

Università degli studi di Bologna

FACOLTA' DI INGEGNERIA

Corso di Laurea in Ingegneria Informatica
Reti di Calcolatori L-A

**PROGETTO DI SERVIZI
RESOURCE-AWARE PER
RETI AD-HOC**

Tesi di laurea di:
Stefano Landi

Relatore:
Chiar.mo Prof. Ing. Antonio Corradi

Correlatore:
Ing. Dario Bottazzi

ANNO ACCADEMICO 2003-2004

Indicazione delle parole chiave:

reti

wireless

MANET

servizio

algoritmo

simulatore

test

INDICE

Introduzione	5
1 - Lo Scenario delle Reti Ad-Hoc	7
1.1 - Le Body Area Networks	8
1.2 - Le Personal Area Networks	9
1.3 - Le Wireless Local Area Networks	9
1.4 - Le Wireless Wide Area Networks	9
1.5 - Alcuni Esempi di Tecnologie	10
1.5.1 - Il Bluetooth	10
1.5.1.1 - Reti Bluetooth: Piconet e Scatternet	10
1.5.2 - IEEE 802.11	12
1.5.2.1 - Architettura 802.11	12
1.5.2.1.1 - IEEE 802.11 DCF	12
1.5.2.2 - Modalità di Operazione	13
1.6 - Modelli per Reti Wireless	14
1.6.1 - Il Modello a Cella	14
1.6.2 - Il Modello a Cella Virtuale	15
1.6.3 - Il Modello Ad-Hoc	16
1.7 - Le Mobile Ad-hoc NETWORK (MANET)	16
1.7.1 - Il Routing nelle MANET	18
1.7.1.1 - Routing Unicast	19
1.7.1.1.1 - I Protocolli Proattivi	19
1.7.1.1.1.1 - Il Destination Sequenced Distance Vector	19
1.7.1.1.2 - I Protocolli Reattivi	20
1.7.1.1.2.1 - Il Dynamic Source Routing Protocol	20
1.7.1.1.3 - I Protocolli Ibridi	22
1.7.1.1.3.1 - Lo Zone Routing Protocol	22
1.7.1.2 - Routing Multicast	24
1.7.1.2.1 - L'On-Demand Multicast Routing Protocol	24
2 - Il Framework AGAPE	27
2.1 - Le Entità AGAPE e il Concetto di Località	29
2.2 - L'Architettura di AGAPE	30
2.2.1 - Il Basic Service Layer	31

2.2.2 - Il Group Management Layer	32
3 - Il View Controlling Service	37
3.1 - Il Problema nello Scenario Specifico Affrontato	40
3.1.1 - Linee Guida del Problema	41
3.2 - Requisiti dell'Algoritmo	43
3.2.1 - L'Algoritmo Cardine del VCS	45
3.2.1.1 - Integrazione nel Framework AGAPE	50
3.3 - Dettagli Implementativi	51
3.3.1 - Località Innestate	52
3.3.2 - Unirsi ad un Gruppo	53
3.3.3 - Propagazione delle Viste	53
3.3.4 - Località Parzialmente Sovrapposte	55
4 - Test del Sistema Sviluppato	57
4.1 - Il Simulatore	58
4.1.1 - L'Architettura del Simulatore	59
4.1.2 - L'interfaccia grafica	62
4.1.2.1 - L'Input	63
4.1.2.2 - Lo Stato del Sistema	64
4.1.2.3 - L'Output	64
4.2 - Alcuni Test e Valutazione delle Performance	65
4.2.1 - Lo Sviluppo di una Piccola Rete	65
4.2.2 - Generazione di Scenari Random	74
Conclusioni	79
Bibliografia	81

INTRODUZIONE

La crescente proliferazione di dispositivi portatili che possono fruire di connettività wireless, i recenti sviluppi delle tecnologie wireless e l'emergere delle Mobile Ad-hoc NETWORK (MANET) aprono uno scenario innovativo nel quale gli utenti non richiedono solo la possibilità di fruire dei tradizionali servizi Internet, quali il web o la e-mail, ma anche la possibilità di beneficiare di servizi collaborativi avanzati. In particolare gli utenti richiedono la possibilità di collaborare ovunque essi siano, in qualunque momento e possibilmente anche quando sono in movimento. Esempi di servizi abilitati dalle tecnologie MANET sono costituiti da applicazioni per il file sharing fra utenti mobili, dal coordinamento di veicoli e da scenari di protezione civile.

Le caratteristiche delle MANET sollevano molti problemi nello sviluppo di servizi collaborativi. La topologia della rete non è determinabile a priori, rendendo impossibili assunzioni sulla disponibilità delle diverse entità interagenti. Disconnessioni, partizioni e merge della rete sono eventi comuni che causano transitori nella collaborazione fra entità nuove e precedentemente sconosciute. Un ulteriore elemento di complessità deriva inoltre dalla richiesta degli utenti di poter collaborare tramite dispositivi eterogenei sia per natura sia per prestazioni, quali lap-top, PDA o telefoni cellulari. Tali dispositivi sono caratterizzati dalla scarsità delle risorse a disposizione, in particolare banda di trasmissione e batteria.

In questa tesi verrà presentato il framework AGAPE, un sistema di gestione dei gruppi che abilita la collaborazione fra utenti che condividono interessi e obiettivi.

In particolare ci focalizzeremo sulle strategie che consentono l'ottimizzazione dell'uso delle risorse in maniera tale da permettere agli utenti di interagire col sistema in maniera efficiente e per il maggior tempo possibile.

La tesi è organizzata come segue: il *Capitolo 1* introduce il nuovo scenario aperto dalle tecnologie MANET mentre il *Capitolo 2* presenta il framework AGAPE.

Il *Capitolo 3* introduce le strategie che consentono l'ottimizzazione dell'uso delle risorse in AGAPE e la loro l'integrazione nel sistema. Infine il *Capitolo 4* mostra i test effettuati sul sistema sviluppato.

CAPITOLO 1

LO SCENARIO DELLE RETI AD-HOC

La proliferazione di dispositivi mobili, quali pc portatili, PDA, palmari, che possono beneficiare di connettività wireless, e la disponibilità ubiqua di questa aprono un nuovo scenario in cui gli utenti richiedono la possibilità di accedere a servizi di interesse ovunque si trovano, in qualunque momento e mentre sono in movimento.

In base alle funzioni per cui è normalmente utilizzata, una classificazione comunemente accettata divide le reti wireless in BAN, PAN, WLAN e WWLAN (fig. 1.1).

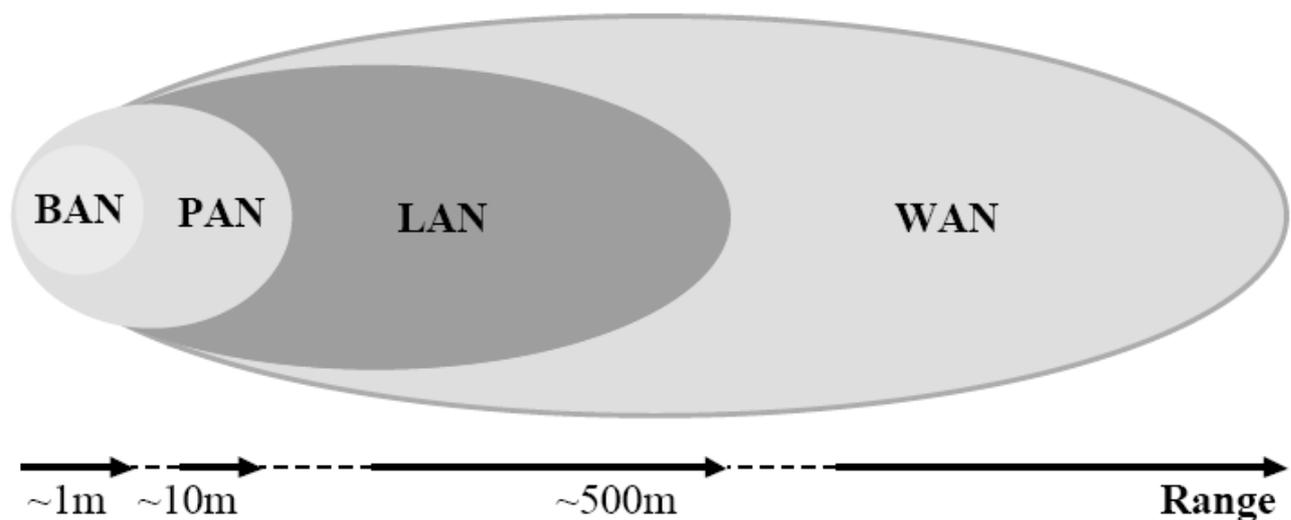


fig. 1.1 – Rappresentazione delle MANET in funzione del raggio di trasmissione

1.1 Le Body Area Networks

Una BAN è strettamente correlata all'uso di dispositivi "indossabili": il suo scopo è quindi quello di fornire connettività tra questi ultimi. Un esempio tipico può essere quello di un telefono che ha necessità di comunicazione con un display o un microfono.

Uno dei primi esempi di Body Area Network (BAN) è costituito dal prototipo sviluppato da T. G. Zimmerman, che utilizzava il corpo umano come canale di comunicazione: egli dimostrò infatti che i dati potevano essere trasferiti attraverso la pelle con l'uso di una piccolissima corrente (nell'ordine di un miliardesimo di amp).

Ulteriori requisiti sono la capacità della BAN di autoconfigurarsi e di fornire la capacità di interfacciarsi con altre BAN. Il tipico raggio

di trasmissione è quindi solitamente limitato al corpo umano, e perciò è dell'ordine di 1 o 2 metri. Per quanto riguarda la banda le BAN hanno a disposizione tipicamente alcune centinaia di Kbps.

1.2 Le Personal Area Networks

Le Personal Area Network (PAN) mirano a consentire la comunicazione dei dispositivi che l'utente possiede (per es.: Pc portatile, PDA, telefono cellulare) con altri presenti in prossimità. Le PAN possono per esempio consentire automaticamente sincronizzazioni e comunicazioni dei dispositivi personali di un utente al variare dell'ambiente nel quale quest'ultimo viene di volta in volta a trovarsi.

Le PAN hanno un raggio di trasmissione nell'ordine della decina di metri e la banda disponibile è tipicamente di ~ 1 Mbps.

Uno standard ormai affermatosi in questo campo è quello del Bluetooth, grazie al basso costo di integrazione di questa tecnologia nei dispositivi più vari.

1.3 Le Wireless Local Area Networks

Le Wireless Local Area Network (WLAN) hanno l'obiettivo di fornire un supporto analogo a quello delle LAN in scenari in cui la copertura è nell'ordine delle centinaia di metri, raggio che consente quindi di coprire un edificio o un blocco di edifici caratterizzati da difficoltà di deployment (es: ambienti domestici) o di deployment limitato nel tempo (es: concerti) o di mobilità degli utenti (es: aeroporti).

La banda a disposizione è nell'ordine delle decine di Mbps.

Una tecnologia largamente utilizzata per le WLAN è IEEE 802.11.

1.4 Le Wireless Wide Area Networks

Le Wireless Wide Area Networks (WWAN) garantiscono copertura geografica (es.: nazionale) e una banda dalle decine alle centinaia di Kbps. Alcuni esempi molto noti sono le reti cellulari quali il GPRS e l'UMTS.

1.5 Alcuni Esempi di Tecnologie

Esempi di tecnologie particolarmente diffuse per le reti wireless sono il Bluetooth e l'IEEE 802.11.

1.5.1 Il Bluetooth

La tecnologia Bluetooth è divenuta uno standard de facto per le comunicazioni a basso costo e a breve raggio di trasmissione tra dispositivi quali PC portatili, telefoni cellulari e altri dispositivi mobili. La banda di lavoro del Bluetooth è quella dello spettro ISM (Industrial, Scientific and Medical) a 2.4 GHz. A causa del basso costo di questa tecnologia si può prevedere che microchip Bluetooth saranno integrati in un numero sempre maggiore di dispositivi elettronici.

1.5.1.1 Reti Bluetooth: Piconet e Scatternet

Dispositivi equipaggiati dello standard Bluetooth possono formare reti temporanee nelle quali una stazione ha il ruolo di master e le altre di slaves. È il master a decidere quale stazione slave deve avere accesso al canale: le unità che condividono lo stesso canale formano una *piconet* (fig. 1.2), il blocco costitutivo fondamentale di una rete Bluetooth. Una piconet ha una velocità di trasmissione nominale di 1 Mbps e può essere composta, oltre che dal master, da un numero massimo di sette slaves.

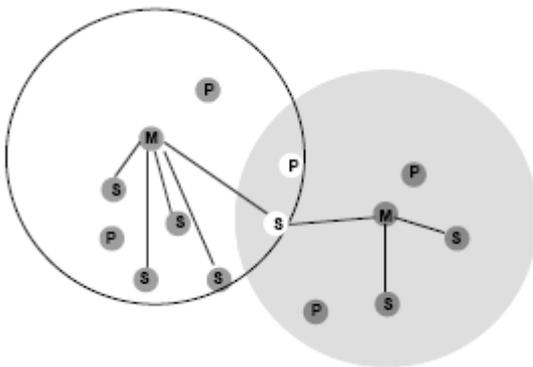


fig. 1.2 – Due piconets parzialmente sovrapposte: M=Master, S=Slave, P=Parking State, ovvero stazioni sincronizzate col M che però non stanno partecipando a scambi di dati

Un'unità Bluetooth consiste in un'unità radio operante nella banda di 2.4 GHz. In questa banda sono definite 79 differenti frequenze radio distanziate di 1 MHz. Il radio layer utilizza come tecnica di trasmissione il *Frequency Hopping Spread Spectrum* (FHSS). La frequenza di hop è una sequenza pseudo-random di lunghezza pari a 79 hop ed è unica per ciascuna piconet (dipende dai parametri locali del master). Questo sistema di trasmissione è stato scelto per evitare possibili disturbi derivanti da altri sistemi vicini operanti nella stessa banda di frequenze.

Prima di iniziare la trasmissione una unità Bluetooth ha bisogno di scoprire se altre unità sono presenti nella sua area operativa. Per fare ciò essa entra nello stato di *inquiry*, nel quale manda continuamente un messaggio di *inquiry*: quando un'unità riceve tale messaggio, se è interessata, può rispondere con uno speciale pacchetto di controllo che contiene l'indirizzo del dispositivo e il suo clock nativo.

Dopo tale fase un'unità Bluetooth beneficia quindi della visibilità degli indirizzi delle unità che si trovano in prossimità. Inoltre ogni entità dispone della stima del clock dei dispositivi ad essa vicini.

Per attivare una nuova connessione è sufficiente quindi che tale unità risponda a sua volta trasmettendo il suo stesso indirizzo e il suo stesso clock.

Nel caso di piconet con aree sovrapposte si può avere la formazione di una scatternet, cioè di una rete più vasta che può esistere nel momento in cui una unità è attiva allo stesso tempo in più piconet. Uno slave può comunicare con nodi appartenenti ad una piconet diversa dalla sua, sfruttando come ponte tali nodi in comune, ma solo in time-multiplexing: un stazione infatti può trasmettere nello stesso istante solo nella piconet con la quale è sincronizzata. Per trasmettere in un'altra deve cambiare i suoi parametri di sincronizzazione.

È d'altra parte necessario però precisare che le attuali specifiche Bluetooth definiscono una scatternet ma non forniscono ancora specifiche implementative complete.

1.5.2 IEEE 802.11

Per l'implementazione delle reti wireless di raggio più ampio (WLAN) nel 1997 nacque un progetto i cui risultati forniscono tuttora lo standard maggiormente utilizzato in questo campo e il cui successo è dovuto alla sua estrema semplicità.

Questo standard IEEE ha preso il nome di IEEE 802.11 ed è stato in seguito esteso da 802.11a e da 802.11b. L'originaria banda nominale di 2 Mb/sec è aumentata fino a 54 Mb/sec.

La rete 802.11 lavora tipicamente nello spettro ISM e quindi a 2.4 GHz ma esistono implementazioni dello standard a 5 GHz (802.11a) ed anche nell'infrarosso (802.11g).

1.5.2.1 Architettura 802.11

802.11 fornisce specifiche di architettura sia per il livello fisico sia per quello di controllo di accesso al mezzo di trasmissione (MAC, acronimo di Medium Access Control).

Per il primo sono previste due metodologie di trasmissione, quella attraverso i raggi infrarossi e quella che sfrutta invece le onde radio.

Per il secondo 802.11 prevede essenzialmente due protocolli, il *Distributed Coordination Function* (DCF) e il *Point Coordination Function* (PCF).

1.5.2.1.1 IEEE 802.11 DCF

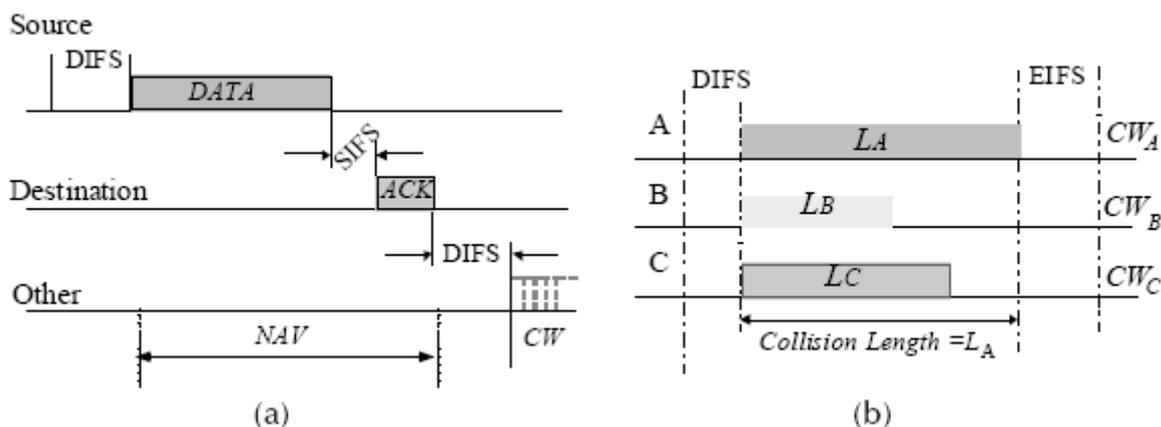


fig. 1.3 – Trasmissione di pacchetti con IEEE 802.11 DCF

(a) Trasmissione con successo

(b) Collisione

Utilizzando il DCF una stazione, prima di iniziare a trasmettere (fig. 1.3a), controlla il canale per accorgersi se un'altra stazione lo sta già utilizzando. Se quest'ultimo viene trovato libero per un tempo che supera una certa soglia, il *Distributed InterFrame Space* (DIFS), allora la stazione continua con la sua trasmissione. Il pacchetto trasmesso ha un campo contenente la lunghezza prevista della trasmissione. Ciascuna stazione attiva mantiene questa informazione in una variabile locale chiamata *Network Allocation Vector* (NAV). Il NAV contiene quindi il periodo di tempo per il quale il canale rimarrà occupato. Dei messaggi di Acknowledgement (ACK) vengono utilizzati per confermare l'avvenuta ricezione del pacchetto trasmesso. Più precisamente il ricevitore, dopo l'arrivo del pacchetto, attende per un certo intervallo di tempo chiamato *Short InterFrame Space*, più breve del DIFS, ed inizia quindi la trasmissione dell'ACK. Quest'ultimo non viene trasmesso se il pacchetto è corrotto o perso a causa di una collisione.

Una collisione avviene (fig. 1.3b) quando due o più stazioni iniziano a trasmettere nello stesso istante. Se l'ACK non viene ricevuto la trasmissione del pacchetto viene considerata fallita e quindi ripetuta. Quando un errore viene rilevato (a causa di collisioni o errori di trasmissione) il canale deve rimanere libero per un tempo detto *Extended InterFrame Space* (EIFS) prima che le stazioni riattivino un meccanismo di ripresa delle trasmissioni. Viene quindi utilizzato un algoritmo di backoff, che ha la seguente modalità di operazione: quando una stazione rileva il canale occupato rimanda la propria trasmissione al termine di quella in corso; quindi inizializza un contatore selezionando un intervallo random (*backoff interval*): trascorso tale lasso di tempo se il canale è ancora rilevato libero la trasmissione inizia, in caso contrario il procedimento viene reiterato.

1.5.2.2 Modalità di Operazione

Una 802.11 WLAN può essere implementata con l'ausilio di access point (ovvero con un'infrastruttura fissa) oppure seguendo il paradigma ad-hoc (senza tale infrastruttura).

Nel primo caso si ha una *Basic Service Set* (BSS), ovvero una rete wireless composta da almeno un *Access Point* (AP) e uno o più nodi

mobili. Tutte le comunicazioni provenienti ed indirizzate a tali nodi passano attraverso l'AP, che si occupa quindi della loro coordinazione.

Il secondo caso definisce invece, nello standard 802.11, una rete ad-hoc: tale rete viene chiamata *Independent Basic Service Set* (IBSS). Un IBSS rende possibile la comunicazione diretta tra due o più stazioni 802.11 senza necessità di un access point centrale o di una infrastruttura fissa: la sincronizzazione è infatti sufficiente a garantire una corretta ricezione dei dati.

802.11 si serve principalmente di due funzioni per raggiungere tale essenziale sincronizzazione: quella di acquisizione e quella di mantenimento della sincronizzazione.

