Routing in MANET

Stefano Tebaldi
(stefano.tebaldi@unimi.it)

Reti di Calcolatori II – A. A. 2003-2004

Classificazione

- Topology-based
  - Flat
    - Proactive (table-driven)
    - Reactive (on-demand)
  - Hierarchical/Hybrid
- Position-based
  - Geographic forwarding
  - Restricted directional flooding
  - Hierarchical/Hybrid

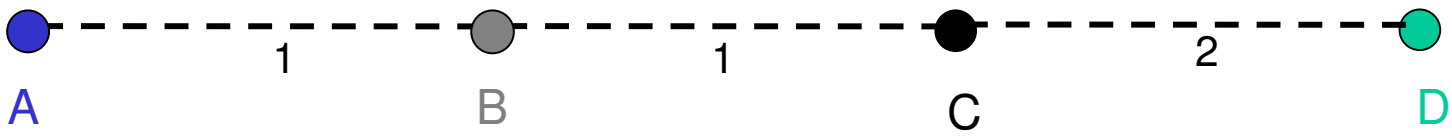
Proactive Routing Protocol

- Derivano dai classici Distance Vector e Link State
- Modificati per adattarli alle caratteristiche delle MANET:
  - Frequenti cambiamenti nella topologia
  - Risorse scarse (batterie, banda, memoria)
- Algoritmi di questo tipo: DSDV, OLSR, TBRPF

Distance Vector Routing

- Ogni router mantiene per ogni altro router:
  - Next hop
  - Metrica (numero di hop, ritardo, qualita' del percorso)
- Aggiornamento delle informazioni di routing tramite scambio dei "vettori di distanza" con i vicini
- Nel determinare la nuova tabella di routing non si tiene conto della vecchia
- **Problema:** soffre del count to infinity

Distance Vector - Tabelle di routing



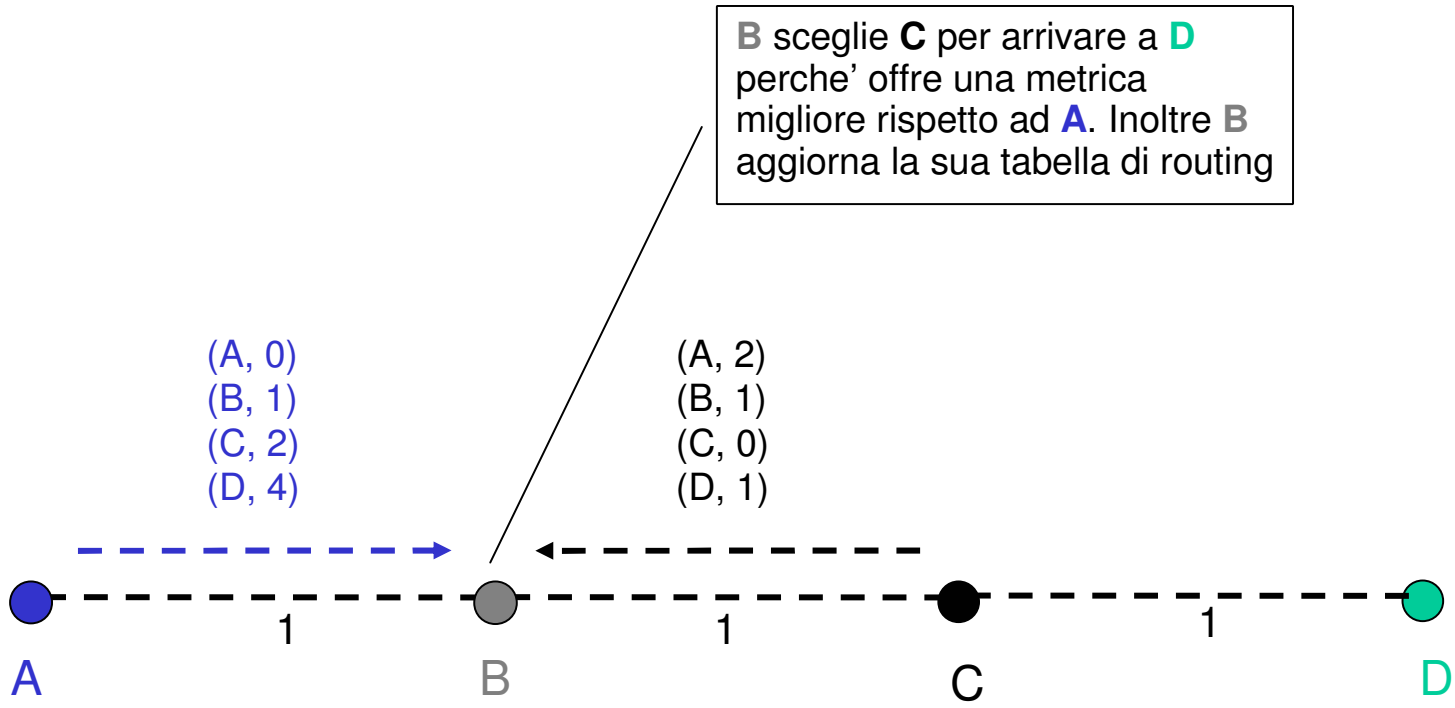
Dest.	Next	Metric
A	A	0
B	B	1
C	B	2
D	B	4

