



Università di Pisa

***Facoltà di Scienze Matematiche, Fisiche e Naturali
Dipartimento di Informatica***

Corso di Aspetti Etici e Sociali dell'Informatica

Digitalizzazione del Campo di Battaglia

Dott. D. Latella

Dott. G.P.Siroli

Realizzato da:

Andrea Batistini

Federico Cappelli

1. Introduzione

Il termine “campo di battaglia” evoca nella mente delle persone le più diverse immagini, e così deve essere. Il campo di battaglia infatti è un luogo diversificato e complesso, può essere un deserto o una giungla, può essere situato in un luogo molto caldo o insopportabilmente freddo, una città popolata o un luogo disabitato; ma le battaglie vengono combattute anche nel cielo e sui mari, quindi anche questi ambienti devono essere considerati dei campi di battaglia. I protagonisti delle battaglie sono soldati, marinai, aviatori e sono combattute su più livelli, tattico, operativo e strategico. Il campo di battaglia però non è soltanto questo, cioè un insieme di luoghi e persone, esso include anche i processi operativi.

Tipicamente sul campo di battaglia opera un'organizzazione strutturata in modo gerarchico la quale si occupa di pianificare le operazioni, comunicare i piani di queste a coloro che le devono eseguire, assegnare le risorse disponibili ai nuclei operativi, monitorare i progressi delle organizzazioni subordinate, tutto al fine di raggiungere l'obiettivo prefissato. L'organizzazione deve inoltre essere in grado di rispondere tempestivamente agli eventi anomali modificando i piani stabiliti.

In un sistema così complesso, applicare una qualsiasi forma di digitalizzazione può apparire un compito superiore alle attuali possibilità della tecnologia, nonostante questo i soli U.S.A. hanno stanziato, solo per il 2008, ben 3.7 miliardi di dollari per portare avanti il loro progetto di digitalizzazione del campo di battaglia denominato Future Combat System (FCS). Per questo progetto che stima di poter mettere in campo le prime unità dotate del sistema FCS per il 2008 e di poter disporre di 32 brigate FCS per il 2014 è stato previsto un costo complessivo di 92 miliardi di dollari e recentemente è stato aumentato del 25% per rispettare le tempistiche prefissate [1].

Un così ingente finanziamento ci dà la consapevolezza del modo in cui i vertici militari statunitensi guardano a le nuove tecnologie con lo scopo di rispondere alla più classica delle domande che si può porre un soldato sul campo di battaglia:

- Dove sono?
- Dove sono i miei alleati?
- Dove sono i miei nemici?
- Qual'è la mia missione?

Caratteristiche del Campo di Battaglia

Da tempo immemorabile i comandanti hanno dovuto prendere decisioni critiche in condizioni di incertezza, e questa è una cosa che è destinata a non cambiare in futuro. A livello pratico è la mancanza di informazioni a creare la condizione di incertezza quindi, normalmente, si attende di avere sufficienti informazioni prima di prendere una decisione. Sul campo di battaglia digitalizzato non esisterà alcun tipo di scarsità di informazioni, anzi, probabilmente, assisteremo alla condizione opposta, cioè ad un sovraccarico di informazioni che dovranno essere filtrate e convertite in dati utili allo scopo, dal ristretto organismo decisionale. Il fardello di dover interpretare la grande mole di dati generata dall'avanzamento nei sistemi di comunicazione, nei computers e nei sensori, ricade ancora sui comandanti e sui loro collaboratori, per i quali risulterà ancora più difficile comprendere le relazioni di causalità e quindi predire gli sviluppi della situazione. In un sistema così complesso è impensabile raggiungere il livello di consapevolezza della situazione (situational awareness) funzionale al raggiungimento degli obiettivi con il

classico sistema di inviare tutte le informazioni disponibili a tutti gli attori del campo di battaglia. Non è necessario in un'infrastruttura sovraccarica di informazioni ed inevitabilmente congestionata che, ad esempio, ad un sergente si consegnino le stesse informazioni che vengono inviate ad un generale, indipendentemente dal fatto che l'architettura di comunicazione lo permetta. Anche le tempistiche dell'azione cambieranno, i mezzi si muoveranno più velocemente, i sistemi d'arma spareranno più lontano, la sensoristica sarà capace di "vedere" in modo più accurato. I comandanti quindi dovranno prendere decisioni in tempi ancora più ristretti degli attuali e creare dei piani efficaci più rapidamente dei loro nemici per poter vincere la battaglia. Per poter mantenere lo stato di situational awareness a ogni livello è necessario quindi sviluppare un sistema innovativo, nei metodi e nelle tecnologie, che possa raccogliere le informazioni dalle varie sorgenti, consegnarle in un formato interpretabile a coloro che ne hanno bisogno, che metta in comunicazione in ogni momento tutti i livelli della gerarchia militare nonché sia in grado di visualizzare il campo di battaglia in un modo comprensibile e diversificato per le varie esigenze implementando anche nuovi sistemi di interazione tra uomo e macchina. La minore disponibilità di tempo e il gran numero di informazioni da dover interpretare ci portano a dire comunque che anche in futuro i comandanti dovranno prendere delle decisioni con una conoscenza imperfetta della situazione.

Il Sistema FCS

Il Future Combat System è il programma di modernizzazione dell'esercito americano consistente in una famiglia di sistemi con equipaggio e senza equipaggio, connessi da una rete comune, e costituenti una forza modulare che fornisce ai soldati e ai loro leader la capacità di prevalere sul nemico in ambienti complessi [2].

Questa è la definizione che l'esercito americano dà di questo complesso programma di riorganizzazione funzionale e tecnologica della sua struttura.

Il sistema consta di un complesso di 14 tipologie di veicoli e di mezzi, di un sistema di comunicazione informatizzato con il soldato stesso come punto centrale di tutta l'architettura.

I mezzi meccanici progettati prevedono le classiche tipologie di mezzi militari terrestri con equipaggio, come:

- Carri armati
- Obici semoventi
- Mortai semoventi
- Mezzi per fanteria meccanizzata
- Veicoli medici e per la riparazione
- Veicoli di ricognizione e sorveglianza
- Veicoli per la logistica,

ed inoltre tutta una serie di nuovi tipi di veicoli, come per esempio un mezzo speciale pensato per poter portare la complessa architettura per il command & control direttamente sul campo di battaglia.

Questi sistemi più tradizionali, almeno come tipo di veicoli, verranno affiancati da innovativi mezzi senza equipaggio denominati con i loro acronimi UGV (Unmanned Ground Vehicle) e UAV (Unmanned Air Vehicle), evoluzione degli analoghi già in uso presso gli eserciti più avanzati, che potranno essere telecomandati da remoto oppure programmati per eseguire movimenti semplici autonomamente (come pattugliare una certa zona) e fungeranno da occhi e da orecchie per i soldati, affiancandosi a reti di sensori disseminate sul campo di battaglia, ed eseguendo compiti ad alto rischio come ad esempio la penetrazione in ambienti ad alto rischio NBC (nucleare, biologico, chimico).

Il soldato di fanteria è l'elemento centrale di tutto il sistema, esso, oltre ai compiti tradizionali, dovrà essere in grado di relazionarsi a tutti quegli strumenti innovativi che saranno forniti nel suo equipaggiamento, come sensori, computer portatili, strumenti di controllo e navigazione per i mezzi UAV e UGV ed inoltre dovrà essere addestrato all'utilizzo delle interfacce di collegamento al sistema informatico.

Il sistema prevede inoltre una complessa architettura informatica strutturata in modo da resistere e adattarsi alle condizioni avverse presenti su un campo di battaglia, formata da una rete di telecomunicazione affidabile e scalabile e di interfacce diversificate per le varie tipologie di utenza e di missione.