La prima è necessaria ad una stazione per entrare in un IBSS già esistente: la scoperta di un IBSS già attiva viene effettuata con un procedimento di scansione del mezzo wireless su diverse frequenze di trasmissione radio. Solo se tale scansione ha esito negativo allora la stazione può inizializzare una nuova IBSS.

La seconda è invece necessaria a causa della mancanza di un nodo centrale che provveda a fornire il proprio segnale di clock come clock comune. La sincronizzazione viene quindi effettuata sfruttando un algoritmo distribuito che deve essere eseguito da ciascun membro dell'IBSS. Tale algoritmo si basa sulla trasmissione di piccoli frame di dati detti *beacon frames*. La stazione che crea l'IBSS decide l'intervallo di trasmissione di tali segnali di sincronismo.

1.6 Modelli per Reti Wireless

Per quanto che riguarda i modelli di comunicazione, in funzione della crescente flessibilità e della diminuzione dell'appoggio alla rete cablata, nelle mobile networks vengono convenzionalmente distinti tre modelli: il modello a cella, il modello a cella virtuale ed il modello ad-hoc.

1.6.1 Il Modello a Cella

Una rete cellulare copre una certa area divisa in celle le quali possono sovrapporsi le une alle altre. Ciascuna cella ha una stazione

base fissa (BS). Le stazioni base sono connesse tra di loro in maniera cablata (fig. 1.4). I nodi mobili (anche detti mobile hosts o MH) possono passare da una cella all'altra. Un MH comunica sempre con gli altri nodi presenti nel sistema attraverso la BS della propria cella. Per fare ciò, un MH deve stabilire un collegamento wireless con la sua BS. Se il destinatario della comunicazione è anch'esso presente nella stessa cella, la BS inoltra il messaggio al destinatario grazie ad un altro collegamento wireless. Se, al contrario, il destinatario si trova in un'altra cella, la BS inoltra il messaggio via cavo alla base station della cella di destinazione che provvederà all'inoltro finale (fig 1.5).

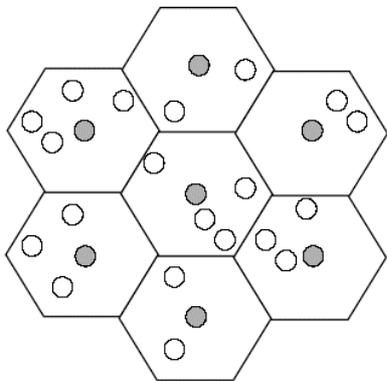


fig 1.4 – Modello a cella

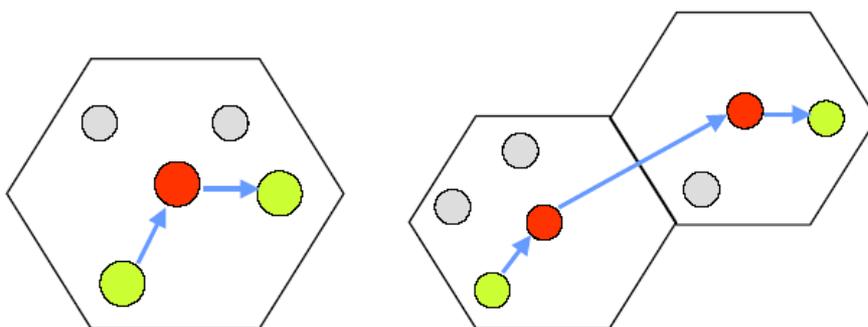


fig. 1.5 – Comunicazione all'interno di una cella e tra celle diverse

1.6.2 Il Modello a Cella Virtuale

Il modello a cella virtuale è simile al precedente, ma in questo modello anche le BS sono mobili e quindi connesse in maniera wireless. Analogamente al modello a cella, anche nel modello a cella virtuale le BS coordinano le trasmissioni fra i vari MH. Contrariamente al modello a cella però, il grafo delle BS può cambiare in funzione del tempo.

Tutte le comunicazioni avvengono sempre tra BS, tranne per ciò che riguarda il tratto iniziale ed il tratto finale della trasmissione. L'MH sorgente invia infatti i dati alla propria BS, che eventualmente (nel caso di trasmissione ad un host appartenente ad una cella diversa) provvede a trasferirli, attraverso le necessarie BS, alla BS della cella di destinazione: a questo punto soltanto avviene il trasferimento finale all'host di destinazione, tutti gli altri tratti sono sempre e solo tra BS.

1.6.3 Il Modello Ad-Hoc

Il modello ad-hoc si ottiene rilassando il modello a cella virtuale: in particolare nel modello ad-hoc sfuma la distinzione dei ruoli di BS e MH.

In una rete ad-hoc quindi tutti i nodi hanno un rapporto cooperativo, dato che non esistono punti di riferimento né fissi (access point) né mobili e che tutti i nodi sono caratterizzati dalla possibilità di spostarsi. Non esistono stazioni mobili che coordinino le attività di un certo sottoinsieme di nodi e di conseguenza tutti gli hosts devono prendere decisioni in maniera collettiva aprendo quindi una serie di nuove problematiche.

A causa della mobilità l'insieme dei nodi vicini a ciascun host varia nel tempo. Poiché gli spostamenti non sono prevedibili, i cambiamenti nella topologia della rete sono nel tempo arbitrari. Tutte le comunicazioni avvengono esclusivamente in maniera wireless. Le reti ad-hoc sono particolarmente adatte a quelli scenari dove non esiste alcun tipo di infrastruttura, quali per esempio missioni belliche o di protezione civile in zone disastrose.

1.7 Le Mobile Ad-hoc NETWORK (MANET)

Una MANET è definita nell'ambito IETF (Internet Engineering Task Force) come un sistema di router mobili che comunicano tra loro tramite tecnologie wireless: l'unione di tali router può formare un grafo arbitrario. Le MANET non richiedono la presenza di un'infrastruttura fissa e predefinita, ma possono formarsi ovunque, in qualunque momento e fra qualunque insieme di dispositivi. Nelle MANET la connettività è basata sulla reciproca

prossimità dei nodi. In ogni istante inoltre possono sopraggiungere partecipanti nuovi e precedentemente sconosciuti.

Ne consegue la possibilità di una conformazione non pianificabile a priori e potenzialmente in continua evoluzione: i nodi possono connettersi e disconnettersi a tempi imprevedibili e mutare frequentemente la loro posizione nella rete: la configurazione di quest'ultima è quindi in continua discussione. Partizionamenti di una MANET in più reti distinte o fusioni di MANET originariamente congiunte sono eventi frequenti e richiedono un adattamento della conformazione: una immediata conseguenza di ciò è il passaggio attraverso fasi di transitorio nella collaborazione tra i nodi.

L'impossibilità di effettuare assunzioni sulla topologia della rete e sulla disponibilità di un nodo rendono impossibile una soluzione di tipo centralizzato.

I collegamenti, contrariamente a quanto accade nelle reti cablate, possono infatti essere unidirezionali: è quindi possibile avere situazioni in cui, sebbene sia possibile la comunicazione dal nodo *A* al nodo *B*, non lo sia dal nodo *B* al nodo *A*.

Dal punto di vista della sicurezza, esse presentano i problemi di base che affliggono qualunque infrastruttura di rete wireless, connessi all'intrinseca debolezza del mezzo di comunicazione, come la facilità di interagire con la rete e di intercettare le informazioni in transito.

Le MANET presentano notevoli problemi in più rispetto, oltre che alle LAN cablate, anche a quelle reti wireless che si appoggiano però anche ad un'infrastruttura fissa (infrastructure-based). In una MANET d'altra parte viene per definizione a mancare il punto di riferimento: i nodi devono quindi auto-organizzarsi in una rete in maniera peer-to-peer.

Vanno quindi presi particolari accorgimenti in più rispetto alle reti cablate: per esempio un algoritmo di aggiornamento della topologia della rete che operi continuamente anche in assenza di traffico spreca inutilmente risorse. D'altro canto la decisione del path lungo il quale instradare un pacchetto non può basarsi solamente sulla

lunghezza del percorso: la strada più breve potrebbe infatti avere vita ben più breve rispetto ad un'altra apparentemente peggiore.

Un ulteriore aspetto da considerare è la parità del contributo fornito alla MANET da ciascun nodo: ogni stazione, nel caso la topologia della rete lo consenta, dovrebbe infatti contribuire in maniera approssimativamente uguale all'instradamento dei pacchetti. In caso contrario si assisterebbe ad uno sfruttamento maggiore delle risorse di un nodo rispetto ad un altro, evenienza da evitare in quanto si ridurrebbe notevolmente la possibilità di partecipazione alle comunicazioni di un nodo rispetto agli altri.

Il consumo delle risorse dei dispositivi per fornire i servizi di comunicazione necessari diviene quindi un problema cruciale: una ridondanza di trasmissione delle informazioni può infatti compromettere il buon funzionamento della rete. Infatti maggiori sono le diffusioni inutili di messaggi, maggiore è il consumo delle batterie dei dispositivi e maggiore è lo spreco di banda disponibile, entrambe risorse essenziali in una rete ad-hoc.

1.1.7 Il Routing nelle MANET

Un problema fondamentale in una MANET è quello del routing: i nodi devono infatti cooperare affinché i pacchetti dalla sorgente possano raggiungere la destinazione: ciò è necessario in quanto ogni stazione può riuscire a comunicare in maniera diretta soltanto nel proprio raggio di trasmissione, mentre tipicamente il ricevitore può trovarsi ad una distanza molto maggiore. Questo è il motivo per il quale le MANET si collocano nella classe delle reti wireless multi-hop.

Poiché in questo caso esistono tipicamente più percorsi possibili tra la sorgente e la destinazione, il protocollo di routing deve svolgere due compiti: il calcolo del miglior percorso possibile (best path) e l'instradamento del pacchetto all'interno della rete. Quest'ultimo può essere realizzato in due diversi modi: mantenendo in ogni nodo una *routing table* locale, che conservi per ogni possibile direzione il prossimo nodo da raggiungere lungo il path, oppure memorizzando all'interno del pacchetto stesso da inviare tutto il percorso e quindi anche tutti i nodi attraverso i quali esso va instradato.

Una classificazione comunemente accettata nella letteratura divide il routing in due grandi famiglie di protocolli: il routing unicast e il routing multicast.

1.7.1.1 Routing Unicast

Pur non esistendo protocolli standard, molti lavori di ricerca sono emersi in tempi recenti. Una suddivisione largamente accettata dei protocolli unicast per le MANET li divide in tre famiglie: i protocolli proattivi, reattivi e ibridi.

1.7.1.1.1 I Protocolli Proattivi

I protocolli proattivi sono nati da tentativi iniziali di adattare alle reti wireless (preservando però al contempo la loro natura) i protocolli convenzionalmente utilizzati per le reti cablate. Questi protocolli determinano tutti i possibili path sorgente - destinazione e li memorizzano nelle routing tables. Il vantaggio di questi protocolli deriva quindi dalla possibilità di invio immediato di un pacchetto senza dover prima calcolare il path, dato che quest'ultimo è già preventivamente memorizzato.

Lo svantaggio è al contrario dato dal maggiore overhead dovuto all'aggiornamento continuo della routing tables.

1.7.1.1.1.1 Il Destination Sequenced Distance Vector (DSDV)

Il protocollo DSDV tiene periodicamente aggiornate le tabelle di routing in modo da mantenere la consistenza all'interno della rete.

Per ottenere ciò viene utilizzato un meccanismo di "invecchiamento" basato sul semplice incremento di numeri di sequenza per indicare la freschezza di un percorso.

L'aggiornamento delle informazioni può avvenire in due modi differenti:

- 1) aggiornamento time driven, in cui i dispositivi periodicamente trasmettono le loro routing table ai nodi vicini per mantenere le tabelle consistenti;
- 2) aggiornamento event driven, dove un nodo invia la sua tabella a tutti gli altri nodi a causa di un improvviso cambiamento della stessa.

Gli aggiornamenti delle table possono essere eseguiti tramite full dump o incremental update: il primo è utilizzato per inviare tutte le informazioni di routing disponibili, il secondo invece viene usato per risparmiare risorse quando sono pochi i cambiamenti rispetto all'ultimo full dump.

Ad ogni fase di aggiornamento una entry in una routing table viene modificata solo se l'informazione ricevuta è più recente o se, a parità di freschezza, ha una valutazione migliore. Inoltre per evitare continue modifiche alle routing tables vengono procrastinati gli aggiornamenti dei paths ritenuti più instabili.

1.7.1.1.2 I Protocolli Reattivi

I protocolli reattivi presentano invece un diverso approccio: i path vengono calcolati solo al momento di un effettivo bisogno: se non vi è traffico generato dai nodi, allora l'attività di routing è totalmente assente, consentendo quindi un notevole risparmio di risorse. Per questo motivo vengono anche chiamati protocolli on-demand. Le routing tables vengono dunque eliminate e i path vengono calcolati con una procedura di flooding.

1.7.1.1.2.1 Il Dynamic Source Routing Protocol (DSR)

Il protocollo DSR invia i pacchetti specificando nell'header il percorso che gli stessi dovranno seguire. In tal modo si hanno diversi vantaggi: innanzitutto si evita la possibilità di creazione di anelli nel percorso, anelli che portano ad un mancato raggiungimento della destinazione finale poiché i pacchetti rimangono per un tempo indeterminato bloccati in un percorso chiuso. Inoltre viene meno in questo modo la necessità di mantenere informazioni di instradamento aggiornate nei nodi intermedi che i pacchetti devono attraversare e, nel caso la memorizzazione di tali informazioni sia necessaria, i nodi intermedi possono ricavare tali dati direttamente leggendo gli header dei pacchetti stessi.

I meccanismi principali di cui questo protocollo si serve sono due: il Route Discovering (scoperta della strada che i pacchetti devono seguire) e il Route Maintenance (mantenimento della medesima).

Il primo funziona nella seguente maniera (fig. 1.6 e fig. 1.7). Utilizzando un flooding ogni nodo che riceve una richiesta di scoperta di un percorso, effettuata grazie a pacchetti di RREQ (Route REQuest), può replicare in tre modi: rispondere alla richiesta con un pacchetto di RREP (Route REPLY) se ha nella propria cache la risposta adatta, scartare la richiesta se è già stata effettuata, aggiungere il proprio codice identificatore e trasmettere a sua volta ai proprio vicini l'interrogazione.

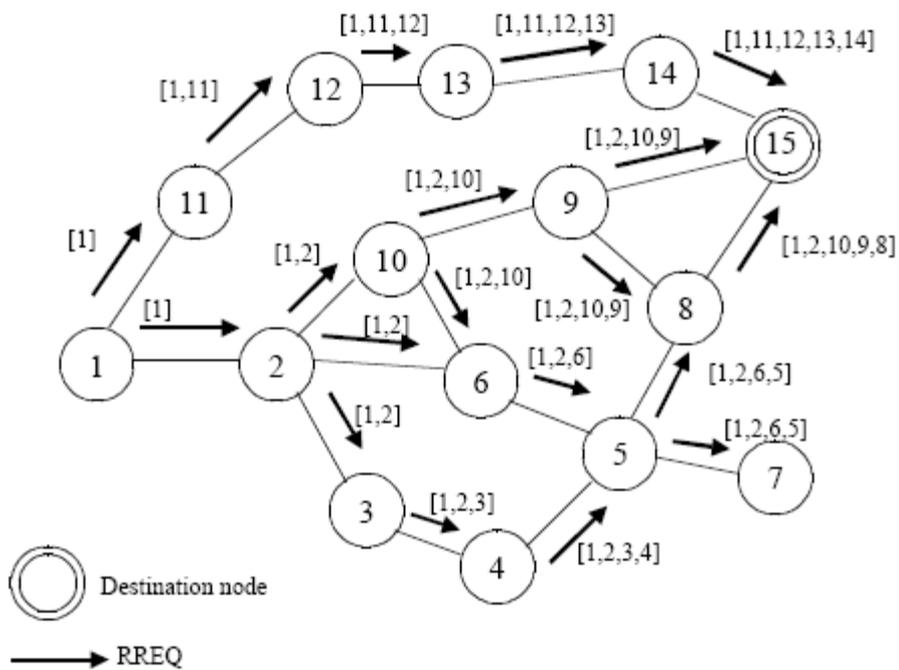


fig. 1.6 – Un esempio di propagazione RREQ del DSR

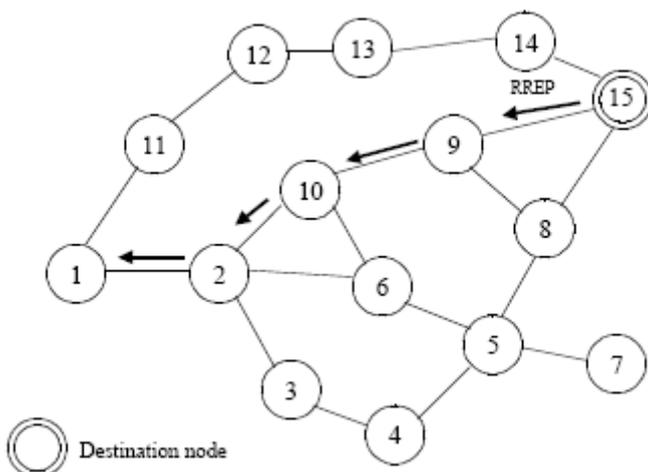


fig. 1.7 - Un esempio di propagazione RREP del DSR

Il secondo invece, rilevata la rottura di un collegamento, fa sì che l'ormai errata informazione sia rimossa dalla cache e la correzione sia propagata.

Per assicurare buone performances utilizzando questo protocollo è inoltre necessario un efficiente meccanismo di gestione delle caches, da utilizzare in maniera massiccia, per memorizzare gli interi percorsi per le varie destinazioni: un uso considerevole delle cache è infatti necessario per evitare ad ogni RREQ l'utilizzo di un dispendioso flooding della rete.

1.7.1.1.3 I Protocolli Ibridi

Dalla fusione delle due tipologie appena descritte nascono quindi i protocolli ibridi, che mirano a combinare i vantaggi dei protocolli proattivi e reattivi, limitando gli svantaggi che derivano rispettivamente dall'overhead introdotto per mantenere aggiornate le routing tables e dalla latenza causata dalla necessità di calcolare al momento un nuovo path prima di poter iniziare una trasmissione.

1.7.1.1.3.1 Lo Zone Routing Protocol (ZBR)

ZBR è un protocollo ibrido che mira a combinare i vantaggi dei protocolli proattivi e reattivi principalmente riducendo la latenza necessaria ad acquisire un nuovo percorso di instradamento da un lato e dall'altro limitando l'overhead introdotto dal protocollo.

ZBR si basa sul concetto di zona $Z(k,n)$ (fig. 1.8), definita come l'insieme di nodi la cui distanza dal nodo centrale n non deve essere superiore a k -hops. Il valore di k è normalmente piccolo se paragonato al diametro della rete e può essere ottimizzato nei vari scenari a seconda del grado di mobilità dei nodi e del traffico presente.

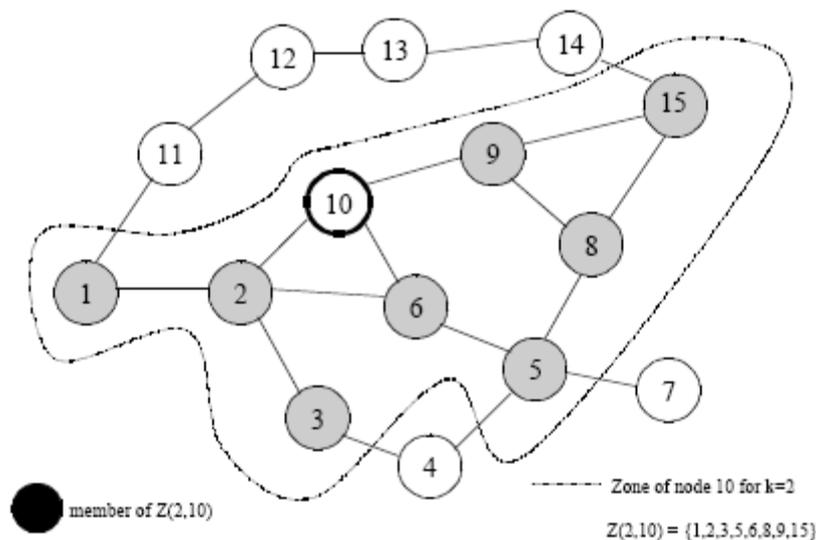


fig. 1.8 – Un esempio di zona ZBR

In pratica, lo ZRP implementa un protocollo proattivo per le comunicazioni con i nodi appartenenti alla sua zona, ed uno reattivo per le comunicazioni con tutti gli altri nodi.

Questo protocollo è organizzato in quattro componenti principali:

- *IntraZone Routing Protocol (IARP)*, il cui compito è fornire in modo proattivo il percorso verso i nodi appartenenti ad una zona.
- *Interzone Routing Protocol (IERP)*, che fornisce *on-demand* il calcolo dei percorsi verso i nodi fuori della zona.
- *Bordercast Protocol (BRP)*, un protocollo di flooding selettivo, che collabora con IARP e viene utilizzato da IERP per la ricerca del routing verso i nodi esterni alla zona.
- *Neighbor Discovery/Maintenance Protocol (NDP)*, utilizzato per rilevare i vicini di un nodo.

Diverse sono le modalità di calcolo del percorso che i pacchetti trasmessi dovranno seguire a seconda che il nodo di destinazione appartenga o meno alla zona del nodo sorgente.