Dest.	Next	Metric
A	A	1
B	B	0
C	C	1
D	C	3

Dest.	Next	Metric
A	B	2
B	B	1
C	C	0
D	D	2

Dest.	Next	Metric
A	C	4
B	C	3
C	C	2
D	D	0

Distance Vector – Route Update



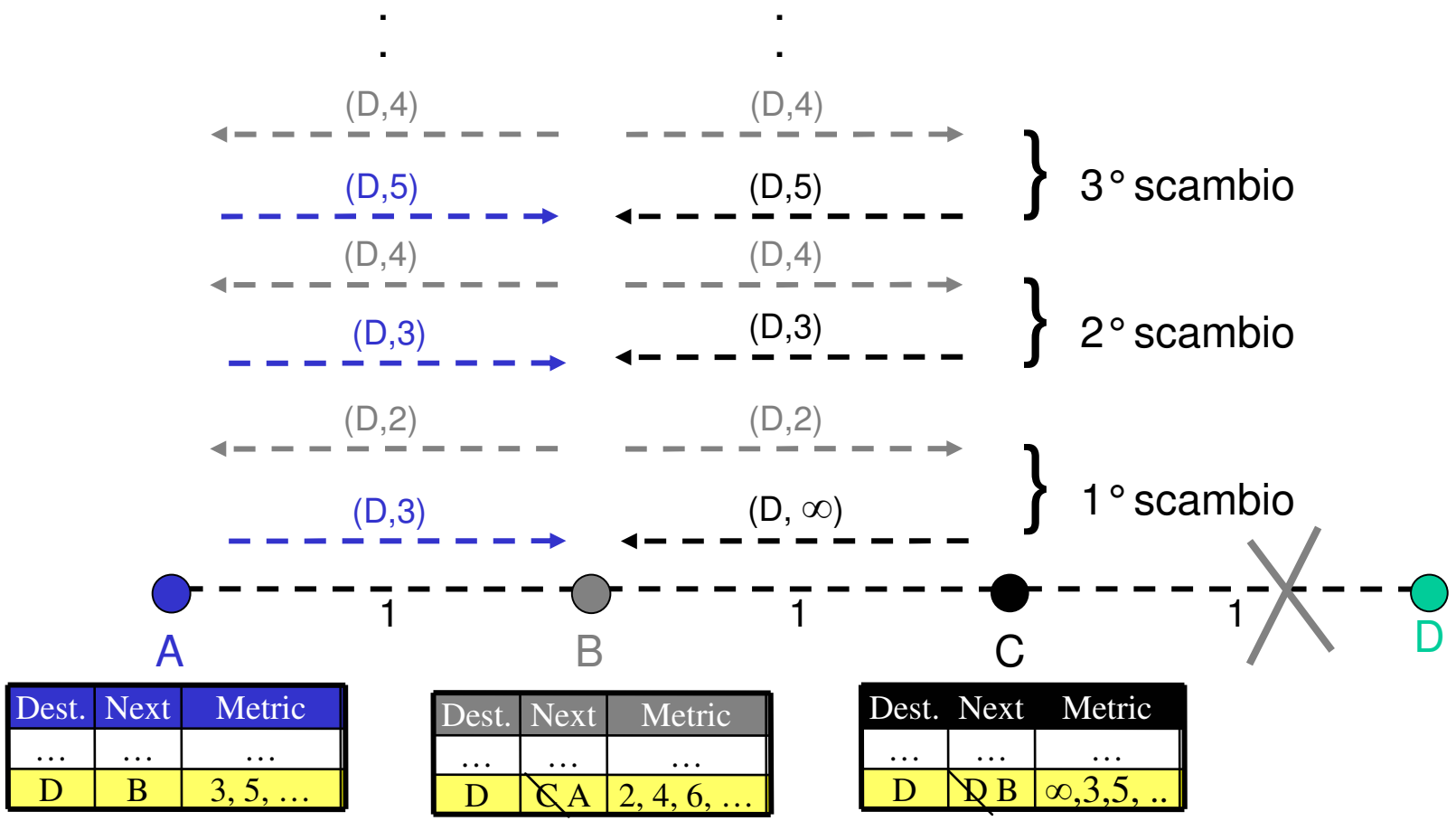
Dest.	Next	Metric
A	A	0
B	B	1
C	B	2
D	B	4

Dest.	Next	Metric
A	A	1
B	B	0
C	C	1
D	C	~~3~~ 2

Dest.	Next	Metric
A	B	2
B	B	1
C	C	0
D	D	1

Dest.	Next	Metric
A	C	3
B	C	2
C	C	1
D	D	0

Distance Vector - Count to Infinity



Destination Sequenced Distance Vector (DSDV)

- Aggiunta di un Destination Sequence Number alle informazioni di routing

- due tipi di route advertisement:
  - ***full dump***, in cui vengono inviate tutte le informazioni di routing
  - ***incremental***, in cui vengono inviate solo le entry che sono state modificate

- Invio immediato oltre che periodico degli aggiornamenti per adattarsi velocemente ai cambiamenti della topologia

