot utilizzati nella trama su tutti i casi in analisi e in particolare sulle istanze più grandi, con guadagni in media del 29% e un massimo del 45%. Questa alternativa può risultare teoricamente applicabile in scenari in cui l'unico obiettivo è quello di ottimizzare le risorse radio e il sistema rende possibile il consumo di maggiore energia. Tuttavia l'impiego della potenza fissa risulta comunque una soluzione inefficiente, soprattutto nella realtà attuale in cui si cerca di sviluppare tecnologie nel modo più eco-sostenibile possibile, in quanto si usa uguale potenza sia per trasmissioni lontane e con alta interferenza, sia per tratte brevi e isolate in cui risulta eccessiva.

Una soluzione più efficace è sicuramente l'introduzione del controllo di potenza in modo da trasmettere a potenza elevata solo quando necessario e mantenere un livello medio comunque ridotto, col vantaggio aggiuntivo di limitare l'interferenza.

A questo proposito sono presenti nella parte relativa della tabella 4.7 i

	N	D	Ist	A	Sol (UB)			Tempi		
					DiSINR	Adcut	EurSP	DiSINR	Adcut	EurSP
Potenza fissa	10	5	I19	431	9	9	9	5,0	2,0	4,1
			I20	363	9	9	9	4,8	2,4	4,3
		10	I4	682	17	17	17	103,9	6,5	18,3
			I22	506	18	18	18	13,8	3,7	9,2
		20	I6	655	34	34	34	810,4	9,0	102,9
	I5		767	27	27	27	198,6	23,4	87,2	
	15	5	I25	775	13	13	13	15,5	8,2	5,3
			I26	1114	12	12	12	432,7	31,6	8,3
		10	I10	1042	33	33	33	322,4	161,7	34,6
			I27	945	29	29	29	760,2	123,6	30,8
		20	I12	1302	43	43	43	3002,4	34,5	78,1
	I29		945	52	53	52	1223,9	103,3	88,0	
	20	5	I31	2911	12	12	12	*7200,0	644,4	20,0
			I32	2133	15	15	15	725,2	187,9	12,5
		10	I33	1777	26	26	26	*7200,0	311,1	44,3
			I34	3239	29	30	30	*7200,0	5155,3	124,1
		20	I18	1777	47	47	47	*7200,0	413,6	151,4
			I36	2463	53	54	53	*7200,0	1777,4	213,5
Potenza variabile	10	5	I19	431	8	9	8	9,0	4,4	5,6
			I20	363	9	9	9	12,3	5,3	5,8
		10	I4	682	16	17	16	135,3	15,0	28,9
			I22	506	18	18	18	59,3	8,6	17,5
		20	I6	655	32	34	32	1709,4	32,4	131,8
	I5		767	24	25	24	5111,5	26,3	350,3	
	15	5	I25	775	13	13	13	35,5	12,8	6,3
			I26	1114	11	12	11	4499,6	45,4	14,8
		10	I10	1042	33	33	33	220,9	49,2	25,5
			I27	945	27	27	27	5245,3	110,8	66,6
		20	I12	1302	41	41	41	3786,2	114,0	158,9
	I29		945	48	50	48	6541,5	95,0	312,3	
	20	5	I31	2911	12	12	12	*7200,0	998,3	24,6
			I32	2133	15	15	15	528,7	207,4	11,3
		10	I33	1777	25	26	26	*7200,0	367,0	32,8
			I34	3239	27	29	28	*7200,0	3981,1	338,5
		20	I18	1777	44	46	45	*7200,0	334,9	227,9
			I36	2463	51	54	50	*7200,0	35,8	732,1

Tabella 4.7: Soluzioni e tempi degli algoritmi con massimo  $P=0,2mW$ (\*: superato il limite di tempo)

risultati ottenuti con un controllo di potenza a sei livelli con un massimo di  $0,2 \text{ mW}$ . In questo caso il peggioramento dei tempi risulta ancora più marcato e specialmente *DiSINR* comincia a diventare lento già nelle istanze con 10 nodi e 20 domande. A differenza del caso a potenza fissa si riscontrano un numero maggiore di casi in cui il modello protocollare ottiene tempi minori dell'euristica sui cammini minimi; tuttavia ciò è giustificato dalla perdita in qualità dei risultati. Infatti *EurSP* continua a ottenere buone soluzioni non ricavando l'ottimo solo in 3 casi con un errore massimo del 4%.