Infatti nel caso la distanza espressa in hops sia minore o uguale a k , viene utilizzato un meccanismo di tipo proattivo table driven, ovvero ZBR si serve per il calcolo del path di tabelle di routing.

In caso contrario ZBR si serve di IERP, che usa a sua volta una forma di flooding selettivo per limitare l'overhead dato dall'invio di

pacchetti, per scoprire on-demand il percorso da seguire all'esterno della sua zona.

1.7.1.2 Routing Multicast

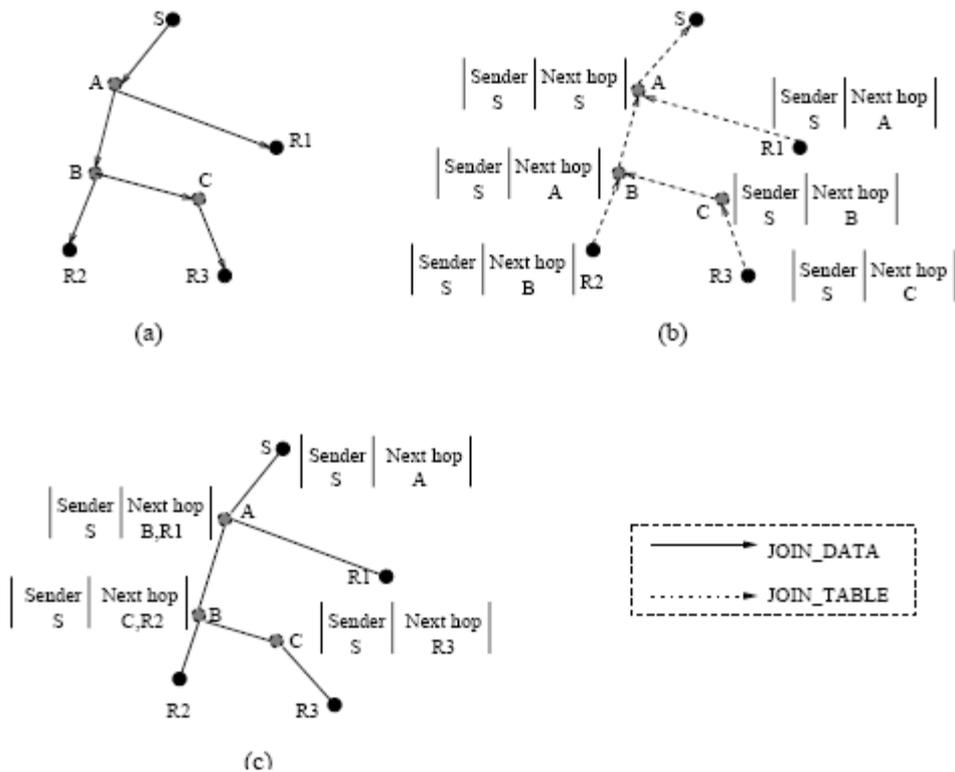
Per multicast si intende la trasmissione di pacchetti ad un gruppo di hosts identificati da un unico indirizzo di destinazione. Tipicamente l'appartenenza ad un gruppo è dinamica, il che significa che un host può unirsi o distaccarsi in qualunque momento. Un nodo può appartenere allo stesso tempo a più gruppi, ma non è necessario che faccia parte di un gruppo per potergli inviare pacchetti.

Nelle MANET ogni nodo ha un raggio di trasmissione limitato e quindi, per provvedere alla consegna di un messaggio, un pacchetto può dover attraversare molti nodi intermedi. I dispositivi comunemente componenti le MANET sono inoltre caratterizzati da scarsa banda di trasmissione, scarsa autonomia e alta mobilità. Rapide riconfigurazione dinamiche sono quindi necessarie: poiché durante tali fasi è possibile la formazione di loop temporanei, l'adattamento della rete deve avvenire nella maniera più semplice possibile per mantenere basso l'overhead introdotto nel canale.

1.7.1.2.1 L'On-Demand Multicast Routing Protocol (ODMRP)

ODMRP usa un concetto di gruppo di forwarding, dove solo un sottoinsieme dei nodi invia a sua volta in multicast i pacchetti ricevuti. Non è richiesto inoltre un controllo esplicito per abbandonare un gruppo.

Con ODMRP l'appartenenza ad un gruppo e i percorsi di instradamento multicast vengono stabiliti e aggiornati dalla sorgente in modalità on-demand.



**fig. 1.9 – Un esempio di ODMRP: (a) Propagazione di pacchetti JOIN_DATA
 (b) Propagazione di pacchetti JOIN_TABLE
 (c) Le multicast tables finali**

Utilizzando il presente algoritmo quando un nodo ha necessità di spedire un messaggio ad un gruppo senza conoscere il percorso per raggiungerlo, invia un pacchetto broadcast, chiamato JOIN_DATA (fig 1.9a), a tutta la rete. Quando un nodo intermedio riceve questo pacchetto controlla che non si tratti di un duplicato (in questo caso tale pacchetto viene scartato). In caso contrario aggiorna la propria tabella di routing, annotando l'ID del nodo da cui è stato ricevuto. In questo modo ogni vicino costruisce il cammino inverso per raggiungere la sorgente. Una volta completata questa operazione il nodo trasmette a sua volta (sempre in broadcast) il pacchetto, nel quale sono ora memorizzate tutte le coppie *<sorgente, nodo precedente>*, oltre ad un numero di sequenza che lo identifica. In questo modo, una volta iterato il procedimento fino a raggiungere il gruppo destinatario, quest'ultimo è in grado di costruire una JOIN_TABLE (1.9b), in cui sono memorizzate le informazioni necessarie per ottenere il percorso di instradamento. Questa tabella viene quindi spedita a tutti i vicini, che controllano se in essa è

presente il loro ID ed in questo caso attivano il campo FG_FLAG (forwarding group flag), comprendendo di far parte del cammino tra la sorgente ed il gruppo destinatario. Conclusa questa operazione il cammino è noto e la sorgente può così inviare il messaggio (1.9c).

CAPITOLO 2

IL FRAMEWORK AGAPE

AGAPE è un framework di gestione di gruppi con cognizione di contesto che consente all'utente di creare uno specifico gruppo, di unirsi ad esso, di lasciarlo e di ricevere le viste degli altri membri siti in prossimità.

L'uso dei gruppi può facilitare la condivisione e lo scambio di informazioni inerenti gli interessi specifici degli utenti e permettere di apprendere se tali utenti si trovano a breve distanza o comunque all'interno di un'area raggiungibile.

L'insieme dei membri che compongono un gruppo AGAPE non è determinato a priori, ma varia dinamicamente a causa degli arrivi e delle partenze delle diverse entità. La mobilità degli utenti e le disconnessioni temporanee dei dispositivi causano cambiamenti solo nelle viste di cui ciascun membro del gruppo è provvisto, ma non nell'appartenenza al gruppo.

Quando un utente entra, lascia o si disconnette da una località della rete, AGAPE riporta il cambiamento della vista del contesto a tutti i membri del gruppo siti in prossimità. Quando un membro di un gruppo si muove da una località ad un'altra, AGAPE ha il compito di fornire a tale nodo la vista della nuova località. Le viste hanno il principale vantaggio di essere rapidamente e direttamente disponibili anche in caso di partizione della rete. Se avviene una partizione della rete infatti (fig. 2.1), almeno i dispositivi appartenenti alla stessa partizione, che quindi si trovano ancora in prossimità e che fanno ancora parte della stessa località, possono continuare a collaborare.

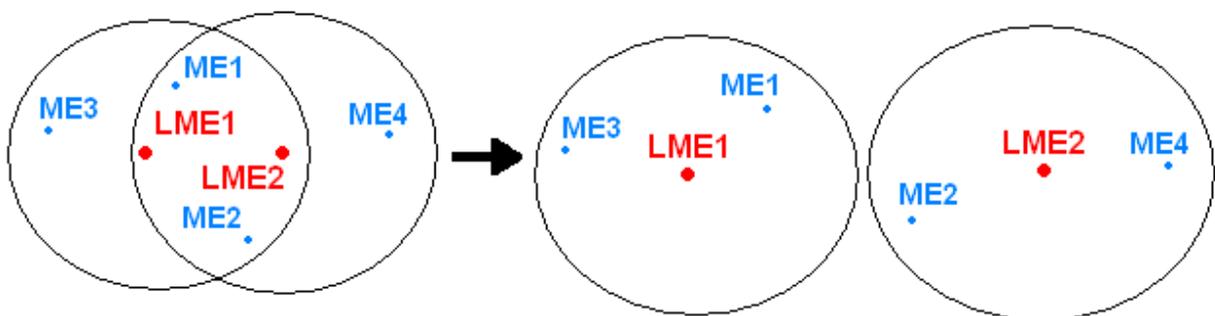


fig. 2.1 – Divisione di una rete in due parti: rappresentati due LME (e loro raggi di trasmissione) e 4 ME

D'altra parte invece una vista globale dei membri di un gruppo può essere fornita, quando necessario, attraverso l'unione di tutte le viste locali ottenute grazie alla coordinazione dei componenti del middleware AGAPE in ciascuna località. In ogni caso è d'obbligo sottolineare che il modello a gruppi di AGAPE restringe il bisogno di coordinazione globale alla consistenza dei membri e delle viste del gruppo. Frequenti merge della rete, disconnessioni da parte degli utenti e cambiamenti di posizione rendono inefficiente e non rilevante assicurare una globale e consistente vista di tutti i membri del gruppo.

Ciascun gruppo è caratterizzato da un unico identificatore di gruppo Group Identifier (GID) e da un profilo di gruppo che specifica interessi, preferenze e attività che dovrebbero essere condivisi da tutti i suoi membri. Ciascun membro di un gruppo ha un identificatore personale Personal Identifier (PID), che è (statisticamente) unico all'interno di un gruppo, e un profilo che esprime gli attributi e le caratteristiche dell'utente e del dispositivo da esso utilizzato, insieme alle preferenze di gruppo in termini di attività, fini e obiettivi.

2.1 Le Entità AGAPE e il Concetto di Località

Come mostra la fig. 2.2, il modello a gruppi di AGAPE riconosce due ruoli per le entità: l'unità gestita (Managed Entity, ME) e l'unità gestore della località (Locality Manager Entity, LME).

Gli ME sono membri di un gruppo che utilizzano i servizi di supporto di AGAPE per collaborare.

Gli LME sono membri di un gruppo che non solo collaborano, ma forniscono anche il supporto per la gestione delle operazioni del gruppo a favore degli ME: essi forniscono infatti la possibilità di creazione a tempo di esecuzione di un nuovo gruppo, permettono alle entità di unirsi ad un gruppo e di mantenere una lista aggiornata dei membri siti in prossimità. In ogni momento gli LME e gli ME possono unirsi o lasciare i gruppi di loro interesse. In ogni caso in un singolo istante di tempo gli ME possono appartenere ad al massimo un singolo gruppo, ma in differenti momenti è loro permesso di mutare la loro appartenenza in favore di altri gruppi.

Questo approccio permette di limitare il consumo di risorse ed in particolare l'uso delle memorie, fattore importante per dispositivi tipicamente forniti di risorse limitate.

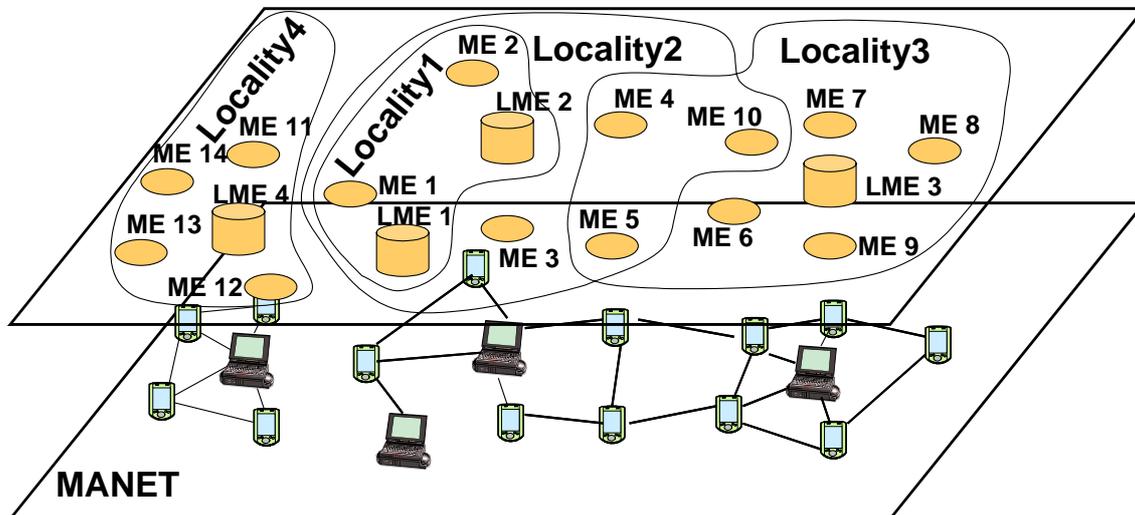


fig. 2.2 – Il modello AGAPE (h=2 hops)

I gruppi AGAPE sono suddivisi in diverse località. Ciascun LME supporta la gestione delle operazioni solo per i membri del gruppo che si trovano nella sua località. Una località è definita come l'insieme delle entità AGAPE i cui dispositivi sono connessi ad un LME da un percorso di instradamento di lunghezza pari al massimo ad h hops. Il valore h esprime il valore massimo del raggio della località misurato in hops della rete. Tale valore h è scelto sulla base dello scenario applicativo.

La fig. 2.2 mostra diverse località, dove ciascuna contiene LME e ME. Come mostrato da tale figura, diverse località possono anche sovrapporsi a causa della prossimità fisica degli LME. Gli ME possono muoversi liberamente e possono appartenere a più di una località in ogni momento. In aggiunta, può accadere che due LME definiscano la stessa località, per esempio nel caso essi siano siti a distanza 1 hop.

2.2 L'Architettura di AGAPE

I servizi AGAPE per la gestione dell'appartenenza a gruppi sono organizzati in due layer logici costruiti sulla Java Virtual Machine (fig. 2.3).

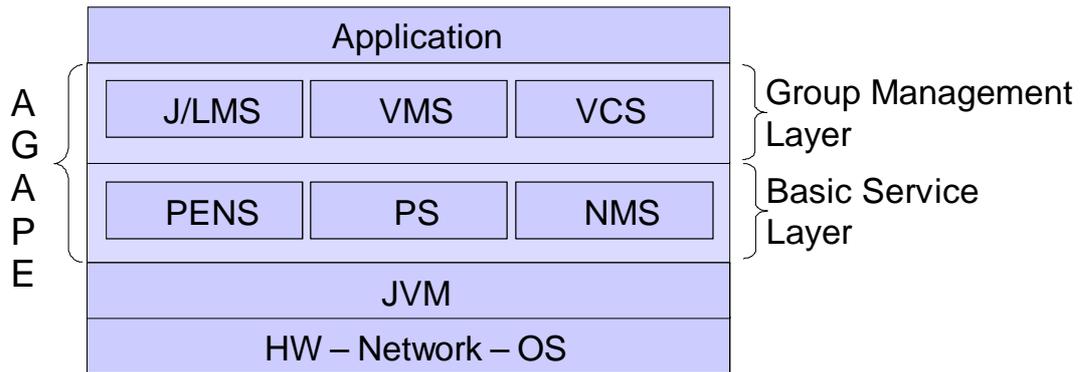


fig 2.3 – L’architettura di AGAPE

2.2.1 Il Basic Service Layer

Il Basic Service Layer include funzionalità di basso livello per permettere ai dispositivi di accedere alla MANET, di supportare i servizi di naming, di discovery dei gruppi e dei membri dei gruppi e di monitorare lo status di disponibilità di un membro.

In particolare il *Network Manager Service (NMS)* provvede a fornire il necessario supporto ai membri di un gruppo per scambiare messaggi attraverso la MANET. L’NMS implementa gli schemi di comunicazione sia punto-a-punto sia multipunto. Il supporto punto-a-punto dell’NMS permette di mandare un messaggio ad un host identificato dal suo IP. Il supporto multipunto permette l’invio broadcast dello stesso messaggio a più entità AGAPE diverse. Esso estende inoltre il protocollo di flooding gossip per limitare la disseminazione di messaggi ad un definito numero di hops introducendo un meccanismo basato sul TTL (Time To Live). In particolare, l’NMS marca i messaggi con un campo TTL che viene decrementato ad ogni hop. Solo i messaggi con TTL maggiore di zero possono essere nuovamente inviati in broadcast. È da notare che la natura probabilistica della disseminazione di dati basata sul gossip fornisce un approccio best effort alla consegna del messaggio a tutti i vicini. L’NMS provvede anche all’accodamento locale dei pacchetti in ingresso e in uscita che vengono gestiti con una politica FIFO (First In First Out).

Il *Proximity Service (PS)* permette ai membri di un gruppo AGAPE, siano essi LME o ME, di segnalare la propria disponibilità all’interno di una località. Ad intervalli regolari il servizio richiede

che l’NMS invii in gossip un messaggio di beacon contenente il GID, il PID e l’indirizzo IP dell’entità. L’intervallo di tempo tra due beacon può essere impostato a seconda dei requisiti specifici dell’applicazione e della mobilità dei nodi. Scendendo nel dettaglio, la frequenza di trasmissione dei beacon dovrebbe essere incrementata di pari passo all’aumento della mobilità dei nodi. Da notare è che il PS si appoggia all’NMS per la trasmissione dei beacon. C’è soltanto una garanzia probabilistica che tutti i membri di un gruppo all’interno di una località ricevano i beacon propagati. La probabilità di ricezione dipende dalla topologia della rete e dalla probabilità di ritrasmissione di un messaggio del protocollo probabilistico di flooding dell’NMS.

Il *Proximity Enabled Naming Service (PENS)* genera in maniera casuale, statisticamente uniforme, identificatori unici di gruppo (GID) e identificatori personali di entità (PID) sfruttando un approccio di naming simile a quello utilizzato per gli ambienti P2P. Il PENS riceve inoltre beacon provenienti da LME e ME siti in prossimità e costruisce una tabella degli LME/ME attualmente disponibili. Ogni entry di una tabella contiene il GID/PID dell’entità AGAPE associata, il ruolo che essa gioca (LME o ME) e il suo indirizzo IP. Inoltre ciascuna entry è associata ad un timestamp. Se il PENS non riceve beacon da un’entità membro entro una certa soglia di tempo, l’entry associata viene rimossa e il dispositivo corrispondente considerato disconnesso.

2.2.2 Il Group Management Layer

Il Group Management Layer fornisce i servizi necessari alla scoperta, alla creazione ed alla distruzione di gruppi.

In particolare il *Join/Leave Manager Service (J/LMS)* permette alle entità AGAPE (LME e ME) di raggiungere, di scoprire, di unirsi ad un gruppo e di staccarsi da esso. Gli LME sfruttano il J/LMS per promuovere dinamicamente la formazione di nuovi gruppi. La creazione di un nuovo gruppo richiede la specifica del profilo del gruppo a livello applicativo. Il J/LMS, deve coordinarsi col PENS per ottenere i GID e i PID dei nodi, col PS per segnalare la disponibilità di un nuovo gruppo e col VMS per creare una nuova

vista di gruppo. Sia gli LME sia gli ME possono utilizzare il J/LMS per scoprire e unirsi ad un nuovo gruppo di interesse. La fase di discovery di un gruppo richiede che il J/LMS si coordini col PENS per ottenere la lista degli LME disponibili del gruppo. Le entità AGAPE non ancora membri del gruppo utilizzano il J/LMS per cercare di unirsi ai gruppi di interesse disponibili. Per unirsi ad un nuovo gruppo l'entità deve specificare il profilo del gruppo desiderato a livello applicativo. Se il profilo specificato combacia col profilo del gruppo disponibile, l'entità è ammessa nel gruppo. Ogni volta che un nuovo membro si unisce al gruppo il J/LMS ritorna il GID/PID (coordinandosi col PENS), insieme al profilo completo del gruppo. Il J/LMS consente inoltre ai membri, siano essi LME o ME, di lasciare un gruppo al quale si erano precedentemente uniti specificando il GID di tale gruppo. Il J/LMS si coordina con gli altri servizi AGAPE (e.g. PS e VMS) per bloccare l'invio da parte del PS dei beacon relativi all'entità distaccata, e di delegare al VMS la cancellazione di qualsiasi informazione relativa a tale entità (e. g. GID/PID, profili utente / dispositivo).

Il *View Manager Service (VMS)* permette agli LME di creare e di diffondere le viste ai membri del gruppo AGAPE ad intervalli regolari. Ciascun membro del gruppo riceve una vista che contiene la lista dei soli membri del gruppo siti nel raggio di visibilità LME responsabile della loro località. In particolare ciascuna vista, dipendente dal contesto, include il GID/PID e il livello di batteria residua del dispositivo che l'ha generata ed è composta dalla lista delle entry che associano ciascun membro del gruppo (ottenuto dal PENS) con l'utente e il profilo del dispositivo e del gruppo (ottenuti dal J/LMS). Quando i membri di un gruppo si connettono o si disconnettono dalla rete o quando cambiano dispositivo di accesso e/o profilo di gruppo, AGAPE riporta i cambiamenti della vista a tutti i membri del gruppo nella località sfruttando il servizio di broadcast implementato dall'NMS. Il VMS si coordina col PENS per ottenere la notifica degli eventi, come per esempio gli arrivi, le dipartite e le disconnessioni di un'entità membro di un gruppo. In particolare questi eventi causano l'aggiornamento da parte del VMS

della vista del gruppo con l'inserimento o la rimozione della entry dell'entità. Inoltre le viste possono cambiare a causa di variazioni dell'utente o del dispositivo utilizzato. In ogni caso questi cambiamenti avvengono tipicamente di rado. Quando un'entità, sia essa LME o ME, muta il suo profilo informativo, delega al VMS l'aggiornamento della vista dipendente dal contesto. In particolare l'entità fornisce il VMS del nuovo profilo da includere nel nuovo aggiornamento della vista. Bisogna notare che il VMS garantisce una consistenza della vista del gruppo di tipo probabilistico. Ciò è dovuto al VMS, che si basa su protocolli probabilistici di gossip per spargere i beacon per testare la disponibilità di membri del gruppo siti in prossimità e per costruire le viste del gruppo.