DSDV - Table Entry

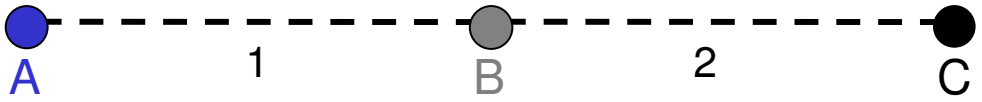
Destination	Next	Metric	Seq. Nr	Install Time	Stable Data
A	A	0	A-550	001000	Ptr_A
B	B	1	B-102	001200	Ptr_B
C	B	3	C-588	001200	Ptr_C
D	B	4	D-312	001200	Ptr_D

- **Sequence Number** creato dalla destinazione. Permette di non avere loop ed evitare il problema del count to infinity
- **Install Time** indica quando e' stata creata una entry (usato per eliminare entry scadute)
- **Stable Data** puntatore ad una tabella contenente una stima del tempo necessario ad un percorso verso la destinazione per essere considerato stabile. Usato per evitare il fenomeno delle fluttuazioni nella rete

DSDV - Destination Sequence Number

- Un nodo aggiorna il **proprio** Sequence Number (usando solo valori pari) ogni volta che invia un update
- Se un nodo si accorge di non poter piu' raggiungere un proprio vicino modifica tutte le entry che hanno il vicino come next hop, incrementandone il Sequence Number (che diventa dispari) e impostando la metrica a ∞
- Un nodo aggiorna la entry per una certa destinazione se riceve un update con un Destination Sequence Number maggiore. A parita' di Sequence Number si sceglie la metrica migliore

DSDV - Tabelle di routing



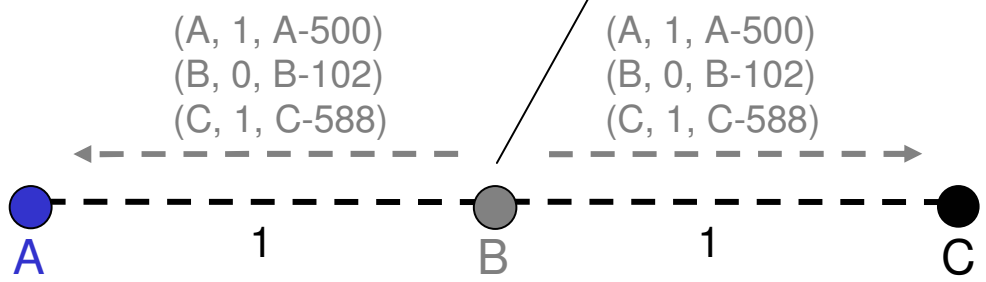
Dest.	Next	Metric	Seq
A	A	0	A-550
B	B	1	B-100
C	B	3	C-586

Dest.	Next	Metric	Seq
A	A	1	A-550
B	B	0	B-100
C	C	2	C-588

Dest.	Next	Metric	Seq.
A	B	1	A-550
B	B	2	B-100
C	C	0	C-588

DSDV – Route Update

B incrementa il proprio Seq. Num. da 100 a 102
B invia le proprie informazioni di routing ai vicini con il nuovo Seq. Num.