La riduzione di lunghezza della trama nel passaggio tra potenza variabile con massimo  $0,1 \text{ mW}$  a quella con potenza massima  $0,2 \text{ mW}$  è in media il medesimo riscontrato per i casi a potenza fissa ovvero del 29%.

Come previsto l'utilizzo della potenza variabile ha un impatto positivo perchè oltre al miglioramento delle soluzioni, in media del 4% ma in alcuni casi fino al 16%, permette un notevole risparmio energetico; infatti con il limite a  $0,1 \text{ mW}$  la potenza effettivamente usata in media sugli archi è circa  $0,07 \text{ mW}$  e con limite a  $0,2 \text{ mW}$  circa  $0,135 \text{ mW}$  quindi del 30% inferiore al tetto massimo imposto 4.6b. Va specificato a tale proposito che gli algoritmi cercano la soluzione ottima con molta più enfasi verso la riduzione degli slot che quella della potenza; modificando la modellizzazione sarebbe quindi possibile trovare un diverso compromesso numero di slots - spesa energetica in base alla necessità.

La seconda valutazione tecnologica che intendiamo condurre riguarda l'effettivo vantaggio che si può ottenere con l'introduzione della collaborazione in una rete wireless.

Per quantificarne l'utilità nel caso più generale riporteremo dei test condotti su istanze casuali come quelle considerate in precedenza, considerando sia i casi a potenza fissa che quelli a potenza variabile.

In questi test utilizzeremo l'euristica basata sui cammini minimi in quanto, dai confronti fatti tra gli algoritmi, ne abbiamo verificato la robustezza e la velocità di esecuzione. Infatti la crescita dei tempi con l'introduzione della collaborazione è consistente ma inferiore alla crescita di complessità del grafo dato che esso non viene utilizzato per intero ma viene ridotto.

	$ N $	$ D $	Ist	Soluzioni (UB)					
				P0,1		P0,2		P0,4	
				NoCoop	Coop	NoCoop	Coop	NoCoop	Coop
Potenza fissa	10	5	I19	11	10	9	9	7	7
			I20	15	12	9	9	8	8
		10	I4	28	26	18	17	14	14
			I22	25	21	18	18	18	16
		20	I6	44	43	40	34	29	27
			I5	34	32	28	27	23	23
	15	5	I25	24	22	15	13	10	10
			I26	20	17	14	12	12	10
		10	I10	44	40	39	33	32	25
			I27	47	45	41	29	23	19
		20	I12	90	72	45	43	41	39
			I29	85	81	73	52	42	37
	20	5	I31	21	18	13	12	11	11
			I32	20	20	20	15	13	12
		10	I33	74	47	28	26	23	22
I34			43	40	35	30	26	25	
20		I18	136	85	50	47	41	39	
		I36	111	81	58	53	44	42	
Potenza variabile	10	5	I19	11	10	8	8	7	7
			I20	15	12	9	9	8	8
		10	I4	28	25	16	16	13	13
			I22	21	20	18	18	15	13
		20	I6	42	41	40	32	29	25
			I5	28	27	25	24	19	19
	15	5	I25	24	22	15	13	10	10
			I26	20	17	14	11	10	10
		10	I10	42	37	33	33	29	23
			I27	45	43	41	27	20	18
		20	I12	84	70	42	41	41	38
			I29	80	76	73	48	37	34
	20	5	I31	19	17	12	12	10	10
			I32	20	20	20	15	12	12
		10	I33	71	45	28	26	23	22
			I34	38	36	31	28	24	23
		20	I18	131	81	47	45	40	38
			I36	105	78	55	50	41	38

Tabella 4.8: Risultati ottenuti senza collaborazione e con collaborazione a potenza fissa e variabile con massimi  $P=0,1\text{mW}$ ,  $P=0,2\text{mW}$  e  $P=0,4\text{mW}$