2. Materiali e Metodi

Questa tesina si propone di dare una visione d'insieme del processo di digitalizzazione del campo di battaglia con particolare riferimento ad alcuni aspetti innovativi e prettamente informatici come la struttura della rete e alcuni servizi e tipologie di software attualmente in sviluppo da parte degli eserciti mondiali. Il materiale utilizzato è stato reperito su internet sotto forma di articoli apparsi su svariate riviste del settore informatico, siti internet di organizzazioni militari e vario materiale universitario di studio e progettazione. Abbiamo cercato di effettuare un rielaborazione e integrazione da diversi tipi di fonti, in molti casi questa è stata resa difficile dalla complessità degli argomenti trattati e principalmente dalla scarsità di materiale reperibile sugli aspetti propriamente militari della questione.

3. Implementazione Fisica e Protocolli di Routing di una Battlefield Network

di Federico Cappelli

Introduzione

Per supportare i livelli superiori e i servizi offerti, una rete da battaglia (da adesso BN, (battlefield network), interconnessa con le varie parti del sistema come: comando, centri di controllo, comunicazioni, computer, reparti di intelligence, sistemi di sorveglianza e ricognizione (C4ISR) [4] ha ovviamente bisogno di dello strato di trasporto, implementato con strutture hardware adeguate, scalabili e con un raggio di copertura adeguato alle esigenze del campo.

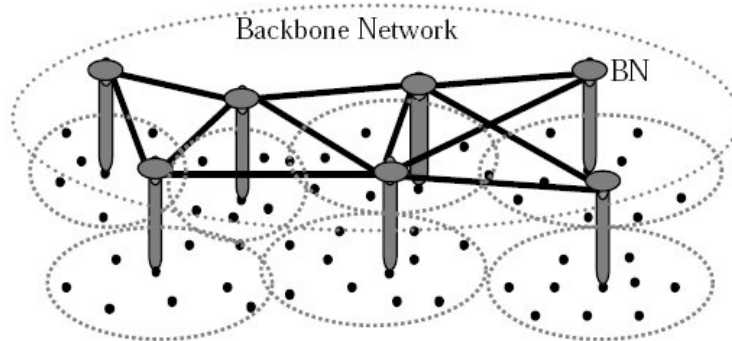
Le normali strutture di rete wireless sono costituite da nodi omogenei, con la stessa potenza dal punto di vista del segnale radio, questo porta a problemi di scalabilità e recenti studi hanno dimostrato che strutture di rete ad hoc su più livelli, con potenze radio diverse, diversa disposizione e organizzazione dei nodi migliora notevolmente le prestazioni [3]. Architetture di rete molto promettenti, proposte per varie applicazioni nei più svariati campi, inseriscono i nodi dinamicamente in gruppi che al loro interno eleggono, con politiche ad hoc, dei nodi principali che andranno a costituire la backbone della rete. Di seguito analizzeremo i vari aspetti critici di una rete così formata, il numero ottimo di nodi di backbone (da adesso NBB) per livello, il sistema di clustering adottato per scegliere e disporre i NBB, il protocollo usato per il routing in una rete così tipizzata ma anche il supporto hardware a tale rete.

Strato di Trasporto e Implementazione Hardware di una Rete da Battaglia

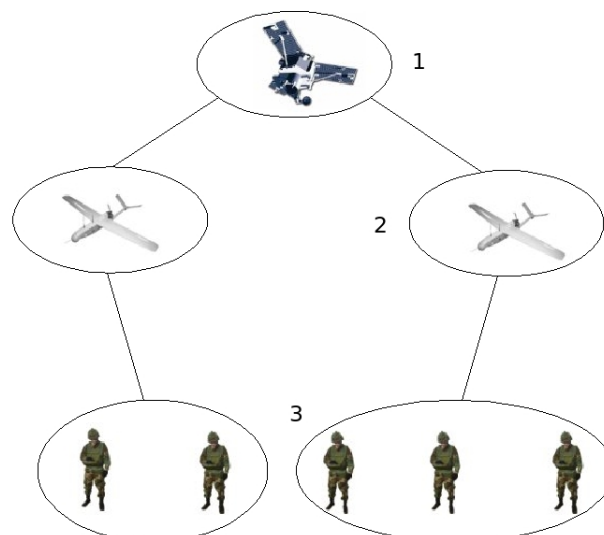
Per ovvi motivi una BN, essendo composta da nodi mobili, è costituita da connessioni wireless basate sul protocollo IEEE 802.11, che in ambito particolare e con particolari requisiti, come quello militare dà grandi vantaggi, come:

- Indipendenza da stazioni fisse, che possono essere sottoposte ad attacchi.
- Dispiegamento immediato delle unità.
- Facilità di riconfigurazione della rete in caso di bisogno.

In più una rete vasta come quelle utilizzate su un campo di battaglia è composta da decine di tipi differenti di nodi, con caratteristiche e potenzialità differenti, una wireless installata su un carro armato sarà sicuramente più potente di quella che può equipaggiare un fante, per questo è necessario strutturare anche l'architettura hardware della rete su più livelli, dove il livello superiore provvede a fornire connettività al livello inferiore, con questo schema utilizzato ricorsivamente:



Un esempio di architettura di una rete multilayer:



Dove ogni layer ha caratteristiche diverse sia dal punto di vista fisico che logico.

- Layer 1: rete satellitare che provvede al più alto livello di comunicazioni, virtualmente molto sicura dal punto di vista fisico e con grande disponibilità di banda.

- Layer 2: UAV di tipo I o di tipo IV [4], a seconda delle esigenze e del teatro operativo, entrambi i dispositivi volanti integrano connettività wireless e sensori per la raccolta dei più svariati tipi di informazioni.
- Layer 3: Fanteria, mezzi blindati, mezzi di trasporto e agenti hardware come sensori (UGV) e robot terrestri teleguidati (UGV).

In questa suddivisione dei nodi esistono dei nodi come i NBB e gli UAV che potenzialmente devono comunicare su due tipi diversi di connessione, diversi sia dal punto di vista fisico, cioè frequenze e ampiezze di banda diverse, che come vedremo in seguito, dal punto di vista dei protocolli, quindi ogni nodo ha a disposizione 2 interfacce di rete distinte.

Architettura, Routing e Deployment di una Rete da Battaglia

Per realizzare una rete su più livelli con le caratteristiche sopra descritte si deve utilizzare un protocollo di routing particolare, come prima cosa analizziamo il protocollo Extended Hierarchical State Routing (EHSR) che però soffre di limitazioni sul numero dei nodi, grandezza della rete, carico sulla rete e routing overhead, per questo è stato sviluppato un protocollo di routing chiamato LANMAR [3], un algoritmo molto efficiente su reti wireless molto ampie, con un grande numero di nodi e un'ottima interazione dei nodi con i backbone; la struttura risultante è una gerarchia di rete su più livelli con tabelle di routing flessibili.

Extended Hierarchical State Routing (EHSR)

L'EHSR è una estensione del protocollo di instradamento gerarchico (HSR) [3], sia HSR che EHSR sono protocolli di routing del tipo "Link State", dove i nodi sono divisi in gruppi (cluster) e per identificare ogni nodo gli viene assegnato un HID (Hierarchical identification number) definito dal MAC Address del nodo e il percorso dal nodo principale.

Consideriamo una rete EHSR strutturata su 3 livelli, rete UAV, rete wireless ad hoc sul terreno i cui nodi interni sono divisi in nodi standard e nodi di backbone, l'HID sarà composto dall'indirizzo della rete UAV, l'indirizzo della rete dei backbone e indirizzo della rete wireless che corrisponde all'indirizzo MAC del singolo nodo, ovviamente l'HID è un indirizzo utilizzato per il routing e quindi definisce il percorso completo (path) del nodo all'interno della gerarchia.