Il *View Controlling Service (VCS)* consente agli LME di decidere quando distribuire o meno le proprie viste. Il VCS riduce in maniera rilevante la propagazione delle viste non necessarie grazie ad un algoritmo che, coordinando ogni LME con gli altri LME all'interno della stessa località, permette di determinare se tale LME è necessario o meno al fine di ottenere una buona copertura dell'area. Il VCS cerca quindi di assicurare i servizi di coordinazione di gruppo necessari in una MANET senza però perdere di vista la necessità di risparmiare risorse, data l'esigua disponibilità di queste ultime che tipicamente caratterizza i dispositivi mobili.



fig. 2.4 – Coordinazione VMS - VCS

Prima di propagare le viste ciascun VMS deve infatti chiedere l'autorizzazione all'istanza locale del VCS (fig. 2.4), fornendo ad

esso le viste attualmente in possesso del dispositivo. Sulla base di queste ultime il VCS determina grazie ad un particolare algoritmo la rilevanza o meno della propagazione delle viste dell'LME in questione.

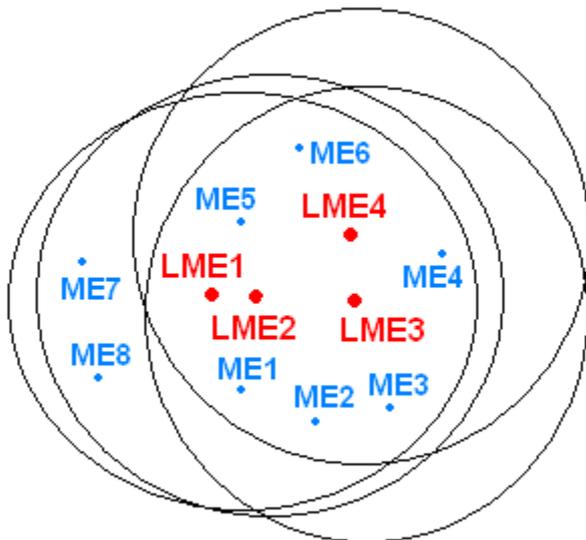


fig. 2.5 – Area ad alta densità di LME: rappresentati 4 LME (e loro raggio di trasmissione) e 8 ME

In un'area ad alta densità di LME per esempio (fig. 2.5), è inefficace ai fini della possibilità di accesso ai servizi del gruppo che tutti gli LME propagano le loro viste. Sotto l'aspetto del risparmio di risorse invece ciò è addirittura controproducente, visto che ripetizioni inutili delle medesime informazioni non possono fare altro che sprecare banda e batteria a disposizione. Grazie al VCS è invece possibile ottenere ottimi risultati in termini di efficacia nel raggiungimento dei vari nodi tenendo però presente al contempo la razionalità nell'uso delle risorse presenti.

CAPITOLO 3

IL VIEW CONTROLLING SERVICE

All'interno del package AGAPE si colloca il View Controlling Service (VCS) che, come mostrato nel capitolo precedente, si coordina col VMS. Il VCS riceve in input dal VMS le viste della località dell'LME e produce in output al VMS l'autorizzazione o meno alla diffusione della propria vista.

Questo componente è essenziale per ottimizzare l'uso delle risorse a disposizione e gioca un ruolo chiave nel mantenimento dell'autonomia della rete. Se tutti i VMS propagassero senza alcun controllo le proprie viste si potrebbe infatti assistere ad una diffusione di informazioni duplicate. Un simile comportamento produrrebbe uno spreco delle risorse, già di per sé limitate nei dispositivi portatili tipicamente utilizzati al giorno d'oggi.

Tale decisione relativa alla necessità o meno di trasmissione viene presa essenzialmente considerando le viste degli LME co-locati: come verrà in seguito illustrato nei dettagli, vengono prese in considerazione le aree di copertura dei singoli LME e viene effettuata una stima di quali potrebbero essere quelli più adatti a trasmettere per massimizzare il rapporto dispendio di risorse / area di copertura raggiunta. La funzione del VCS è infatti la ricerca della minimizzazione dell'uso delle risorse a disposizione dei dispositivi, mantenendo comunque sostanzialmente invariate le possibilità di comunicazione tra i nodi.

È qui necessario precisare però come per area di copertura di un LME non si intenda esattamente la regione all'interno della quale un LME riesce a inviare o a ricevere trasmissioni, ma un'astrazione di tale concetto applicato al problema affrontato. All'interno di tale regione fisica contano infatti, ai fini del lavoro, solo i nodi presenti e non, come nell'uso comune, la reale completa sovrapposizione delle aree. Possiamo infatti definire copertura di un dato LME l'insieme dei nodi raggiunti da tale entità in maniera diretta. In questo contesto i due LME in fig. 3.1, per esempio, hanno la stessa copertura.

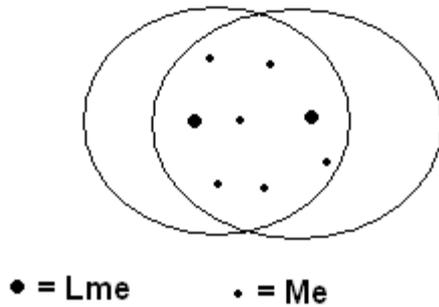


fig. 3.1 – Aree di copertura ritenute equivalenti

Per determinare quale valore fornire in output il VCS incorpora un algoritmo che ricerca il minor dispendio di risorse possibile lasciando però allo stesso tempo invariata o quasi la raggiungibilità dei vari nodi. L'insieme delle istanze della classe VCS presente nei vari LME di una località cerca quindi, in maniera distribuita, di determinare nella maniera più accurata possibile quale può essere la migliore (in termini di rapporto copertura / risparmio raggiunti) configurazione di LME trasmettenti. Tali istanze forniranno poi l'autorizzazione alla propagazione delle viste solo a quell'insieme di LME ritenuto necessario per assicurare la comunicazione.

Da precisare come per configurazione degli LME si intenda come la rete reagisce alla presenza di una certa topologia, ovvero quali LME decidono di trasmettere, e di conseguenza quale copertura viene raggiunta, data una certa disposizione dei nodi nello spazio. Una riconfigurazione è d'altra parte un adattamento di una configurazione causato da una variazione della topologia stessa.

A causa della scarsa quantità di informazioni a disposizione delle singole istanze di VCS (ogni VCS ha infatti a disposizione le sole viste, ricevute dal VMS, della propria località, tipicamente di dimensione ben minore rispetto alla rete) non è possibile in molti casi raggiungere la configurazione migliore, cioè quella che si potrebbe ottenere operando con a disposizione la completa conoscenza della rete. Con tale cognizione sarebbe infatti possibile determinare quali sono gli LME più adatti a trasmettere per raggiungere i risultati desiderati, ma tale conoscenza non è data e ci si deve quindi accontentare di una configurazione che si avvicini alla conformazione ideale.

Nel proseguimento della tesi si intenda dunque come soluzione ottima la configurazione ideale, verso la quale i VCS tendono a portare la rete, mentre per soluzione sub-ottima quella realmente computabile data l'esiguità delle informazioni a disposizione delle singole istanze di VCS.

3.1 Il Problema nello Scenario Specifico Affrontato

Al fine di evitare lo spreco di risorse preziose è essenziale minimizzare il numero delle trasmissioni senza per questo perdere la possibilità di raggiungere tutti o quasi i nodi presenti nella rete: è necessario quindi che solo gli LME necessari propaghino le loro viste. In caso contrario si assisterebbe infatti ad un elevato numero di trasmissioni / ricezioni inutili, con conseguente spreco di risorse.

Un primo approccio di soluzione al problema potrebbe portare a pensare ad una diffusione su ogni singolo nodo di tutte le informazioni necessarie a descrivere la topologia della rete. Una volta che ogni nodo avesse ricevuto tali dati potrebbe, con un opportuno algoritmo, determinare quali LME debbano propagare le viste e quindi decidere sulla utilità o meno della propria trasmissione.

L'assunzione di onniscienza delle diverse entità del sistema non è però accettabile in scenari ad-hoc perché richiederebbe la soluzione di un problema di agreement distribuito: notoriamente problemi di questo tipo non hanno soluzione in tempo finito in reti asincrone inaffidabili con process failure.

Qualora fosse anche possibile la soluzione a tale problema, sarebbe comunque necessaria una dispendiosa fase di diffusione tramite flooding delle informazioni indispensabili per descrivere la rete. Se pensiamo ad una rete di notevoli dimensioni tale fase potrebbe già di per sé introdurre un notevole overhead, anche se il problema maggiore di un approccio di questo tipo è però la mancanza di reattività rispetto alla velocità, potenzialmente molto elevata, con cui possono variare le condizioni della rete in scenari dinamici. Un approccio di questo genere non consentirebbe infatti di reagire in maniera efficace alle modifiche che potrebbero sopraggiungere: una minima variazione nella topologia, quale l'arrivo di un nuovo nodo

o la scomparsa di uno prima presente, comporterebbe l'intera riconfigurazione della stessa. In uno scenario dove i nodi sono in continuo movimento una soluzione di questo tipo in pratica non porta ad alcun risparmio ed è quindi inutile per un efficiente sfruttamento delle risorse.

L'approccio al problema che è stato adottato rinuncia invece alla sicurezza della determinazione della soluzione ottima, accontentandosi di un approccio best effort, guadagnando però d'altro canto una notevole rapidità decisionale. Ogni nodo deve infatti limitarsi alla conoscenza della topologia della propria località e deve quindi cercare di determinare, in base a informazioni che sono quindi inevitabilmente parziali, se la propria trasmissione delle viste è utile o meno per fornire i servizi AGAPE di gestione del gruppo.

L'algoritmo che si occupa di prendere tale decisione deve quindi tenere conto essenzialmente di tre ostacoli: la scarsità dell'autonomia dei dispositivi, la limitatezza della banda e la parzialità delle informazioni a disposizione, oltre che della necessità di funzionare in ambiente distribuito. Le informazioni che ogni dispositivo possiede sono inevitabilmente parziali, in quanto ogni nodo ha conoscenza soltanto della propria località, ma nulla sa della topologia globale della rete.

Il VCS ha quindi il compito di decidere se la trasmissione di ciascun LME è ritenuta utile per assicurare la massima copertura in rapporto al massimo risparmio di risorse, ovvero garantire che tutte le entità in una località vengano incluse nelle viste.

3.1.1 Linee Guida del Problema

Da un punto di vista funzionale il problema affrontato consiste nel consentire la comunicazione al maggior numero possibile di nodi presenti in una certa area, tenendo però conto di alcune problematiche fondamentali.

Innanzitutto la rete deve autoconfigurarsi, ovvero ogni istanza di VCS deve essere in grado, autonomamente e localmente al dispositivo, di prendere una decisione, ossia se autorizzare o meno il proprio VMS alla diffusione della propria vista.

L'algoritmo progettato deve minimizzare il numero di riconfigurazioni necessarie. In altre parole la struttura della rete deve essere il più stabile possibile, ovvero capace di evitare continui adattamenti e, nel caso essi siano necessari, di far sì che coinvolgano il minor numero possibile di nodi. In altre parole è un obiettivo ottenere una rete che reagisca rapidamente ai mutamenti quando questi siano di una certa entità, ma che al contempo non disperda forze inutilmente per adattarsi a variazioni trascurabili della topologia.

La robustezza della rete è quindi fondamentale per far sì che, nonostante la mobilità dei nodi, sia assicurata per quanto possibile la continuità della comunicazione, in maniera trasparente all'utente: questo significa che la rete deve cercare una copertura il più ampia possibile ed eventualmente prendere veloci contromisure agli spostamenti dei nodi.

Vista la necessità di ottimizzare l'uso delle risorse ed in particolare della banda e della batteria dei dispositivi, l'area di copertura ottenuta va valutata in relazione al dispendio di risorse necessario per conseguirla: si ricerca quindi il miglior rapporto spesa effettuata-copertura ottenuta. Da ciò deriva la volontà di rinunciare alla soluzione ottima pur di ottenere un più efficiente utilizzo delle risorse. Non viene quindi per esempio valutata una buona scelta l'introduzione di molte propagazioni di viste duplicate per raggiungere un solo nodo (o comunque un numero di nodi percentualmente trascurabile) in più, lontano dagli altri (fig. 3.2).

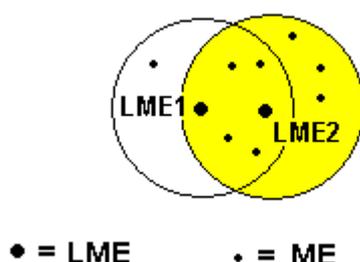


fig. 3.2 – Due LME e alcuni ME: l'area di copertura è rappresentata colorata (solo LME2 propaga la propria vista)

3.2 Requisiti dell'Algoritmo

Come precedentemente sottolineato, l'algoritmo massimizza il rapporto tra area coperta e numero di LME trasmettenti necessari per ottenere tale risultato.

Uno dei fini dell'algoritmo è conseguire la maggiore stabilità possibile: in particolare riguardo vengono quindi tenuti gli LME che hanno batteria residua scarsa. Si cerca dunque una copertura efficace senza considerare l'eventualità di una trasmissione di tali LME. In questo modo si evita che questi ultimi cessino di trasmettere, costringendo la rete ad una riconfigurazione.

Vengono poi evitate le trasmissioni di più LME che coprono la stessa area: uno solo dei due è infatti necessario, il secondo introduce solo ridondanze senza aggiungere nulla alla copertura.

Se vi sono LME le cui viste contengono totalmente le viste di altri, l'algoritmo fa in modo che l'LME contenitore trasmetta e l'LME contenuto si accorga della inutilità di un suo invio di messaggi.

Nel caso sia possibile, vengono vagliati possibili avvicendamenti nella trasmissione in maniera tale che, pur restando invariata la copertura, si minimizzi il numero di LME attivi.

Importante è inoltre cercare di rinunciare a quei nodi troppo lontani, il cui raggiungimento peggiorerebbe l'uso efficiente delle risorse, ovvero a quei nodi che costringerebbero a trasmissioni ridondanti migliorando di poco la copertura: meglio è, come da scelta progettuale, abbandonare tali nodi, seguendo il principio secondo il quale chi più lontano è, meno interessato dovrebbe essere alle informazioni inviate.

I problemi evidenziati guidano perciò lo sviluppo di una euristica che consenta di determinare la configurazione che fornisca il miglior rapporto area coperta / numero di LME trasmettenti, considerando comunque una soglia di copertura minima ritenuta necessaria.

È indispensabile innanzitutto preservare per il maggior tempo possibile l'autonomia dei dispositivi: ciò è infatti fondamentale per consentire agli utenti il maggior tempo possibile di interazione con la rete.

Occorre inoltre risparmiare la banda a disposizione: da un'efficiente utilizzo di quest'ultima dipende infatti la velocità di trasmissione della rete, condizione indispensabile per qualsiasi applicazione.

È necessario poi tenere particolarmente conto dei dispositivi coi livelli più bassi di batteria residua: è quindi essenziale sfruttare maggiormente le risorse dei nodi più ricchi per garantire la massima stabilità alla rete. Basandosi sui nodi con autonomia minore si comprometterebbe infatti la durata della rete nel tempo: quando un dispositivo esaurisce l'alimentazione e quindi si disconnette, la variazione della topologia della rete può portare, oltre alla perdita di un nodo, anche alla necessità di una riconfigurazione.

Bisogna quindi mantenere il più a lungo possibile la rete stabile per minimizzare il numero di riconfigurazioni necessarie: un riassetamento provoca infatti invii e ricezioni di molti messaggi con conseguente dispendio di risorse.

Una piccola variazione dello scenario non deve assolutamente modificare tutto l'insieme degli LME che stanno trasmettendo: solo se strettamente necessario si deve ricorrere a modifiche nella configurazione e in ogni caso queste ultime non devono ripercuotersi sull'intera rete.

È inoltre opportuno evitare aree con inutili sovrapposizioni delle medesime trasmissioni: dove la densità di LME è elevata è necessario che pochi (possibilmente anche uno solo) LME trasmettano. Ne consegue la volontà di ricercare la più alta copertura possibile utilizzando a questo scopo il minor numero di LME trasmettenti possibile: se ad esempio la vista di un LME è contenuta in quella di un altro, tale LME deve evitare l'invio di messaggi ridondanti per la copertura e controproducenti per il consumo delle risorse.

Un altro obiettivo è fare sì che, per lo meno all'interno di una località, gli LME riescano a prendere decisioni coordinate: a tale scopo ogni LME deve valutare se trasmettendo può realizzare la copertura di due o più altri LME, ovvero tentare di avvicendare nella trasmissione altri LME mantenendo comunque invariata la copertura risultante. Infatti se tale avvicendamento risulta possibile le trasmissioni possono come minimo essere dimezzate.

Avendo scelto la volontà di lavorare a *best effort* non è possibile avere come meta la soluzione migliore in assoluto, i costi per ottenerla sono troppo alti, mentre lo scopo è al contrario un efficiente sfruttamento delle risorse, anche a costo di discostarsi leggermente dal risultato ottimo. Conseguenza di questa scelta è per esempio la rinuncia alla copertura di alcuni nodi troppo lontani e il cui raggiungimento comporta uno sforzo troppo ingente in rapporto all'impatto in positivo sulla copertura.

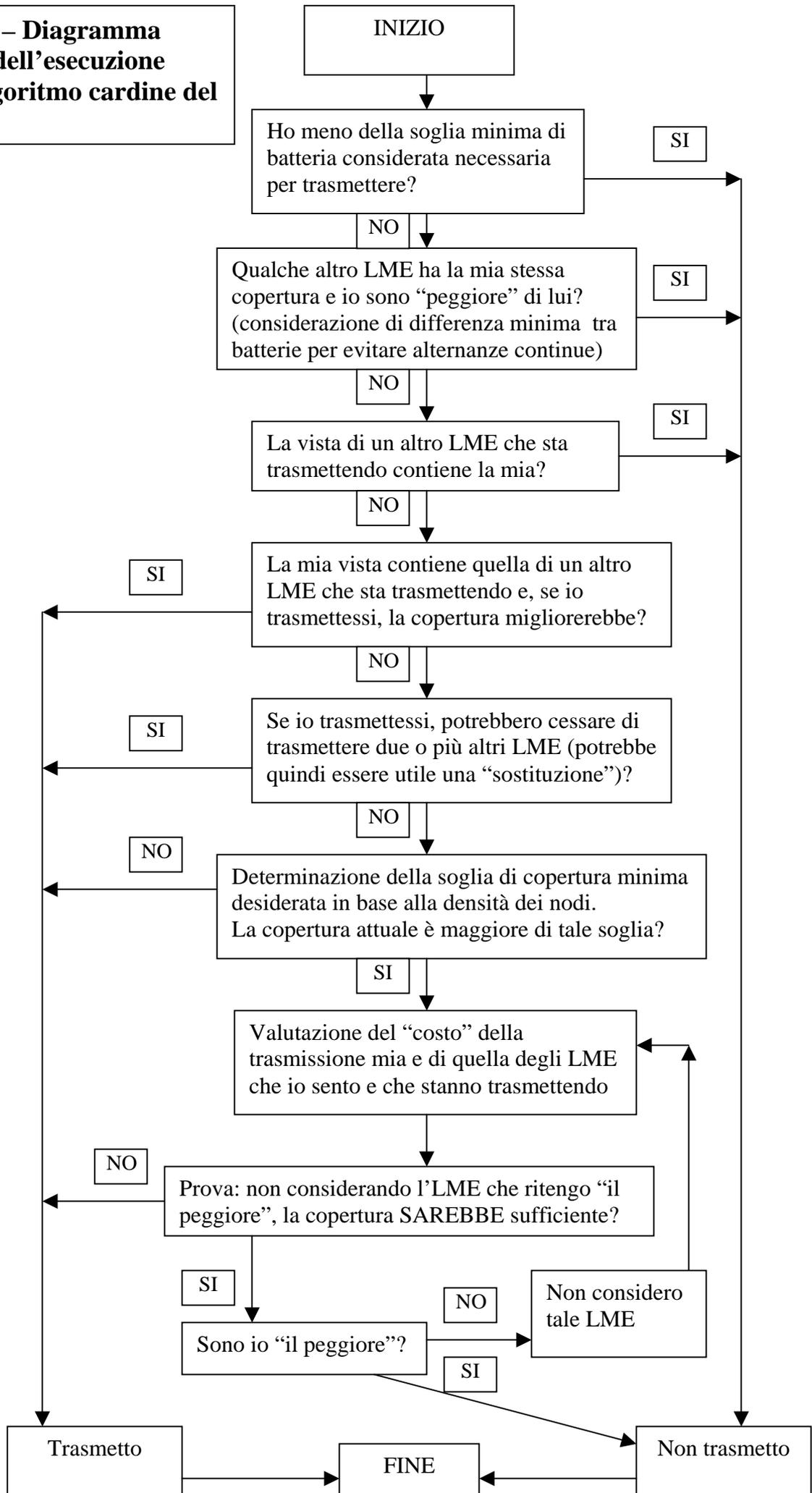
Quanto è stato fin qui esposto va considerato in funzione della dimensione della rete (o meglio della densità locale: un LME non ha infatti conoscenze globali). In una rete di dimensioni limitate una riconfigurazione è infatti ben meno problematica che in una rete vasta: modifiche allo status vanno quindi tanto più scongiurate quanto più aumenta il raggio della MANET.

