



SAPIENZA
UNIVERSITÀ DI ROMA

Internet of Underwater Things

Prof.ssa Chiara Petrioli

Christian Cardia – Gabriele Saturni

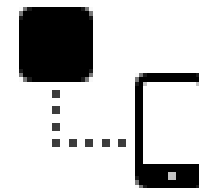
Che cosa significa
“Internet of Things”
?

Internet of Things

- Estensione di internet su differenti tipi di “oggetti” (ad esempio auto, elettrodomestici ecc..)
- Ad esempio i dati rilevati da appositi sensori possono essere scambiati attraverso l'utilizzo di internet
- Gli oggetti possono essere monitorati e gestiti da remoto
- Si pensi ad un insieme di dispositivi con dei sensori, in grado di misurare temperatura, qualità dell'aria, rumorosità ecc.
- Questi dispositivi, grazie all'IoT sono in grado di trasmettere i dati rilevati e di ricevere comandi (per il controllo remoto).
- I dispositivi che possono fare uso dell'IoT possono essere di qualsiasi tipo: piante dotate di sensori, pacemaker di pazienti ecc.
- Questi dispositivi devono soltanto avere due tipi di requisiti: un indirizzo IP per essere identificati e devono essere in grado di scambiare informazioni sulla rete

Un esempio ... la domotica

- Grazie all'IoT è possibile avere delle case “intelligenti”.
- Si possono controllare una serie di oggetti da remoto come l'illuminazione, attivare o spegnere elettrodomestici, regolare la temperatura dell'ambiente





IoUT

Internet of Underwater Things

IoUT – Perché è importante?

- Circa il 71% del pianeta è ricoperto dagli oceani.
- Le temperature degli oceani determinano il clima e i venti che influenzano la vita sulla terra.
- L'obiettivo principale dell'IoUT è quello di estendere l'Internet of Things nel mondo Underwater
- Grazie a questo è possibile monitorare costantemente gli oceani, gli inquinamenti (e non solo...).





IoUT – Applicazioni

- Campionamento e monitoraggio degli oceani
- Monitoraggio ambientale
- Esplorazione
- Prevenzione dei disastri ambientali
- Assistenza alla navigazione
- Sorveglianza distribuita
- Detenzione di mine
- Ecc...

Come possono comunicare gli “oggetti” in Underwater?

Onde radio

Si propagano a lunghe distanze soltanto a frequenze estremamente basse (30 – 300 Hz) -> richiedono enormi antenne molto costose che implicano un grande consumo energetico.

Con alte frequenze → corto raggio di trasmissione (ad esempio 2.4 Ghz usato dal WiF, circa 1 cm.) Soffrono il problema dell'attenuazione.

Onde ottiche

Trasmissione molto veloce (fino a 10Mbps) ma possono essere usate per comunicazioni a corto raggio (esempio 10 metri) e richiedono un'elevata precisione tra la sorgente e la destinazione (non semplice in underwater).

Soffrono il problema dello *scattering*.

Come possono comunicare gli “oggetti” in Underwater?

Onde Acustiche

Questa tecnologia consente una comunicazione a lungo raggio, fino a diversi chilometri. Nonostante questo, presenta comunque delle problematiche difficili da affrontare:

Basso Data Rate: 80bps – 64kbps

Ritardi di propagazione variabili e molto lunghi

Larghezza di banda molto bassa

Elevato BER (Bit Error Rate)

Una rete Acustica sottomarina è composta da un insieme di nodi che possono fungere da nodo sorgente, nodo destinazione o nodo relay.

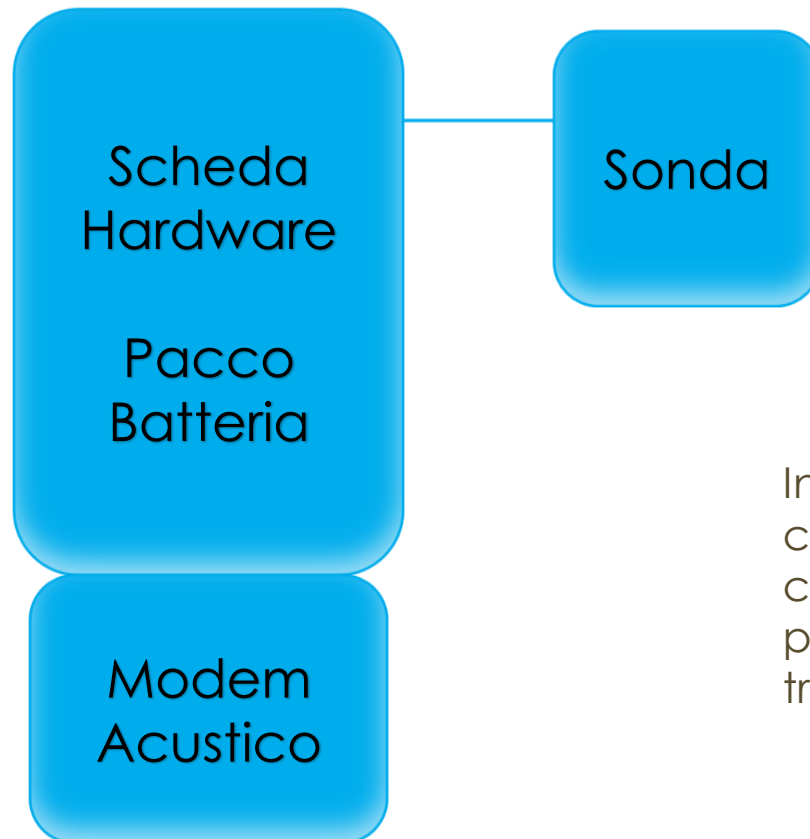
Cosa intendiamo con nodo?



All'interno di questa "scatola" è presente un hardware (che ad esempio potrebbe essere una scheda Arduino) con installati i protocolli da utilizzare (Applicazioni, protocolli di routing, mac ecc..) e un pacco batteria.

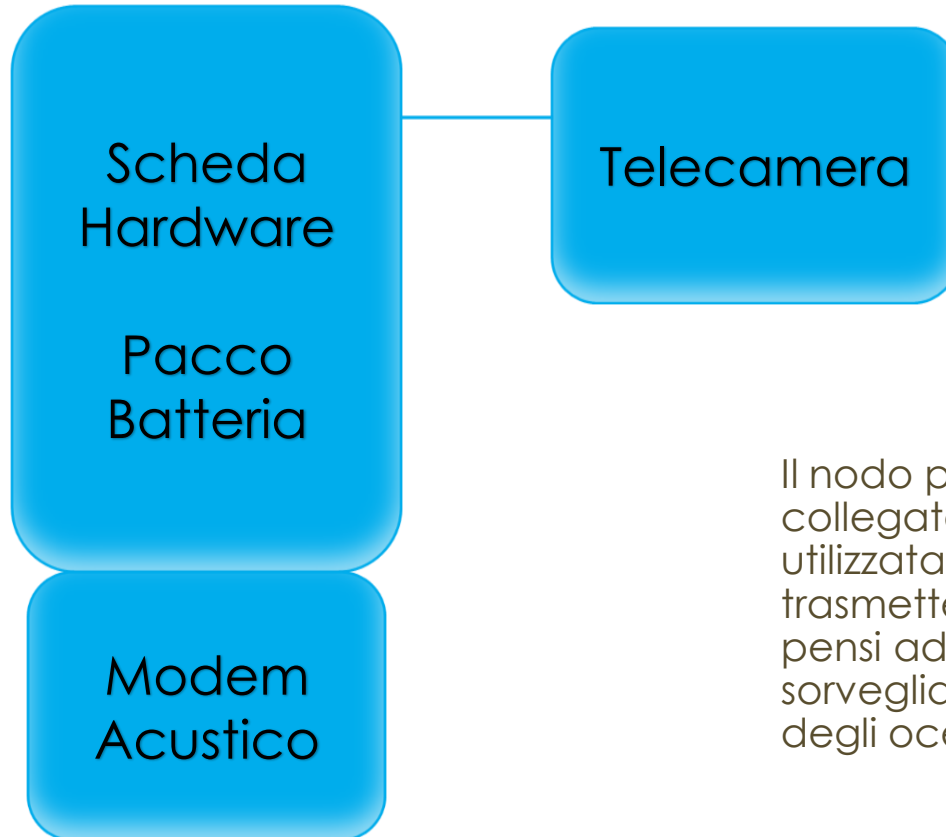
La "scatola" sopra descritta è collegata ad un modem acustico utilizzato per ricevere o inviare dati tramite onde acustiche.

Cosa intendiamo con nodo?



Inoltre il nodo potrebbe avere collegate una o più sonde che utilizza per raccogliere parametri ambientali e trasmetterli.

Cosa intendiamo con nodo?

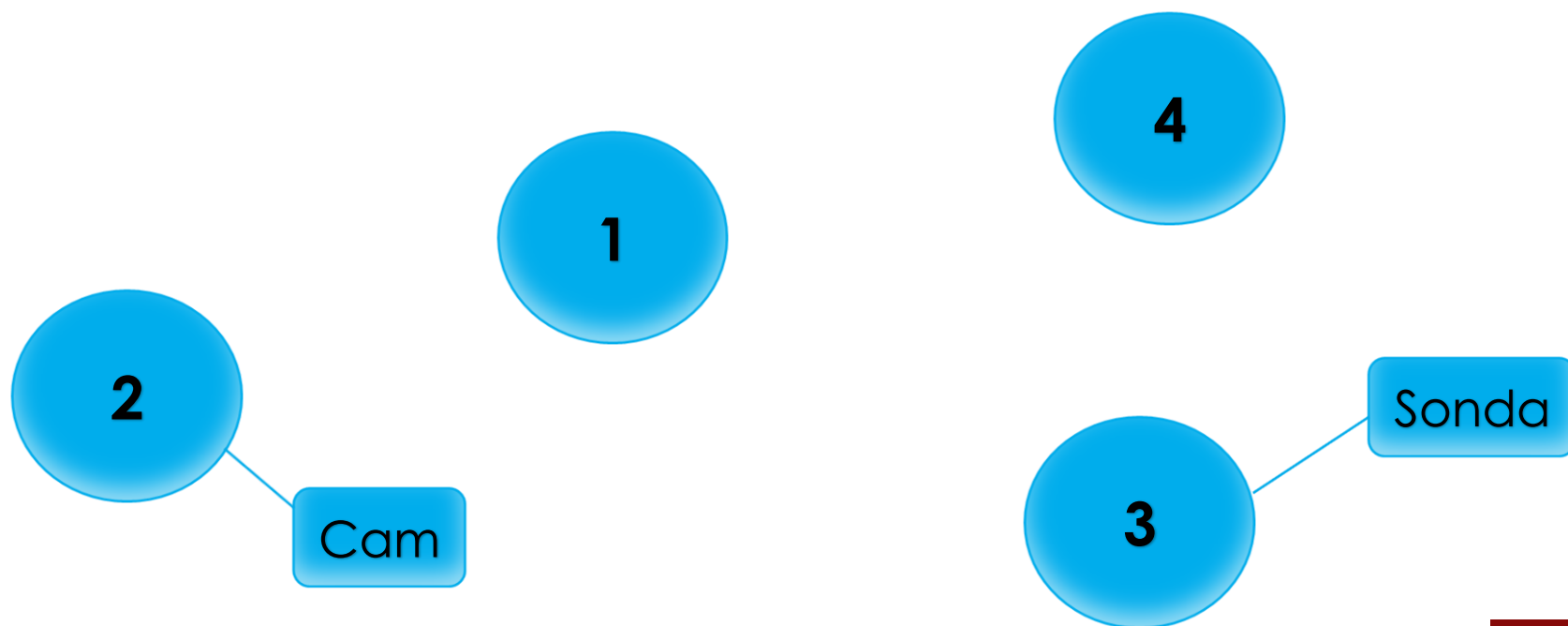


Il nodo potrebbe anche avere collegata una telecamera, utilizzata per scattare foto o trasmettere video in diretta (si pensi ad applicazioni di sorveglianza o di esplorazione degli oceani).

Un esempio di topologia

In questo esempio la rete è composta da 4 nodi, di cui uno con collegata una telecamera (nodo con id 2) ed un altro una sonda (id 3). Solitamente i dati raccolti, come ad esempio fotografie e/o misurazioni vengono trasmesse ad un nodo destinazione che potrebbe trovarsi in superficie ed essere in grado di comunicare con l'esterno.

Inoltre nella rete possono anche trovarsi nodi "relay" il cui unico compito è quello di inoltrare i pacchetti.



Perché è importante che i nodi trasmettano le informazioni anziché memorizzarle in locale?

- Possibilità di avere informazioni in tempo reale (altrimenti l'accesso ai dati avverrebbe soltanto al recupero degli strumenti).
- Possibilità di riconfigurazione
- Possibilità di rilevare un fallimento
- Missioni limitate dalla capacità di memoria

Underwater - Comunicazione Acustica

Un esempio di alcuni modem acustici - Evologics



Alcune problematiche sui protocolli di rete



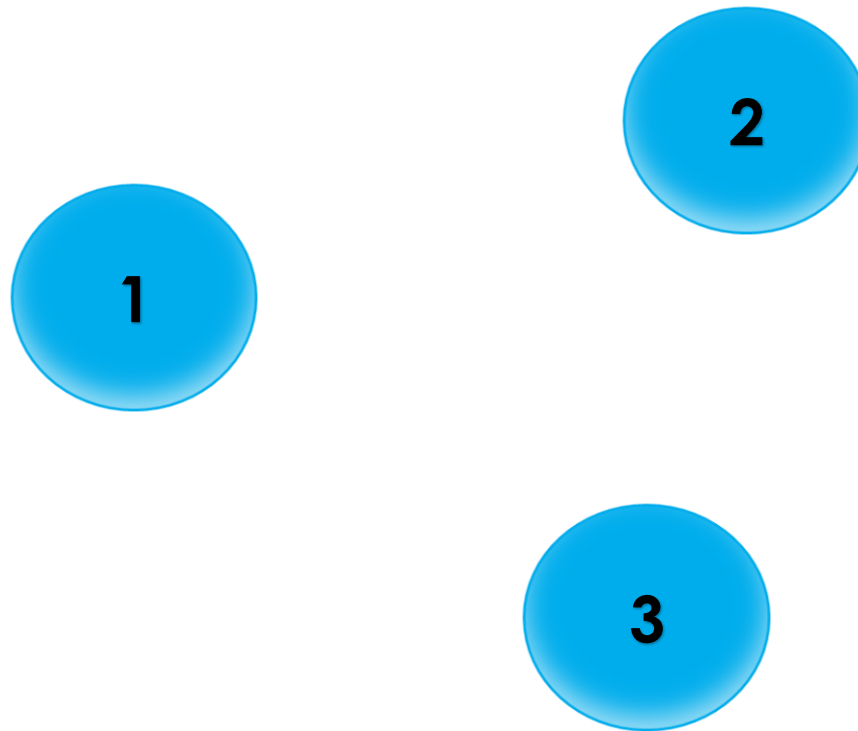
Rispetto alla comunicazione radio nel mondo Terrestre, la comunicazione Acustica presenta delle grandi differenze e degli svantaggi notevoli.