Data la mobilità intrinseca della rete i nodi possono cambiare cluster in continuazione, quindi l'HID di un nodo deve essere modificato molto spesso e gli altri nodi devono essere informati di questo cambiamento, per realizzare questo ogni nodo ha un indirizzo logico permanente e un home agent (HA) per registrare il proprio HID, viene naturale che gli UAV fungano da HA per i nodi nella loro area. Ogni nodo della backbone trasmetterà un segnale radio di controllo, per segnalare la propria presenza. Una volta ricevuto questo segnale un nodo potrà scegliere a quale NBB (CH) sia il più vicino e quindi appoggiarsi ad esso tramite una registrazione, se il NBB risulta diverso dall'ultima volta il nodo standard deve aggiornare il proprio HID e segnalarlo al suo Home Agent.

Negli alti livelli di una rete gerarchica risulta molto semplice l'utilizzazione dell'EHSR, ma quando si prova ad applicare questo protocollo ai livelli più bassi si riscontrano delle notevoli limitazioni che degradano la sua efficienza:

1. Tutto il traffico della rete deve passare attraverso i NBB, anche per la comunicazione tra nodi dello stesso gruppo, questo può causare congestione e conflitti sui NBB.

2. Una rigorosa applicazione dell'EHSR richiede che ogni nodo sia in grado di raggiungere un NBB in un salto, infatti un cluster con nodi non direttamente connessi al NBB (multi-hop nodes) può risultare inefficiente, questo nelle applicazioni reali limita troppo il numero e la posizione di nodi sotto un unico NBB e innesca continui cambiamenti di gruppo dei nodi, facendo di conseguenza aumentare le modifiche di HID, accrescendo la congestione.
3. Tutti i nodi che fungono da contatto tra layer, i cosiddetti nodi gerarchici, sono vulnerabili agli attacchi, quindi la distruzione di uno di questi nodi porta alla paralisi di tutto il suo cluster, uno svantaggio insostenibile in una rete da battaglia.

A causa di queste limitazioni è stato studiato un protocollo di routing che prende come base il LANMAR.

Landmark Ad Hoc Routing (LANMAR)

Il LANMAR è un protocollo di routing efficiente in reti wireless “piatte”, cioè non gerarchizzate, questo protocollo assume che la rete ad hoc su larga scala sia suddivisa in sotto reti logiche (cluster), dove i membri hanno “interessi” in comune e sono interessati a muoversi coerentemente (per esempio un gruppo di tank, un battaglione di fanti, un gruppo di colleghi a una conferenza).

Il LANMAR usa il concetto di confini (landmarks) per tenere traccia delle varie sotto reti, un approccio molto simile al “landmark hierarchical routing” proposto in [5] per reti cablate.

Analizzando più in dettaglio questo protocollo gli indirizzi di rete dei nodi mobili conterranno questo ID della sotto rete: ID: <Group ID, Host ID>.

Ciascun gruppo logico ha un nodo che ha la funzione di “confine”. Il routing per un confine (NC, nodo confine) è propagato attraverso la rete con un sistema Distance Vector [6] che gestisce dinamicamente le tabelle di routing e come schema di routing usa il Fisheye State Routing (FSR) con il concetto di “scope” per le operazioni locali, cioè ogni nodo ha una sua “area di competenza”, generalmente la portata dei suoi sistemi trasmissivi, che vede come una sua sotto-rete e che può essere gestita in modo diverso dal resto della rete.

Il LANMAR all'interno del Fisheye scope (FS) per il routing utilizza un algoritmo link state, invece per i nodi esterni al FS viene utilizzato solamente il landmark distance vectors.

All'interno del LANMAR gli aggiornamenti dei percorsi di routing avvengono in questo modo: Ogni nodo invia periodicamente informazioni sulla topologia della rete ai nodi più vicini, in questi aggiornamenti, un nodo, invia tutte le informazioni nell'ambito del proprio FS e ogni aggiornamento da questa fonte è numerato sequenzialmente, incapsulato nel messaggio di aggiornamento, il nodo inserisce il proprio vettore delle distanze di tutti i NC (landmarks).

Questo processo di scambi garantisce che ogni nodo abbia una mappatura molto dettagliata dei nodi all'interno del suo FS, e una vettore delle distanze e di routing di tutti gli altri nodi confine.

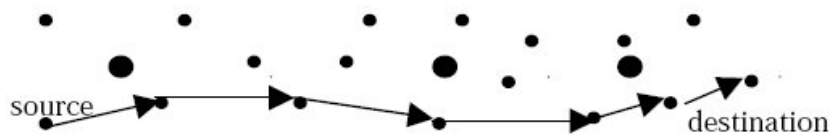
Quando un nodo ha bisogno di recapitare un pacchetto, se il destinatario è all'interno del suo FS sono disponibili accurate informazioni per il routing dalla FSR e il pacchetto può essere inoltrato direttamente, altrimenti il pacchetto potrà essere inoltrato attraverso il corrispondente NC corrispondente all'indirizzo della sotto rete logica di destinazione, contenuto nell'intestazione del pacchetto, In ogni caso se il pacchetto transita nel FS del nodo destinazione prima di raggiungere il NC viene instradato direttamente.

Lo schema del LANMAR riduce considerevolmente l'ampiezza della tabella di routing e il traffico sulla rete dovuto agli aggiornamenti delle tabelle di routing, quindi previene il traffic overhead, cosa molto apprezzata in reti mobili ad hoc su larga scala.

Tuttavia il LANMAR come descritto non può avere il vantaggio dei potenziali accorciamenti di percorso (short cut) offerti dall'architettura di rete gerarchica con backbone fisici e UAV, proprio a causa della sua concezione "piatta".

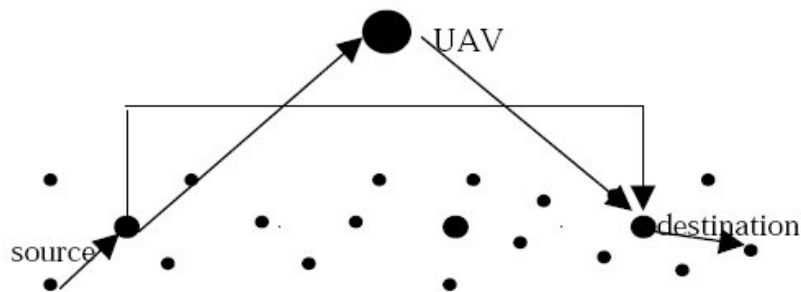
LANMAR su una Rete Gerarchica

Il LANMAR applicato a una rete wireless piatta, come visto nel paragrafo precedente per inoltrare pacchetti a un nodo destinazione deve far passare necessariamente il pacchetto attraverso il rispettivo NC lungo un percorso con molti salti nodo-nodo.



LANMAR routing in a "flat" ad hoc wireless network without UAVs

Ma il LANMAR in virtù del fatto che ha una logica intrinsecamente gerarchica può essere integrato in una struttura fisica gerarchica basata su UAV e NBB, dove il protocollo può inoltrare i pacchetti in un modo più efficiente: Il nodo sorgente inoltra attraverso il proprio NBB, che inoltrerà al NBB remoto vicino al NC, tutto attraverso il link di livello superiore, infine il NC inoltrerà direttamente al nodo destinazione al suolo. Questo diminuirà sensibilmente il numero di salti e quindi il sovraccarico della rete.



Per utilizzare i vantaggi degli "short cut" descritti sopra dobbiamo così modificare il LANMAR con i seguenti punti:

Primo, ogni nodo a terra, compresi i NBB utilizzano come algoritmo di routing nella rete wireless al suolo, che ha una copertura ristretta, il LANMAR originale. Questa è la base per poter ritornare a un modello di rete piatto nel caso venisse a mancare una connessione verso il livello superiore (l'unico UAV che connetteva un gruppo di nodi viene abbattuto).