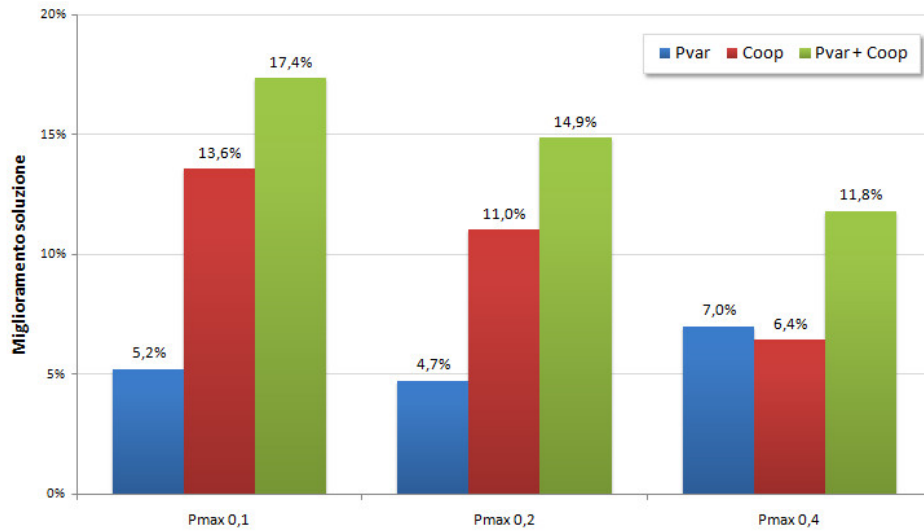
Per quanto riguarda i risultati riportati nella tabella 4.8 vediamo che con potenza massima a  $0,1\text{ mW}$  la collaborazione porta a un risparmio di risorse, in termini di time slot nella trama, in tutti i casi tranne uno, e questo è abbastanza indipendente dal fatto che venga utilizzato o meno il controllo di potenza. Il guadagno ottenuto cresce all'aumentare del numero di nodi e domande con una media del 14% con potenza fissa e 13% con quella variabile, con alcune istanze da 20 nodi dove si supera il 30%.

Intuitivamente si potrebbe pensare che utilizzando maggiore potenza e diventando il grafo più connesso, i cammini diventino più brevi e si riduca l'utilità dell'inoltro cooperativo. Questo è poco visibile considerando la potenza massima a  $0,2\text{ mW}$  dove il miglioramento introdotto dalla collaborazione permane, con una media dell'11% a potenza fissa e 10% a potenza variabile. Introducendo invece la potenza massima a  $0,4\text{ mW}$  si nota che il guadagno apportato si riduce al 6% e 5% in media nei due casi.

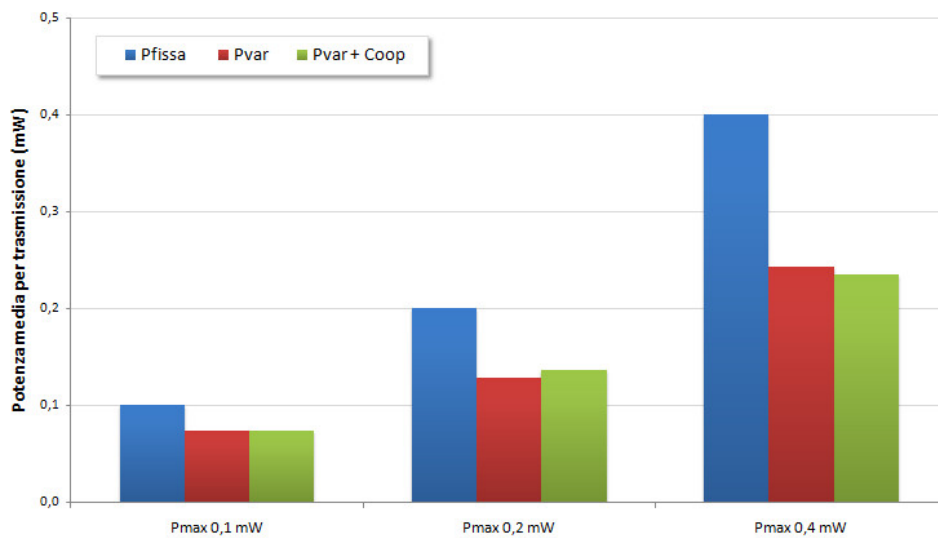
A bassa potenza la collaborazione è utile per aumentare il range delle trasmissioni e porta miglioramenti superiori a quelli del controllo di potenza mentre a potenza più alta, dove molti nodi sono già in visibilità, i rapporti si invertono poiché il controllo è determinante nel limitare l'interferenza. Nel caso a  $0,4\text{ mW}$  esso porta da solo a un guadagno del 7% medio contro il 6% circa della collaborazione. Inoltre la potenza media utilizzata è di  $0,24\text{ mW}$ , ossia del 40% inferiore al limite massimo 4.6b.