Questo comporta che le stesse soluzioni utilizzate nel mondo Terrestre non possono essere utilizzate.

Per tali motivi, è fondamentale sviluppare nuove soluzioni ad hoc che tengano conto di tutte le problematiche descritte sopra.

Livello MAC

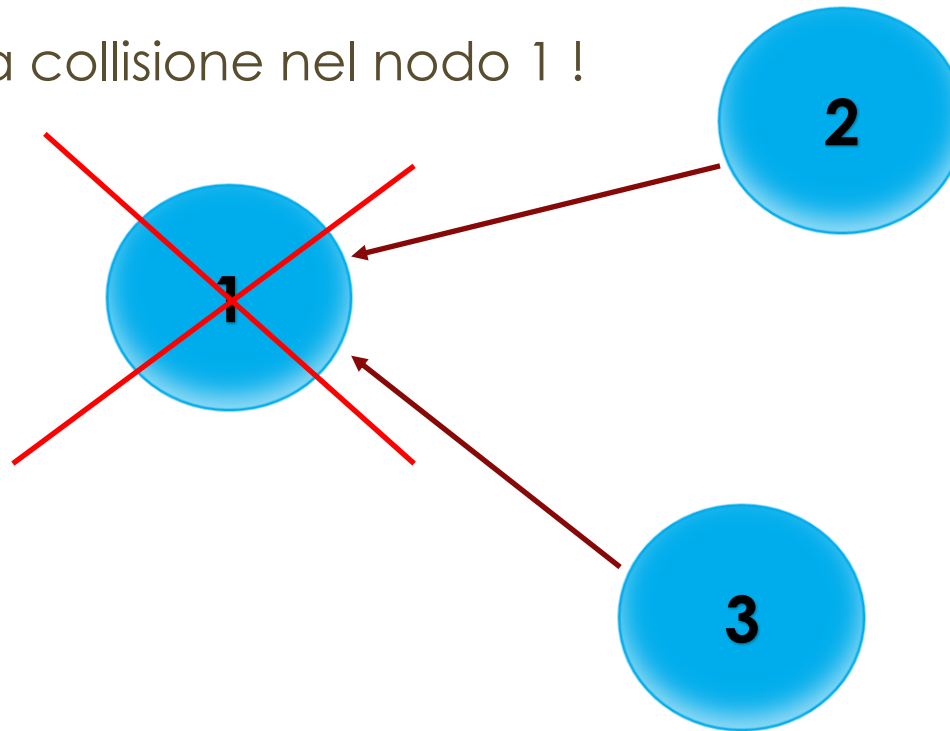
Che cosa succede se il nodo 2 e il nodo 3 trasmettono un pacchetto dati allo stesso istante al nodo 1?



Livello MAC

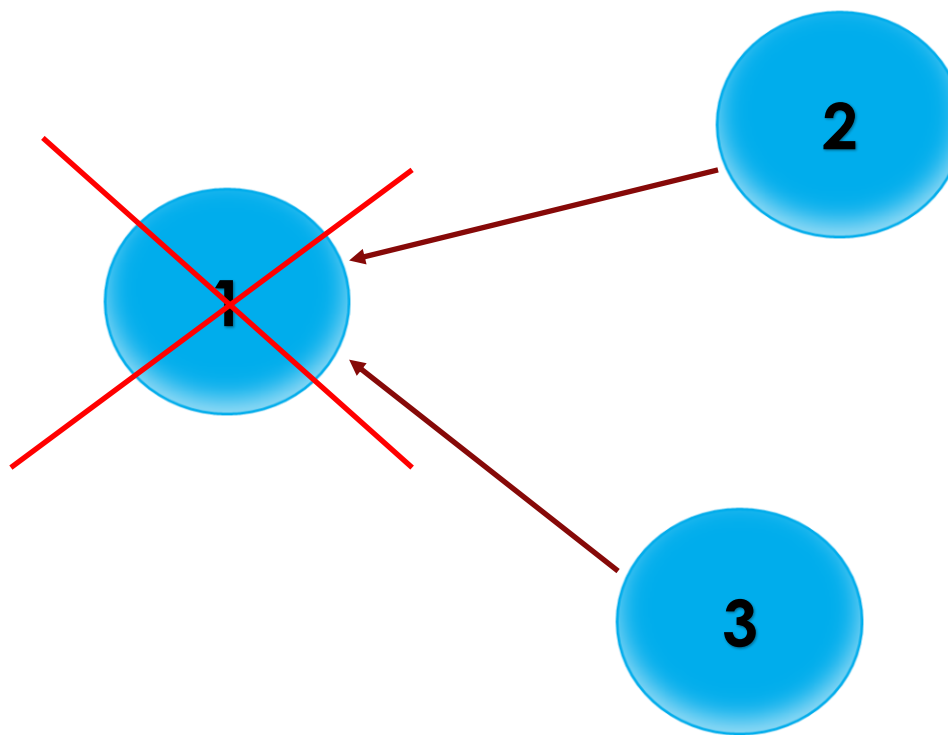
Che cosa succede se il nodo 2 e il nodo 3 trasmettono dati allo stesso istante?

Avviene una collisione nel nodo 1 !



Livello MAC

Che cosa significa collisione?



Che cosa significa collisione?

I nodi della rete condividono il canale trasmissivo. Se non utilizzano nessun meccanismo per accedere al canale e quindi trasmettono allo stesso istante, il nodo con id 1 subisce una collisione, ovvero riceve contemporaneamente i due segnali e non riesce ad interpretarli.

Questo richiede una nuova trasmissione da entrambi i nodi.

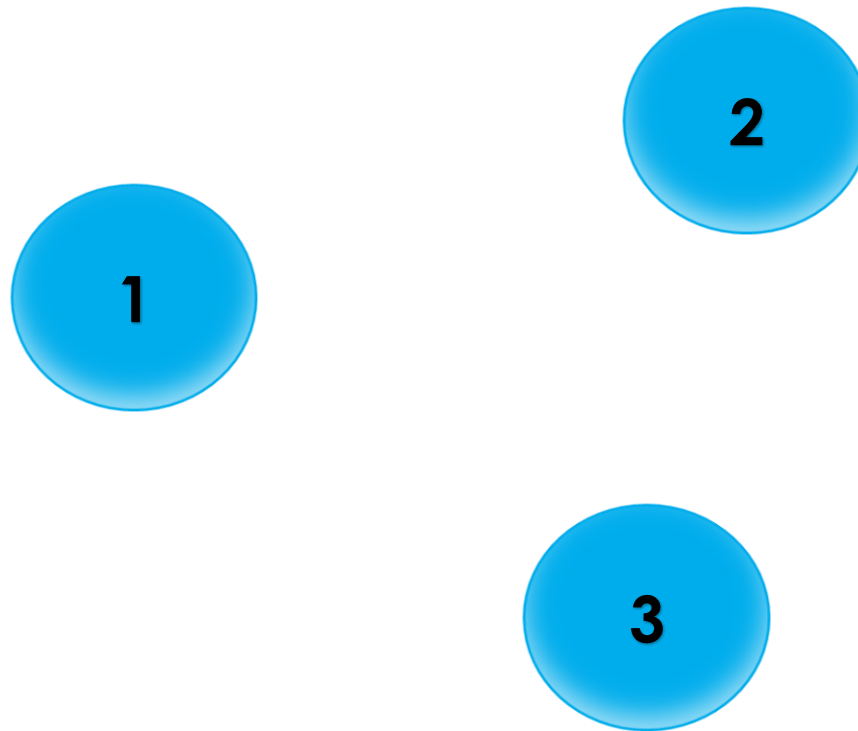
Livello MAC

I protocolli MAC (Medium Access Control) sono dei protocolli appartenenti al livello Dati dello stack protocollare.

Hanno l'obiettivo di evitare le collisioni o meglio, concordare l'accesso al canale.

Livello MAC

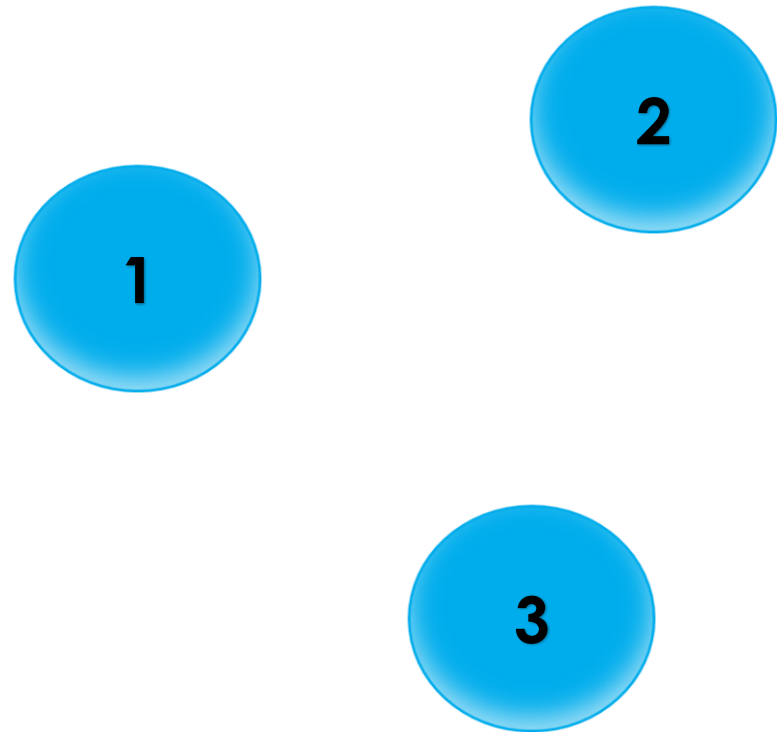
Tornando al problema precedente, ovvero i nodi 2 e 3 devono trasmettere un pacchetto dati, come possono evitare una collisione?



MAC – CSMA (Carrier Sensing Multiple Access)

L'idea alla base del Csma è: ascolta prima di trasmettere, se trovi il canale occupato riprova più tardi.

Esiste anche la variante Csma:cd (collision detection): in questo caso, mentre un nodo sta trasmettendo, può ascoltare rapidamente il canale e rilevare un'altra trasmissione: in tal caso provvede immediatamente a bloccare la trasmissione.

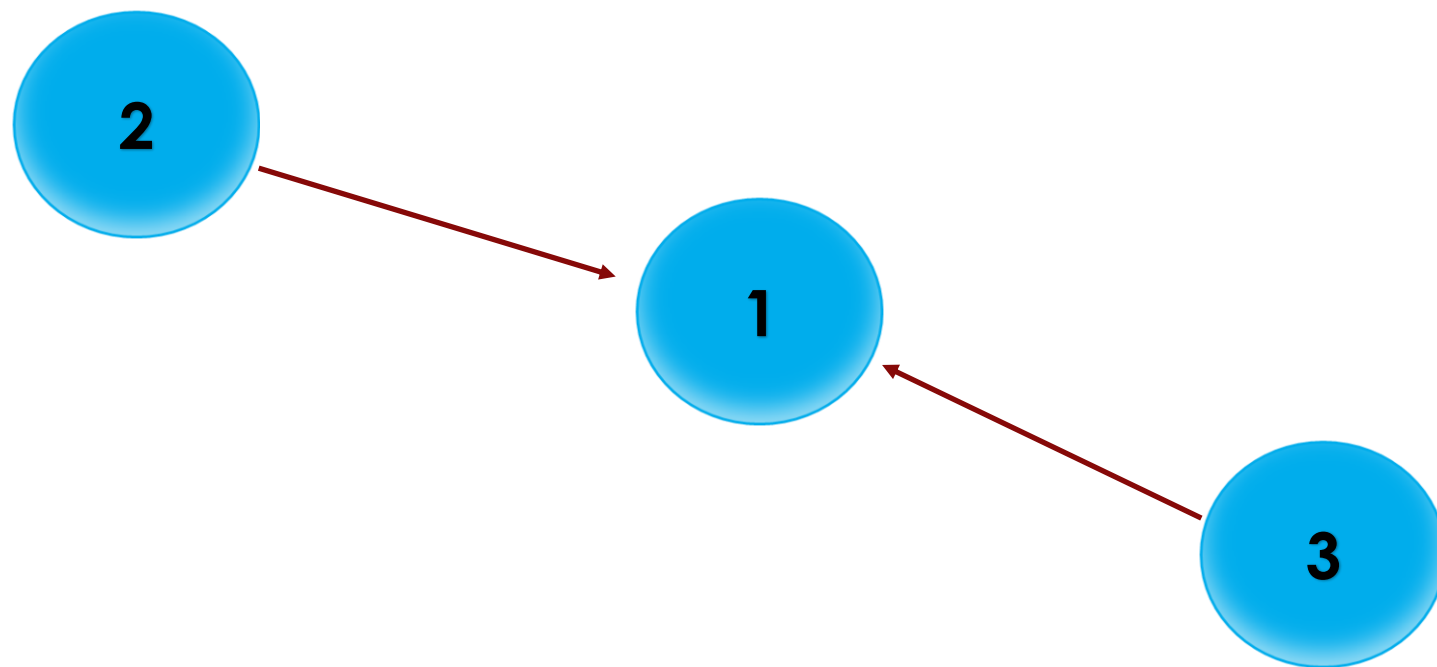


MAC – CSMA (Carrier Sensing Multiple Access)

Utilizzando il Cdma, quali problemi potrebbero esserci?