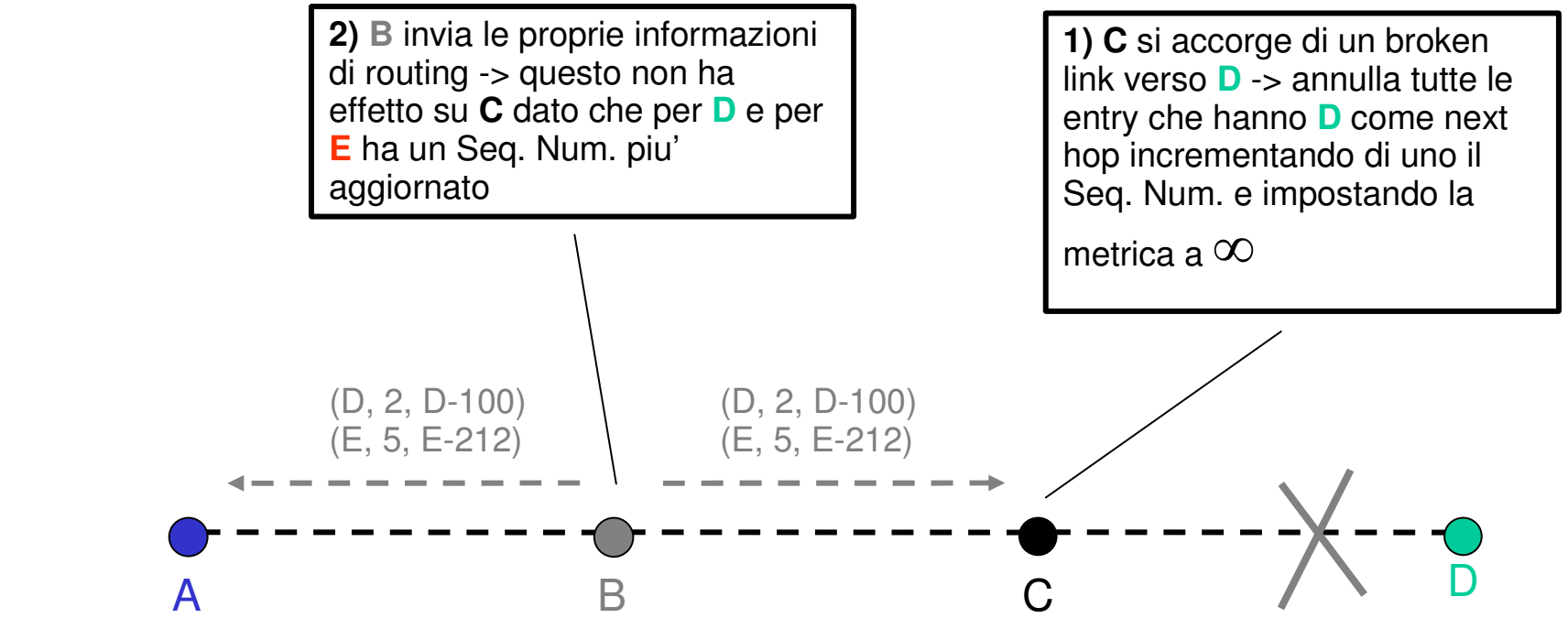


Dest.	Next	Metric	Seq
A	A	0	A-550
B	B	1	B-102
C	B	2	C-588

Dest.	Next	Metric	Seq
A	A	1	A-550
B	B	0	B-102
C	C	1	C-588

Dest.	Next	Metric	Seq.
A	B	2	A-550
B	B	1	B-102
C	C	0	C-588

DSDV - no loop, no count to infinity

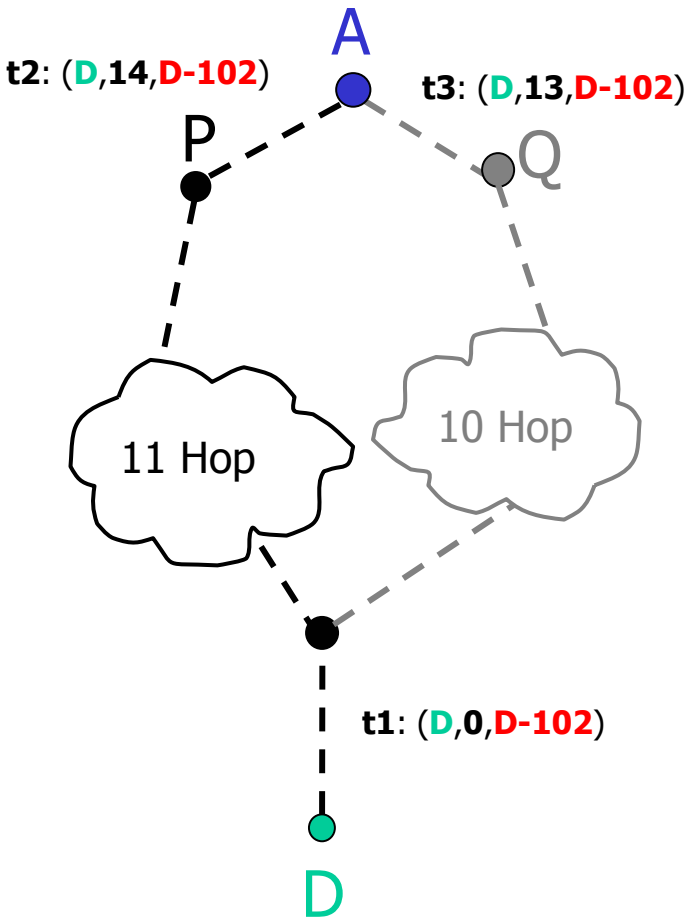


Dest.	Next	Metric	Seq.
…	…	…	…
D	B	3	D-100
E	B	6	E-212

Dest.	Next	Metric	Seq.
…	…	…	…
D	C	2	D-100
E	C	5	E-212

Dest.	Next	Metric	Seq
…	…	…	…
D	D	∞	D-101
E	D	∞	E-213

DSDV - Problema delle fluttuazioni



t0: A ha una entry per D con **D-100** come Seq. Num.

t1: D invia un update con Seq. Num. aggiornato (**D-102**)

t2: A riceve da **P** l'aggiornamento (D, **14**, **D-102**) -> la entry per D in A sara': [D, **P**, **15**, **D-102**]
A propaga questa informazione immediatamente

t3: A riceve da Q l'aggiornamento (D, **13**, **D-102**) -> la entry per D in A diventa: [D, Q, **14**, **D-102**]
A propaga questa informazione immediatamente

Problema: la prima route advertisement poteva essere evitata!!!