Osserviamo quindi che l'impiego congiunto del controllo di potenza e della collaborazione risulta vantaggioso ottenendo, in media, una riduzione totale della trama del 17%, del 15% e del 12% rispettivamente con potenza massima  $0,1\text{ mW}$ ,  $0,2\text{ mW}$  e  $0,4\text{ mW}$ . In 4.6a riportiamo in modo sintetico i vantaggi ottenuti in media con la collaborazione, il controllo di potenza e il loro utilizzo congiunto.





(a) Grafici dei guadagni apportati dal controllo di potenza e dalla collaborazione con diversi livelli di potenza massima



(b) Potenze medie utilizzate nei casi a diversa potenza massima

Figura 4.6: Grafici riassuntivi sull'uso di collaborazione e controllo di potenza

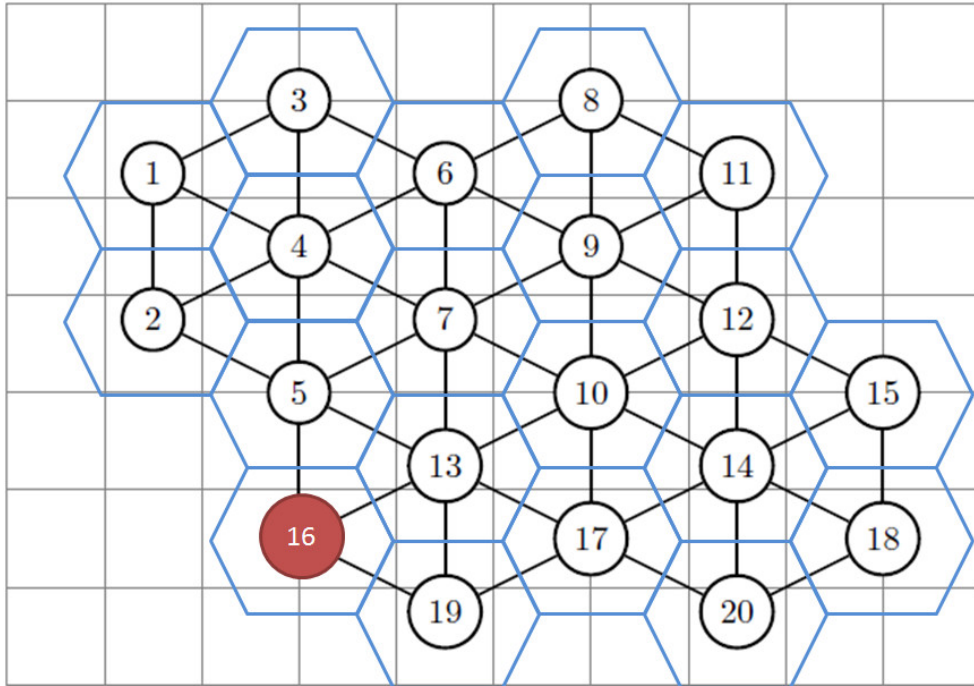
## 4.7 Esempio di applicazione

In questa sezione presenteremo un esempio su un'istanza creata ad hoc per valutare l'applicabilità e le prestazioni della collaborazione in uno scenario tipico di reti wireless.

Un possibile impiego delle reti mesh è quello di sostituire l'infrastruttura cablata nelle reti cellulari per le comunicazioni tra stazioni base e verso la dorsale. In 4.7 consideriamo il backhauling del traffico imponendo 10 domande che partono da varie stazioni base, ciascuna responsabile di una cella, e convergono verso un gateway centrale (nodo 16 in figura 4.7). Una configurazione del genere vale anche per una ampia area coperta con Wi-Fi con distribuzione WDS (Wireless Distribution System) in cui il traffico converge verso un gateway che connette la WLAN a internet.

In 4.7 abbiamo rappresentato una struttura di rete a celle regolari con il gateway nel nodo 16 e effettuato prove con varie combinazioni da 10 domande dirette verso di esso. Abbiamo riscontrato che la collaborazione offre effettivamente un guadagno in termini di slot che va dal 17% al 24% in base alle domande in gioco; l'introduzione della potenza variabile porta, in alcuni casi, ad un ulteriore miglioramento. La collaborazione risulta quindi una possibilità interessante da valutare per l'impiego in questo contesto.