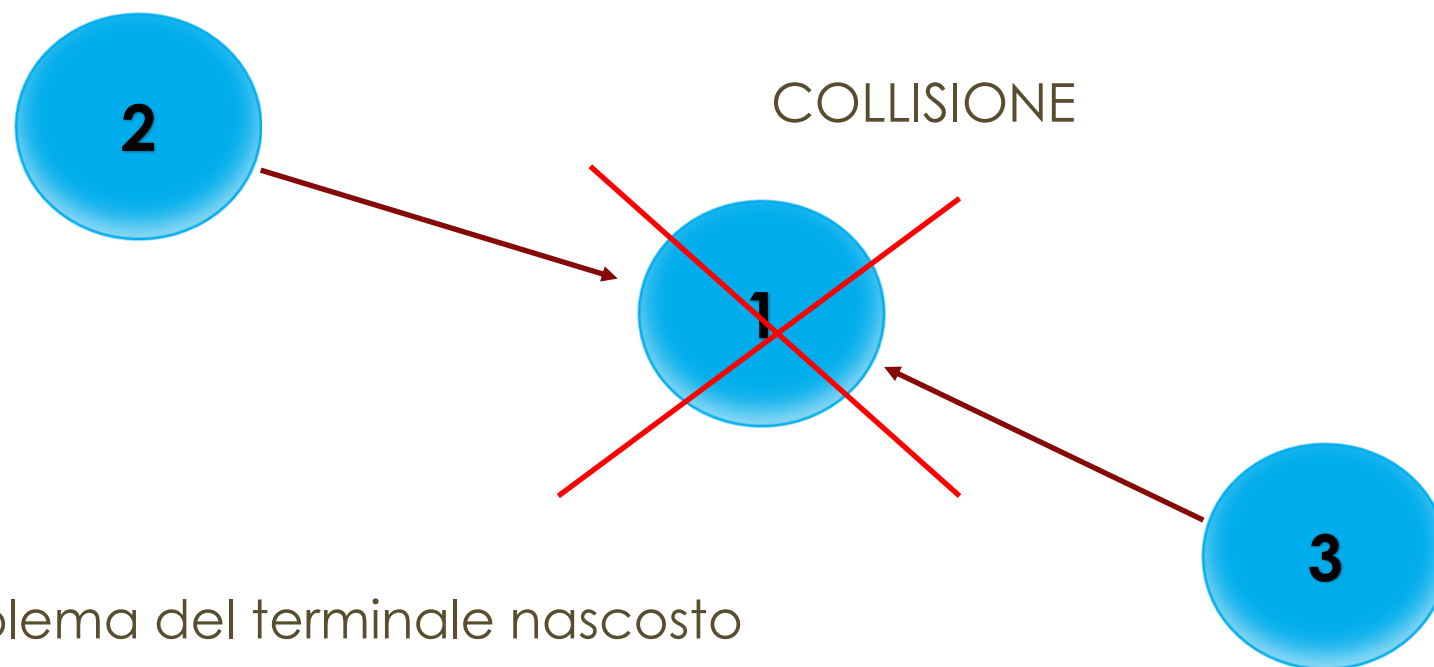
MAC – CSMA (Carrier Sensing Multiple Access)

Supponiamo che i nodi 2 e 3 non possono comunicare tra loro (quindi non possono neanche ascoltare la trasmissione di uno o dell'altro) ed entrambi vogliono trasmettere dati al nodo 1.



MAC – CSMA (Carrier Sensing Multiple Access)

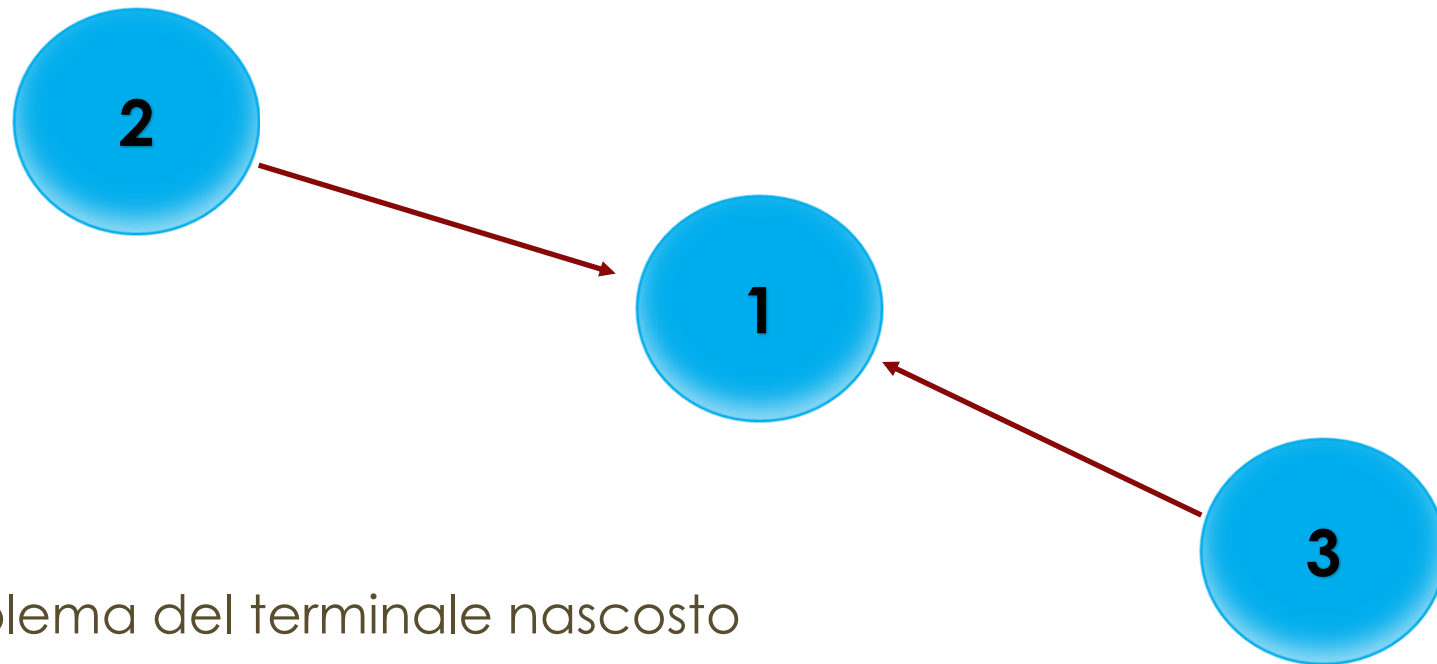
Supponiamo che i nodi 2 e 3 non possono comunicare tra loro (quindi non possono neanche ascoltare la trasmissione di uno o dell'altro) ed entrambi vogliono trasmettere dati al nodo 1.



Problema del terminale nascosto

MAC – CSMA (Carrier Sensing Multiple Access)

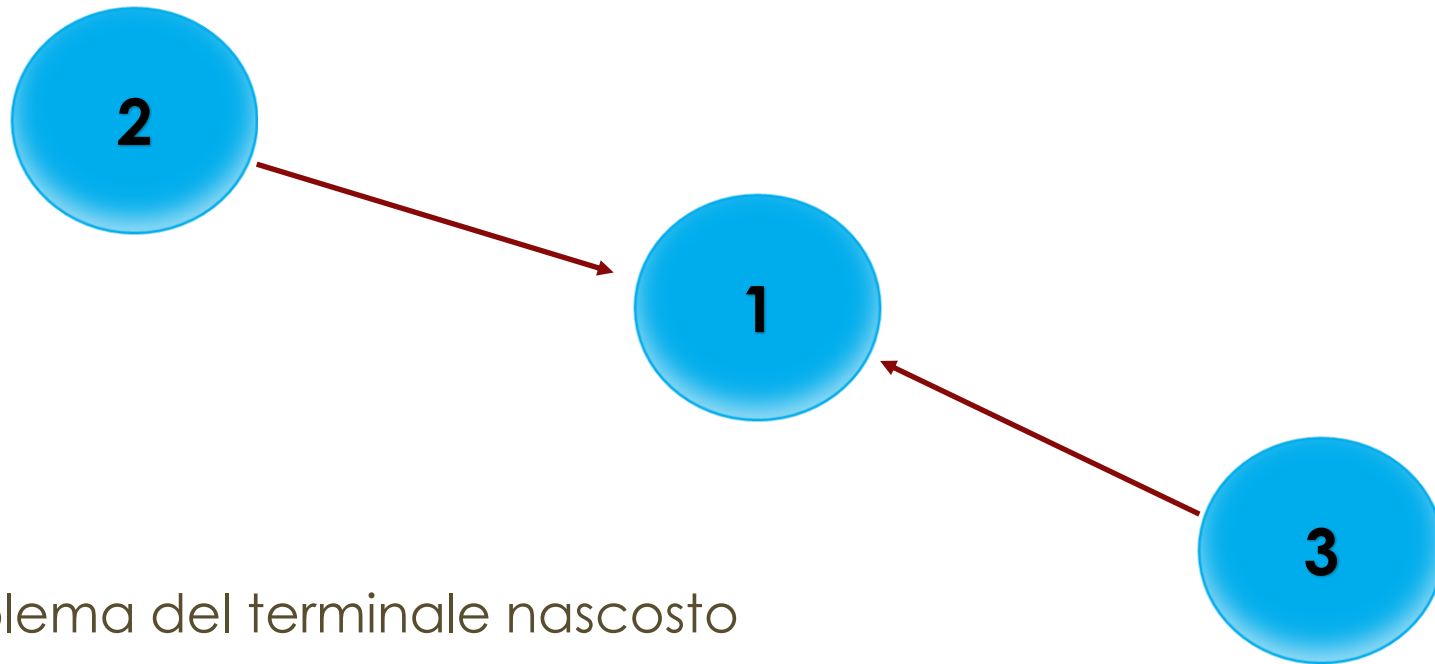
Come può essere risolto questo problema?



Problema del terminale nascosto

Rts (Request to Send) – Cts (Clear to send)

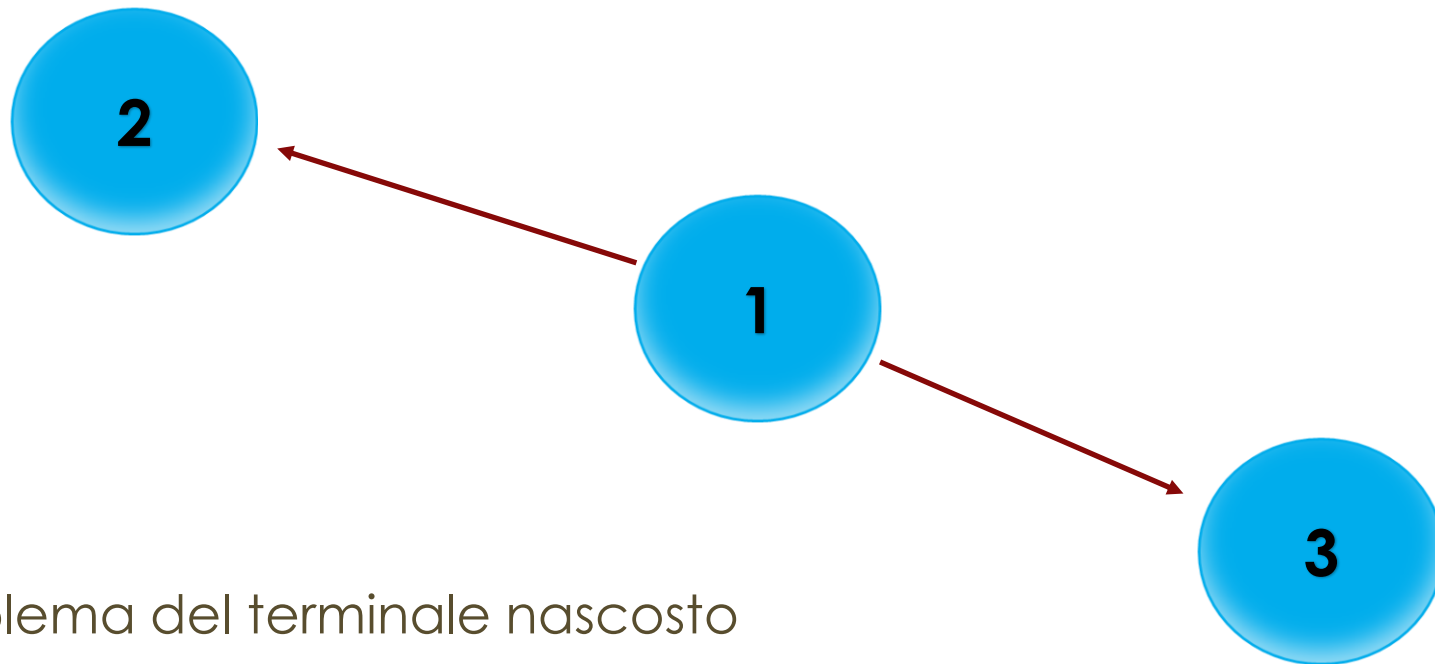
I due nodi (2 e 3) inviano un piccolo pacchetto chiamato Request to Send.



Problema del terminale nascosto

Rts (Request to Send) – Cts (Clear to send)

Il nodo 1 invia un piccolo pacchetto di Cts che contiene l'id del nodo che può trasmettere



Problema del terminale nascosto

Un esempio di protocollo che utilizza il meccanismo di RTS/CTS

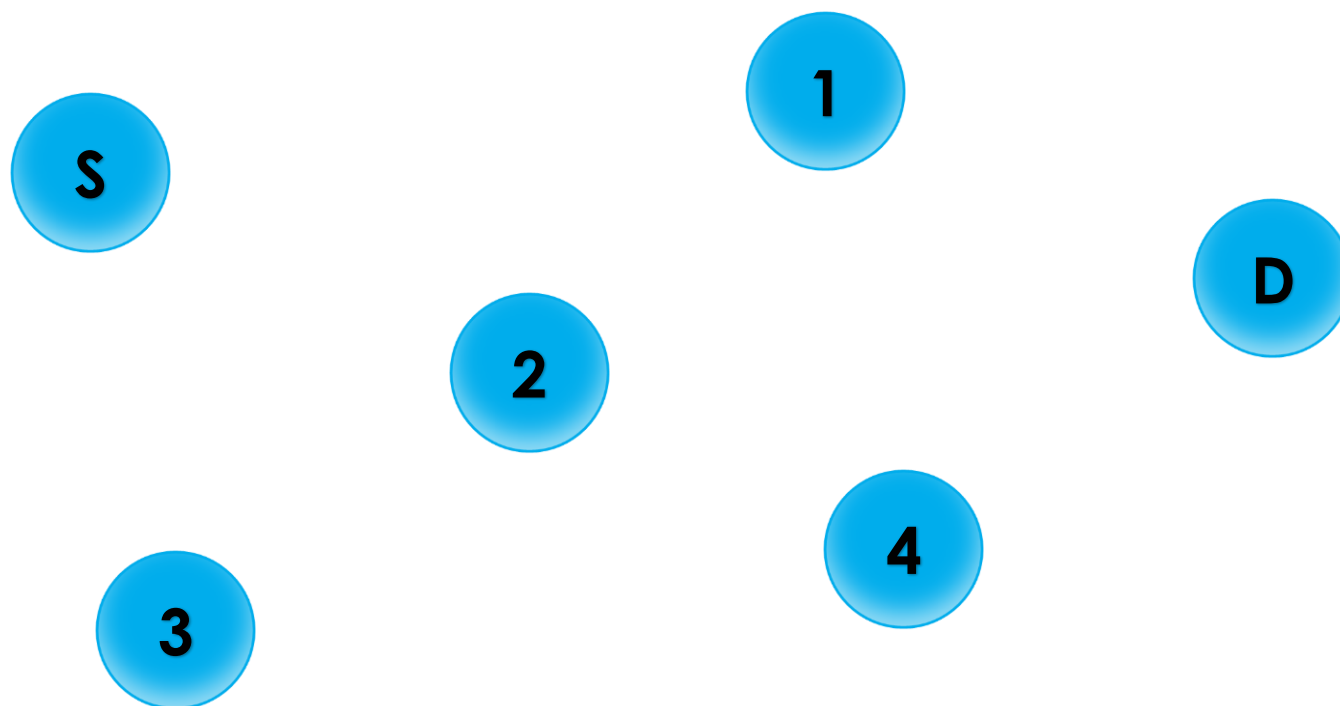
- **DACAP** (Distance Aware Collision Avoidance Protocol): Il nodo sorgente invia un pacchetto di RTS. La destinazione risponde con un pacchetto di CTS ed attende il pacchetto dati: se durante questa attesa rileva un'altra trasmissione, invia un piccolo pacchetto di WARNING al nodo sorgente. Il nodo sorgente, una volta ricevuto il pacchetto CTS, attende prima di trasmettere dati (WARNING time). In questo tempo, se rileva la trasmissione di un altro pacchetto di controllo o di un pacchetto WARNING non trasmette il pacchetto dati. Il WARNING time viene calcolato in base al *round-trip time* dello scambio dei pacchetti RTS/CTS.