Sempre in funzione della rete è inoltre necessario assegnare differenti priorità alle risorse da risparmiare: in caso di dimensioni limitate essenziale è infatti la copertura globale raggiunta, mentre in una rete più vasta, supponendo una copertura comunque sufficiente, da ottimizzare è in particolar modo la banda disponibile.

3.2.1 L'Algoritmo Cardine del VCS

L'algoritmo che deriva da tutte le riflessioni sopra esposte segue quindi il percorso logico descritto nel seguente diagramma (fig. 3.3):

fig. 3.3 – Diagramma logico dell'esecuzione dell'algoritmo cardine del VCS



Come evidenziato dal diagramma in fig. 3.3, ogni nodo valuta innanzitutto la propria batteria residua: se tale valore è inferiore ad una certa soglia, decidere di trasmettere porterebbe ad un rapido esaurimento delle stesse e, una volta spentosi il nodo, alla necessità di riconfigurazione della rete: volendo evitare ciò, in questo caso si ritiene opportuno non trasmettere. Importante è infatti mantenere la rete il più stabile possibile.

Procedendo nella valutazione si considera l'area di trasmissione degli LME: se la copertura di due o più nodi è equivalente, solo uno di questi nodi deve trasmettere: gli altri introdurrebbero solo ridondanze. Tra questi nodi viene scelto il "migliore": in questa analisi si tiene conto della batteria residua di ciascun LME, chi ne ha di più trasmette. Se fosse considerato però solo questo fattore si potrebbe però avere uno spiacevole effetto collaterale: infatti se due LME avessero la stessa batteria residua o comunque se ci fosse poca differenza tra le due, chi trasmettesse consumerebbe più del secondo, portando il proprio livello residuo sotto quello dell'altro. Dopo poco si assisterebbe quindi ad uno scambio delle parti, che diverrebbe in seguito continuo. Questa alternanza con continua necessità di riconfigurazione è un fattore negativo e quindi da evitare: viene quindi introdotta una soglia minima di differenza tra le batterie residue prima di effettuare un avvicendamento: in questo modo tali avvicendamenti (e quindi l'instabilità dovuta a riconfigurazioni) vengono drasticamente ridotti.

Sempre nel tentativo di ridurre le sovrapposizioni di aree di trasmissione, con conseguenti controproducenti diffusioni di dati, viene valutato se la vista di un LME è interamente contenuta in quella di un altro LME che sta trasmettendo (cioè se l'insieme dei nodi di cui è percepita la presenza è un sottoinsieme). In questo caso il primo dei due deve evitare di trasmettere: verrebbero infatti introdotte soltanto ridondanze senza nulla aggiungere alla copertura raggiunta.

Nel caso la vista di un LME contenga strettamente quella di un altro LME, allora il primo dei due deve iniziare a trasmettere: il secondo capirà poi di dover terminare la trasmissione. Un accorgimento è però necessario: bisogna controllare che con questa sostituzione la

copertura aumenti. Ciò è importante dato che in caso contrario la vista più ampia introdurrebbe soltanto controproducenti trasmissioni duplicate.

Se invece la vista di un LME contiene quelle di almeno altri 2 LME, allora è corretto che il primo inizi la trasmissione anche se la copertura non aumenta: infatti in tale modo si sostituisce un LME ad almeno altri due risparmiando quindi almeno una fonte di trasmissioni non necessaria.

Per determinare la soluzione non ci si può però soltanto basare su casi notevoli, ma risulta necessario anche valutare le situazioni nello specifico, scendendo in alcuni casi a compromessi. Come da scelta progettuale infatti si ritiene importante evitare sforzi elevati per raggiungere nodi troppo lontani che migliorerebbero di poco la copertura effettuata. La volontà è infatti quella di lavorare a *best effort*: se per introdurre piccole migliorie nella copertura le risorse rischiano un peggiore utilizzo, si è disposti a rinunciare a tale esiguo numero di ME pur di aumentare l'efficienza della rete.

Per questo motivo è stata quindi introdotta una soglia percentuale limite di copertura richiesta: se la copertura attuale è superiore a tale soglia, allora si valuta se rinunciare alla trasmissione, perdendo eventualmente anche una piccola percentuale di area coperta, oppure no.

Nel caso la soglia minima non sia ancora stata varcata viene quindi valutato qual è l'LME "peggiore" tra quelli nella propria area di copertura. In tale funzione di valutazione del "costo" si tiene conto di diversi elementi:

- quanti nodi vengono coperti in maniera "esclusiva" da ciascun LME (ossia quanti nodi si perderebbero se tale LME cessasse le trasmissioni): viene quindi assegnato un bonus per ogni nodo esclusivo
- quanti nodi vengono coperti inutilmente (ovvero quanti nodi ricevono trasmissioni ridondanti consumando quindi risorse per nulla): viene quindi assegnato un malus per ogni nodo coperto in maniera ridondante proporzionale al numero di intersezioni con aree di altri LME trasmettenti

- per ciascuno dei nodi di cui ai due punti precedenti viene attribuito un peso diverso a seconda della batteria residua
- se l'LME stava già trasmettendo viene assegnato un bonus: la stabilità della rete è infatti un fattore positivo e si è disposti anche ad accettare una configurazione leggermente peggiore pur di ottenerla.

Tale bonus varia in funzione della densità di nodi presenti nell'area: viene infatti seguito il principio secondo il quale una riconfigurazione dello status quo è tanto più da evitare quanto maggiore è il numero dei nodi. In altre parole se la densità è bassa, una variazione di scenario (e quindi un avvicendamento nella trasmissione tra LME) non comporta grosso dispendio di risorse, è accettata senza particolari accorgimenti e quindi tale bonus di stabilità è minore; al contrario, tanto maggiore è la densità, tanto più una riconfigurazione può creare spreco di risorse: finché lo status è accettabile non si procede quindi ad avvicendamenti e il bonus di stabilità assegnato è perciò maggiore.

Una volta effettuata tale calcolo viene quindi valutata la nuova situazione, nell'ipotesi che l'LME più indicato a non trasmettere effettivamente cessi di inviare la propria vista: se ciò avvenisse a quanto scenderebbe l'area di copertura? Se sotto questa condizione si è oltrepassata la soglia minima di copertura richiesta allora la trasmissione viene ritenuta essenziale.

Se al contrario la soglia minima di cui sopra non è stata ancora raggiunta, allora si procede non considerando più tale LME e aggiornando la percentuale di copertura attuale.

Se l'LME la cui trasmissione non è considerata essenziale è quello sul quale è in esecuzione l'algoritmo, allora viene presa la decisione di rinunciare a trasmettere. In caso contrario si riprende dalla valutazione dell'LME "peggiore".

Da notare come, nel caso un LME non avverta alcuna trasmissione dagli altri LME a lui vicini, comprenda, come da diagramma, che senza di esso la copertura della sua area è pari allo 0%: immediatamente quindi inizia a trasmettere. In questo modo si supera ogni possibile deadlock che si sia eventualmente creato,

iniziando una reazione a catena che sblocca la situazione e porta l'algoritmo a convergere verso la corretta soluzione.

Ogni LME deve effettuare una valutazione basandosi solo sulle informazioni ad esso note, cioè solo sui dati relativi alla propria località: data la scarsità di tali informazioni, e dato che la rete al contrario può avere dimensione arbitraria e non nota a nessun singolo LME, tipicamente anche molto maggiore della singola località, non è sempre possibile raggiungere la soluzione migliore in un solo passo.

Alla prima esecuzione dell'algoritmo gli LME tendono infatti ad avvicinarsi molto alla configurazione ritenuta migliore; nella maggior parte dei casi questa risulta poi essere definitiva, ma in casi particolari si assiste ad alcune successive variazioni: in base alla situazione creatasi inizialmente infatti, alcuni LME possono poi cambiare la loro posizione da trasmettenti a non e vice versa. Tali mutamenti coinvolgono però in ogni caso un numero limitato di nodi e portano ad un rapido (al massimo altri uno o due passi) convergere della rete verso la soluzione migliore evitando successivi continui avvicendamenti tra LME.

3.2.1.1 Integrazione nel Framework AGAPE

Una volta implementato in Java, il VCS è stato integrato nel framework AGAPE: poiché tale codice è però costituito dall'algoritmo in precedenza testato, tale fase è stata svolta senza notevoli modifiche se non per le classi accessorie richieste.

L'unica coordinazione necessaria con altri componenti di AGAPE è quella col VMS: da esso infatti il VCS riceve in input le viste, ed è ad esso che il VCS restituisce un valore corrispondente all'autorizzazione o meno alla trasmissione della vista dell'LME.

Due aggiunte necessari sono state l'implementazione della classe Vista e della classe Membro, effettuate come dal seguente diagramma (fig. 3.4).

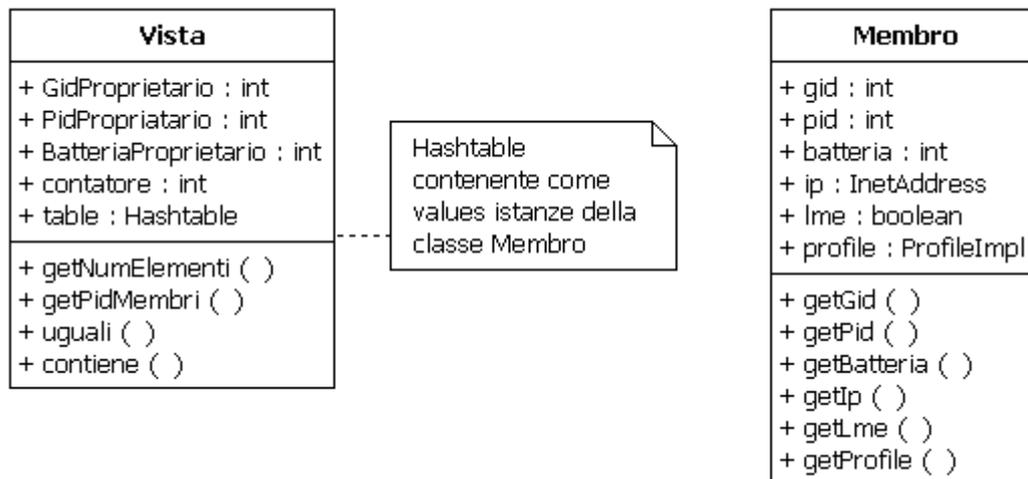


fig. 3.4 – Diagrammi uml delle classi Viste e Membro

Ogni LME riceve infatti dall’VMS un vettore di Vista, dove ogni Vista contiene un vettore di Membro, cioè l’insieme dei nodi nel raggio di trasmissione di ciascun LME.

Da notare inoltre come ogni Membro contenga un campo ProfileImpl, cioè l’implementazione concreta del profilo dell’utente che sta utilizzando tale dispositivo. Tali profili saranno poi la base per il livello applicativo soprastante il Middleware.

3.3 Dettagli Implementativi

Per quanto riguarda il middleware AGAPE, ne sono stati sviluppati due differenti prototipi.

Una release supporta le operazioni di gestione degli LME ed include tutti i servizi AGAPE. Tale release si adatta a dispositivi portatili come i lap-top, ricchi di risorse computazionali (grandi quantità di memoria, file system, potenti CPU e batterie dalla lunga durata). L’altra versione disponibile supporta le operazioni degli ME e include solo un sottoinsieme dei servizi supportati da AGAPE: J/LMS e VMS lato client, insieme a PS, PENS e NMS.

Questa sezione fornisce i dettagli implementativi e funzionali di alcuni servizi di gestione dei gruppi AGAPE che forniscono supporto la formazione dinamica e la gestione dei gruppi ad-hoc, i. e. J/LMS, VMS e VCS. Il principio che ha guidato l’implementazione dei servizi AGAPE è l’ottimizzazione delle risorse per far fronte alle limitazioni degli attuali dispositivi

portatili. In particolare viene qui spiegato come il J/LMS, il VMS e il VCS AGAPE lavorano in alcuni scenari di particolare importanza, come nel caso di LME, siti in prossimità, appartenenti allo stesso gruppo, come nel caso di LME, non siti in prossimità, che appartengono allo stesso gruppo e che definiscono località in parte sovrapposte e come nel caso di località partizionate.

Da notare come queste condizioni operative siano comuni negli scenari reali, come in MANET dense in scenari di emergenza civile dove un grande numero di utenti operano vicini grazie a dispositivi eterogenei.

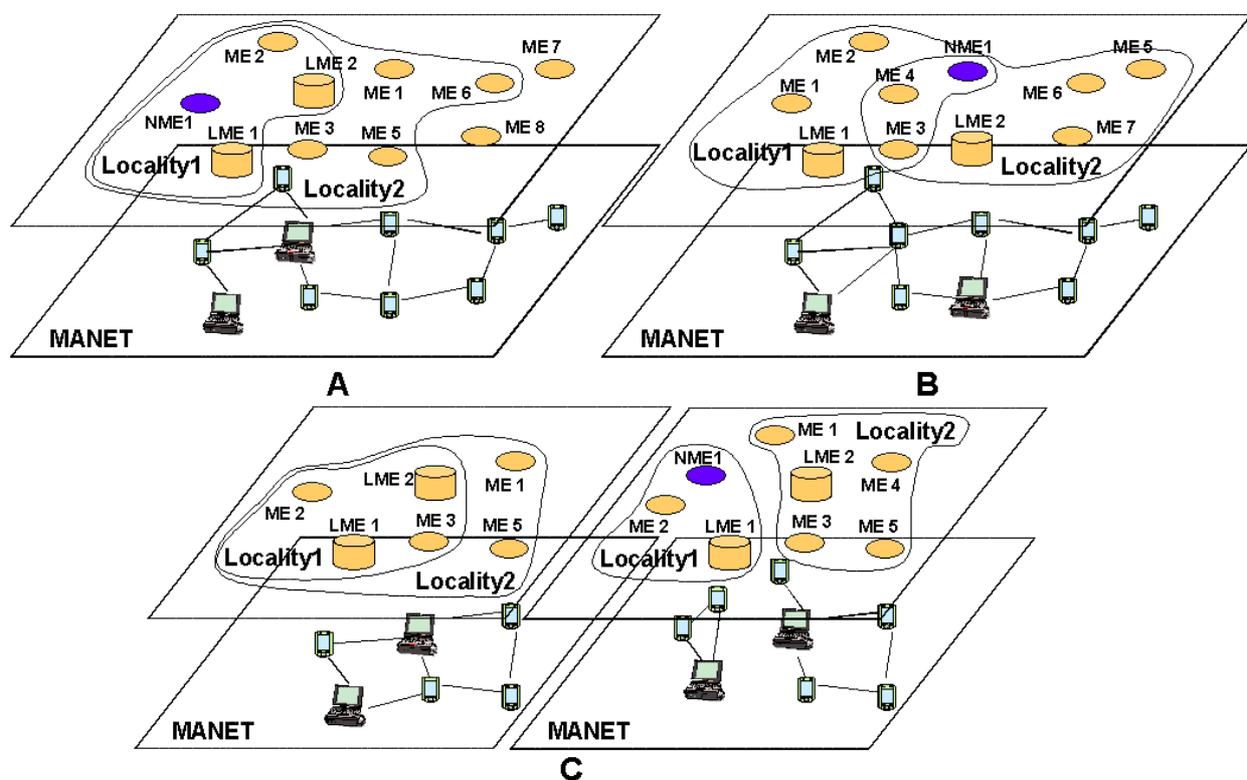


fig. 3.5 – AGAPE in esecuzione in differenti condizioni

3.3.1 Località Innestate

La fig. 3.5 mostra una situazione operativa nella quale due LME (LME1 e LME2) appartengono allo stesso gruppo, i. e. hanno lo stesso GID, e definiscono due località innestate, con la Località 1 contenuta nella Località 2. Da notare che i dispositivi LME eseguono tutti i servizi AGAPE mentre i dispositivi ME eseguono solo un sottoinsieme dei servizi supportati da AGAPE (J/LMS e VMS lato client, insieme a PS, PENS e NMS).

3.3.2 Unirsi ad un Gruppo

Consideriamo il caso dell'entità non membro NME1 che entra nella Località 1 e desidera unirsi al gruppo di suo interesse. L'istanza J/LMS sul dispositivo NME1 ottiene la lista di tutti gli LME nella Località 1, i. e. LME1 e LME2, insieme ai loro GID, PID, IP. Il J/LMS di NME1 scansiona la lista ottenuta e riconosce che sia LME1 sia LME2 appartengono allo stesso gruppo poiché hanno lo stesso GID. Il J/LMS seleziona in maniera casuale uno dei due, e. g. LME1, per unirsi al gruppo. L'istanza J/LMS di NME1 manda un messaggio di richiesta di unione a LME1 che include le preferenze di gruppo (in termini di obiettivi di gruppo e attività), insieme ai profili dell'utente e del dispositivo NME1.

In seguito alla ricezione del messaggio, il J/LMS di LME1 controlla se gli attributi richiesti del gruppo corrispondono alle preferenze di gruppo espresse da NME1. Se LME1 appartiene al gruppo di interesse di NME1, LME1 ammette l'entità di nel gruppo e risponde a NME1 con un messaggio di acknowledge che include il GID/PID del nuovo membro, ottenuto dal PENS, insieme al profilo di gruppo completo. Inoltre il J/LMS di LME1 si coordina con il VMS per includere il profilo di NME1 nella vista del gruppo.

Da notare come il protocollo di unione di un nuovo elemento ad un gruppo esistente è disegnato per preservare le risorse degli ME (cpu, memorie e batterie).

Quando possibile le operazioni di unione vengono delegate all'LME, e. g. tutta la computazione di confronto del profilo di un non membro che vuole unirsi al gruppo. La delegazione gioca un ruolo particolarmente importante quando un ME vuole unirsi ad un gruppo di interesse ed un gran numero di LME sono disponibili.

3.3.3 Propagazione delle Viste

Sia LME1 sia LME2 mantengono una vista di gruppo aggiornata di tutti i membri correntemente disponibili nella loro località. A intervalli di tempo regolari entrambe le istanze dei VMS di LME1 e di LME2 cercano di propagare le viste di gruppo a tutti i membri in Località 1 e in Località 2. Per propagare tali viste ciascun VMS deve chiedere l'autorizzazione all'istanza locale del VCS.

In particolare prima che NME1 si unisca al gruppo, LME2 ha il permesso di spargere la sua vista di gruppo a tutti i membri in Località 2, mentre LME1 non ha la possibilità di farlo. L'istanza del VCS di LME1 riconosce infatti che la sua vista di gruppo è un sottoinsieme di quella ricevuta da LME2.

Quando NME1 si unisce al gruppo (contattando LME1), l'LME che governa la propagazione delle viste cambia. Quando il VMS di LME1 include la nuova entità (NME1) nella sua vista di gruppo, il VCS di LME1 riconosce che NME1 non è contenuto nella vista di gruppo ricevuta da LME2 e consente anche ad LME1 di propagare la sua vista ai membri in Località 1. Una fase di transizione è però necessaria: sia LME1 sia LME2 trasmettono le loro viste a tutti i membri.

In particolare, LME2 riceve la vista da LME1. Il VMS di LME2 scansiona la vista ricevuta, riconosce che una nuova entità membro NME1 non fa parte della sua vista di gruppo mantenuta localmente e verifica che NME1 si trovi in Località 2 (coordinandosi col PENS). Come rappresentato in fig. 3.5, NME1 si trova in Località 2 e quindi LME2 include NME1 nella sua vista di gruppo.

Quando LME2 dissemina la sua vista a tutti i membri in Località 2, anche LME1 la riceve (trovandosi infatti nella stessa località). Quando il VMS di LME1 tenta di propagare la sua vista, viene bloccato poiché la vista di gruppo ricevuta da LME2 include NME1. Considerazioni simili si applicano sia nel caso NME1 giochi il ruolo di ME sia di LME.

In condizioni operative normali tutti i membri di gruppo siti in Località 2 (inclusi quelli appartenenti a Località 1) ricevono la vista da LME2. In ogni caso quando NME1 si unisce al gruppo, tutti i membri in Località 1 sperimentano una fase transitoria quando ricevono le viste di gruppo sia da LME1 sia da LME2. Durante questa fase i membri in Località 1 fondono le viste provenienti da LME1 e da LME2. Questa soluzione permette di risparmiare memoria in dispositivi che ne posseggono una quantità scarsa, ma porta ad un maggior consumo delle batterie dovuto alla computazione aggiuntiva necessaria per unire le viste in arrivo. Differenti soluzioni sono comunque possibili, e. g. per mantenere

due tabelle distinte associate rispettivamente alle viste provenienti da LME1 e da LME2. La scelta dipende dalla priorità che si vuole dare al risparmio delle risorse, se si ritiene più importante preservare memoria, CPU o batteria.