Per dare un'idea più completa di come la cooperazione e il controllo di potenza possano agire per riportare i guadagni visti proponiamo nella figura 4.8 l'inoltro di due sole domande di traffico sul grafo cellulare, indicando gli slot utilizzati nella configurazione ottima. Nella prima immagine viene utilizzato l'inoltro hop-by-hop semplice in cui risulta chiaro come la scelta di cammini non minimi possa portare a un riuso spaziale più efficiente; nella seconda è evidente come la collaborazione possa ridurre il numero di salti per giungere alla destinazione; infine nella terza immagine la combinazione di cooperazione e controllo di potenza riesce a ridurre di un ulteriore slot la lunghezza della trama.



Domande	NoCoop	Coop	CoopPvar
D1	25	19	19
D2	23	19	17
D3	24	19	18

Figura 4.7: Grafo di scenario cellulare e domande analizzate

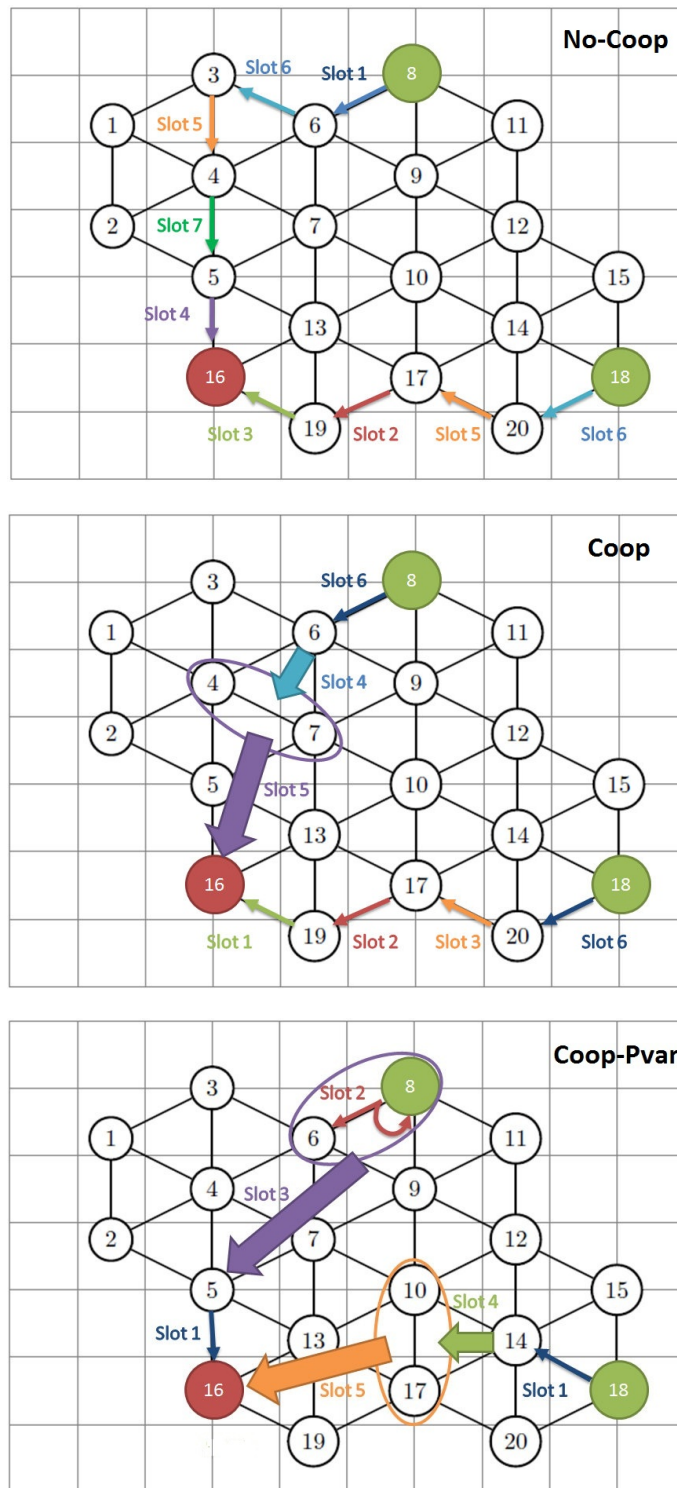


Figura 4.8: Inoltro di due domande su grafo cellulare

## 4.8 Scalabilità

Gli algoritmi euristici proposti hanno ottenuto buone performance sia in tempi che in risultati, soprattutto l'euristica basata sui cammini minimi che raggiunge l'ottimo nella maggior parte dei casi analizzati in tempi molto bassi.