Livello di Routing



Livello di Routing

- Si supponga che il nodo sorgente S deve trasmettere un pacchetto dati al nodo destinazione D. Come è possibile scegliere il percorso che il pacchetto deve seguire?





Livello di Routing

- In questo scenario è importante considerare molte problematiche derivanti dal canale. Due esempi potrebbero essere:
 - Il collegamento tra due nodi potrebbe essere molto disturbato a causa di diversi fattori (questo comporta la richiesta di più trasmissioni dello stesso pacchetto)
 - Un collegamento potrebbe essere asimmetrico

Categorie di protocolli di routing Internet of Things

- **Protocolli Proactive**
 - **Protocolli Reactive**
- **Protocolli Geografici**

Categorie di protocolli di routing Internet of Things

- **Protocolli Proactive:** ogni nodo mantiene delle tabelle di routing verso tutti i nodi della rete. In questo approccio ogni nodo scambia pacchetti di controllo in broadcast con gli altri nodi al fine di mantenere le tabelle aggiornate. Quindi, quando un nodo deve inviare un pacchetto dati conosce il percorso che dovrà seguire.

Vantaggi e svantaggi?

Categorie di protocolli di routing Internet of Things

- **Protocolli Reactive:** un nodo che deve inviare un pacchetto dati per prima cosa inizia una fase chiamata “Route Discovery”. In questa fase vengono inviati pacchetti di controllo ed una volta terminata, il nodo conosce il percorso verso la destinazione e può quindi inviare il pacchetto dati.

Vantaggi e svantaggi?

Categorie di protocolli di routing Internet of Things

- **Protocolli Geografici:** ogni nodo conosce la propria posizione e quella dei suoi “vicini”. In base a queste informazioni può decidere quale *next hop* scegliere.

Vantaggi e svantaggi?

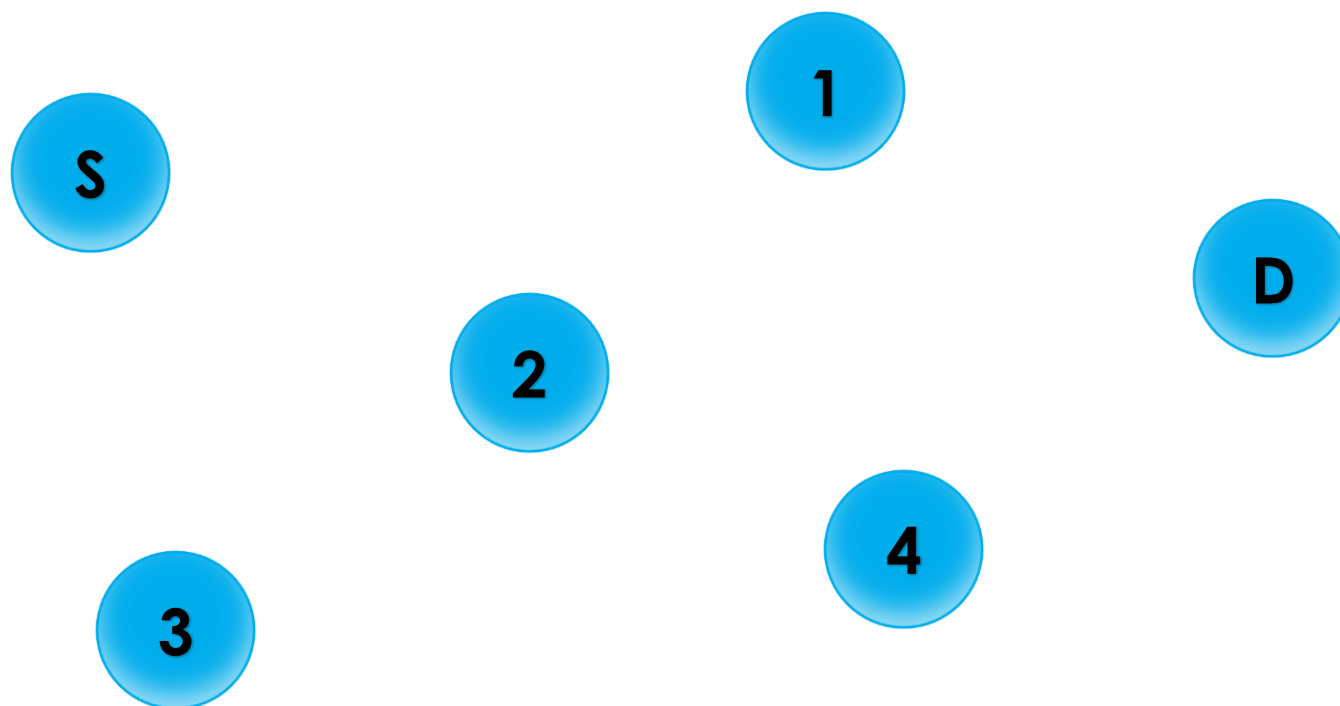
Livello di Routing

Alcuni fattori da tenere in considerazione per quanto riguarda le performance di un protocollo sono:

- **PDR:** numero pacchetti ricevuti dalla destinazione / numero pacchetti inviati
- **Ritardo end-to-end:** tempo impiegato dal pacchetto dal momento in cui viene inviato dal nodo sorgente al momento in cui viene ricevuto dal nodo destinazione
- **Energia:** energia consumata per trasmettere il pacchetto

Flooding

- Si supponga che il nodo sorgente S deve trasmettere un pacchetto dati al nodo destinazione D. Come funziona il flooding?



Flooding

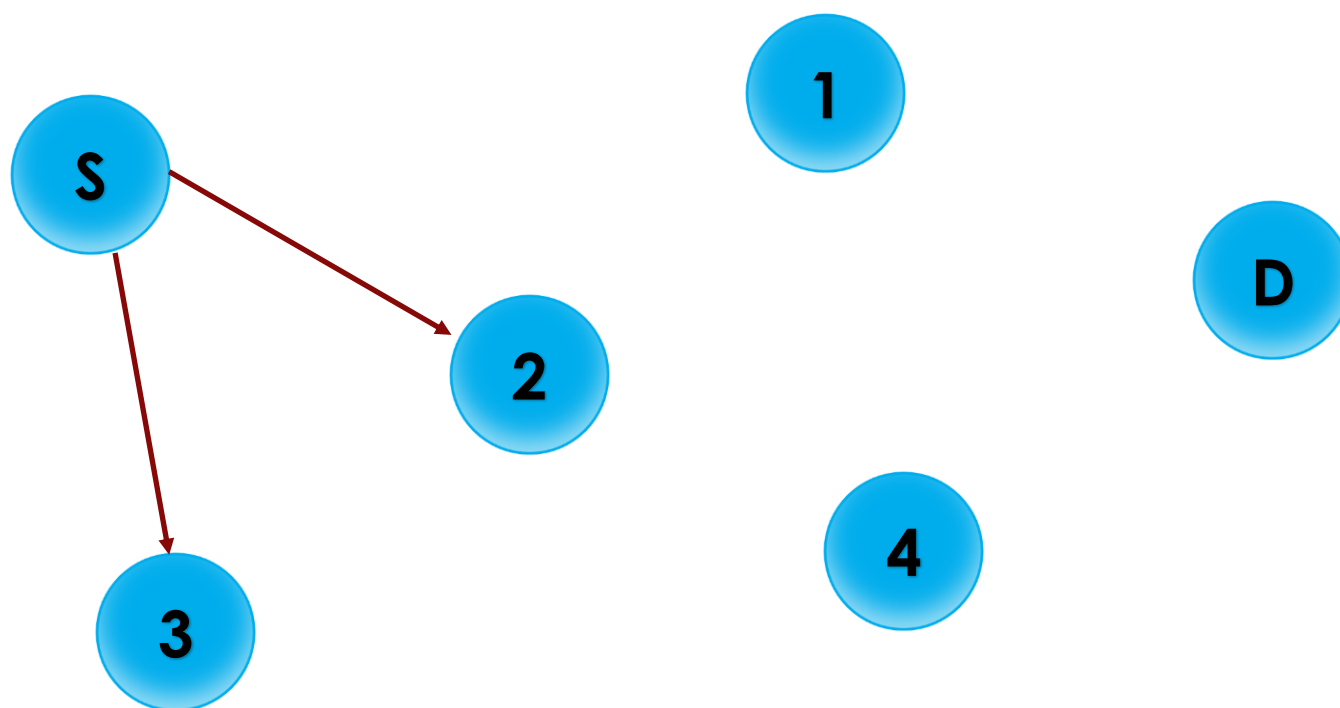
- Nel flooding ogni nodo trasmette il pacchetto a **tutti** i suoi vicini. Ogni nodo *relay* che riceve un pacchetto dati, esegue le seguenti operazioni:

Se il pacchetto è già stato ricevuto (quindi è un duplicato), scartalo.

Se non è stato ricevuto inoltralo ai tuoi vicini

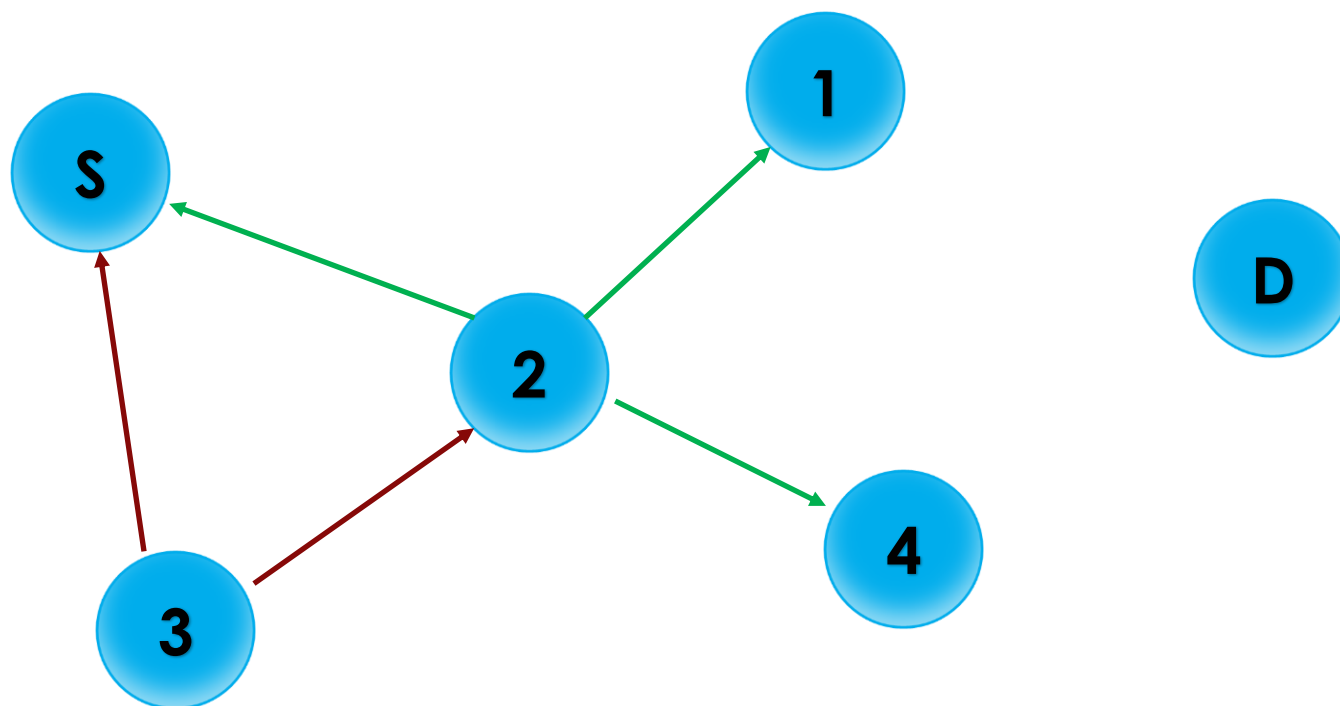
Flooding

- Il nodo S trasmette il pacchetto a tutti i suoi vicini



Flooding

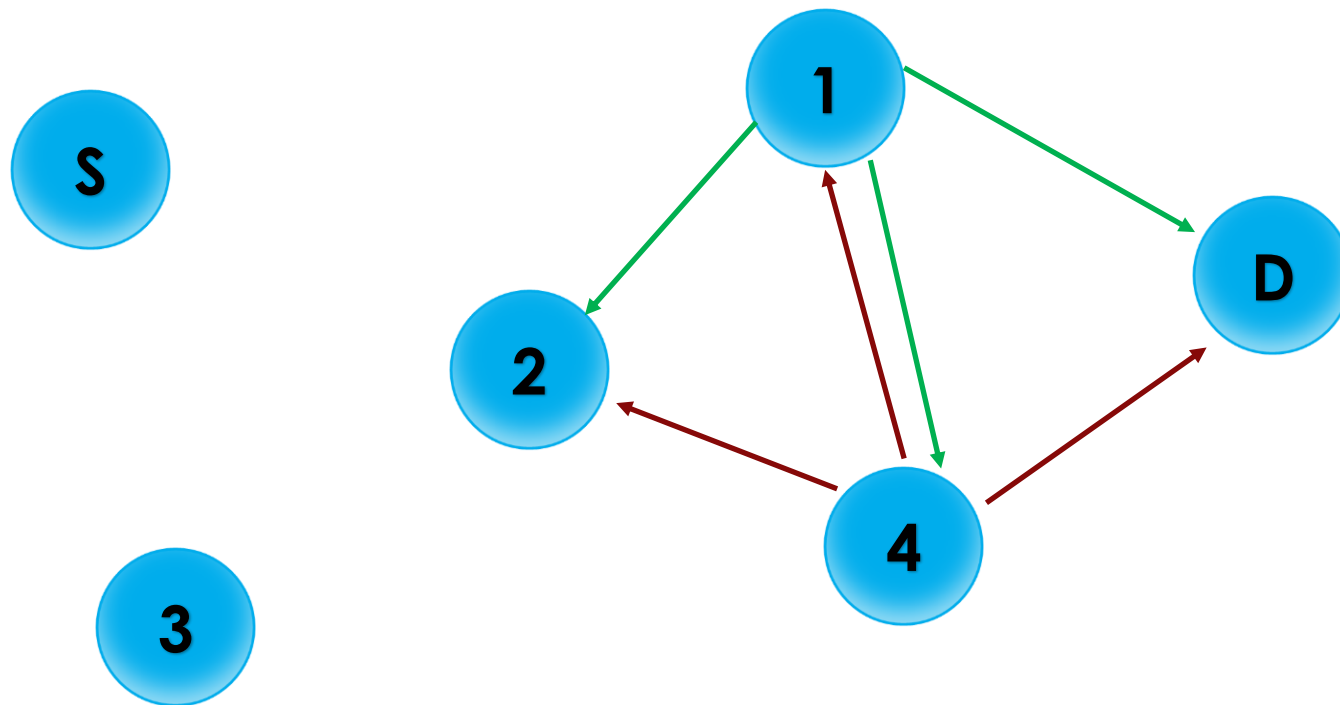
- I nodi 2 e 3, non avendo mai ricevuto il pacchetto lo inoltrano. Il nodo S lo riceve due volte e lo scarta.