3.3.4 Località Parzialmente Sovrapposte

La fig. 3.5B mostra come LME1 e LME2 non abbiano visibilità l'uno dell'altro e di conseguenza non possano comunicarsi le rispettive viste. Il VCS di entrambi gli LME permette infatti al VMS di ciascuno dei due la propagazione delle viste ai membri di Località 1 e di Località 2.

È facile notare come tutti i membri nell'intersezione delle due località ricevano le viste sia di LME1 sia di LME2. Questi membri operano poi una fusione di tali viste nella stessa table sulla base dei campi GID e PID inclusi in ciascuna entry delle viste.

Possiamo inoltre notare come la nuova entità membro NME1 è inclusa solo nella vista di LME1 in Località 1. Quando NME1 riceve la vista propagata da LME2 riconosce di non esservi incluso: esso manda quindi il suo completo profilo (ottenuto dal suo J/LMS) a LME2. Una volta ricevuto il messaggio LME2 include NME1 nella sua vista.

CAPITOLO 4

TEST DEL SISTEMA SVILUPPATO

Al fine di validare il sistema sviluppato prima di inserirlo nel framework AGAPE, abbiamo condotto diversi test: è stato quindi sviluppato un simulatore per verificare i risultati ottenuti dall'algoritmo. Un'altra necessità era inoltre avere dati sperimentali concreti sulla qualità dei risultati conseguiti. Il codice inserito in tale simulatore è lo stesso che, una volta collaudato, è stato poi integrato nel framework AGAPE.

La scelta di una verifica attraverso un simulatore software è stata adottata a causa delle difficoltà relative al deployment di una rete MANET costituita da un ampio numero di dispositivi e dalle difficoltà relative all'analisi dei dati provenienti dai vari file di log creati sui singoli nodi.

Data la specificità dei test da effettuare non sono stati usati strumenti standard quali NS2 (Network Simulator 2), ma si è scelto di realizzare un piccolo simulatore che fosse di rapido sviluppo ma che al contempo, con un approccio semplice al problema, offrisse la possibilità di testare tutte le soluzioni trovate in fase di realizzazione. L'uso di strumenti standard inoltre non era adatto anche alla luce della necessità di testare direttamente il codice che è poi stato inserito nel package AGAPE.

Per il simulatore si è scelto il livello applicativo per un più rapido sviluppo e per riuscire a mostrare in maniera semplice ma efficace i risultati raggiunti.

Il programma viene azionato dall'utente attraverso l'interfaccia grafica ad eventi, ma poiché ogni run dell'algoritmo costituisce un nuovo istante di trasmissione delle viste e poiché tali trasmissioni avvengono ad istanti regolari, il simulatore lavora a tutti gli effetti in real-time. In questo modo è inoltre possibile verificare i risultati ottenuti e modificare di conseguenza lo scenario per testare particolari aspetti altrimenti difficili da evidenziare.

4.1 Il Simulatore

L'input deve consentire la simulazione di scenari con un numero di nodi anche elevato, nell'ordine di grandezza del centinaio o anche più. È inoltre necessario un metodo sufficientemente veloce di generazione random di località, mostrando (e quindi testando) tutti

gli aspetti evidenziati nel capitolo precedente. D'altra parte deve però consentire la realizzazione di scenari particolari, magari meno comuni nella realtà, ma che presentino in maniera semplice tutte le sfaccettature della dinamica dell'algoritmo.

Nei test va inoltre simulata la differenza di batteria residua dei vari dispositivi, che può causare una diversa scelta nella configurazione, oppure una diversa evoluzione dello scenario nel tempo.

Un altro aspetto rilevante da valutare è inoltre la necessità di verificare sperimentalmente la reattività dell'algoritmo al variare della topologia della rete. Essenziale nel lavoro è infatti la capacità di adattarsi rapidamente a scenari in continua evoluzione, con i nodi liberi di muoversi nello spazio. È stata quindi simulata la possibilità dei nodi di mutare la loro posizione nel corso dell'evoluzione della topologia simulata.

4.1.1 L'Architettura del Simulatore

Come da fig. 4.1, il cuore dell'algoritmo è dato dalla funzione *determinaStatus()* che, eseguendo l'algoritmo, setta il valore del campo *status* che indica appunto la decisione o meno di propagare la propria vista.

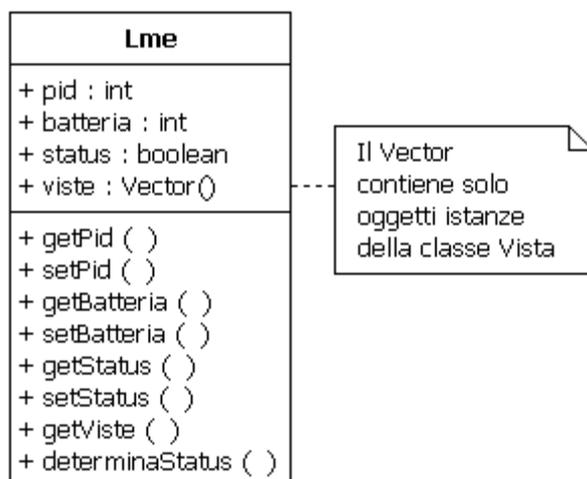


fig. 4.1: - Diagramma UML della classe LME
(N.B.:compaiono solo i metodi pubblici principali)

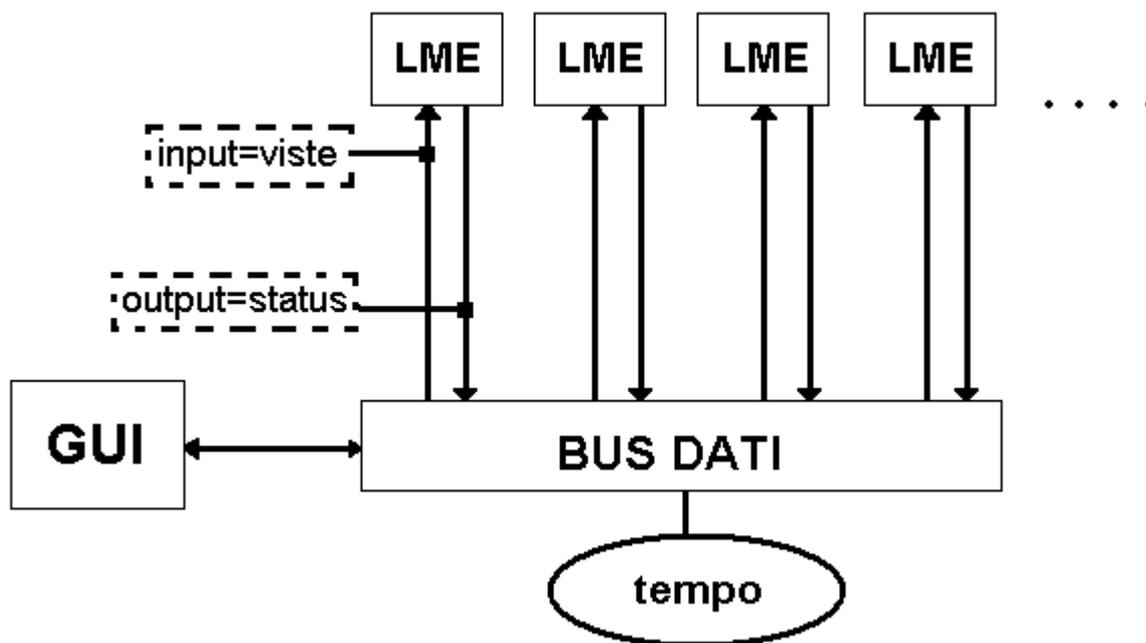


fig. 4.2 – L'architettura del simulatore

Il simulatore (fig. 4.2) è fornito di una interfaccia grafica (Graphic User Interface, GUI) che permette all'utente di generare gli scenari prima e le evoluzioni degli stessi poi. A partire da tale interfaccia vengono forniti in ingresso al BUS i dati relativi allo scenario nel quale si desidera verificare il funzionamento del sistema.

Come da fig. 4.2 il programma crea vari LME, i quali hanno necessità di comunicare tra loro come avviene negli scenari reali: tale simulazione viene effettuata da un bus dati, che si occupa della coordinazione dei vari nodi gestendo sia la trasmissione delle viste sia l'output.

Il BUS è stato introdotto per poter seguire le rapide variazioni della topologia aggiornando di pari passo le viste e l'insieme dei nodi che devono riceverle.

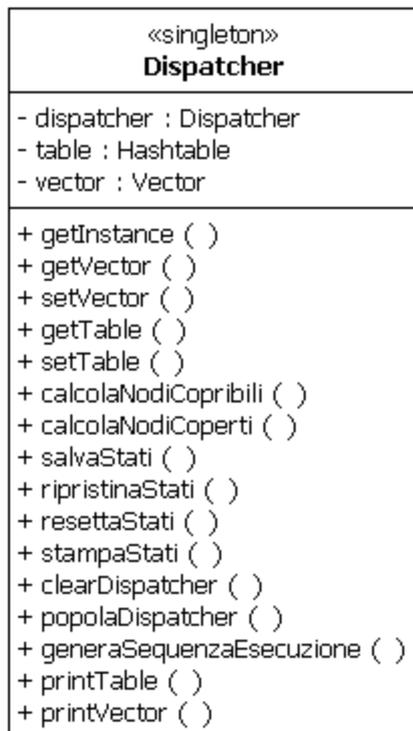


fig. 4.3 – Diagramma uml della classe Dispatcher

Tale simulazione del BUS di trasmissione dati viene effettuata grazie alla classe Dispatcher (fig. 4.3). Tale classe ha il compito di coordinare in maniera opportuna gli LME, simulando il concetto di località, e a questo scopo ha due strutture: la prima contiene gli oggetti istanze della classe LME che rappresentano le varie entità AGAPE gestori dei gruppi. La seconda struttura invece memorizza per ciascuna di tali entità tutti gli altri LME da essa raggiunti, ovvero i membri della località con i quali viene poi effettuato il confronto per stabilire quali siano gli LME più adatti (dal punto di vista del rapporto copertura / dispendio di risorse) a propagare le viste.

Il problema della possibile inconsistenza delle viste dei vari LME viene automaticamente risolto grazie alla modalità di input: la GUI, infatti, memorizza le coordinate dei punti corrispondenti ai vari LME / ME e sulla base di queste procede in seguito alla creazione delle viste. Queste ultime vengono prodotte calcolando la distanza tra ciascun LME e tutte le altra entità: se un'entità risulta essere compresa nell'area di trasmissione dell'LME in questione, allora viene inserita nella sua vista. In tal modo è inoltre possibile per l'utente modificare lo scenario in qualsiasi momento senza

possibilità di un conseguente errore nella modifica delle viste: è infatti sempre compito della GUI ricalcolare le nuove posizioni e procedere a fornire al Dispatcher le viste aggiornate, relative alla nuova situazione. Sarà poi il Dispatcher a fornirle in ingresso ai singoli LME che possono quindi mettere in esecuzione l'algoritmo presentato nel capitolo precedente.

In input ogni istanza della classe LME deve ricevere le viste dei soli altri LME che si trovano nella sua località; compito del bus è quindi anche la selezione dei dati da inviare ai vari LME: ognuno infatti deve avere in input solo i dati che gli competono.

Il BUS riceve poi d'altra parte l'output dei vari LME, cioè la decisione presa in seguito all'esecuzione dell'algoritmo, ovvero se trasmettere o meno: tale valore rappresenta la concessione o meno dell'autorizzazione alla propagazione delle viste che il VCS fornisce al VMS.

È stato poi preso un ulteriore accorgimento per rendere più realistico il simulatore: nella realtà infatti le istanze della classe VMS interrogano le istanze della classe VCS in momenti diversi e quindi in ordine sempre differente e imprevedibile. Per simulare tale comportamento e per evitare che l'algoritmo fosse eseguito dai singoli LME sempre nello stesso ordine, cosa che avrebbe potuto falsare i risultati, è stato introdotto un metodo di generazione casuale della sequenza nella quale viene effettuato il run dell'algoritmo da parte dei vari LME.

Come precedentemente accennato viene inoltre gestito il tempo sfruttando il modello ad eventi fornito dall'interfaccia grafica: il reale trascorrere del tempo viene infatti perfettamente riprodotto dai run dell'algoritmo. Ogni run corrisponde infatti ad un istante di trasmissione delle viste ed alla conseguente decisione del proprio status da parte di ciascun LME.

4.1.2 L'interfaccia Grafica

L'interfaccia grafica è stata disegnata seguendo il pattern Model / View / Controller: è stata realizzata infatti una partizione della frame in tre pannelli, in maniera tale da facilitare sia l'inserimento dei dati necessari a creare lo scenario, sia la visualizzazione della

rete, che viene presentata sotto forma di grafico, sia i risultati, leggibili nell'apposita area di testo.

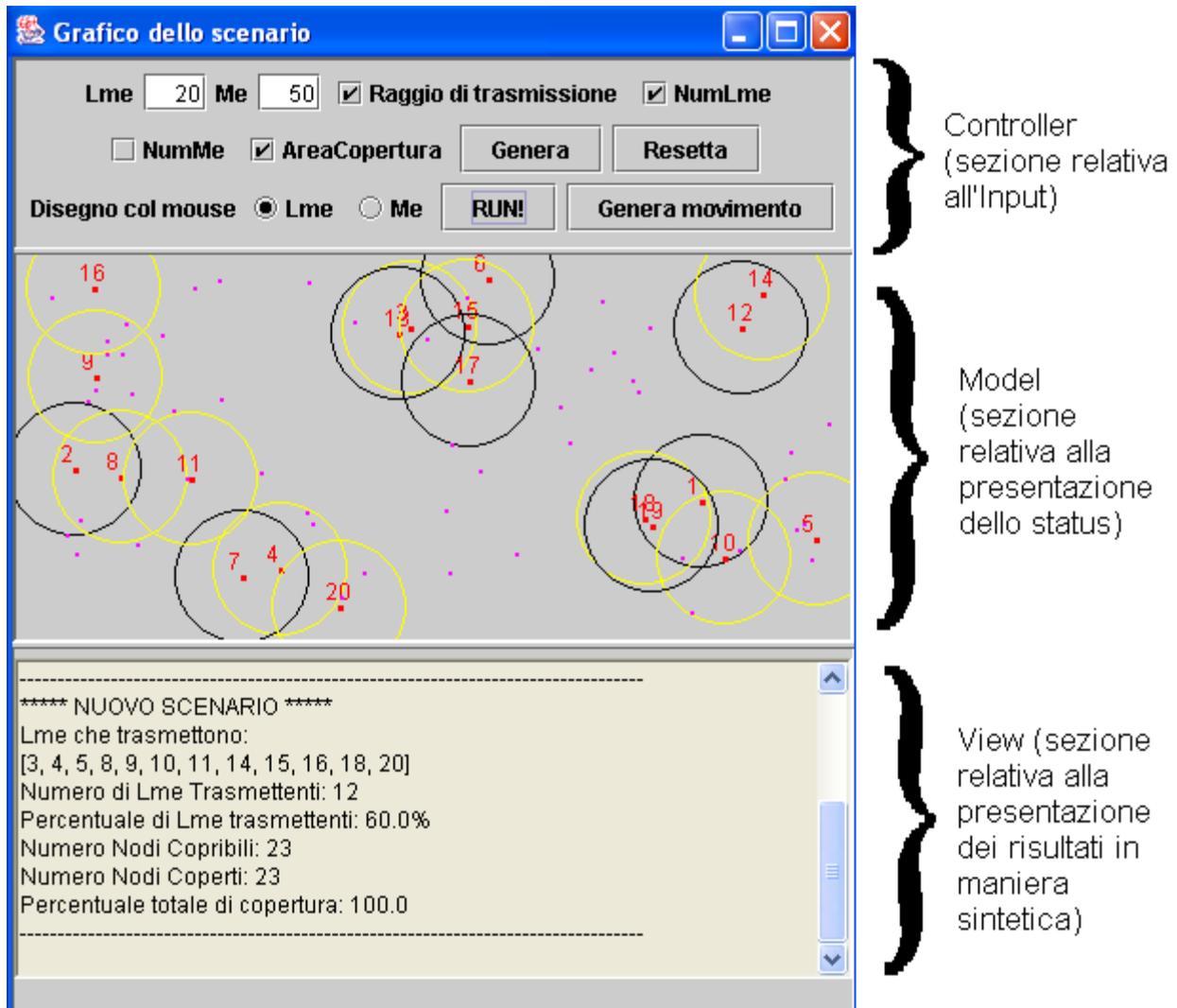


fig. 4.4 – Interfaccia grafica dopo l'esecuzione dell'algoritmo applicato ad uno scenario generato in maniera casuale

4.1.2.1 L'Input (fig. 4.4 – sezione alta)

Scendendo nel dettaglio due sono le modalità di creazione di uno scenario: l'utente può disegnare manualmente i nodi semplicemente cliccando nell'area grafica, scegliendo con l'apposito radio button se aggiungere LME o ME oppure può generare in maniera automatica uno scenario con i nodi posizionati in maniera casuale, dopo aver specificato il numero di LME e di ME desiderati.

La prima modalità è di più lenta realizzazione ma permette di creare la rete esattamente come l'utente desidera ed è quindi adatta a test specifici, la seconda è al contrario casuale e quindi

intrinsecamente non precisa, ma consente d'altro canto di realizzare test con molti nodi in pochi attimi.

Il button *Genera Movimento* consente poi, una volta creato lo scenario, di simulare la variazione di posizione degli LME e degli ME: tale variazione viene immediatamente visualizzata nel pannello ed è subito attiva la possibilità di eseguire nuovamente l'algoritmo per verificare l'adattamento della rete alla nuova configurazione.

Inoltre è possibile generare più movimenti prima di effettuare un nuovo run. In questo modo è simulabile sia un movimento lento dei nodi (con poche "generazioni di movimento" prima del successivo run) sia un movimento più rapido (più "generazioni di movimento").

4.1.2.2 Lo Stato del Sistema (fig. 4.4 – sezione centrale)

Generando lo scenario (o anche successivamente) l'utente può scegliere con quale livello di dettaglio grafico visualizzare quanto creato: con le apposite check box possono essere infatti disegnate o meno le aree di trasmissione degli LME, i numeri identificativi di LME e/o ME e, una volta messo in azione l'algoritmo, evidenziare con un colore diverso le aree di trasmissione degli LME che effettivamente hanno determinato la necessità di propagare la propria vista. Tutte queste possibilità vengono date per mettere in rilievo l'aspetto al quale un utente è maggiormente interessato nel test: tutte le opzioni infatti, se selezionate contemporaneamente, possono in molti casi (specialmente con molti nodi rappresentati) complicare la visualizzazione e la comprensione grafica del risultato.

4.1.2.3 L'Output (fig. 4.4 – sezione bassa)

La terza sezione del pannello è riservata all'esposizione riassuntiva dei dati ottenuti dall'esecuzione dell'algoritmo.

Vengono elencati i PID (numeri progressivi identificanti in maniera univoca i vari nodi) degli LME trasmettenti, il numero totale degli stessi e la percentuale di attivi sul totale dei rappresentati. Segue quindi il numero di ME che sono potenzialmente all'interno dell'area di trasmissione di almeno un LME (tale numero non corrisponde a quello degli ME visualizzati, dato che, specialmente

nella creazione random, molto frequentemente più ME possono trovarsi in aree totalmente scoperte), il numero di quelli effettivamente coperti dalla configurazione attuale e la percentuale di copertura raggiunta. Come controllo dell'effettivo funzionamento del programma vengono elencati anche i PID dei nodi rimasti esclusi dalle trasmissioni (per problemi di risparmio di risorse, viene seguita una politica *best effort*) anche se potenzialmente raggiungibili. Nel caso si eseguano più Run consecutivi dell'algoritmo, avendo o meno modificato nel frattempo lo scenario, vengono sottolineate le variazioni avvenute, ovvero quali LME hanno iniziato o cessato la trasmissione.

4.2 Alcuni Test e Valutazione delle Performance

Vengono in questo capitolo illustrati alcuni esempi di evoluzione di scenari, per fornire una valutazione quantitativa del comportamento dell'algoritmo e dei risultati che esso può raggiungere: come supporto visivo vengono fornite le schermate stesse presentate in output all'utente dal simulatore, tralasciando la sezione relativa all'input (il controller).

Nel seguito si confronteranno i risultati raggiunti grazie al VCS con quelli che si conseguirebbero senza di esso: viene quindi definita naive la soluzione che si avrebbe dando a tutti i VMS l'autorizzazione alla propagazione delle viste.

4.2.1 Lo Sviluppo di una Piccola Rete

Prendiamo come esempio il caso di studio di una piccola rete in espansione e analizziamo come le entità presenti in essa reagiscono, grazie al VCS, al mutare della topologia.