Soluzione: la tabella **stable data** contiene per ogni destinazione una stima del tempo (settling time) necessario a far si che un percorso per la destinazione possa essere considerato il migliore. Un nodo aggiorna la propria routing table, ma aspetta 2 * settling time prima di propagarla

Reactive Routing Protocol

- Si evita che ogni nodo conosca e mantenga un percorso per ogni altro nodo della rete
- Le informazioni di routing vengono create e mantenute solo quando serve (da qui reactive/on-demand)
- Un nodo mantiene informazioni di routing solo per le comunicazioni a cui partecipa e solo per la loro durata
- Algoritmi di questo tipo: AODV, DSR, TORA

Reactive routing: schema generale

- Quando un nodo sorgente (S) deve comunicare con una destinazione (D) e non ha per essa informazioni di routing inizia il processo di Route Discovery:
  - Una Route Request (RREQ) contenente l'identificativo di D viene inviata nella rete in broadcast
  - Qualsiasi nodo che ha informazioni di routing valide per D puo' rispondere con una Route Reply (RREP)

- Durante la fase di Route Discovery vengono create le informazioni di routing necessarie a permettere la comunicazione fra S e D

- Se un nodo partecipante al percorso che va da S a D si accorge di un link failure sul next hop, avverte S inviandole una Route Error (RERR)

Ad hoc On Demand Distance Vector (AODV)

- Puo' essere visto come la versione reactive di DSDV
- Le informazioni di routing mantenute sono praticamente le stesse
- Le entry nella routing table di un nodo vengono create o aggiornate in base alle RREQ e alle RREP che un nodo vede passare
- L'aggiornamento viene fatto sempre in base al valore del Destination Sequence Number e della metrica (numero di hop)

AODV - aggiornamento routing table

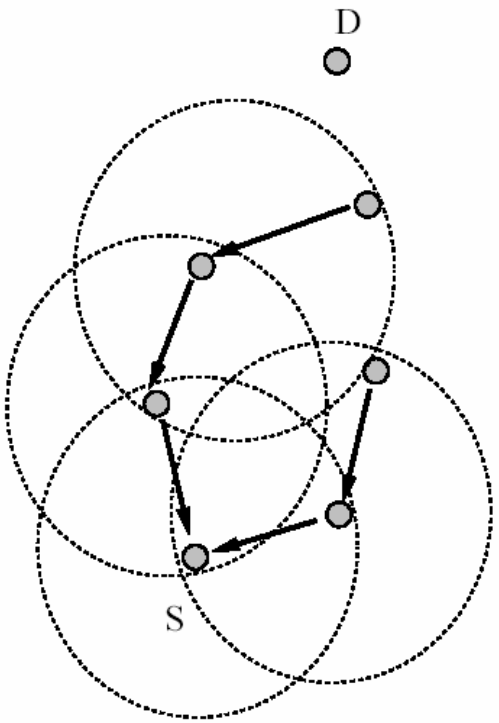


Figure 1. Reverse Path Formation

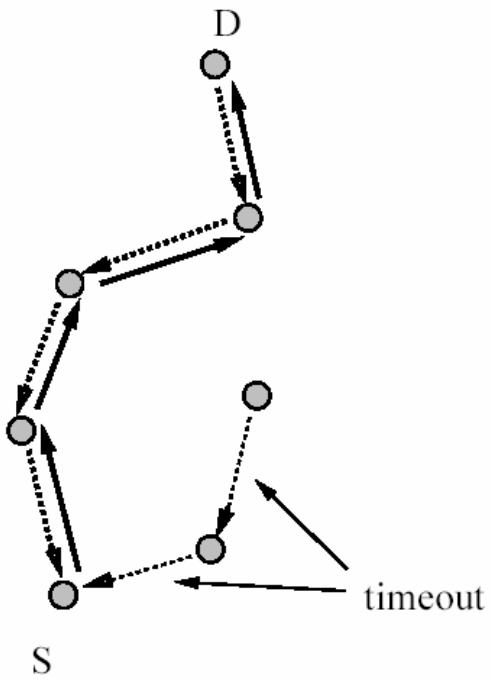


Figure 2. Forward Path Formation

1) La ricezione di una RREQ permette ad un nodo di creare/aggiornare l'informazione di routing per S

Una RREQ contiene:
<source_addr, source_seq_num, broadcast_ID, dest_addr, dest_seq_num, hop_count>

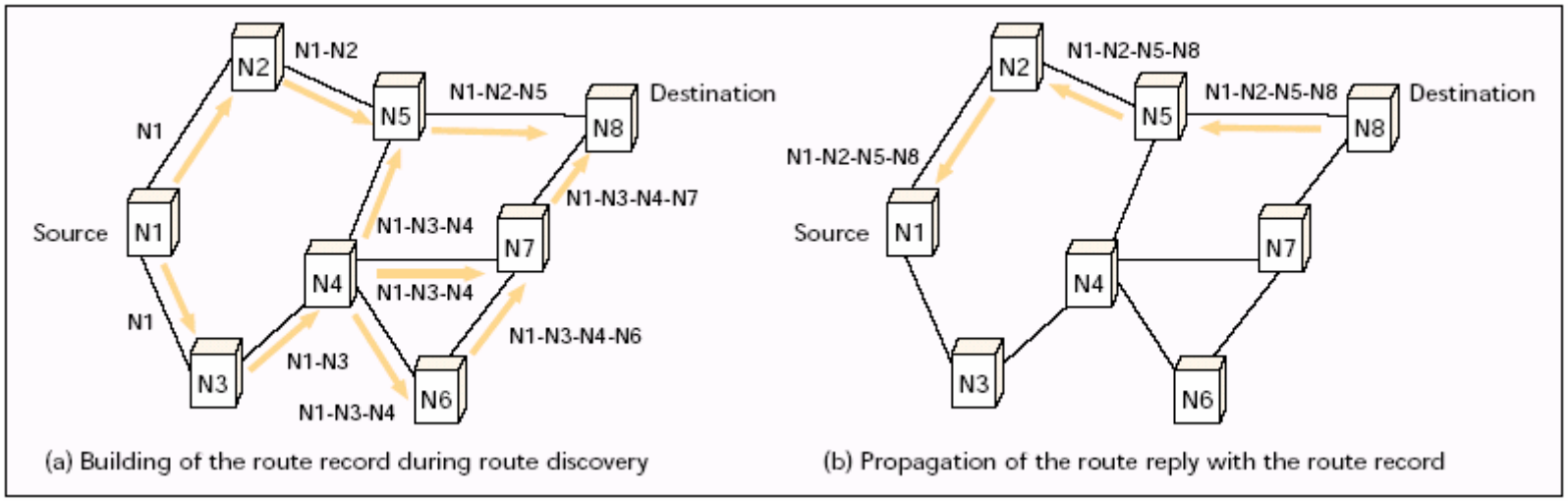
2) La ricezione di una RREP permette ad un nodo di creare/aggiornare l'informazione di routing per D