Flooding



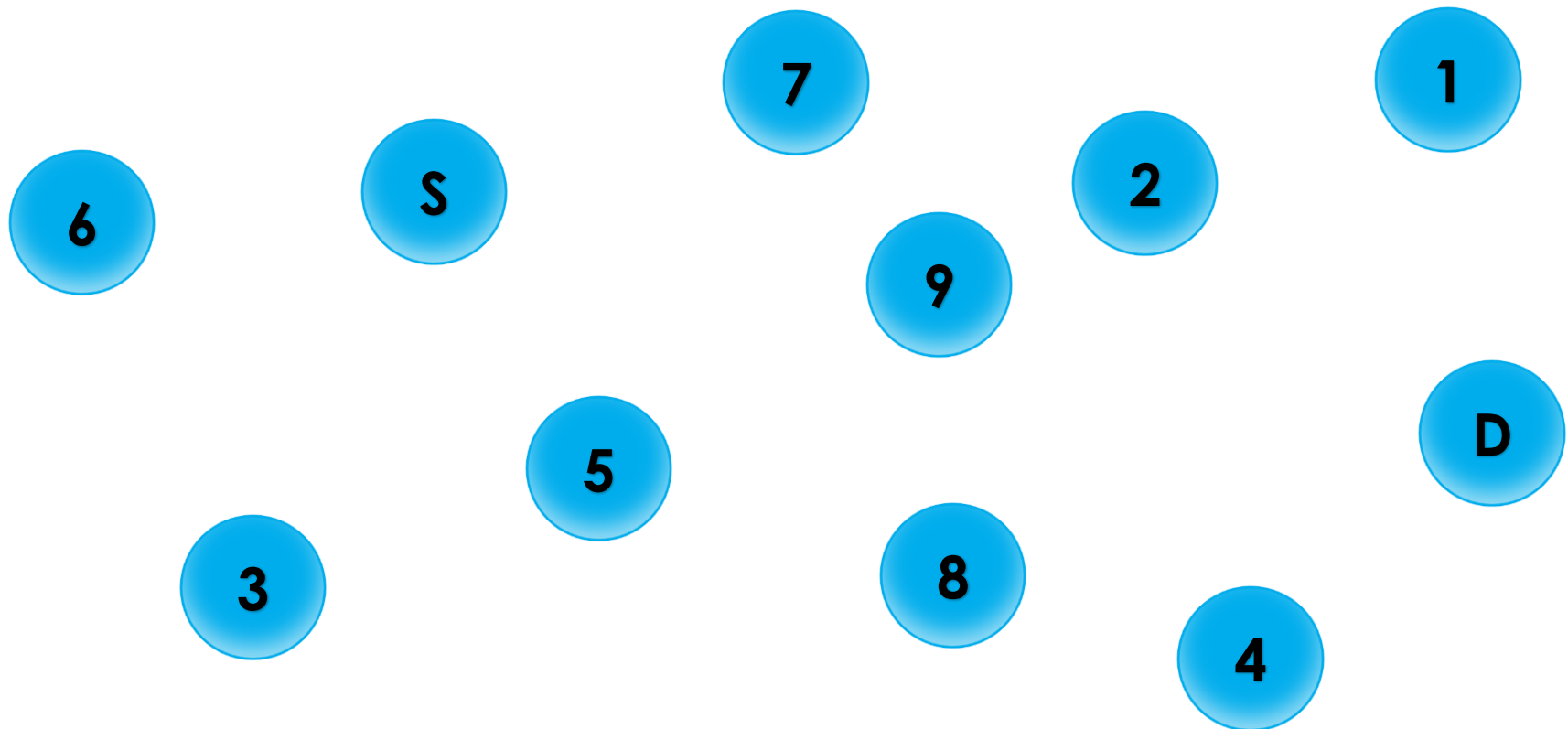
- I nodi 1 e 4, non avendo mai ricevuto il pacchetto lo inoltrano. Successivamente riceveranno altri duplicati dello stesso pacchetto che saranno scartati.
- Il nodo destinazione D riceve due copie dello stesso pacchetto



Flooding



- Supponiamo ora di avere una rete più grande. Cosa succede se il nodo Sorgente S deve inviare nuovamente un pacchetto dati al nodo destinazione D?



Flooding

- Se la rete è composta da 40 nodi?
- Se diversi nodi devono inviare molti pacchetti?

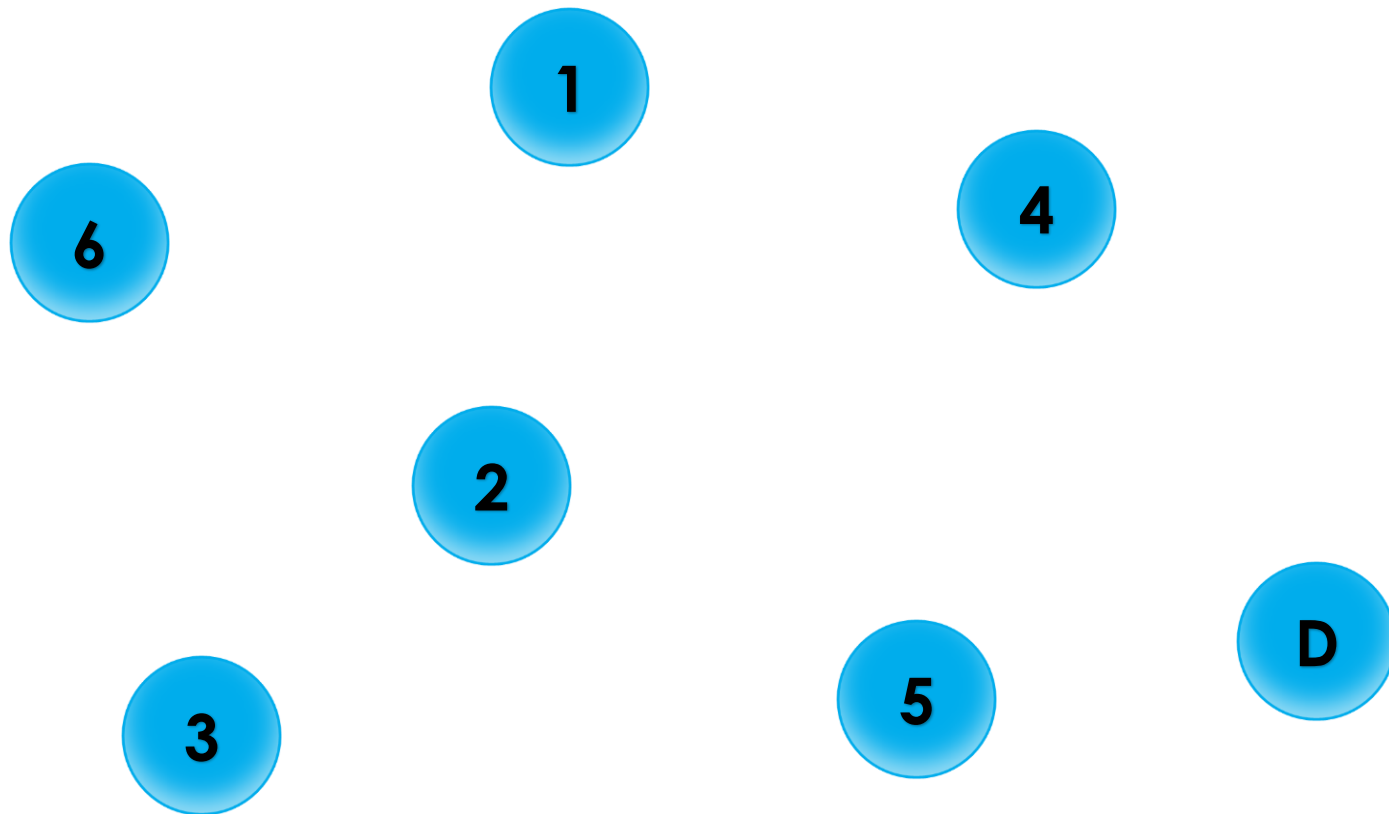


Analizziamo i vantaggi e gli svantaggi del flooding...

- Protocolli su diversi livelli dello stack si scambiano informazioni al fine di prendere decisioni
- Grazie a questi protocolli si possono prendere decisioni che riguardano la qualità dei link

Un esempio - CARP

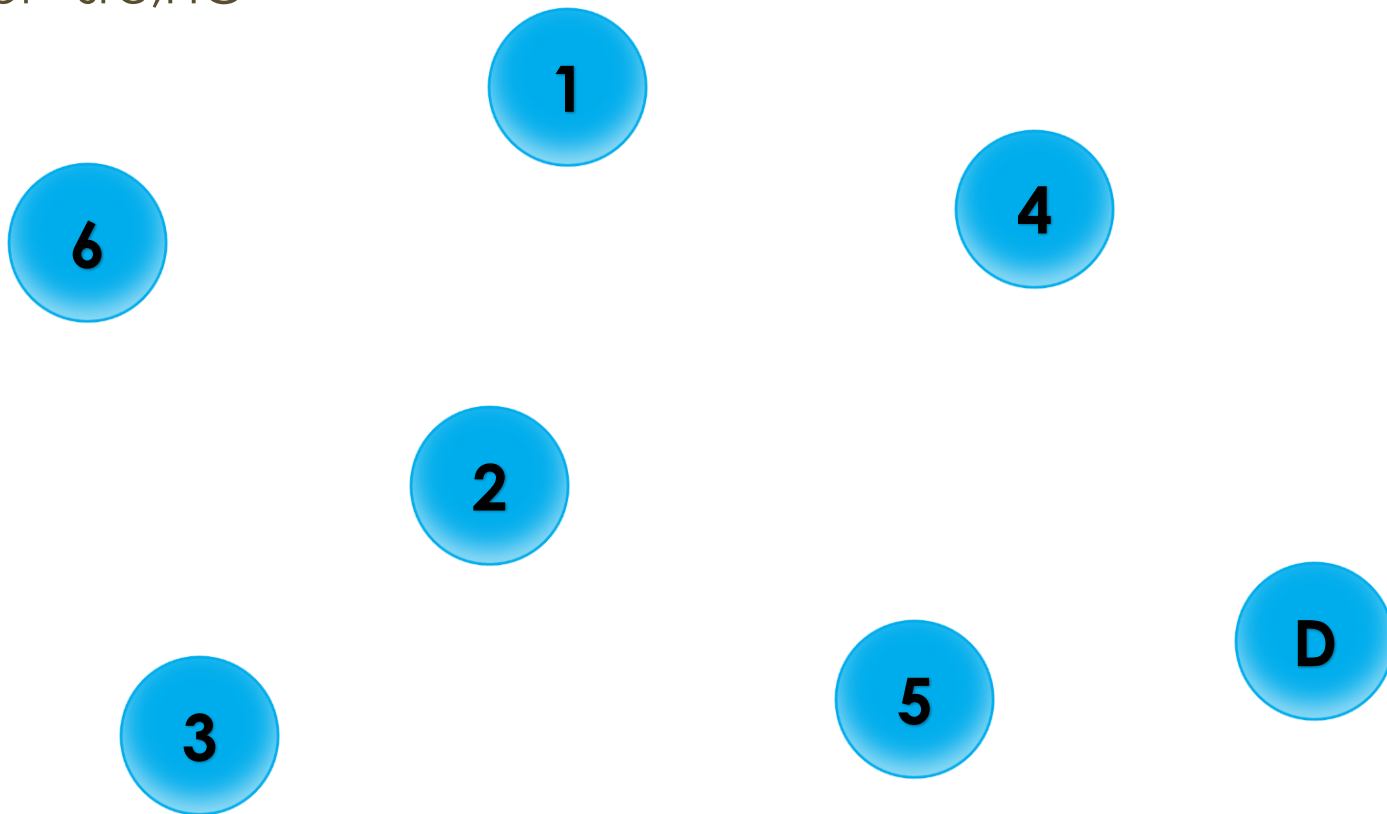
- CARP (Channel-Aware Routing Protocol)