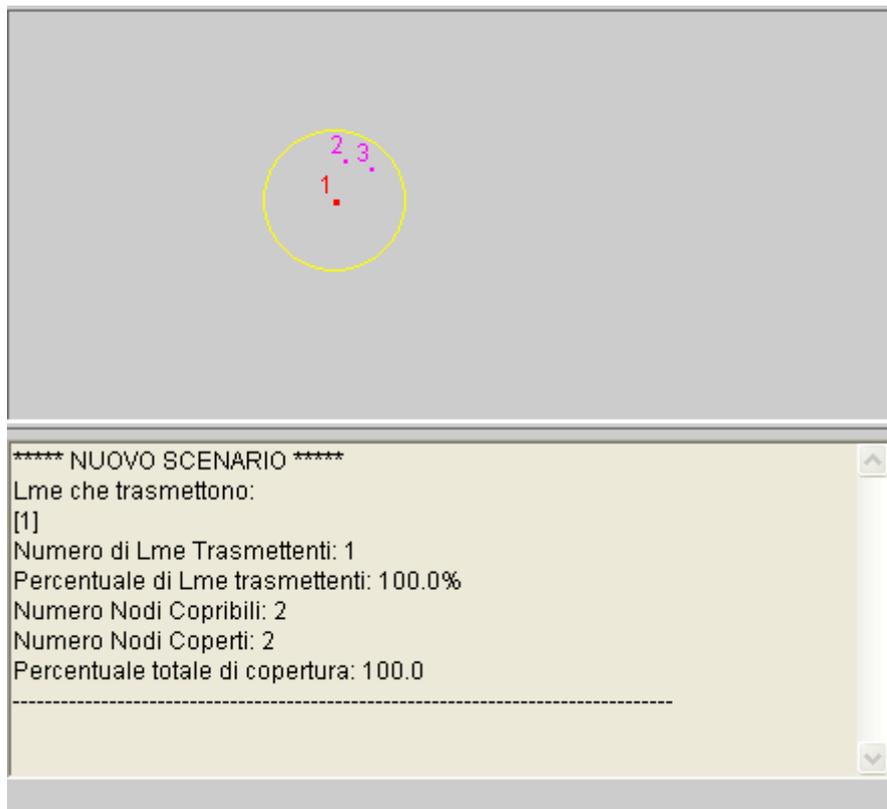


fig. 4.5 – Una elementare rete composta da un LME e due ME

Una piccola rete può, per esempio (fig. 4.5), essere composta da un solo LME e da alcuni ME, nel caso in figura soltanto due.

Nell'area di output possiamo leggere i risultati forniti dall'esecuzione dell'algoritmo da parte del VCS: l'unico LME presente sta fornendo i servizi necessari al gruppo e quindi la copertura raggiunta è naturalmente del 100%. In questo caso elementare notiamo come il VCS non apporti alcuna miglioria rispetto al caso naive.

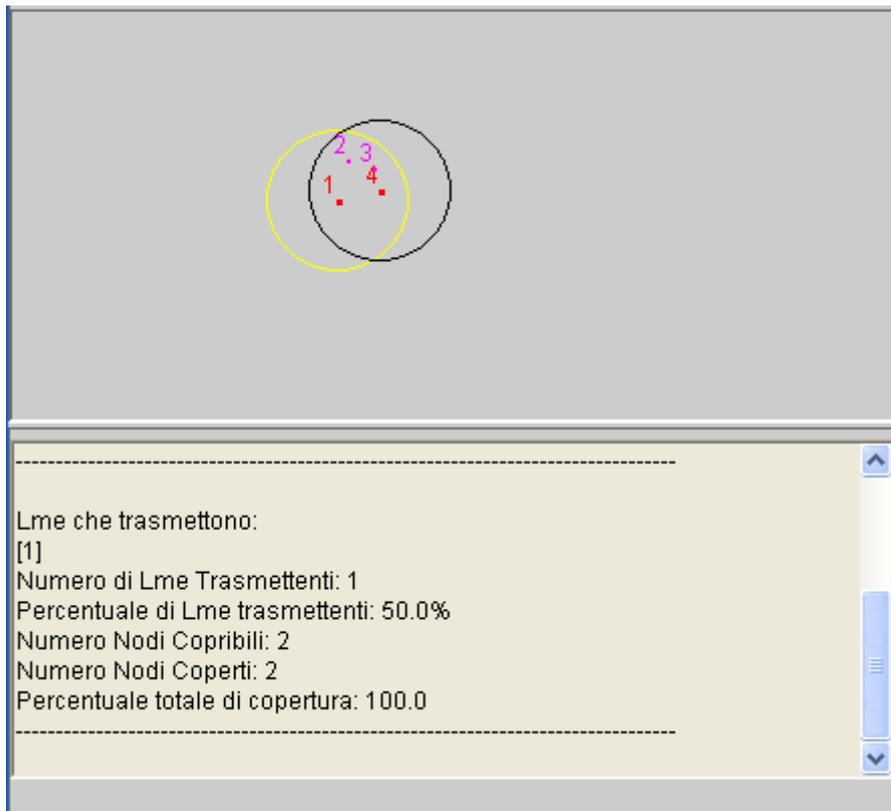


fig. 4.6 – Arrivo di un LME

Una possibile evoluzione dello scenario può essere rappresentata dal sopraggiungere nell'area presa in considerazione di un secondo LME (fig 4.6).

Quando il VMS di tale LME chiede l'autorizzazione alla propagazione della vista al suo VCS, questi gliela nega: è infatti inutile al fine della copertura, e al contrario controproducente dal punto di vista del corretto uso delle risorse, che due LME trasmettano contemporaneamente all'interno della stessa area. Il nuovo LME non farebbe altro infatti se non ribadire informazioni già espresse dall'LME di PID=1.

È indicativo notare come la stessa copertura sarebbe stata raggiunta a ruoli invertiti, ovvero se fosse stato autorizzato alla trasmissione l'LME di PID=4 al posto dell'LME di PID=1. Un requisito essenziale è però, come illustrato nei capitoli precedenti, la stabilità della rete, per evitare riconfigurazioni inutili che non possono fare altro che causare provvisori transitori senza alcun beneficio: quando è possibile viene quindi mantenuto invariato, come in questo caso, lo stato presente.

Anche solo con questa minima variazione possiamo già notare come la copertura rimanga percentualmente invariata, ma il numero di LME che propagano la propria vista sia già calato al 50%, introducendo notevole risparmio di banda e batteria dei dispositivi rispetto al caso naive, dove avremmo invece entrambi gli LME trasmettenti.

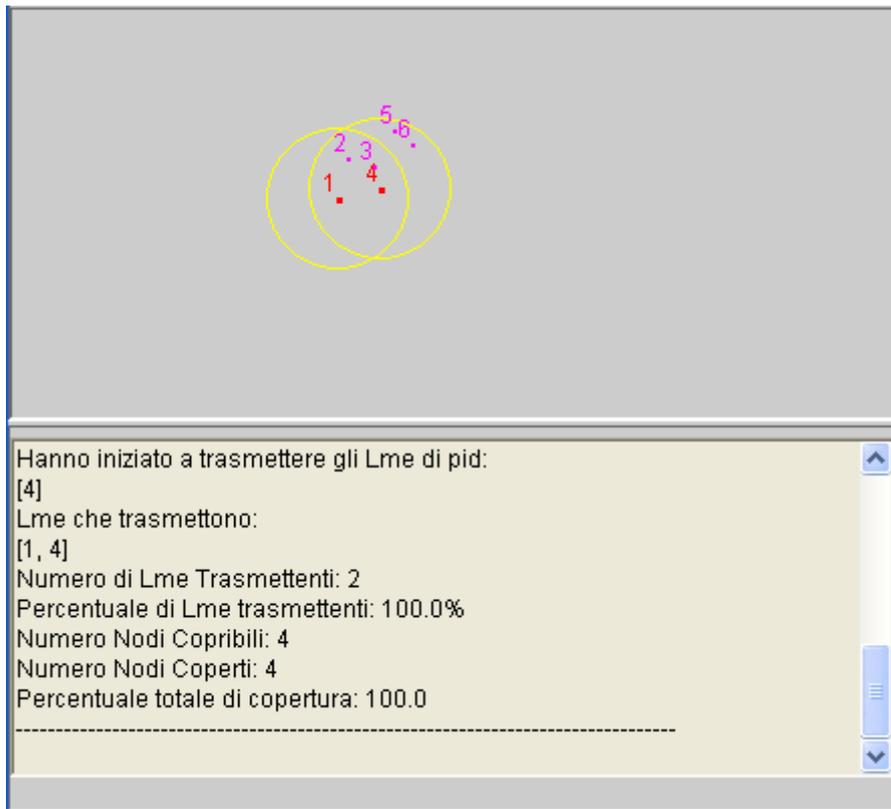


fig. 4.7 – Sopraggiungere di altri due ME

Al sopraggiungere (fig. 4.7) di altri due ME nell'area coperta dall'LME di PID=4, che era precedentemente interdette dal proprio VCS alla propagazione delle viste, si ha una modifica nella configurazione. Attraverso l'uso di pacchetti beacon infatti, i due nuovi ME vengono rilevati dall'LME di PID=4, il quale d'altra parte, leggendo la vista in suo possesso proveniente dall'LME di PID=1, capisce che tali nodi non hanno un'entità gestore del gruppo. Il VCS dell'LME di PID=4 autorizza quindi il suo VMS e la propagazione della vista di tale LME ha inizio.

La grafica segnala puntualmente, nell'area di output sottostante, tale variazione: è però necessario notare come questa configurazione non sia stabile, ma come costituisca soltanto una fase di transitorio.

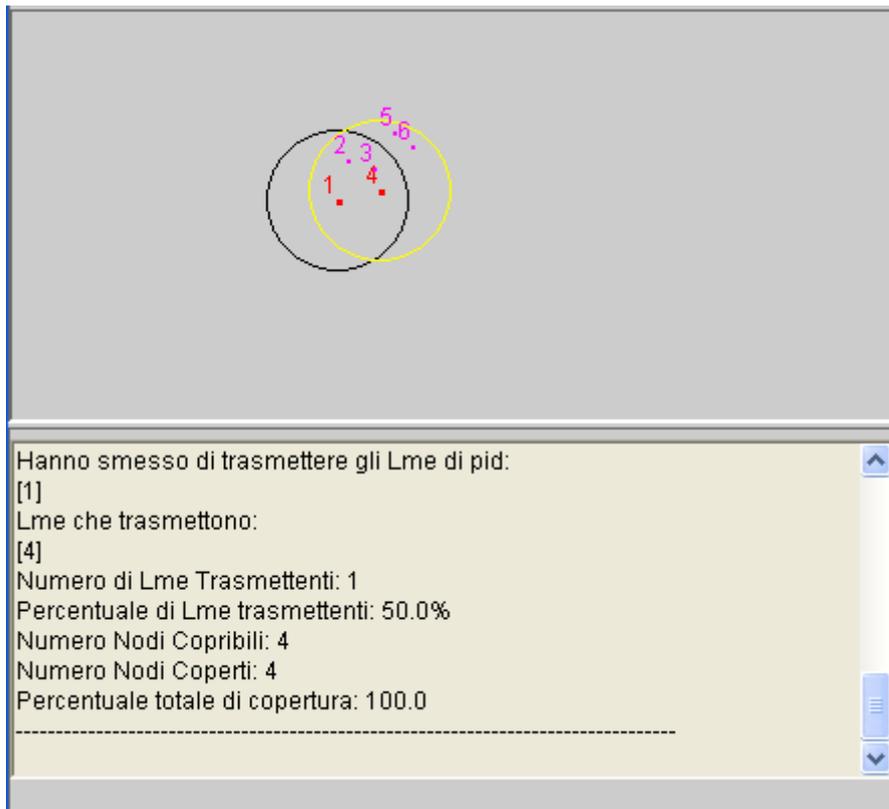


fig. 4.8 – Seconda fase del transitorio

Come mostrato in fig. 4.8 infatti, si assiste ad una seconda variazione nella configurazione senza variazione della posizione o del numero dei nodi.

L’LME di PID=1, che gestiva precedentemente i servizi di gruppo, riceve la nuova vista dell’LME di PID=4. In tale vista sono presenti tutti i nodi raggiunti dall’LME di PID=1 più due ulteriori nodi. Il VCS di tale LME, operando un confronto, si accorge che la propagazione della propria vista non è più necessaria poiché quella dell’altro LME è migliore. L’autorizzazione alla trasmissione al VMS viene quindi revocata.

Nuovamente, nonostante l’aumento del numero di nodi, assistiamo ad un risparmio di trasmissioni, rispetto al caso naive, pari anche in questo caso al 50%.

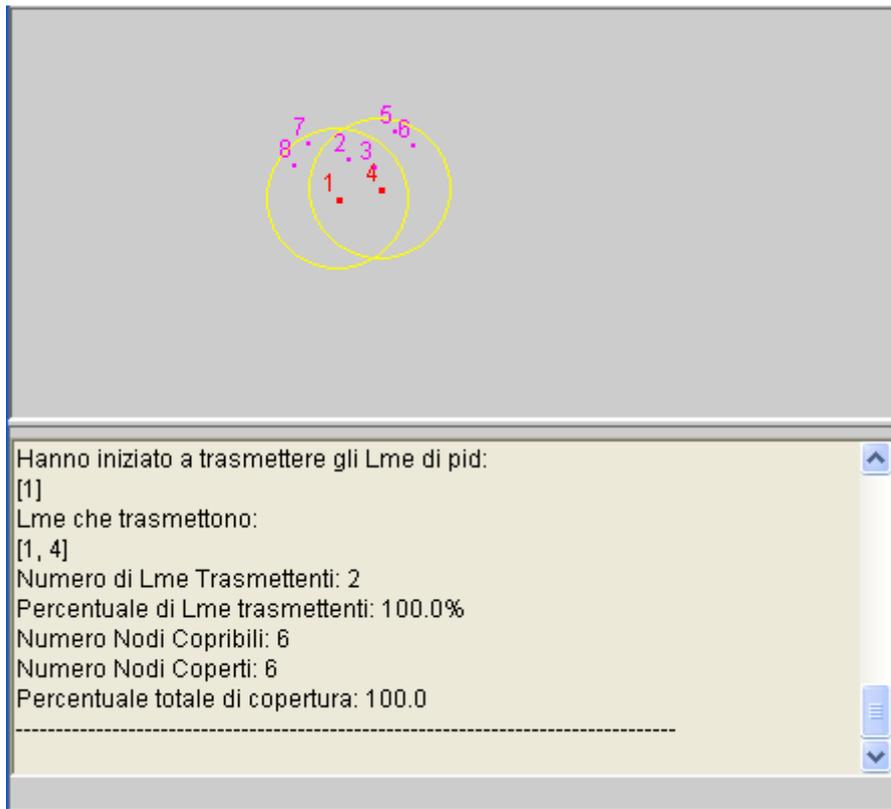


fig. 4.9 – Ulteriori due ME sopraggiungono

Una nuova reazione della rete si ha nel caso in fig. 4.9, ovvero all'arrivo di altri due ME nell'area coperta però dall'LME il cui VCS sta interdicensi il suo VMS dalla propagazione della propria vista.

Sempre grazie ai pacchetti di beacon, l'LME di PID=1 rileva i nuovi nodi e di conseguenza il VCS locale autorizza il VMS alla trasmissione; non sono in seguito necessarie ulteriori fasi di stabilizzazione.

È necessario notare come la copertura sia del 100%, ma come in questo caso gli ME di PID=2 e di PID=3 ricevano viste da entrambi gli LME presenti nello scenario: si ha quindi un consumo maggiore di risorse, che risulta però inevitabile, dato che se uno soltanto dei due LME propagasse la propria vista la copertura sarebbe troppo scarsa. In uno scenario con un così esiguo numero di entità non si può infatti rinunciare a due nodi, che rappresentano una percentuale di partecipanti troppo alta.

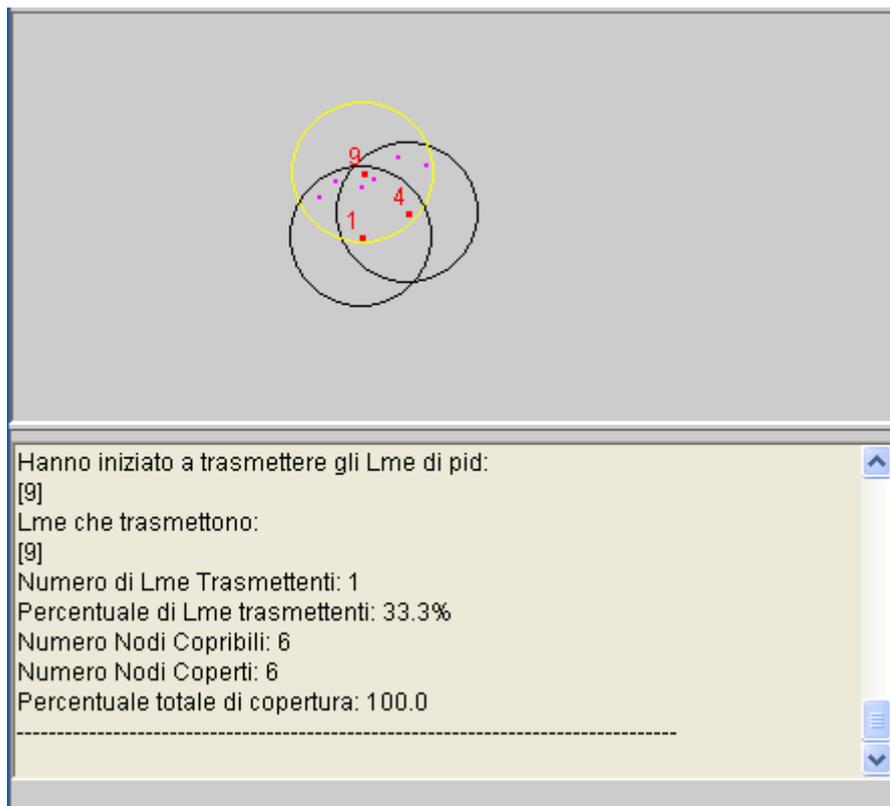


fig. 4.10 – Un nuovo LME presente nell’area

Nell’immagine in fig 4.10 (nella quale sono stati omessi, utilizzando l’apposita opzione del controller, i PID degli ME per maggiore chiarezza grafica) assistiamo ad un esempio di “sostituzione”.

Il nuovo LME sopraggiunto infatti è compreso nell’area di copertura degli LME di PID=1 e di PID=4. La propagazione della sua vista non è quindi necessaria ai fini dell’aumento della zona coperta.

Grazie al confronto delle informazioni provenienti dai pacchetti di beacon e di quelle contenute nelle viste diffuse dagli altri due LME, il VCS del nuovo LME raggiunge però un notevole risultato: comprende infatti che l’entità sulla quale è in esecuzione può raggiungere da sola la stessa soluzione degli altri due LME messi insieme. In altri termini, l’unione delle viste degli LME di PID=1 e di PID=4 corrisponde alla sola vista del nuovo LME arrivato.

L’LME di PID=9 viene quindi autorizzato dal proprio VCS alla trasmissione. In seguito alla ricezione di tale vista i VCS degli altri due LME si accorgono che le proprie viste sono contenute in quella trasmessa dall’LME di PID=9 e revocano l’autorizzazione alla diffusione delle proprie informazione ai corrispondenti VMS. In fig. 4.10 vediamo la configurazione a transitorio esaurito: da evidenziare

come tale transitorio duri in ogni caso un ciclo di propagazione delle viste soltanto e come il numero di LME trasmettenti passi, rispetto al caso naive, da tre a uno soltanto, con un risparmio del 67% di entità che inviano la propria vista. La copertura resta invece, nonostante ciò, invariata.

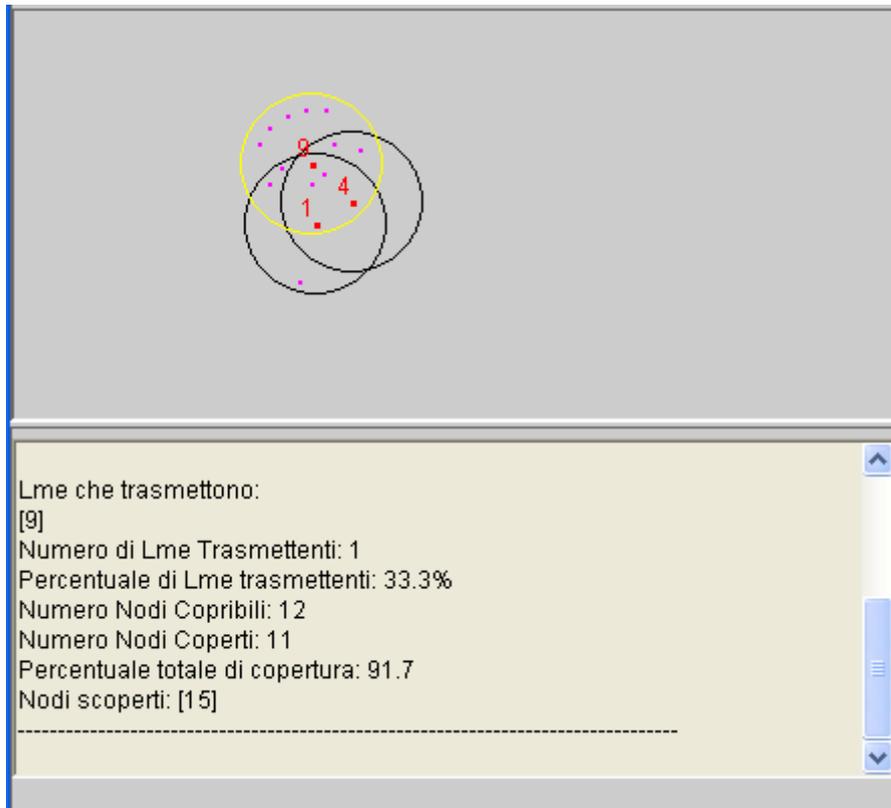


fig. 4.11 – Aumento degli ME presenti

Aumentando il numero dei nodi presenti nello scenario (fig. 4.11) entrano in gioco altri fattori che possono portare a scelte diverse: è ora necessario infatti effettuare una valutazione non solo sull'area coperta, ma anche su quanti nodi ricevono viste in maniera ridondante. Come mostrato in figura infatti, l'LME di PID=1 potrebbe propagare la propria vista per raggiungere il nuovo ME giunto in basso a sinistra, ma vi sarebbero troppi nodi che riceverebbero informazioni inutilmente duplicate, con conseguente spreco di risorse. Questo è il motivo per il quale nello scenario presente non avviene alcuna variazione alla configurazione attuale e il VCS dell'LME di PID=1 continua a bloccare l'invio della propria vista: la copertura è infatti sufficientemente elevata per poter permettere un ragionamento best effort e ritenere trascurabile un ME

isolato. Con la presente soluzione viene inoltre evitata una eventuale fase di transizione.