Una RREP contiene:
<source_addr, dest_addr, dest_seq_num, hop_count, lifetime>

Dynamic Source Routing (DSR)

- Le RREQ, viaggiando per la rete in cerca di un percorso per la destinazione, memorizzano gli ID dei nodi attraversati
- Quando un nodo risponde ad una RREQ, nella RREP inserisce il percorso memorizzato dalla RREQ stessa (completandolo nel caso di un nodo intermedio)
- Per inviare una RREP alla sorgente un nodo puo':
  - Usare il percorso trovato (se i link sono bidirezionali)
  - Usare un percorso che conosce (route cache)
  - Eseguire a sua volta una RREQ per la sorgente (problema ricorsioni infinite)
- Ricevuta la RREP, la sorgente memorizza il percorso trovato in una route cache e lo inserisce nell'header dei pacchetti di dati da inviare alla destinazione (source routing)

DSR - esempio



Una RREQ puo' generare piu' RREP, questo permette ad un nodo di collezionare piu' percorsi alternativi verso una destinazione.

I percorsi presenti nella route cache hanno una validita' limitata, allo scadere di un expiration time vengono scartati

DSR - Route Cache

- Per migliorare le prestazioni del protocollo un nodo cerca di imparare piu' percorsi possibili
- Il percorso *[S,E,F,J,D]* per D permette a S di conoscere anche il percorso *[S,E,F]* per F
- Un nodo puo' conoscere nuovi percorsi dalle RREQ, dalle RREP e dai pacchetti di dati che instrada
- Un nodo puo' addirittura ascoltare le comunicazioni degli altri nodi (in modalita' promiscua) e da queste imparare ulteriori percorsi

DSR - analisi

- L'uso della route cache puo' permettere a DSR di limitare il piu' possibile l'uso della procedura di route discovery
- Di contro, i percorsi memorizzati possono diventare presto obsoleti
- Inserire il percorso nei pacchetti "ruba" spazio per i dati, aumenta l'occupazione di banda e limita la scalabilita' del protocollo a reti di piccole dimensioni
- La route cache puo' occupare molta memoria nei nodi

Proactive VS Reactive

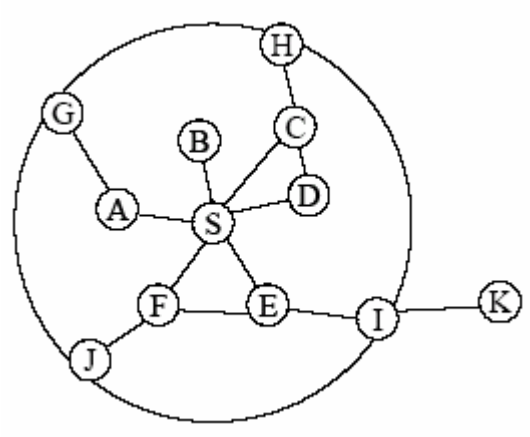
	Proactive	Reactive
Gestione informazioni di routing	Per ogni possibile destinazione e da aggiornare costantemente	Solo per le destinazioni per cui si partecipa alla comunicazione. Create e aggiornate quando serve
Costo sulla rete	Elevato. Cresce all'aumentare dei nodi e della mobilita'	Minore dei proactive. Cresce all'aumentare della mobilita' e del traffico
Occupazione di memoria nei nodi	Puo' essere elevata. Dipende dalla dimensione della rete	Minore dei proactive. Dipende dal numero di comunicazioni a cui un nodo partecipa
Ritardo nelle comunicazioni	No. Informazioni di routing subito disponibili	Sì. Dovuto alla fase di Route Discovery

Approccio Ibrido: Zone Routing Protocol (ZRP)

- ZRP e' costituito sia da una parte proactive che da una reactive
- L'obiettivo e' di combinarle insieme per sfruttarne i pregi limitandone il piu' possibile i difetti
- Ogni nodo mantiene informazioni di routing in "modalità" proactive solo per una certa zona della rete
- Il concetto di zona viene sfruttato anche per ottimizzare la ricerca in "modalità" reactive di una destinazione che risiede al di fuori della zona stessa