Un esempio - CARP

■ Fase di Hello

Il nodo destinatario D invia un pacchetto di Hello contenente i campi <src,HC>

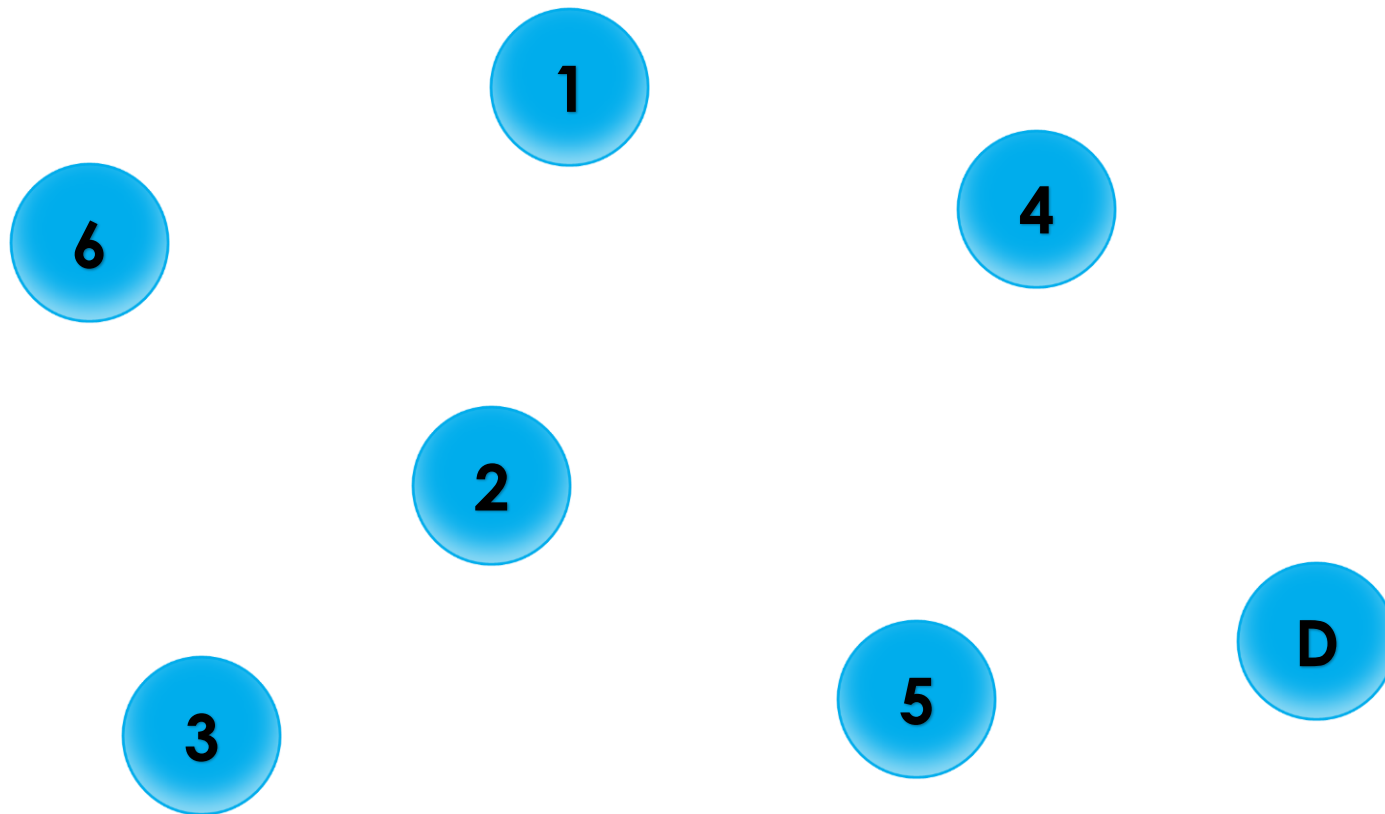


Un esempio - CARP

■ Fase di Hello



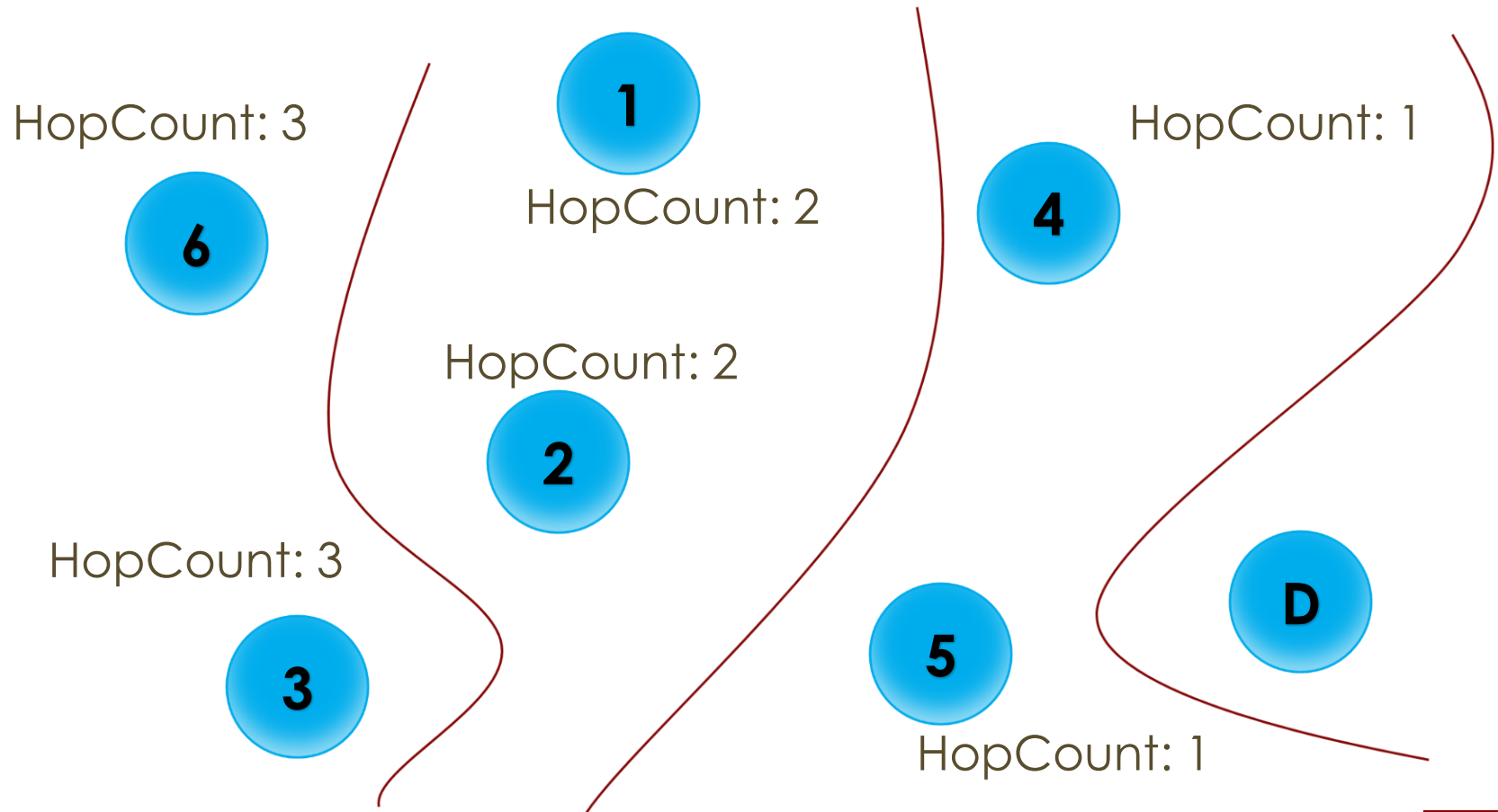
Ogni nodo controlla il proprio HopCount: se minore a quello del pacchetto lo scarta, altrimenti aggiorna il proprio HopCount ed inoltra il pacchetto inserendo il proprio id e l'hop count aggiornato



Un esempio - CARP

■ Fase di Hello

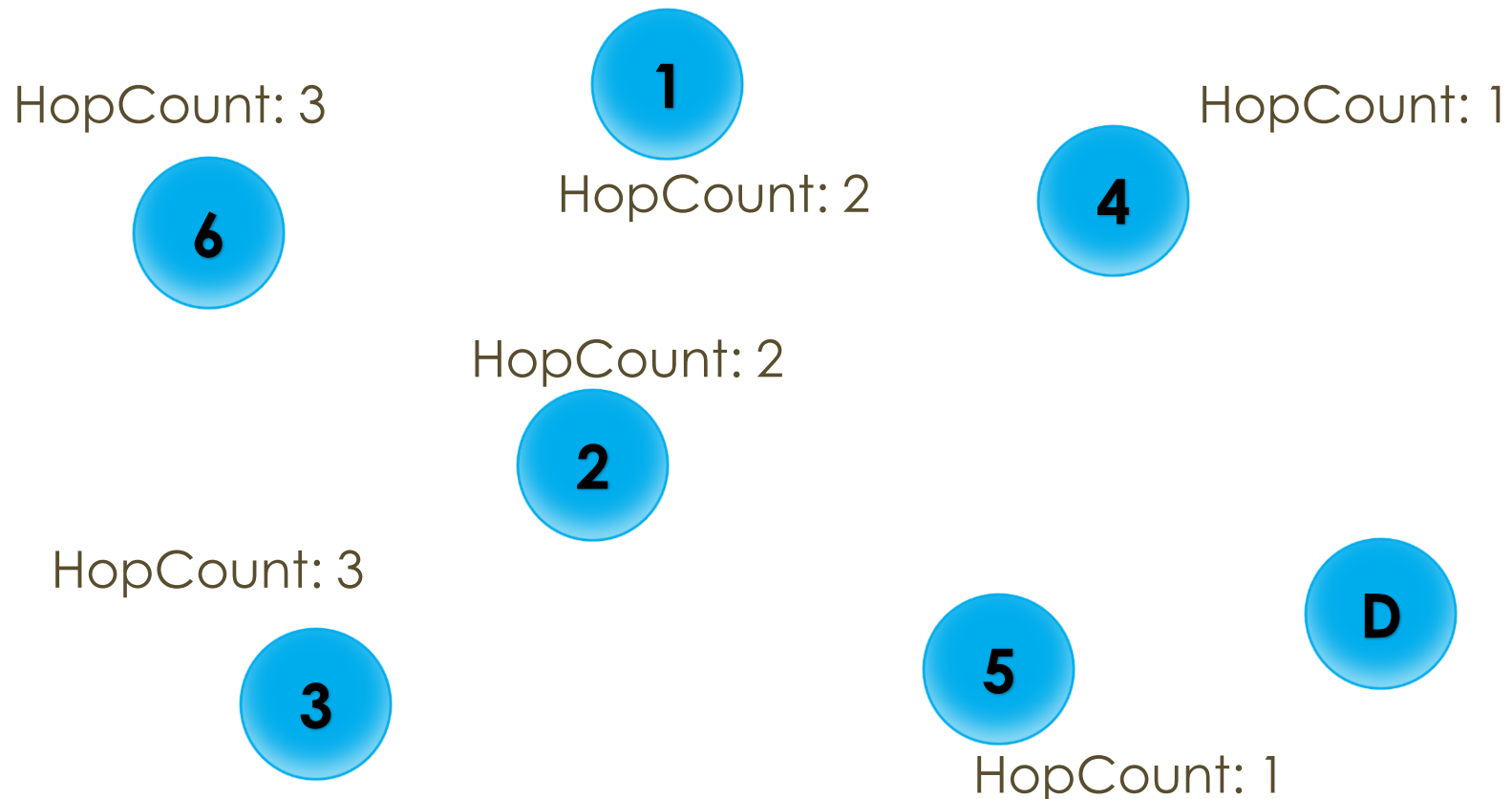
Dopo questa fase ogni nodo conosce la distanza in numero di hop dalla destinazione



Un esempio - CARP

- Il nodo 2 deve inviare dei pacchetti

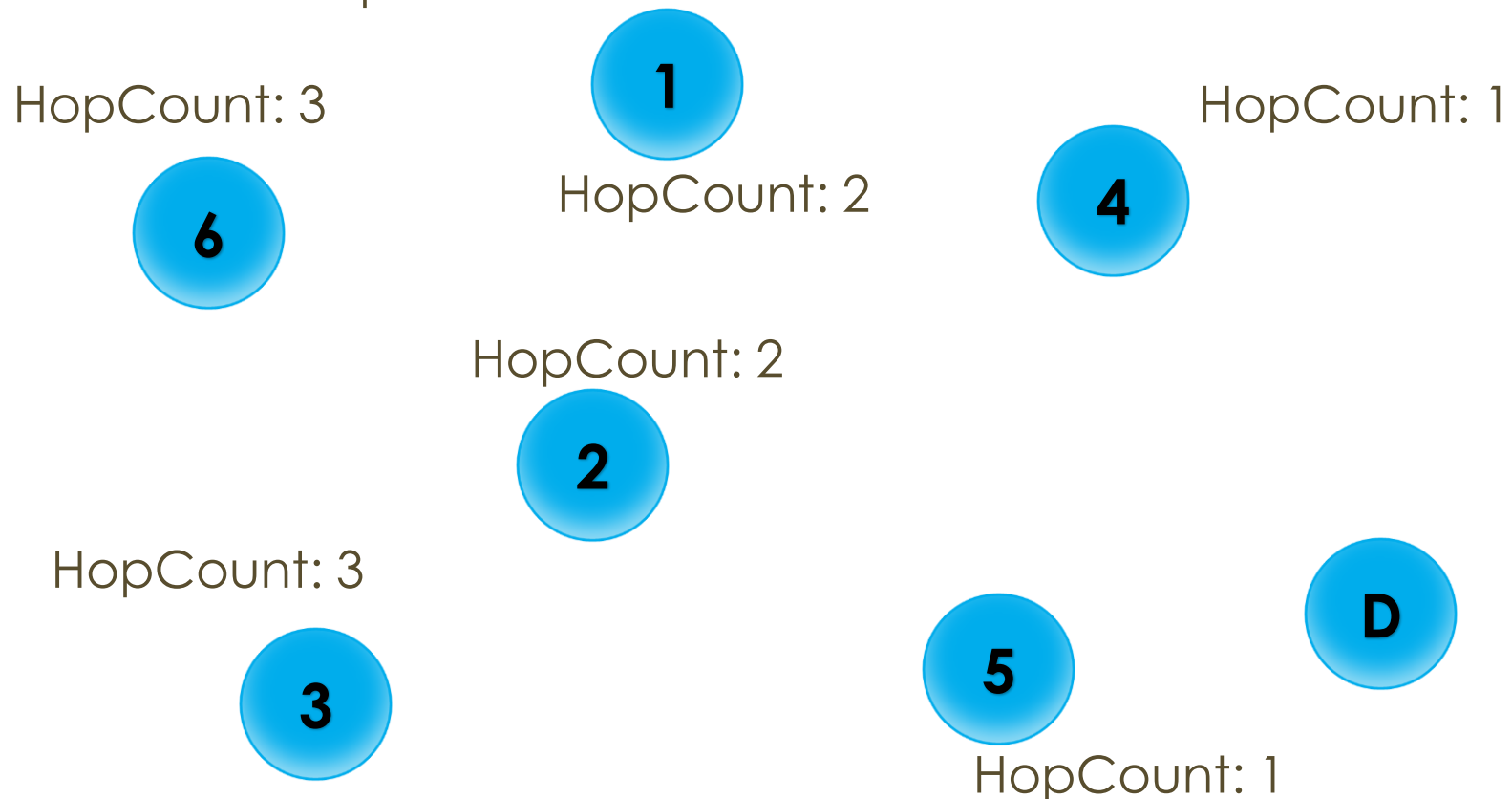
Il nodo 2 invia un pacchetto chiamato PING dove inserisce il proprio id, hop count e la lista dei pacchetti che deve inviare.