Secondo, un NBB che ha una copertura radio molto ampia deve inoltrare ai NBB vicini e agli UAV il vettore delle distanze dei NC, attraverso la rete tra NBB. Il contenuto di questo pacchetto è identico al pacchetto di aggiornamento dei NC, quindi i NBB vicini interpreteranno questo pacchetto come un normale messaggio di aggiornamento, quindi i path del livello dei NBB, ovviamente molto corti essendo genericamente i NBB molto inferiori di numero rispetto ai nodi standard, andranno a sostituire i path del livello sottostante, molto più lunghi a causa dei percorsi con molti salti. Dai NC aggiorna i nodi al suolo, in modo da fargli conoscere il miglior percorso per i NC remoti, incluso il path che

utilizzano per la rete di livello più alto. Per inoltrare i pacchetti usando l'interfaccia radio corretta ogni NBB deve conoscere l'interfaccia radio e il path del prossimo nodo. Una caratteristica importante del protocollo LANMAR ampliato alle reti multilayer è l'affidabilità e la tolleranza agli errori, i nodi ordinari non conoscono il link al nodo più alto, ma questo viene scoperto automaticamente dai NBB.

Facciamo un esempio di cosa accadrebbe nel caso che un NBB di un gruppo venga distrutto, il path corto che passava attraverso questo nodo diviene inutilizzabile, allora le nuove informazioni, diramate sulla rete attraverso i nodi al suolo sostituiscono questo path scaduto con un nuovo path, quindi questi nodi tornano al vecchio protocollo di routing e usano path multi-hop per raggiungere i NC. Inoltre il NBB responsabile della comunicazione con l'UAV inoltra le informazioni di routing a tutti gli altri NBB attraverso gli UAV. Il path risultante sarà trattato come se fossero 2 salti.

Quindi, quando due NBB non si possono connettere a tutti gli altri nodi direttamente il path a due salti che passa attraverso gli UAV è la soluzione migliore. Nel peggiore dei casi, quando sia i NBB che gli UAV non sono operativi o non raggiungibili, questo sistema ritorna al protocollo originario per il routing su reti wireless ad hoc "piatte", cioè il LANMAR originale.

Lo schema di questa rete è di facile dispiegamento dato che le uniche possibili modifiche vengono apportate ai NBB.

Numero Ottimale di Nodi di Backbone

All'interno di una rete multilayer, come quella analizzata nei paragrafi precedenti un parametro importante è il rendimento ottimo della rete, dato da [3]:

$$\Theta\left(\frac{W}{\sqrt{N}}\right)$$

dove W è l'ampiezza della banda, Θ è la teta della notazione di Knuth [7] e N il numero totale dei nodi, compresi i NBB; apparentemente questo valore declina rapidamente all'aumentare del numero dei nodi, quindi per migliorare il rendimento dovremmo mantenere il numero dei nodi dipendenti da un singolo NBB abbastanza piccolo (per esempio sotto i 100).

Più specificatamente il rendimento di una rete al livello dei nodi sul terreno è dato da:

$$R_{local} = \Theta\left(\frac{W_1}{\sqrt{N/m}}\right)$$

dove m è il numero dei NBB. Invece il rendimento della rete di livello superiore composta da soli NBB è dato da:

$$R_{backbone} = \Theta\left(\frac{W_2}{\sqrt{m}}\right)$$

Ovviamente W_1 e W_2 saranno diversi dato che l'ampiezza di banda tra la rete di NBB e la rete tra nodi standard è probabilmente diversa.

Conoscendo il rendimento ottimo per le due interfacce di rete che un NBB utilizza possiamo calcolare il numero ottimale di NBB e quindi di cluster di nodi standard:

$$M = \frac{W_1}{W_2} \cdot \sqrt{N} + 1$$

con un numero basso di nodi standard

$$M = \frac{W_1}{W_2} \cdot \sqrt{N}$$

con un numero alto di nodi standard, più probabile in reti mobili reali.

Disposizione dei NBB e Clustering dei Nodi Standard

Dopo aver calcolato il numero ottimo di NBB il secondo parametro critico è il capire come posizionarli, il modo più semplice sarebbe quello di spargere uniformemente i NBB sul territorio interessato, sfortunatamente il posizionamento statico ha due grandi problemi, prima di tutto dato che ogni nodo della rete presa in considerazione è mobile anche i NBB saranno mobili, quindi può succedere che molti NBB si muovano verso uno stesso punto creando interferenze o conflitti o, al contrario, una certa area può essere lasciata senza copertura causando buchi nella rete. Il secondo problema è quello della tolleranza ai difetti, un NBB può subire guasti oppure venire distrutto, in questo caso dovrebbe venir disposto sul campo un nuovo NBB per sostituirlo, operazione che nel caso delle reti da battaglia è estremamente difficile se non impossibile.

Per risolvere questi problemi è stata attuata una politica molto semplice, se un qualsiasi nodo standard ha connettività con un altro NBB questo nodo è candidato ad essere "eletto" NBB e quindi creare un proprio cluster, per fare questo ogni nodo standard dovrà avere due interfacce di rete, la prima per comunicare sulla rete tra nodi standard e la seconda, utilizzata solo nel caso il suddetto nodo sia eletto, per comunicare nella rete superiore dei NBB, se poi un nodo è un apparato sprovvisto della seconda, e probabilmente, più potente interfaccia di rete, magari perché è un apparato molto semplice, questo nodo non sarà eleggibile. In questo modo se un NBB viene distrutto o diventa irraggiungibile viene rimpiazzato da un altro NBB scelto tra l'insieme di nodi standard eleggibili.

L'elezione di un NBB avviene dinamicamente, di conseguenza possono essere dinamicamente distribuiti sull'area da coprire. [8]

Clustering dei Nodi

Tra tutti gli schemi per decidere in che modo raggruppare i nodi, sono stati adottati per la loro semplicità il Lowest ID (LID) e l' Highest Degree (HD) [9] [10]. Fino ad adesso la ricerca si è concentrata su come ottenere buoni cluster, questo consiste principalmente in non sovrapporre i NBB, tuttavia la stabilità è un serio problema nelle applicazioni reali del clustering, specialmente quando il clustering è usato per supportare il routing, cosa che accade nelle reti da battaglia, dove la stabilità dei NBB è una richiesta imprescindibile. Purtroppo gli schemi visti non possono soddisfare questa richiesta, per questo è necessario utilizzare uno schema adatto a tale scopo, tipo il Random Competition based Clustering (RCC) [3]

L'idea principale di questo schema è che ogni nodo che non appartiene a un cluster può iniziare la formazione di un proprio cluster mandando sulla rete un pacchetto (cluster head claim packet, CHCP) dove afferma di essere un Cluster Head (CH), di conseguenza il primo nodo che invia un messaggio del genere viene eletto, ricorsivamente se un nodo senza cluster riceve notizia, tramite i pacchetti di aggiornamento delle tabelle di routing, che esiste un CH nelle vicinanze diventa membro di questo cluster.

I CH devono mandare periodicamente un CHCP per mantenere il loro stato avvertendo gli altri nodi della loro presenza. Dato che c'è un ritardo da quando un nodo manda un CHCP

a quando il pacchetto viene ricevuto dai vicini, è possibile che in questo tempo molti altri nodi mandino lo stesso pacchetto, per evitare questo ogni nodo prima di inviare un CHCP attende un tempo random, quindi viene applicata implicitamente la politica che il nodo più veloce vince e diventa un CH. Purtroppo la sola introduzione del timer random non elimina completamente il problema della concorrenza delle trasmissioni e nel peggiore dei casi si possono avere ancora conflitti sull'elezione dei CH, per questo quando ci sono due nodi eleggibili si controlla semplicemente l'ID del nodo e si avvantaggia quello con l'ID più basso.