Per questo motivo abbiamo deciso di provare ulteriormente la scalabilità di entrambi gli algoritmi su istanze da 25 e da 30 nodi con 5, 10 e 20 domande, dove il modello *DiSINR* è di difficile applicabilità per questioni di tempo e di memoria.

$ N $	$ D $	Ist	$ A $	$ A' $	Potenza fissa		Potenza variabile	
					UB	Tempi	UB	Tempi
25	5	I37	4762	170	9	50,8	8	117,9
		I38	6784	163	9	104,0	9	188,9
	10	I39	5220	365	11	223,1	11	263,1
		I40	6495	373	18	418,4	17	4257,8
	20	I41	8528	670	28	*7200,0	26	*7200,0
		I42	7192	693	31	*7200,0	29	*7200,0
30	5	I43	8751	193	7	77,9	7	133,7
		I44	11876	197	10	69,6	10	109,2
	10	I45	10435	356	17	526,7	16	652,2
		I46	11255	368	11	2626,1	10	1726,5
	20	I47	11585	689	33	*7200,0	31	*7200,0
		I48	13948	704	28	*7200,0	26	*7200,0

Tabella 4.9: Risultati e tempi ottenuti con l'euristica SP su istanze da 25 e 30 nodi(\*: superato il limite di tempo)

Riportiamo in tabella 4.9 per l'algoritmo *EurSP* la riduzione degli archi prodotta, i risultati e i tempi ottenuti su un totale di 12 istanze, sia a potenza fissa che variabile. Essendo la riduzione del grafo proporzionale al numero delle domande, sulle istanze da 5 i tempi sono ancora molto ridotti, su quelle da 10 crescono moderatamente, mentre diventano critici su quelle da 20 dove superano sempre il limite imposto. Tuttavia l'algoritmo ci offre un modo per arginare questo problema: diminuendo il numero di cammini minimi generati si può ridurre ulteriormente il grafo pagando in termini di

N	D	Ist	A	A'	Potenza fissa		Potenza variabile	
					UB	Tempi	UB	Tempi
25	20	I41	8528	367	28	751,5	26	1880,3
		I42	7192	372	32	901,9	29	2037,8
30	20	I47	11585	392	34	1714,4	31	3573,3
		I48	13948	376	28	2738,1	26	*7200,0

Tabella 4.10: Risultati e tempi ottenuti con l'euristica SP sulle istanze da 20 domande utilizzando  $k=10$ (\*: superato il limite di tempo)

N	D	Ist	A	Potenza fissa		Potenza variabile	
				UB	Tempi	UB	Tempi
25	5	I37	4762	9	2925,9	9	3924,7
		I38	6784	10	*7200,0	10	*7200,0
	10	I39	5220	11	4450,6	11	*7200,0
		I40	6495	19	*7200,0	19	3709,8
	20	I41	8528	30	*7200,0	30	*7200,0
		I42	7192	32	*7200,0	32	*7200,0

Tabella 4.11: Risultati e tempi ottenuti con l'euristica protocollare su istanze da 25 e 30 nodi(\*: superato il limite di tempo)

qualità delle soluzioni. In tabella 4.10 sono presenti i risultati con 10 cammini minimi per le istanze precedentemente andate in timeout. I tempi si riducono sensibilmente e solo l'istanza con più archi continua a superare il limite di tempo; le soluzioni peggiorano solo in due casi mentre negli altri rimangono le medesime.

Per quanto riguarda il modello protocollare, in tabella 4.11 sono presenti i risultati ottenuti e i tempi impiegati sulle istanze da 25 nodi, dove risulta già evidente la difficoltà incontrata dall'algorithm. Infatti la maggior parte delle istanze impiegano un tempo di esecuzione di oltre 7200 secondi ed è quindi inutile riportare quelle da 30 nodi. E' da notare che sulle istanze in cui entrambe le euristiche raggiungono il limite di tempo l'upper bound migliore è ottenuto da quella basata sui cammini minimi quindi questa, a parità di tempo, progredisce maggiormente nel processo risolutivo.

Osserviamo che per entrambi i modelli i tempi di risoluzione del problema riportati in tabella cominciano a divergere in modo consistente dal tempo

reale, che contiene anche quello speso da AMPL per gestire i modelli e le soluzioni. L'impatto maggiore è riscontrabile sul modello protocollare per il quale sono necessari un numero maggiore di controlli e strutture dati. Questo problema risiede nel linguaggio utilizzato e per scalare ulteriormente con le dimensioni delle istanze riducendo questo overhead sarebbe utile passare a un'implementazione in un linguaggio più efficiente.