Un esempio - CARP

- Il nodo 2 deve inviare dei pacchetti

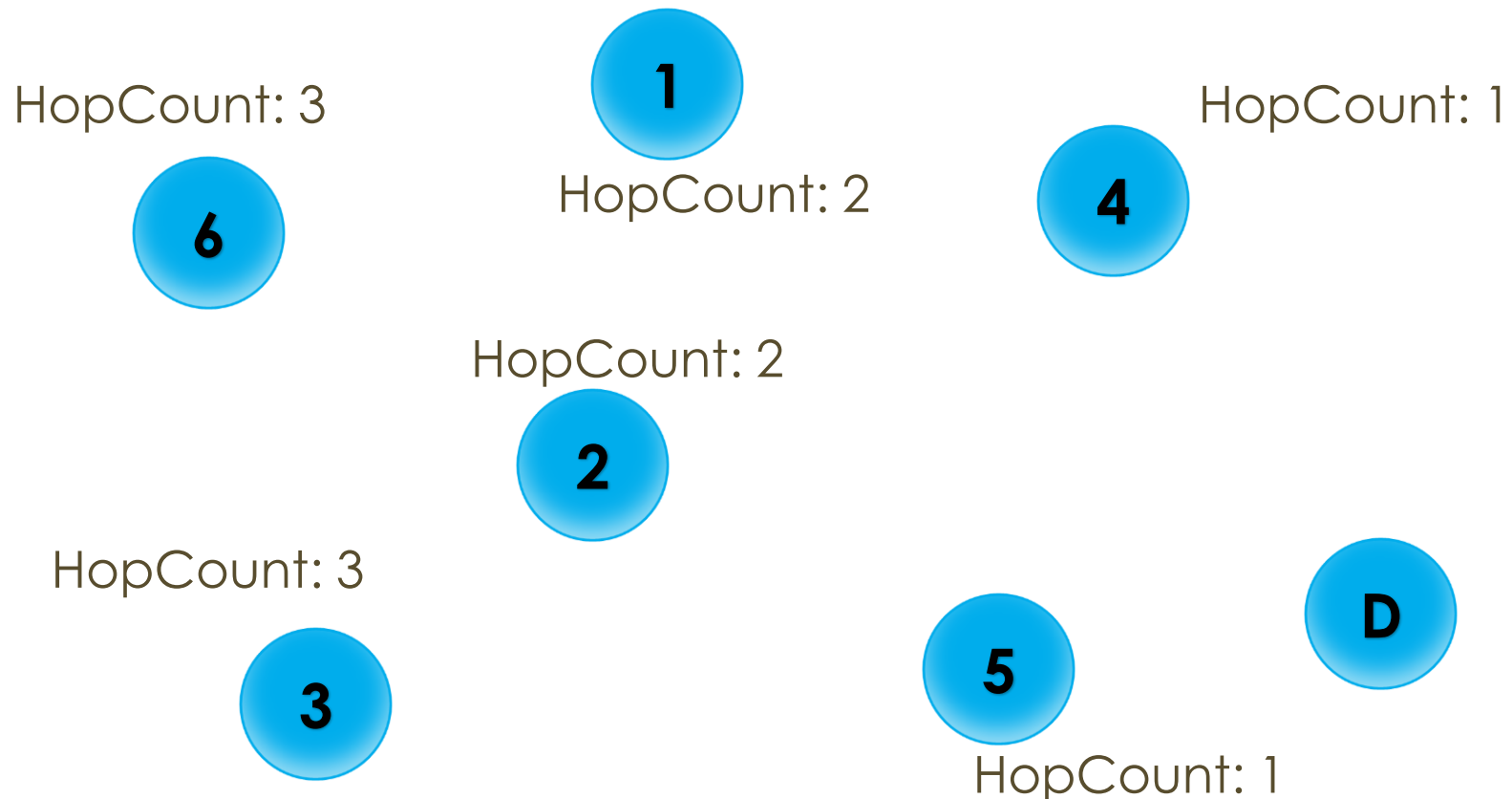
Tutti i nodi che hanno un hop count maggiore a quello contenuto nel pacchetto di PING lo scartano. Gli altri invece, rispondono al nodo 2 con un pacchetto chiamato PONG.



Un esempio - CARP

- Il nodo 2 deve inviare dei pacchetti

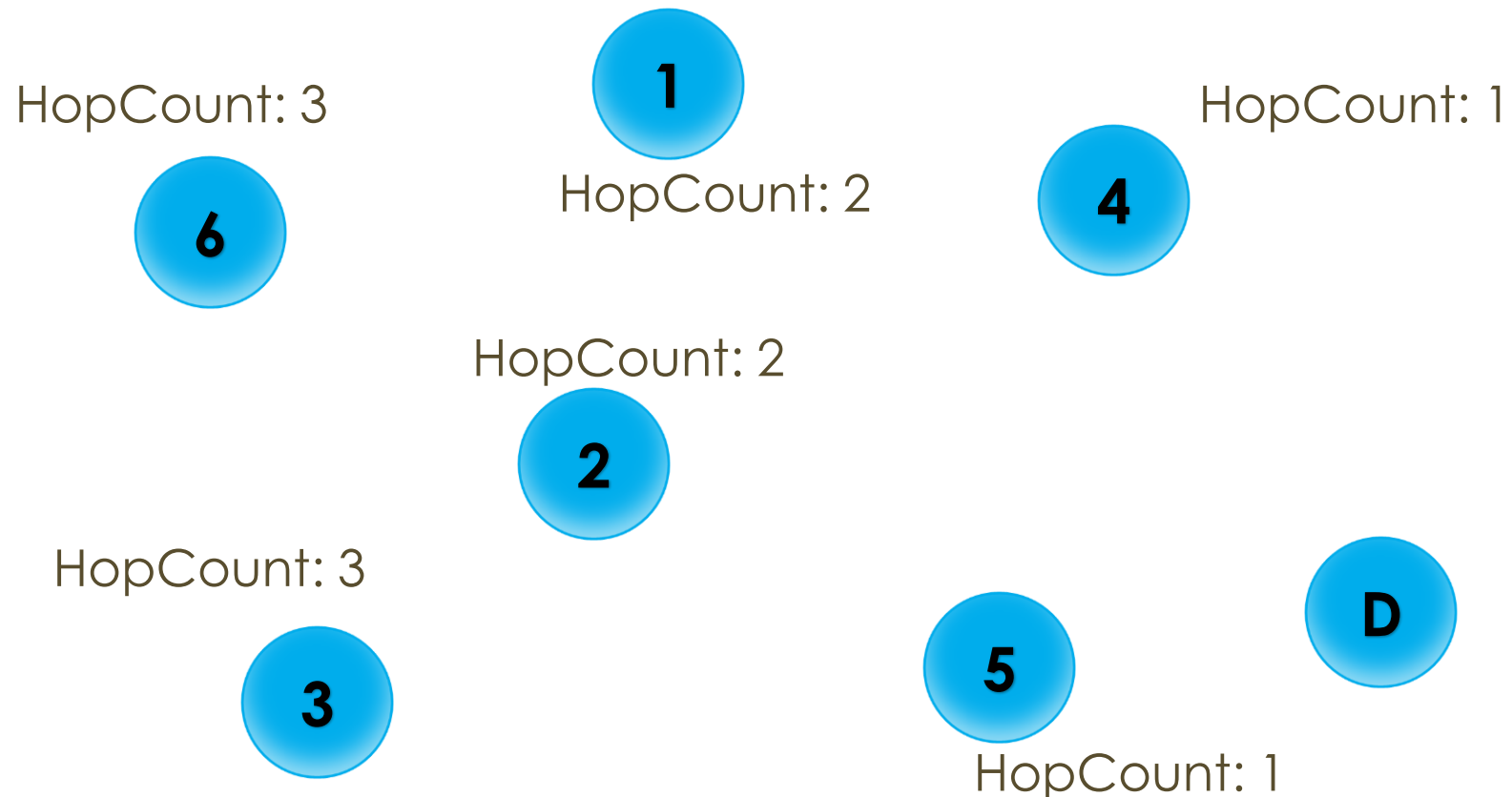
Nel pacchetto di PONG vengono inserite alcune informazioni tra cui: l'hop count del nodo, lo spazio disponibile in questo nodo, l'energia residua, un'indicazione sulla qualità del link (e altre informazioni...)



Un esempio - CARP

- Il nodo 2 deve inviare dei pacchetti

Il nodo 2, una volta ricevuti diversi pacchetti di PONG potrà selezionare il next hop sulla base dello spazio in memoria, energia residua e qualità del link.



Un esempio - CARP

- Quella mostrata è una descrizione molto sintetica del protocollo.
- Per chi fosse interessato di seguito viene riportato l'articolo:

S. Basagni, C. Petrioli, R. Petroccia and D. Spaccini. "CARP: A Channel-aware Routing Protocol for Underwater Acoustic Wireless Networks"

Una reale applicazione dell'IoUT

Una reale applicazione dell'Internet of Underwater Things (oltre ad avere sistemi per il monitoraggio ambientale o per la sorveglianza) potrebbe essere quello di creare un Sistema per diver.

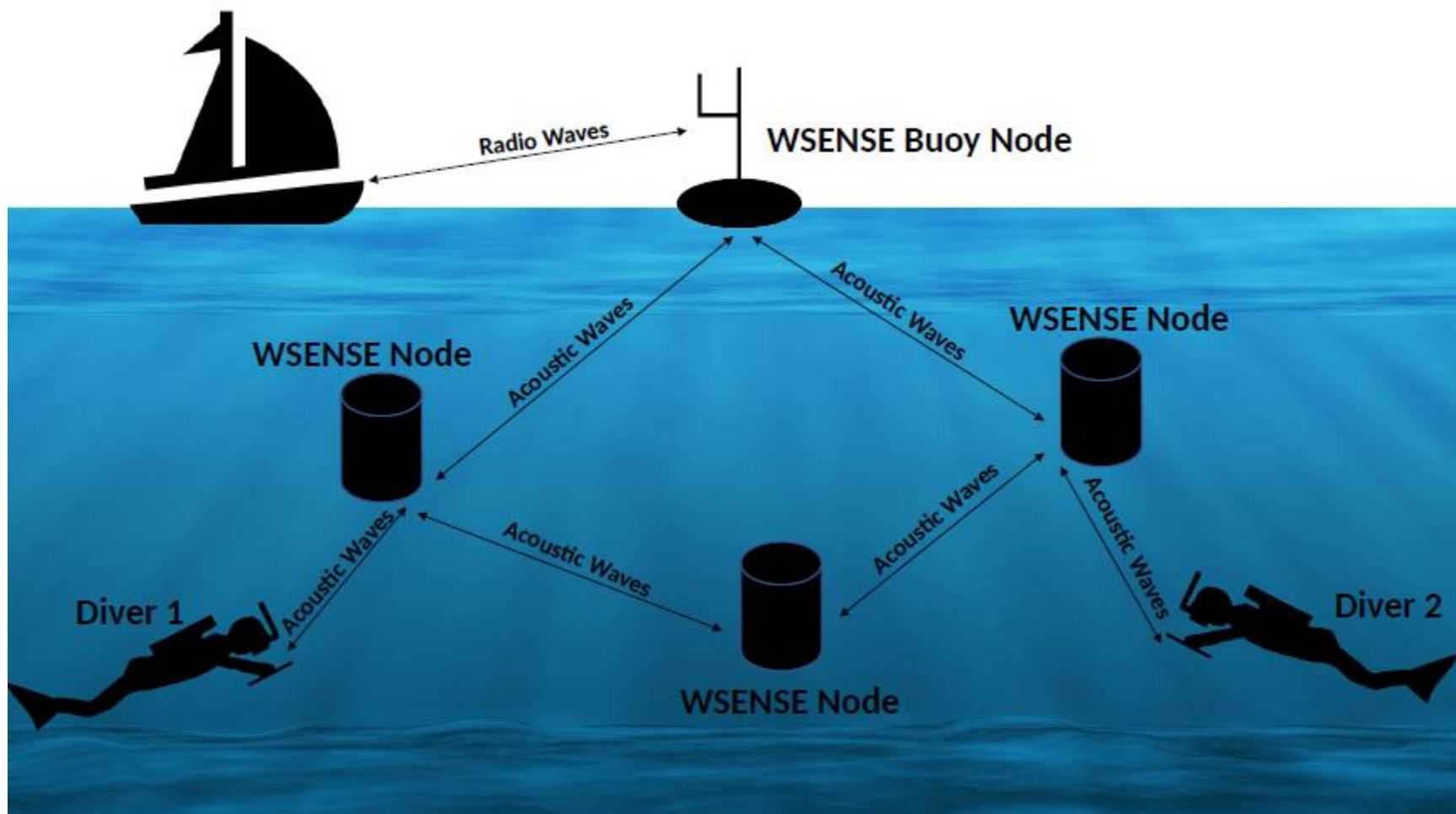
L'idea principale è quella di permettere ad un diver di:

- Inviare e ricevere messaggi con altri diver e con operatori esterni
- Scattare ed inviare immagini ad operatori esterni
- Avere un Sistema di localizzazione

Una reale applicazione dell'IoUT



■ Tipico scenario



Una reale applicazione dell'IoUT



SAPIENZA
UNIVERSITÀ DI ROMA

■ Sistema per diver

Modem acustico

Nodo underwater

Tablet sottomarino

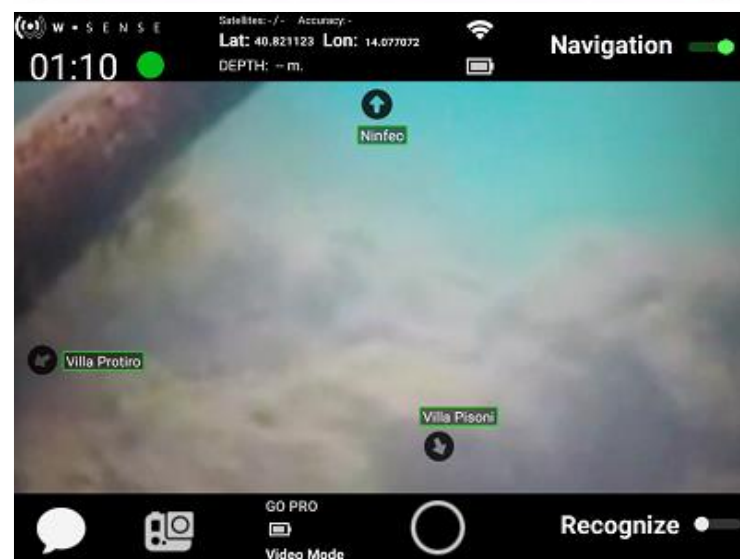


Una reale applicazione dell'IoUT

■ Applicazione per diver

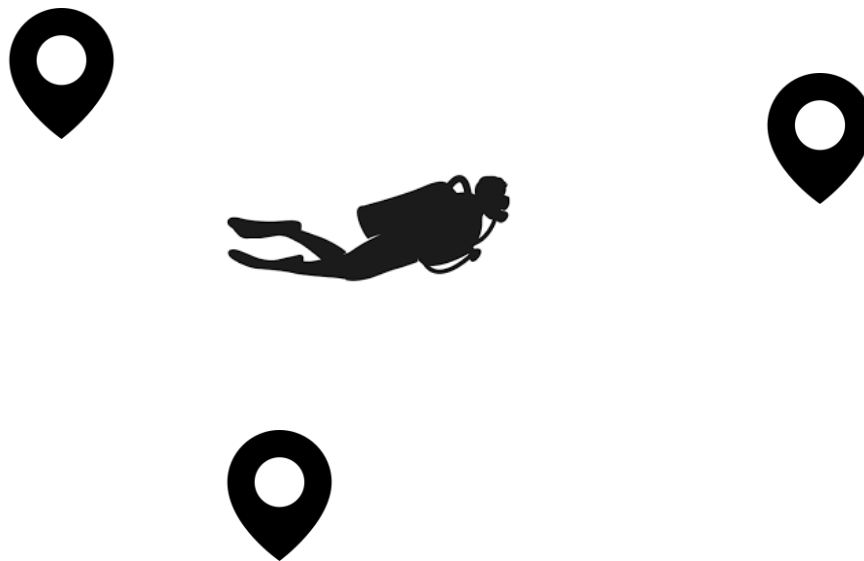


SAPIENZA
UNIVERSITÀ DI ROMA



Sistema di localizzazione

Il Sistema è composto da 3 (o più) nodi con un gps. Il nodo diver misura il tempo di arrivo del segnale per avere una stima sulla distanza da questi nodi.