Questo schema per il clustering risulta molto stabile e performante del LID e dell'HD, dato che ad esempio nello schema LID un CH riceve un CHCP da un nodo con ID più basso lascia il suo ruolo a questo nodo, allo stesso modo nello schema HD quando appare un nodo con molti vicini il cluster viene riformato ed essendo nodi mobili questo può succedere anche con una frequenza molto elevata, tale da degradare l'efficienza della rete. Al contrario nell'RCC un CH cede la sua carica solamente nel caso che un altro CH si avvicini, e poiché generalmente due CH sono ad almeno due hop di distanza questo schema risulta estremamente stabile.

Clustering Multi-Hop per la Elezione dei Nodi di Backbone

Di solito gli schemi per il clustering tra nodi standard non sono basati sugli hop perché è sempre possibile raggiungere un nodo con un solo passaggio, questo non è fattibile però per l'elezione dei NBB, anche perché è necessario decidere il numero approssimativo di NBB da creare, coerentemente a come visto in precedenza. Per fare questo dobbiamo estendere lo schema di clustering aggiungendo il concetto di k-hop clustering, questo significa che ogni CH può raggiungere un nodo standard all'interno del suo cluster con al più k salti. Aggiustando il valore di K possiamo approssimativamente controllare il numero dei CH, più alto è K meno saranno i CH e quindi i NBB.

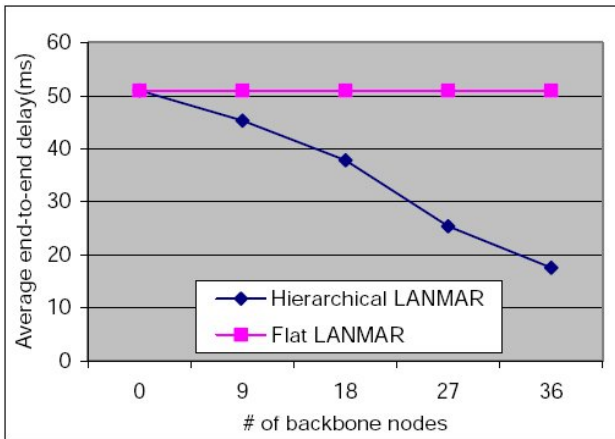
È da notare che ogni nodo eleggibile a NBB partecipa a questa elezione.

Per rendere questo schema di clustering multi-hop, ogni nodo deve inoltrare i messaggi CHCP al proprio CH e ogni nodo mobile deve selezionare il suo prossimo CH in base al scope del k-hop, se il nodo che fa richiesta di diventare CH è nel raggio di k salti viene accettato, altrimenti no.

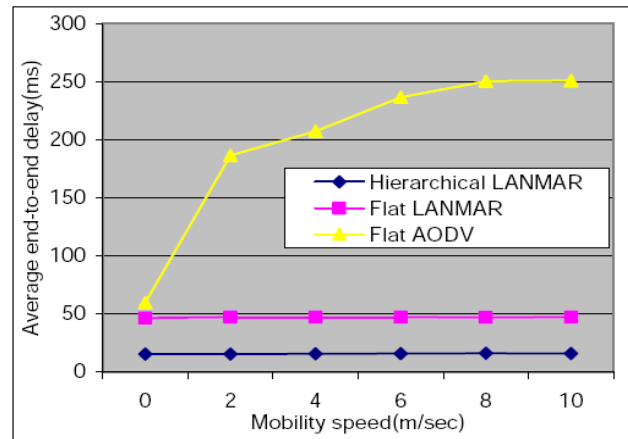
Valutazione delle Prestazioni

Per la valutazione delle prestazioni dell'algoritmo di routing LANMAR su reti piatte, LANMAR su reti gerarchizzate e schema di clustering RCC sono disponibili test con prodotti come il GloMoSim[11][12], sistema per la simulazione di reti ad hoc.

Di seguito riportiamo solamente alcuni grafici di esempio, per approfondimenti rimandiamo a [3].



Ritardo end-to-end in funzione del numero # di NBB



Ritardo end-to-end in funzione della velocità di spostamento dei nodi

4. Quality of Service nelle Reti Mobili ad Hoc su Larga Scala.

di Andrea Batistini

Introduzione

Nell'ambito delle future operazioni militari e della progressiva digitalizzazione del campo di battaglia la disponibilità di protocolli di tipo quality of service (QoS) efficienti diventerà un fattore chiave per determinare il successo di tutto il sistema dell'Internet Tattico applicato a situazioni operative.

Al contrario delle reti cablate, dove la disponibilità di banda normalmente non costituisce un problema, in una rete mobile ad hoc su larga scala da implementarsi su un campo di battaglia la scarsità della banda e la necessità di assicurare la trasmissione delle informazioni critiche necessitano di un generale ripensamento dei tradizionali metodi di QoS.

Innanzitutto che cos'è il QoS? Il Quality of Service viene definito dall'ITU (International Telecommunication Union) come "un insieme di requisiti di qualità connessi al comportamento di uno o più oggetti".

Nella fattispecie delle reti di computer il termine QoS si riferisce ad un meccanismo di controllo che può assegnare delle priorità, nell'utilizzo della banda disponibile, a determinati flussi di dati o a determinati utenti. È immediatamente chiara la necessità di implementare un protocollo di questo tipo in una struttura gerarchica come l'ambito militare.

Il problema dell'implementazione di politiche di QoS nelle reti mobili ad hoc non è nuovo e negli ultimi anni sono state proposte diverse soluzioni tecniche per risolverlo come lo SWAN [13], l'INSIGNIA [14] e il CEDAR [15]. I normali approcci che vengono utilizzati per le politiche di QoS come l'IntServ, in cui le applicazioni usano il Resource Reservation Protocol (RSVP) per richiedere e riservarsi risorse sulla rete, o il più innovativo DiffServ [16], nel quale i singoli pacchetti vengono marcati in accordo al tipo di servizio che necessitano, da soli, non riescono a soddisfare le richieste dell'architettura della rete in oggetto, rete costituita da migliaia di nodi eterogenei, con differenti capacità di trasmissione e in rapido movimento in un ambiente ostile.

Pare quindi interessante analizzare più nel dettaglio almeno una delle soluzioni più innovative proposte, come quella scaturita dallo studio effettuato da Kaixin Xu, Ken Tang e Rajive Bagrodia della Scalable Network Technologies Inc., con la collaborazione di Mario

Gerla del Computer Science Department di UCLA e di Michael Bereschinsky del CECOM RDEC STCD.

Il modello da loro proposto condivide molti dei concetti chiave con il protocollo SWAN, come il “probing” per individuare la minima banda disponibile su un dato percorso o il “controllo di ammissione” sul nodo sorgente, inoltre un’ulteriore comunanza è data dall’uso del “rate control” sui flussi best-effort (cioè quei flussi che utilizzano tutta la banda disponibile a meno che essa non venga richiesta da flussi a priorità più elevata) per rispondere ai problemi di congestione della rete.

La differenza fondamentale tra i due approcci sta nell’utilizzo da parte del modello in esame del protocollo di routing LANMAR di cui costituisce di fatto un’estensione e della struttura di rete gerarchica di tipo MBN (mobile network nodes).