ZRP – Routing zone e Intrazone Routing

- La routing zone di un nodo e' l'insieme dei nodi la cui **minima** distanza in hop da esso e' <= del "raggio della zona"
- I nodi la cui distanza minima e' uguale al raggio sono chiamati nodi periferici
- All'interno della zona viene utilizzato un protocollo proactive chiamato IntrAzone Routing Protocol (IARP)

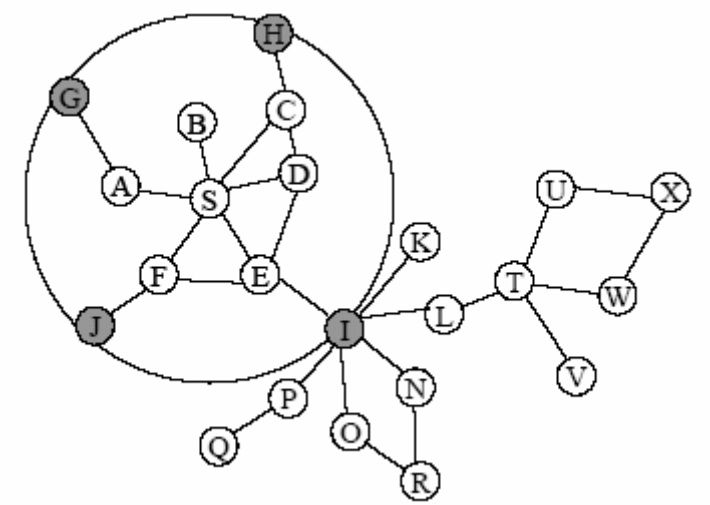


Routing Zone di raggio 2 per il nodo S

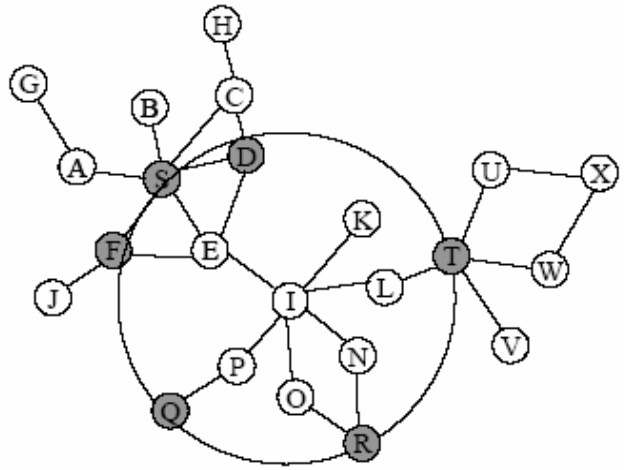
ZRP – Interzone Routing

- Nel caso in cui la destinazione non sia presente nella zona di un nodo si passa ad utilizzare la componente reactive, chiamata IntErzone Routing Protocol (IERP)
- Le RREQ non vengono inviate in broadcast, ma solo ai nodi periferici della zona (bordercasting)
- Dell'instradamento delle RREQ se ne occupa una componente chiamata Bordercast Resolution Protocol (BRP) che ottiene le informazioni sui nodi periferici dallo IARP
- Ricevuta una RREQ, un nodo periferico controlla che la destinazione sia nella sua zona e in caso affermativo invia alla sorgente la RREP, altrimenti reinstrada la RREQ ai propri nodi periferici

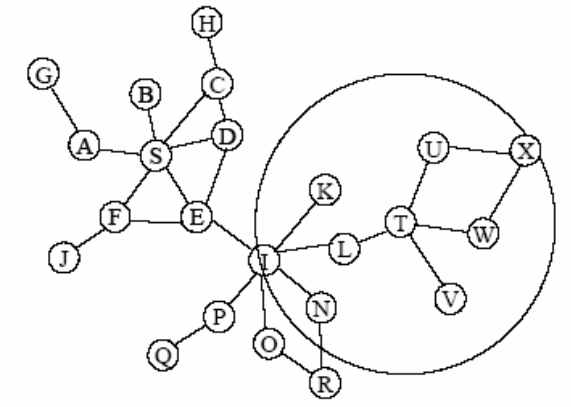
ZRP - esempio



1) S vuole comunicare con W.
Invia la RREQ ai propri nodi periferici



2) I non trova W nella sua zona
e quindi invia a sua volta la RREQ ai propri nodi periferici



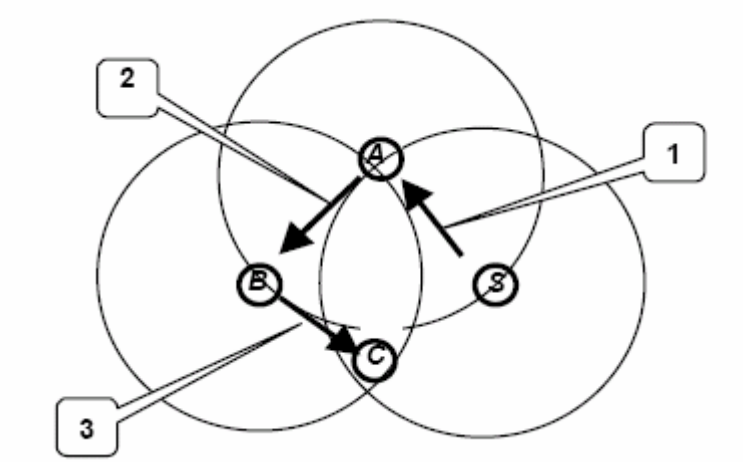
3) T si accorge che W e' nella sua zona e quindi invia ad S la RREP