# Capitolo 5

## Conclusioni

In questa tesi abbiamo analizzato uno scenario di reti wireless multi-hop, le quali costituiscono un importante strumento di comunicazione con applicazioni in numerosi ambiti commerciali. Per questo motivo sono un problema di grande attualità, verso cui la ricerca si sta spingendo allo scopo di produrre nuove soluzioni tecnologiche, sempre più efficienti e a costi competitivi sul mercato.

La nostra attenzione si è concentrata sullo studio del meccanismo della collaborazione applicato a questo tipo di reti. Basandoci su un approccio risolutivo presente in letteratura abbiamo messo a punto degli algoritmi per ridurre la complessità del problema e poterlo scalare su reti di dimensioni maggiori. In questo modo abbiamo ridotto notevolmente i tempi di esecuzione e osservato che l'approssimazione sulla qualità delle soluzioni è modesta in tutti i casi considerati. Inoltre abbiamo esteso il modello al caso di potenza variabile per valutare il miglioramento ottenibile a livello di soluzioni e i benefici dal punto di vista energetico.

Dai risultati riportati nel capitolo 4 emerge che, considerando la potenza di trasmissione fissa, la cooperazione permette di ottenere un guadagno nella maggior parte dei casi affrontati, tuttavia questo è più consistente quando la rete in esame è meno densa, tenendo presente che la densità è dovuta al rapporto fra le potenze e le distanze in gioco. Infatti in queste condizioni l'interferenza aggiuntiva causata da trasmissioni contemporanee ha un im-



patto ridotto sul livello di riuso riscontrabile, mentre si ottiene un vantaggio considerevole dall'aumento del raggio di trasmissione. Al contrario nelle reti molto dense l'utilizzo della collaborazione è meno consigliabile dato l'alto livello di interferenza presente nel sistema.

Il controllo di potenza risulta sempre utile, in quanto è fondamentale nel limitare l'interferenza e rende quindi possibile una maggiore applicabilità della collaborazione mitigandone gli effetti negativi al crescere delle potenze utilizzate. Alzando il limite massimo della potenza si possono ottenere ulteriori riduzioni della trama e, dato che i livelli massimi vengono usati solo in casi strettamente necessari, il consumo energetico medio aumenta in modo più contenuto.

Nonostante l'andamento generale riscontrato la struttura della rete in analisi ha un peso consistente nel determinare l'applicabilità o meno delle soluzioni tecnologiche considerate, quindi gli strumenti da noi proposti risultano utili per una valutazione veloce delle prestazioni nel singolo caso e per organizzare al meglio l'attività di progetto.

Se i risultati ottenuti ci permettono di asserire che la collaborazione può essere utile in molti casi negli scenari wireless, resta però da verificare quanto l'effetto delle problematiche implementative consenta di sfruttare il guadagno teorico stimato. Dato che il nostro modello è focalizzato sulla riduzione della trama si potrebbe modificare in modo da rendere centrale il controllo di potenza e ricercare un compromesso migliore tra consumo energetico e numero di slot utilizzati. Inoltre un possibile altro ambito di sviluppo è quello di studiare algoritmi che si avvalgano in modo dinamico e distribuito delle potenzialità della collaborazione.

# Bibliografia

- [1] M. Abolhasan, T. Wysocki, and E. Dutkiewicz. A review of routing protocols for mobile ad hoc networks. *Ad Hoc Networks*, 2(1):1–22, 2004.
- [2] A. Behzad and I. Rubin. Optimum integrated link scheduling and power control for multihop wireless networks. *IEEE Transactions on Vehicular Technology*, 56(1):194–205, 2007.
- [3] R. Bhatia and M. Kodialam. On power efficient communication over multi-hop wireless networks: Joint routing, scheduling and power control. In *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM 2004*, volume 2, 2004.
- [4] P. Bjorklund, P. Varbrand, and D. Yuan. A column generation method for spatial TDMA scheduling in ad hoc networks. *Ad hoc networks*, 2(4):405–418, 2004.
- [5] A. Bletsas, H. Shin, and MZ Win. Cooperative communications with outage-optimal opportunistic relaying. *IEEE Transactions on Wireless Communications*, 6(9):3450–3460, 2007.
- [6] A. Capone and G. Carello. Scheduling optimization in wireless MESH networks with power control and rate adaptation. In *Proc. of 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks, 2006. SECON 2006*, 1, 2006.
- [7] A. Capone, G. Carello, I. Filippini, S. Gualandi, and F. Malucelli. Solving a resource allocation problem in wireless mesh networks: A compa-