Panoramica dell’Architettura

L’architettura proposta nel modello consta di 5 componenti principali che in seguito analizzeremo più in dettaglio. Essi sono: la gestione adattiva della banda (adaptive bandwidth management), un protocollo scalabile di instradamento (scalable QoS routing), un controllo di ammissione dei flussi di dati (admission control) e un controllo per la congestione dei singoli nodi (congestion control). Essa viene implementata sul protocollo di trasmissione IEEE 802.11 MAC. [17]

Vengono presi in considerazione due differenti algoritmi, uno a livello dei nodi sorgente e uno a livello dei nodi intermedi. A livello dei nodi sorgente i flussi real-time (quindi a priorità più alta) vengono filtrati attraverso il componente di admission control, successivamente vengono marchiati come pacchetti real-time (questo in quanto il protocollo IEEE 802.11 MAC non fa distinzione fra pacchetti best-effort e real-time, anche se esistono degli studi volti ad ampliare questo protocollo [18] e affidati al protocollo di instradamento per la consegna, il traffico di tipo best-effort, invece, viene immesso direttamente sulla rete, ma per garantire la priorità al traffico real-time viene modulato in base alla disponibilità di banda. Ai nodi intermedi, invece, viene affidata semplicemente la rilevazione e la gestione delle congestioni sulla rete nella loro prossimità. Analizziamo ora più dettagliatamente i singoli componenti dell’architettura.

Gestione Adattiva della Banda

Dato che la banda radio usata per la comunicazione, intorno ad un singolo nodo, è di fatto in condivisione con tutti i nodi vicini, in un dato istante; la misurazione accurata della disponibilità di banda per un singolo nodo non è un problema di facile soluzione all’interno di una rete senza fili con un grande numero di nodi. Per effettuare questa misurazione, quindi, sulla quale si basa la scelta della metrica di instradamento, si deve non solo considerare la capacità trasmittiva del singolo nodo, ma integrarla con la capacità dei nodi a lui vicini. Il modello esaminato propone una soluzione del tipo $BW_t = W(1 - R_t)$,

basandosi sull’analisi delle transizioni degli apparati tra i 4 stati tipici di una ricetrasmittente 802.11, cioè “busy”, “carrier sensing channel busy”, “virtual carrier sensing busy” e idle, considerando i primi 3 come stato di “occupato” e il quarto come “libero”

Dove, W è la banda dell’apparato (2 Mbps per una ricetrasmittente standard) e R_t è il rateo di utilizzazione della banda al tempo t , compreso tra 0 e 1.

L’informazione sulla quantità di banda disponibile viene generata periodicamente utilizzando il sistema di cui sopra ed utilizzata, da ogni singolo nodo, per determinare attraverso il componente call admission control se un dato flusso di informazioni

appartenente ad una particolare classe di servizio può essere accettato o meno dal nodo stesso. Se il flusso viene accettato la quantità di banda da esso utilizzata va a modificare dinamicamente la quantità di banda disponibile per gli altri flussi in arrivo o in uscita.

Instradamento del QoS

Dopo essere riuscito a misurare correttamente la banda disponibile il modello si propone di estendere il protocollo di routing per inglobare le informazioni sulla disponibilità della banda. I protocolli per le reti ad hoc si dividono in 2 categorie fondamentali, instradamento proattivo nominale (namely proactive routing) e l'instradamento su richiesta (on-demand routing), entrambi però manifestano alcuni problemi per reti di grandi dimensioni e con componenti in movimento, l'orientamento degli autori quindi si è indirizzato su un protocollo a struttura gerarchica. Il protocollo che viene utilizzato in questo modello è il LANMAR (Landmark Ad Hoc Routing). Esso fondamentalemente consiste di due protocolli indipendenti di instradamento: il local scoped routing e il landmark distance vector routing. A livello locale si può usare un qualsiasi protocollo di instradamento di tipo "flat" (un protocollo di tipo flat distribuisce le informazioni a seconda del bisogno ad ogni router che può essere raggiunto o che sia in grado di ricevere informazioni) come il Fisheye [19], con il quale il singolo nodo può calcolare la quantità di banda disponibile tra se stesso e tutti gli altri nodi nel suo raggio d'azione. Per raggiungere invece i diversi gruppi di nodi che compongono il protocollo LANMAR viene usata la propagazione del vettore di distanza che contiene informazioni sull'instradamento verso i vari gruppi, piuttosto che verso i singoli nodi dei rispettivi gruppi. L'informazione sulla quantità di banda viene inclusa in questo modo, ogni landmark (nodo principale e riferimento del singolo gruppo di nodi) calcola la minima (Min_{BW}) e la massima (Max_{BW}) quantità di banda verso ogni nodo del suo gruppo utilizzando il protocollo locale, dopodiché calcola la quantità di banda disponibile verso tutti gli altri landmark raggiungibili e a questo punto propaga l'informazione sulla rete. In questo modo ogni singolo nodo conosce in ogni momento, la banda disponibile verso i singoli nodi del suo gruppo (attraverso il protocollo locale), la banda disponibile verso tutti i landmark e la banda minima e massima verso ogni nodo appartenente ad il gruppo di nodi di uno specifico landmark.

Controllo di Ammissione

Attraverso il sistema sopra descritto ogni tipo di informazione sulla quantità di banda è già disponibile presso ogni singolo nodo, in questo modo il controllo di ammissione (Call Admission Control) può decidere se un determinato flusso è in quel momento accettabile o dovrà essere posticipato. L'algoritmo è molto semplice:

- Se la destinazione è un nodo nello stesso gruppo di nodi della sorgente e la banda è sufficiente il flusso di dati verrà trasmesso oppure in caso contrario rifiutato
- Se invece la destinazione fa parte di un altro gruppo il primo discriminante sarà la banda disponibile verso il landmark di destinazione se la banda fosse insufficiente il flusso verrà rifiutato
- Se la banda verso il landmark fosse sufficiente verranno esaminati i due parametri Min_{BW} e Max_{BW} , se la richiesta fosse minore di Min_{BW} il flusso verrà accettato, se invece la richiesta fosse maggiore di Max_{BW} il flusso verrà rifiutato, se in ultima istanza la richiesta di banda cadesse nell'intervallo tra il minimo e il massimo verrà inviato un pacchetto sonda per ottenere l'informazione sulla effettiva quantità di

banda disponibile verso il nodo destinazione e la decisione sarà presa in base al risultato.

Controllo di Congestione

Dato che la rete analizzata è composta da numerosi oggetti in rapido movimento e quindi la sua topologia cambia da un istante ad un altro, un singolo controllo, benché efficiente, all'atto dell'immissione di un flusso di dati sulla rete è ovviamente insufficiente per garantire la qualità del servizio desiderata in applicazioni di tipo militare, in quanto problemi di congestione di traffico possono verificarsi in qualsiasi momento. In questi casi il modello si propone di individuare questi problemi e nel caso di preferire i flussi di dati real-time riducendo la banda disponibile per i flussi best-effort.

Per ovviare a questo problema il sistema deve essere in grado di identificare le congestioni. In una rete cablata quando si verifica un collo di bottiglia i pacchetti iniziano a creare una coda fino al superamento dello spazio disponibile e alla conseguente perdita del pacchetto. In una rete dinamica però il parametro della lunghezza della coda non rappresenta più un valido sistema per determinare la congestione di un nodo, in quanto questa potrebbe ancora non essersi formata negli stadi iniziali della congestione. Il modello sfrutta quindi le informazioni fornite dal componente di gestione adattiva della banda; monitorando il rateo di utilizzo del canale radio del singolo nodo e definendo un valore di soglia possiamo affermare che un nodo sta entrando in uno stato di congestione quando il valore di utilizzo del canale radio supera il valore di soglia.

Nel momento in cui si verifica una congestione di un nodo verrà ridotta in esso la banda disponibile per i flussi di dati di tipo best-effort che fino a quel momento stavano utilizzando tutta la banda lasciata libera dai pacchetti real-time, questa operazione viene effettuata secondo lo schema Additive Increase Multiplicative Decrease (AIMD) [20]. Per assicurare che non avvenga un eccessivo deperimento dei flussi best-effort viene garantita una piccola frazione della banda disponibile a questi flussi in ogni momento.