ZRP - Query Control Mechanism

- Le zone dei vari nodi si possono sovrapporre
- Una RREQ puo' quindi ritornare in una zona che e' gia' stata "coperta"
- Per evitare questo problema tre meccanismi di controllo possono essere utilizzati:
  1) Loop-back Termination
  2) Early Termination
  3) Selective Bordercasting

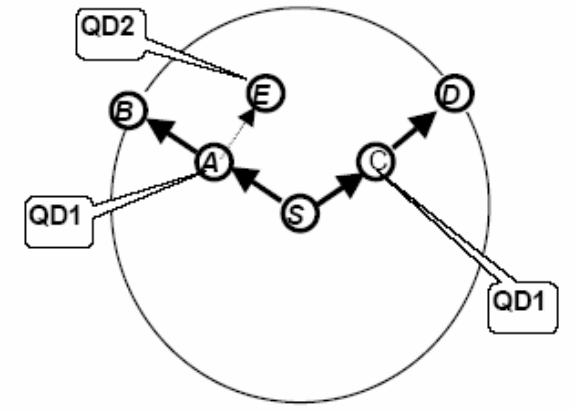
ZRP – Loop Back Termination

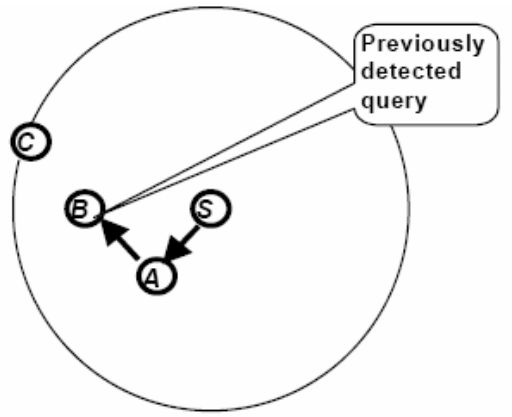
Nel caso in cui le RREQ accumulino i nodi attraversati come in DSR, un nodo periferico puo' controllare se qualcuno di questi nodi risiede nella sua routing zone. In caso affermativo scarta la RREQ



ZRP – Early Termination

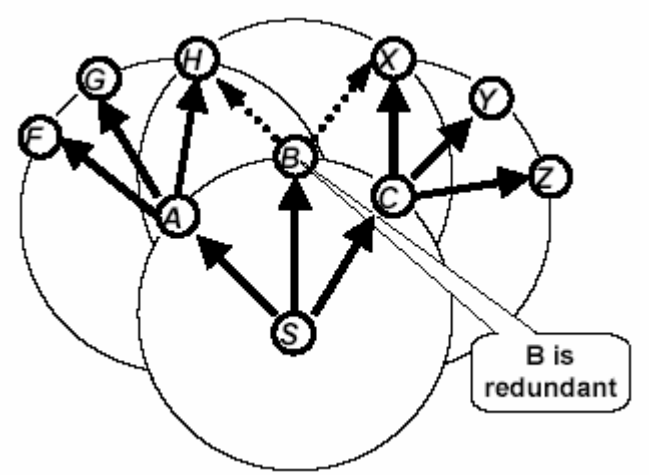
I nodi memorizzano le RREQ gia' instradate (identificate da <sorg_ID, RREQ_ID>) o "ascoltate" (in modalita' promiscua). Se vedono passare una RREQ gia' incontrata la scartano





ZRP – Selective Bordercasting

Tecnica utilizzata per prevenire il problema. Il raggio di azione dello IARP deve essere esteso ad una zona doppia di quella normale. Questo permette ad un nodo di conoscere i nodi periferici dei propri nodi periferici. Il nodo puo' così decidere quali dei propri nodi periferici scartare o indicare a questi quali dei loro nodi periferici non utilizzare



Costo del meccanismo:

1) estendendo la zona aumenta il costo dello IARP

2) l'indicazione dei nodi periferici da utilizzare e' inserita nella RREQ sprecando spazio

ZRP – Scelta del raggio della zona

- L'obiettivo e' di trovare una dimensione ottima che consenta di avere un traffico di controllo minimo
- Piu' aumenta il raggio, piu' aumenta il costo dello IARP (soprattutto per reti dense)
- Per lo IERP piu' le zone sono grandi e meglio e' (soprattutto al crescere della dimensione della rete)
- Se la topologia della rete cambia frequentemente e le comunicazioni sono occasionali, il costo maggiore e' dovuto alla parte proactive -> zone di raggio piu' piccolo
- Se la topologia della rete cambia frequentemente e le comunicazioni sono frequenti, il costo maggiore e' dato dalla parte reactive -> zone di raggio maggiore