- parison between a CP-based and a classical column generation. *Networks*, 55(3):221–233, 2010.
- [8] A. Capone, S. Gualandi, and D. Yuan. Performance Evaluation of Cooperative Forwarding Schemes in Wireless Networks with General Topology and Multiple Traffic Flows: an Optimization Approach.
- [9] T. Cover and AE Gamal. Capacity theorems for the relay channel. *IEEE Transactions on Information Theory*, 25(5):572–584, 1979.
- [10] RL Cruz and AV Santhanam. Hierarchical link scheduling and power control in multihop wireless networks. In *In proc. of the annual allerton conference on communication control and computing*, volume 40, pages 1543–1554, 2002.
- [11] R.L. Cruz and A.V. Santhanam. Optimal routing, link scheduling, and power control in multi-hop wireless networks. In *IEEE INFOCOM 2003*, volume 1, pages 702–711, 2003.
- [12] M. Dohler, A. Gkelias, and H. Aghvami. 2-hop distributed MIMO communication system. *IEEE Electronics Letters*, 39(18):1350–1351, 2003.
- [13] M. Dohler, F. Said, and H. Aghvami. Concept of virtual antenna arrays. In *In Proc. of IEEE Globecom 2002*.
- [14] P. Gupta and PR Kumar. The capacity of wireless networks. *IEEE Transactions on information theory*, 46(2):388–404, 2000.
- [15] P. Herhold, E. Zimmermann, and G. Fettweis. Cooperative multi-hop transmission in wireless networks. *Computer Networks*, 49(3):299–324, 2005.
- [16] M. Johansson and L. Xiao. Cross-layer optimization of wireless networks using nonlinear column generation. *IEEE Transactions on Wireless Communications*, 5(2):435–445, 2006.

- [17] J.N. Laneman, D.N.C. Tse, and G.W. Wornell. Cooperative diversity in wireless networks: Efficient protocols and outage behavior. *IEEE Transactions on Information theory*, 50:12, 2004.
- [18] Y. Li and A. Ephremides. A joint scheduling, power control, and routing algorithm for ad hoc wireless networks. *Ad Hoc Networks*, 5(7):959–973, 2007.
- [19] C. Molle, F. Peix, S. Pérennes, and H. Rivano. Optimal Routing and Call Scheduling in Wireless Mesh Networks with Localized Informations. *Trustworthy Global Computing*, pages 171–185, 2009.
- [20] K. Papadaki and V. Friderikos. Approximate dynamic programming for link scheduling in wireless mesh networks. *Computers & Operations Research*, 35(12):3848–3859, 2008.
- [21] M. Qin and RS Blum. Capacity of wireless ad hoc networks with cooperative diversity: a warning on the interaction of relaying and multi-hop routing. In *In Proc. of IEEE International Conference on Communications, 2005. ICC 2005*, volume 2, 2005.
- [22] G. Scutari, S. Barbarossa, and D. Ludovici. Cooperation diversity in multihop wireless networks using opportunistic driven multiple access. In *In Proc. of IEEE Workshop on Signal Processing Advances in Wireless Communications, 2003. SPAWC 2003*, pages 170–174, 2003.
- [23] Y. Shi, S. Sharma, Y.T. Hou, and S. Kompella. Optimal relay assignment for cooperative communications. In *In Proc. of the 9th ACM international symposium on Mobile ad hoc networking and computing*, pages 3–12, 2008.
- [24] J.B. Si, Z. Li, L.J. Dang, and Z.J. Liu. Joint optimization of relay selection and power allocation in cooperative wireless networks. In *In Proc. of 11th IEEE Singapore International Conference on Communication Systems, 2008. ICCS 2008*, pages 1264–1268, 2008.

- [25] J. Tang, G. Xue, C. Chandler, and W. Zhang. Link scheduling with power control for throughput enhancement in multihop wireless networks. *IEEE Transactions on Vehicular Technology*, 55(3):733–742, 2006.
- [26] E.C. Van Der Meulen. Three terminal communication channels. *Advances in Applied Probability*, pages 120–154, 1971.
- [27] Y. Zhu and H. Zheng. Understanding the impact of interference on collaborative relays. *IEEE Transactions on Mobile Computing*, 7(6):724–736, 2008.