Valutazione

Questo modello è stato valutato dai loro sviluppatori utilizzando il simulatore QualNet [12], esso si è rivelato estremamente scalabile sia per quanto riguarda la dimensione della rete sia considerando il movimento dei nodi. Il protocollo di quality of service, ideato dagli autori succitati, permette di portare i tempi di "average end-to-end packet delay" (tempo medio di transito del singolo pacchetto dalla sua sorgente alla sua destinazione), dagli oltre 300ms per una semplice rete LANMAR a circa 40ms, introducendo un ritardo, dovuto al tempo necessario al controllo di ammissione dei flussi per eseguire il suo compito, di mediamente 60 ms (per una rete di tipo MBN con 400 nodi). Questo tipo di implementazione del servizio QoS va a migliorare sensibilmente il già ottimo protocollo di routing LANMAR rendendo realmente possibile la gestione di una complessa rete di sistemi mobili, come quella che può esistere su un campo di battaglia, in modo efficiente e garantendo che le informazioni vitali siano privilegiate rispetto alla numerosissima quantità di dati che una rete di tipo Tactical Internet è in grado di generare e trasportare.

5. Agenti Mobili sul Campo di Battaglia

di Andrea Batistini

Introduzione

Da sempre il successo di un'operazione militare è dipeso in una qualche misura dall'accuratezza e dalla velocità con cui le informazioni venivano trasmesse a coloro che ne avevano bisogno per prendere le giuste decisioni. Al giorno d'oggi questo è ancora più vero. I nuovi sistemi elettronici con cui sono equipaggiati i mezzi da combattimento e addirittura i soldati stessi, forniscono, insieme a tutte le altre tecnologie di informazione e di rilevamento come satelliti e reti di sensori, una quantità di informazioni e di dati tali da rendere sempre più difficile e dispendioso, in termini di tempo e di capacità di calcolo, l'atto di ottenere l'informazione desiderata.

Di fronte ad un proliferare così incontrollato di dati si è reso necessario studiare, per un ambito critico come quello militare, una nuova serie di strumenti per l'estrazione di dati dai vari database presenti sul campo di battaglia. Attualmente la risposta a questa esigenza sembra essere fornita dagli agenti software mobili. Verrà data una breve panoramica introduttiva degli agenti software mobili e verrà preso in esame un modello pratico, sviluppato in ambito militare, del loro utilizzo.

Gli Agenti Software Mobili

Un agente software è un pezzo di software che esegue dei compiti seguendo le istruzioni dell'utente. Più sofisticato è un agente, minori saranno le istruzioni di cui avrà bisogno per eseguire il compito dato. Gli agenti mobili hanno l'abilità di "viaggiare" da un host all'altro in una rete e di svolgere i loro compiti direttamente sul sistema remoto. Hanno l'abilità di comunicare con altri agenti e sistemi e la capacità di muoversi su reti eterogenee.

Una loro fondamentale proprietà è l'autonomia, un agente infatti deve essere in grado di prendere tutte le decisioni necessarie per raggiungere l'obiettivo prefissato e piuttosto che essere pilotato dall'utente deve "cooperare" con esso [21].

Un agente mobile è formato da due cose: i dati (informazioni collezionate e lo stato del processo) e il codice (istruzioni che regolano il comportamento); esso si muove da host a host portando con sé i suoi dati e il suo codice, dopo essere arrivato sul nuovo host esso continua l'esecuzione del processo da dove si era fermato al momento di lasciare il vecchio host. Un agente deve essere in grado di andare in esecuzione su qualsiasi tipo di macchina su una rete, indipendentemente dalla sua architettura o dal tipo di processore installato su di essa. [22]

Vantaggi nell'Utilizzo di Agenti Mobili e loro Implementazione

Gli agenti mobili vengono utilizzati per implementare sistemi distribuiti orientati agli oggetti che siano flessibili e scalabili.

Essi presentano numerosi vantaggi:

- in caso di calcoli complessi questi possono essere spesso spezzettati in diverse parti discrete, gli agenti potrebbero occuparsi di portare queste parti su diverse unità di calcolo, inizializzare l'operazione e riportare indietro i risultati che potrebbe essere in seguito aggregati per ottenere il dato finale.

- in caso di applicazioni che necessitino di una risposta immediata ad un flusso di dati real-time, indipendentemente dalla disponibilità dell'utente
- in tutti quei sistemi inseriti in reti a scarsa affidabilità o parzialmente disconnesse sulle quali è spesso difficile ottenere l'informazione desiderata a meno di non scaricare grandi quantità di dati, molti dei quali si rivelerebbero inutili. Un agente in questo caso potrebbe accedere alle sorgenti di dati per filtrare le informazioni desiderate e riportarle indietro al richiedente senza la necessità che esso sia collegato alla rete durante l'elaborazione.

Uno degli aspetti più importanti degli agenti mobili riguarda la loro citata capacità di poter essere eseguiti su qualsiasi tipo di architettura, per questo sono stati utilizzati numerosi linguaggi di programmazione come il TCL, lo Scheme, il Rosette e anche il C, il linguaggio Java sembra essere però il più promettente sia per la sua semplicità di implementazione su praticamente ogni sistema sia per alcune peculiarità, come la RMI (Remote Method Invocation), che permette di chiamare dei metodi sugli oggetti attraverso la rete, e la serializzazione degli oggetti stessi.

Il server

La teoria dell'implementazione degli agenti mobili ci dice che essi dovrebbero contenere tutto il codice necessario al loro funzionamento, questo però non è facilmente realizzabile nella pratica, principalmente per motivi di sicurezza (il comportamento degli agenti mobili non è molto lontano dal comportamento di un classico virus), per questo nasce il bisogno di un server per gli agenti mobili. Il server risiede nel sistema host e fornisce un ambito operativo all'agente di modo da fornirgli le risorse necessarie a svolgere le sue operazioni e al tempo stesso limitarlo, nel consumo di risorse, nel numero di accessi ecc...

Il server si occupa anche di trasferire l'agente dove esso desidera oltre che di regolare l'accesso degli altri agenti che stanno facendo richiesta all'host dove il suddetto server è installato, molteplici agenti possono coesistere ed essere in esecuzione sullo stesso server.

L'agente

La parte più importante di tutto il sistema è però l'agente stesso, esso si sposta tra i server cercando di completare il compito assegnato, esso quindi deve essere in grado di programmare il miglior itinerario possibile e reagire di fronte alle difficoltà eseguendo azioni aggiuntive oppure spostandosi in altri ambienti dove potrebbe trovare le informazioni necessarie.

L'agente possiede un ciclo di vita che comincia con la sua *creazione* nel quale all'agente viene assegnato uno stato iniziale, un identificativo e viene preparato per le istruzioni successive. Segue l'*attivazione* che viene eseguita ogni qual volta l'agente raggiunge un nuovo host, qui viene inizializzato dal server ed esegue i propri compiti, al completamento del compito si ha la *disattivazione*, l'agente interrompe i suoi calcoli e salva il suo stato e i suoi risultati rendendosi disponibile per un eventuale trasferimento su un nuovo host o per l'esecuzione di procedure rischiose dal punto di vista della stabilità dell'agente. L'ultimo stadio del ciclo di vita dell'agente è la sua *terminazione*, in questo stadio l'agente termina ogni attività e libera le risorse che stava occupando, dopodiché il suo stato è perduto per sempre. Il singolo agente ovviamente può essere clonato conservando i dati dell'originale e il suo stato (ma con un id diverso) allo scopo di raggiungere il risultato in maniera più efficiente.