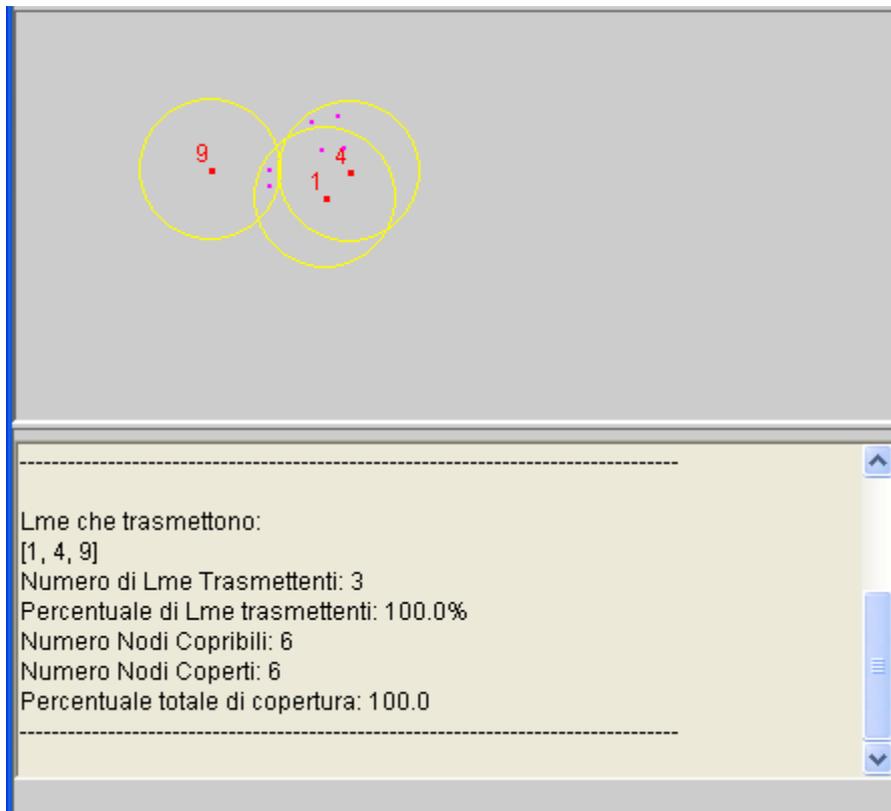


fig. 4.12 – Partizione della rete

A causa del movimento dei nodi può accadere come in fig. 4.12 una partizione della rete in due o più gruppi: in tal caso gli LME non ricevono più gli uni le viste degli altri e di conseguenza iniziano a propagare la propria vista. Nel caso mostrato l’LME di PID=9 e gli altri due LME non sono più nei rispettivi raggi di trasmissione e non rilevano più di conseguenza la diffusione delle altrui viste.

L’LME di PID=9 inizia quindi la trasmissione non sentendo nessun altro LME e lo stesso fanno gli altri due LME, che rilevano l’uno la presenza dell’altro, ma sono entrambi necessari per poter raggiungere tutti gli ME presenti.

Possiamo ora osservare le differenze nei vari casi appena illustrati con e senza l’ausilio del VCS (fig. 4.13).

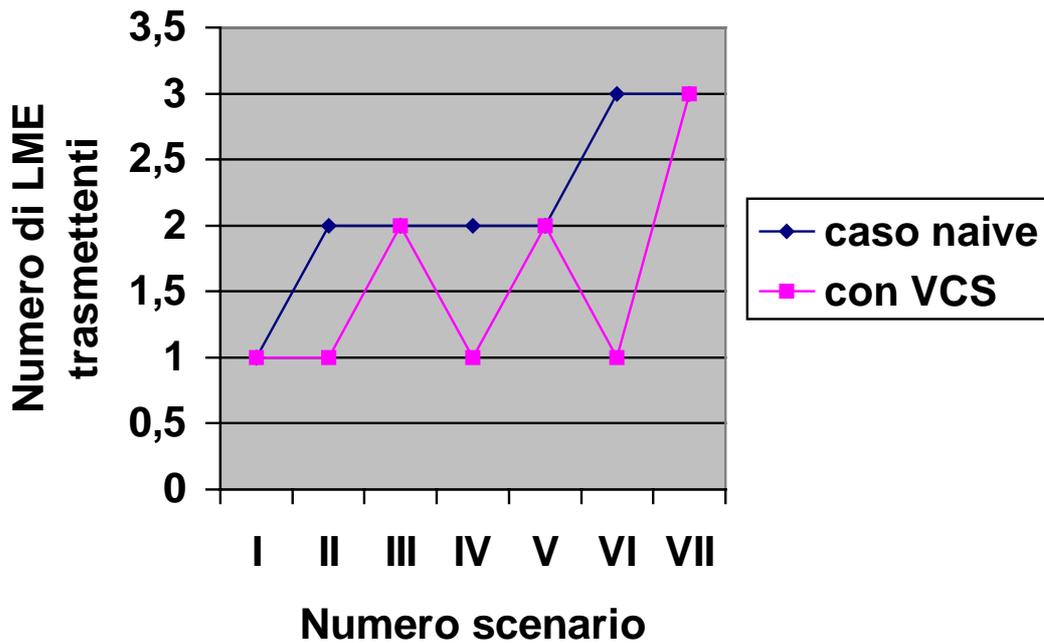


fig. 4.13 – Confronto tra le prestazioni con e senza VCS

4.2.2 Generazione di Scenari Random

Grazie alla funzione di generazione casuale di scenari è possibile creare reti con un elevato numero di nodi in poco tempo: ciò è particolarmente utile per evidenziare come, con l'aumentare del numero dei dispositivi, l'impatto del VCS sia sempre maggiore.

Sono quindi stati generati 10 scenari casuali con 35 LME e 100 ME, numeri che rappresentano una MANET di dimensioni già molto elevate, ottenendo i risultati presenti nei seguenti grafici.

Le dimensioni degli scenari generati corrispondono nella realtà approssimativamente ad un'area di 3 Km x 1.5 Km.

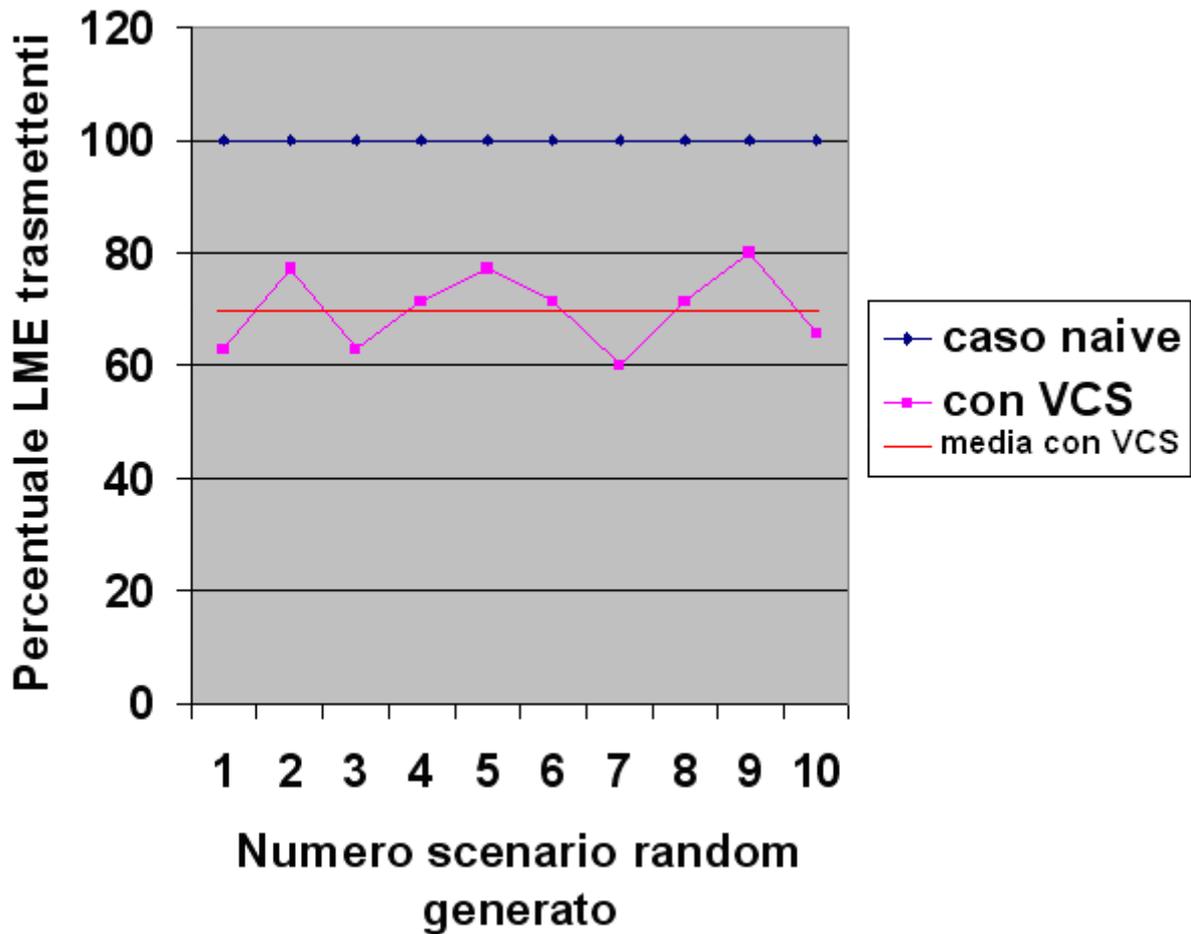


fig. 4.14 – Confronto tra VCS e caso naive: percentuale di LME trasmettenti

Come evidenziato in fig. 4.14 la percentuale di LME necessari per ottenere una buona copertura è ben inferiore al 100%: il risparmio in termini di risorse è inoltre di gran lunga ancora maggiore. Per comprendere la motivazione di ciò è sufficiente osservare come normalmente non propagano le proprie viste gli LME presenti in aree con alta densità di LME e di nodi in generale. Non trasmettendo tali LME, è quindi elevato il numero di dispositivi che evitano la ricezione di informazioni non necessarie e quindi alto anche il risparmio di risorse.

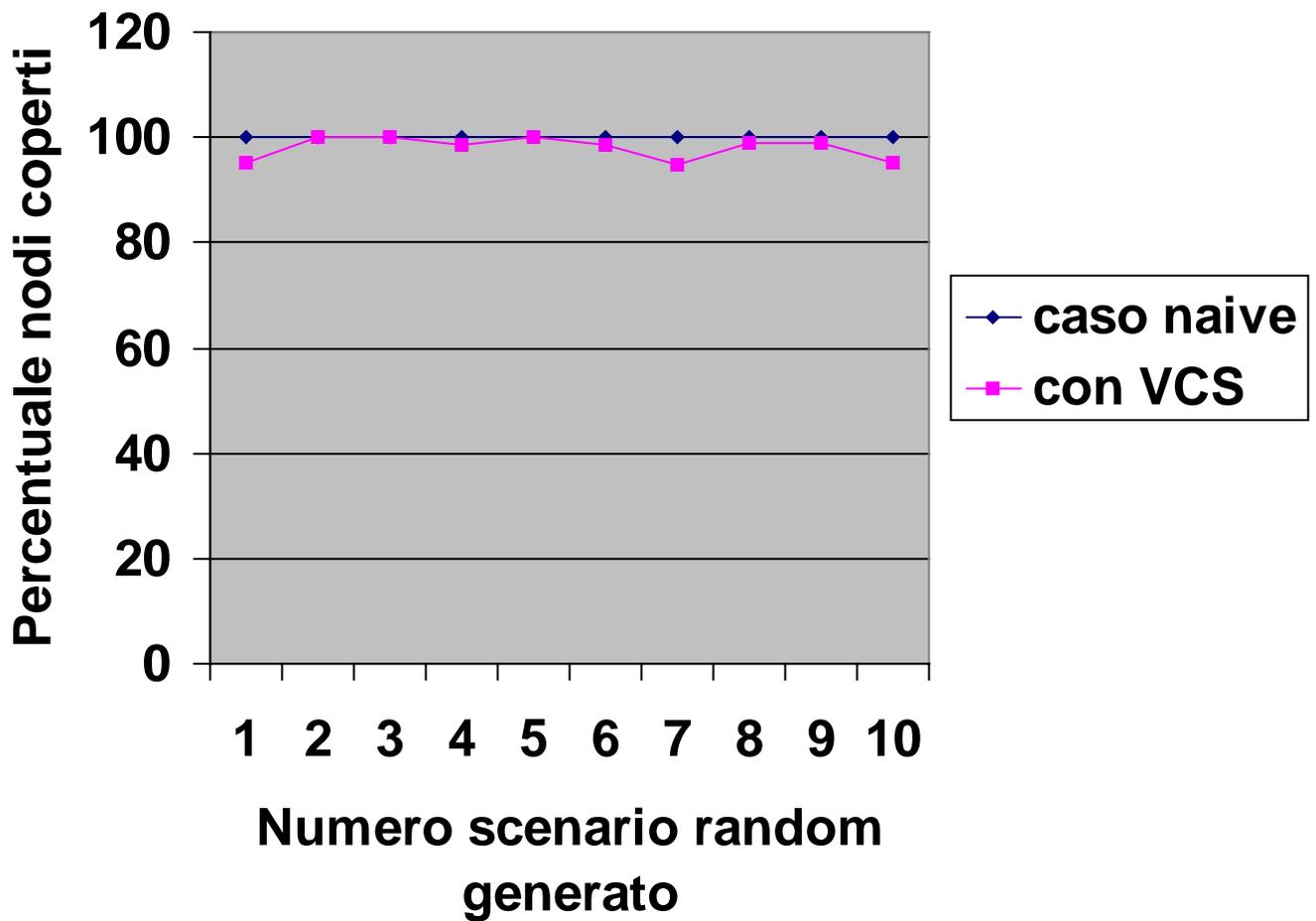


fig. 4.15 – Confronto tra VCS e caso naive: percentuale di nodi coperti

Nonostante l'elevato risparmio di risorse raggiunto, come mostrato in fig. 4.15, la copertura conseguita è molto elevata: quasi sempre superiore al 95%, in alcuni casi equivalente addirittura al 100% del caso naive senza VCS.

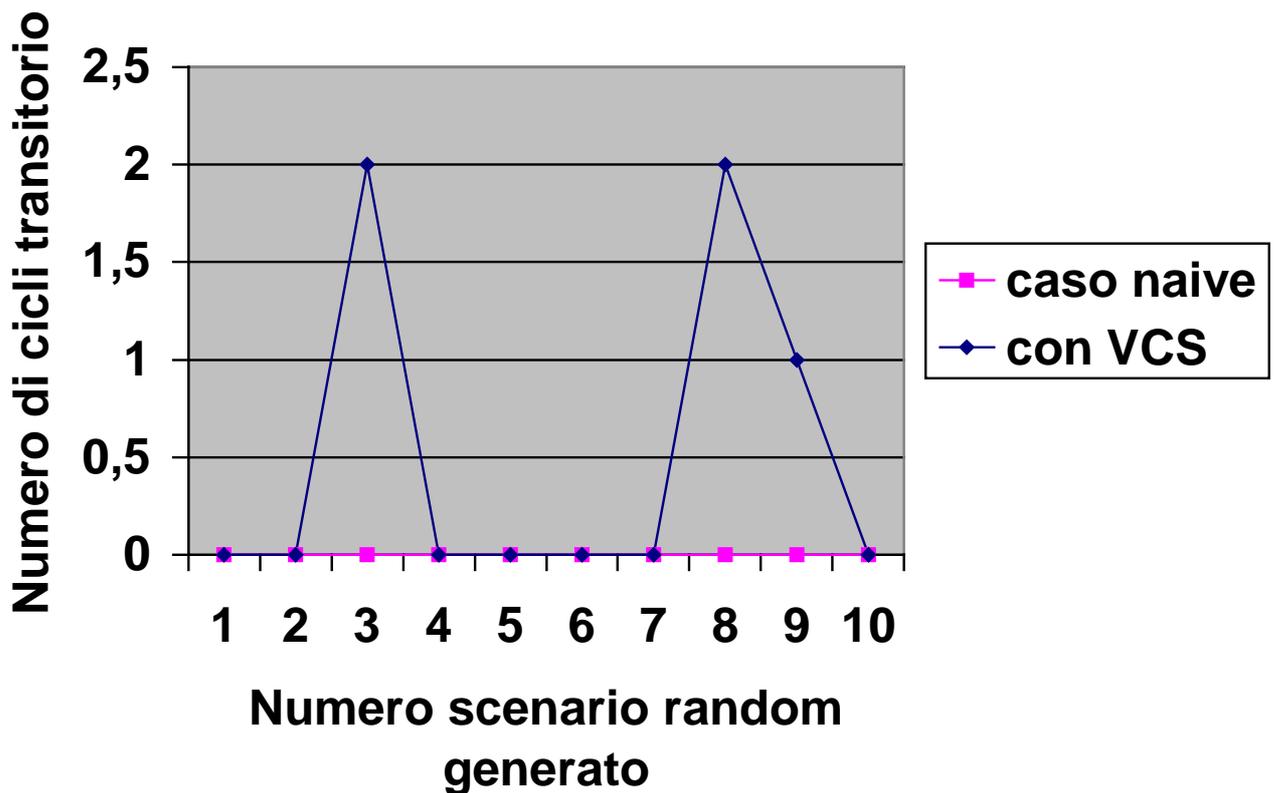


fig. 4.16 – Cicli di esecuzione impiegati dai VCS per raggiungere una configurazione stabile

Come è possibile notare dal grafico in fig. 4.16, solo in pochi casi è stato necessario un transitorio prima di raggiungere la configurazione definitiva. È inoltre necessario osservare come tali transitori coinvolgano in ogni caso solo zone piccole (ovvero pochi LME) rispetto alla dimensione globale della rete e come di conseguenza non creino particolari problemi di spreco di risorse.

È inoltre possibile fare un discorso analogo per quanto concerne la valutazione delle performances inerenti la mobilità dei nodi: provando a generare movimenti dei nodi casuali la rete torna nuovamente in uno stato stabile dopo una riconfigurazione che dura pochi cicli (tipicamente uno o due al massimo).

CONCLUSIONI

Lo sviluppo di servizi collaborativi avanzati in scenari MANET solleva problemi complessi che richiedono l'adozione di nuove linee guida per il design e l'implementazione di servizi collaborativi.

In questa tesi è stato introdotto AGAPE, un sistema di gestione dei gruppi context-aware.

In particolare, in questa tesi sono stati mostrati il progetto e l'implementazione del View Controlling Service di AGAPE, un servizio che consente il coordinamento delle diverse entità di AGAPE al fine di effettuare la gestione dei gruppi tenendo conto di varie informazioni di contesto.

Il servizio è stato testato per via simulativa in diversi scenari applicativi ottenendo risultati incoraggianti che possono stimolare ulteriori attività di ricerca.

Possibili linee di sviluppo futuro sono costituite dai miglioramenti dell'algoritmo di razionamento delle risorse, considerando anche altri aspetti, quali per esempio i profili degli utenti e dei dispositivi stessi.

RIFERIMENTI

R. Bruno, M. Conti, and E. Gregori, *WLAN Technologies for mobile Ad Hoc Networks*, Proc. HICSS-34, Maui, January 3-6, 2001;

M. Conti and S. Giordano, Special issue on “Mobile Ad Hoc Networking”, *Cluster Computing Journal*, 5 (2), April 2002;

M. Conti, E. Gregori, and L. Lanzini, Metropolitan Area Networks, Sprinter Limited Series on Telecommunication Network and Computer System, Novembre 1997;

M.S. Corson, J.P. Maker, and J.H. Cornicione Internet-based Mobile Ad Hoc Networking, *IEEE Internet Computing*, July-August 1999, pp.63-70;

Gerd Kortuem, Proem: A Middleware Platform for Mobile Peer-to-Peer Computing, *Mobile computing and communications reviews*, 6:62-64, 2002;

W. Stallings, *Local & Metropolitan Area Networks*, Prentice Hall, New York, 1996;

K. Van Dam, S. Pitchers, and M. Barnard, *Body Area Networks: Toward a Wearable Future*, Proc Wireless Word Research Forum, Munich, March 2001;

K. Van Dam, S. Pitchers, and M. Bernard, *From PAN to BAN: Why Body Area Networks*, Proc. Wireless Word Research Forum, Munich, March 2001;

D. Bottazzi, et. al., *AGAPE: a Location-aware Group Membership Middleware for Pervasive Computing Environments*, in Proc. of the 8th IEEE International Symposium on Computers and Communication (ISCC 2003), IEEE Press, Turkey, July 2003.