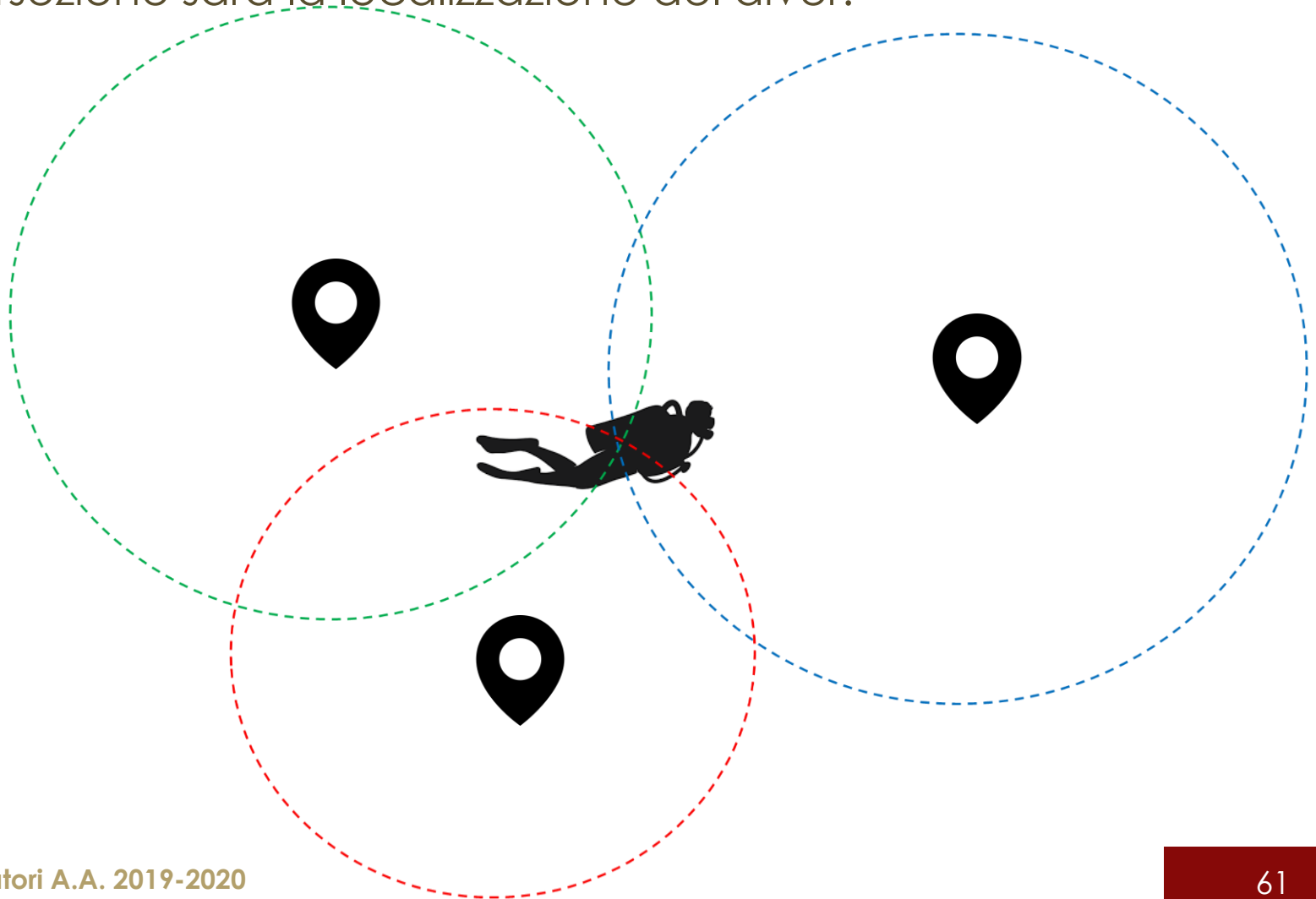


Sistema di localizzazione



SAPIENZA
UNIVERSITÀ DI ROMA

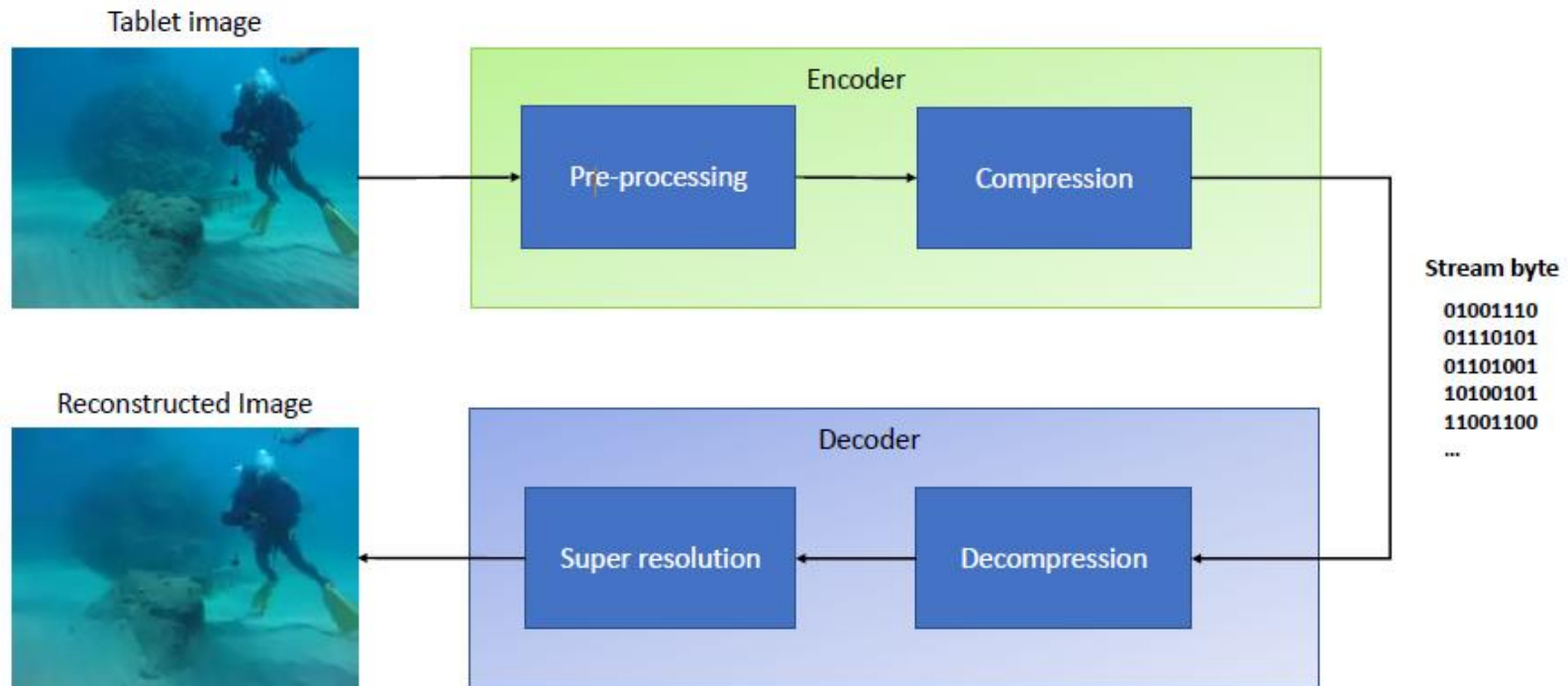
Grazie alla distanza con questi nodi può tracciare tre circonferenze.
La loro intersezione sarà la localizzazione del diver.



Trasmissione delle immagini



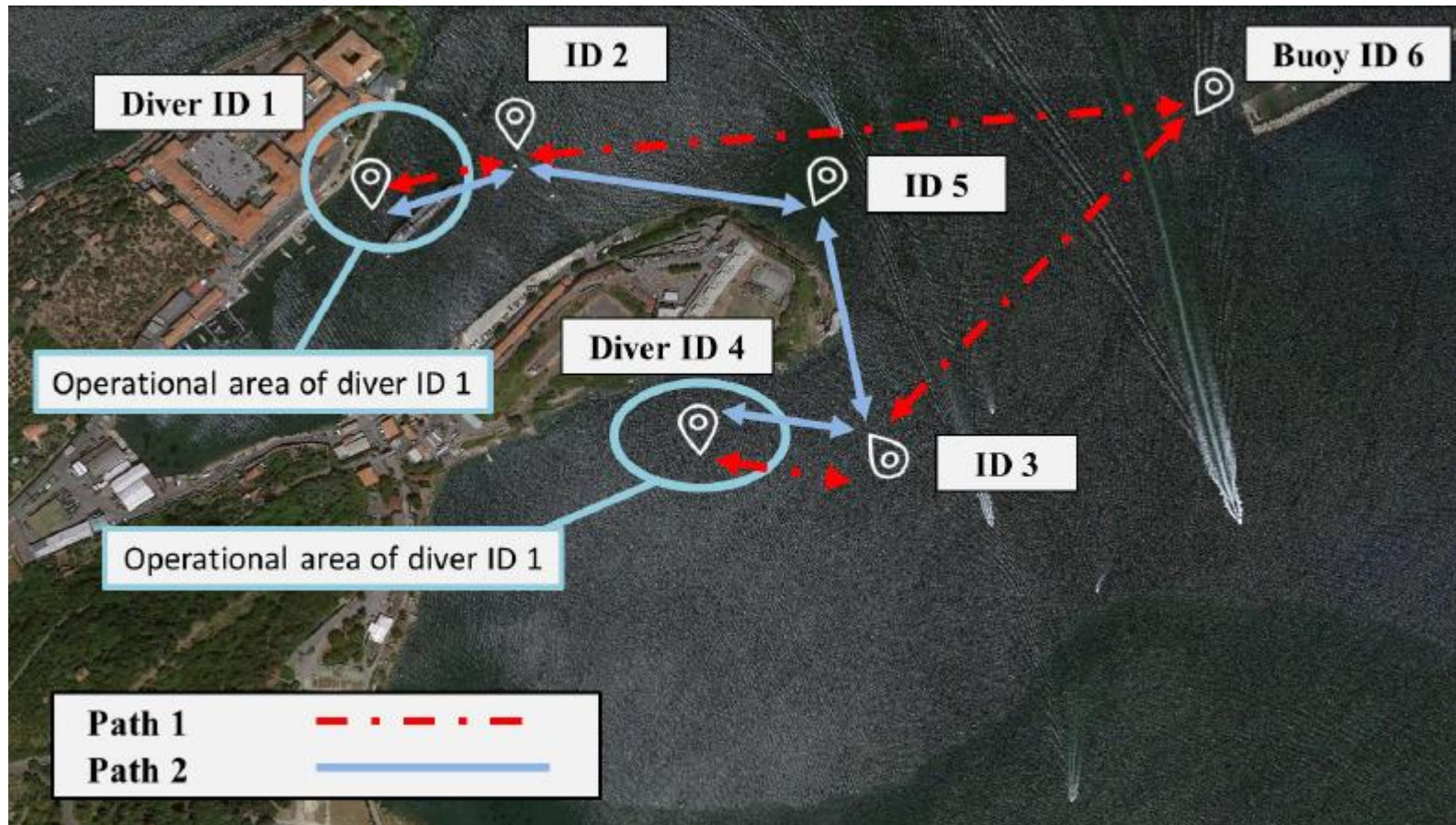
Una volta scattata una foto, il Sistema utilizza degli algoritmi di compressione per minimizzare il più possibile la dimensione dell'immagine e quindi il tempo di trasmissione. Questi possono essere di tipo *lossless* o *lossy* (in base alle attuali condizioni del canale).



Alcuni esperimenti reali

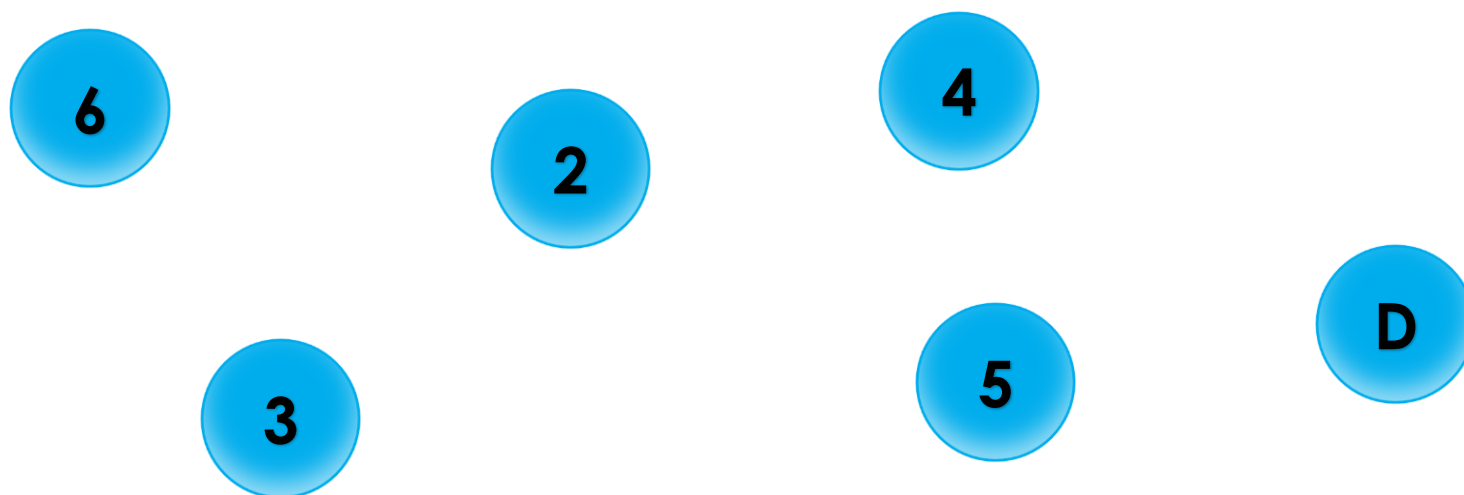


Una rappresentazione della topologia utilizzata per consentire la comunicazione tra diver e centro di controllo utilizzando percorsi multi-hop.



Temi di ricerca

- Sviluppo di protocolli adattivi per consentire di trasmettere immagini in una rete acustica sottomarina
 - Supponiamo che il nodo 6 deve trasferire un'immagine (un grande numero di pacchetti) al nodo destinazione D



Temi di ricerca

- Obiettivi
 - PDR: è importante che il nodo destinatario riceva tutti i pacchetti per poter ricostruire l'immagine
 - Tempo di trasmissione: la trasmissione dei pacchetti deve impiegare meno tempo possibile
 - Consumo energetico: è fondamentale consumare meno energia possibile