The Domain Adaptive Information System (DAIS)

Gli Agenti Mobili in Campo Militare

Il Domain Adaptive Information System (DAIS) sviluppato da M.O.Hofmann e K.R.Whitebread dei Lockheed Martin Advanced Technology Laboratories e A.McGovern del Computer Science Department dell'università del Massachusetts è un sistema di agenti mobili per il recupero e la disseminazione di informazioni in una rete militare. Il DAIS è un sistema studiato e costruito specificatamente per l'ambito militare e mostra i vantaggi di un sistema ad agenti mobili nel recupero e nella disseminazione di informazioni all'interno di una rete intrinsecamente inaffidabile, con scarsa banda disponibile e connessa ad un grande numero di sistemi diversi, che spaziano dai computer palmari ai grandi server, come quella di un campo di battaglia. L'implementazione del sistema trae ispirazione dai vari tipi di sistemi attualmente in uso per quanto riguarda gli agenti mobili, come l'Aglets dell'IBM [23] o l'Odissey della General Magic [24].

I principi base sui quali si è basata la struttura del DAIS sono semplici:

- spostare l'attività di calcolo in prossimità dei dati che devono essere lavorati. Dato che in una rete militare la quantità di dati che viene generata è immensa, lo spostamento della piccola parte di codice necessaria per delineare un agente riduce notevolmente il carico sulla rete rispetto al movimento della massa dei dati.
- essere un sistema di tipo "Launch & Forget" (lancia e dimentica). L'agente opera in maniera autonoma nella rete, questo è un requisito essenziale sul campo di battaglia dove la connessione alla rete è necessariamente sporadica e inaffidabile. L'agente può essere lanciato e dimenticato ed, essendo esso persistente, attenderà pazientemente che le sorgenti di dati diventino disponibili, per elaborarli, consegnando puntualmente i dati desiderati al richiedente nel momento in cui questo tornerà a connettersi.

Architettura del DAIS

L'approccio costruttivo del DAIS separa gli agenti mobili in due categorie:

- gli agenti di esecuzione (task agent), il cui compito è viaggiare attraverso la rete ed eseguire compiti su richiesta dell'utente
- gli agenti di servizio (service agent), che risiedono su un host e sono incaricati di preprocessare i dati locali e renderli disponibili ai task agent.

I service agent risiedono sul computer host in un dock cioè in una specie di ricettacolo software apposito che si occuperà di ricevere e gestire i task agent. Nella sua accezione più globale il dock, detto esteso, è formato dal dock vero e proprio (dock proper), dall'agente di servizio per l'infrastruttura (infrastructure service agent) e dall'agente di servizio per la gestione dei dati (information service agent).

Il servizio per l'infrastruttura contiene a sua volta due servizi, il servizio di monitoraggio della rete (network monitor service) e il servizio per il codice (code service); analogamente anche il servizio per i dati è suddiviso in due sottoservizi, il servizio per i database (database service) e il meta-servizio per il dominio (domain meta-service).

Ogni servizio svolge un compito specifico che andremo ad analizzare più nel dettaglio.

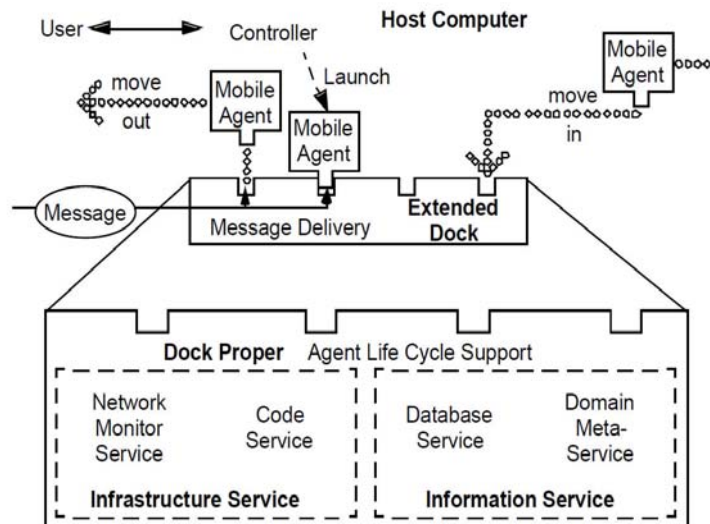


Figura 1. Architettura del sistema

Network Monitor Service

Monitorizza lo stato delle connessioni sulla rete verso gli altri host, notifica ai task agent se l'host richiesto è disponibile o meno e più in generale media le loro richieste di connessione verso gli altri host.

Code Service

Permette agli agenti mobili, al cambio di host, di portarsi dietro soltanto il codice necessario per la loro ricostituzione e reinizializzazione, nonché i risultati raggiunti, fornendo agli agenti tutte le parti di codice che rimangono invariate da un host all'altro. Questo permette un ulteriore risparmio di banda riducendo ulteriormente le dimensioni dell'agente mobile. L'impronta di un tipico agente sulla rete si attesta infatti tra 8 e 15 KBytes.

Database Service

Si occupa di interagire con i vari database rendendo così indipendente il lavoro dell'agente mobile che non necessita di conoscere in anticipo il tipo di database che dovrà interrogare. Questo permette di poter ottenere informazioni con un'unica richiesta da diversi tipi di database sia che essi siano SQL, Microsoft Access o di tipo militare come il Warrior/Warlord.

Domain Meta-Service

Fornisce una mediazione tra le richieste degli utenti e le fonti di informazione. All'interno del dominio viene creato un vocabolario di termini concettuali e di modelli convenzionali di rappresentazione che vengono usati per costruire degli schemi che convertano concetti astratti in campi effettivamente ricercabili nei vari database. Questo permette di utilizzare un'interfaccia uniforme ma capace di accedere a diverse tipologie di sorgenti di informazione. Il servizio lavora in modo dinamico aggiornandosi quando vengono creati nuovi database o vengono rilevati nuovi host contenenti database aggiuntivi.

Gli Agenti Mobili

Sono stati sviluppati diversi tipi di agenti mobili secondo le necessità tipiche di gestione dell'informazione su una rete militare. Tra i vari agenti disponibili citiamo gli agenti "add" e "forward" che si occupano di inserire informazioni nei database, l'agente "SQL query" che sfruttando il database service agent si occupa del recupero di informazioni, l'agente

“abstract query” che guidato e supportato dal domain meta-service agent è in grado di scovare informazioni rilevanti da diverse tipologie di database distribuiti, agenti che cancellano i dati o i record di un database ecc...

Il Meccanismo di Query Astratta

Il meccanismo di query astratta traduce la query (richiesta di una informazione in un database) dell'utente, espressa in termini e concetti generali, in molteplici query SQL secondo gli specifici schemi dei vari database da interpellare. Il domain meta-service interviene appunto fornendo una lista di termini astratti utilizzabili e mantiene un indice per riferirli agli schemi propri dei vari database. Una volta che dai vari database sono state estratte le informazioni necessarie l'agente di query astratta si occupa anche di presentarle in modo uniforme.

I Risultati e lo Sviluppo del DAIS

Il DAIS è stato collaudato con successo in alcune esercitazioni militari e dopo i test è diventato addirittura parte integrante del sistema di gestione delle informazioni di alcuni reparti operativi dell'esercito americano come il 14th CI/HUMINT Battalion (dove CI sta per Counter Intelligence e HUMINT per Human Intelligence). I suoi risultati sono stati abbastanza sorprendenti, infatti di fronte ad un tempo di addestramento per il personale di circa 4 ore, ha portato il Delivery Delay (tempo intercorrente tra la generazione di un rapporto e il momento in cui questo raggiunge tutti i suoi destinatari) da una media di 4 ore a circa 10 minuti ed un Delivery Rate (tasso di rapporti che riesce a raggiungere i destinatari indicati) da meno del 50% a quasi il 100%. Questi ed altri dati incoraggianti mostrano che sistemi come il DAIS, nonostante abbisognino di alcune modifiche e correzioni, rappresentino una strada innovativa ed estremamente efficiente per relazionarsi al problema della gestione delle informazioni in sistemi distribuiti.

La migrazione verso architetture ad agente comune (UMA – Universal Mobile Agent), con agenti riprogrammabili dinamicamente ed in modo intelligente, capaci di generare “figli” al fine di risolvere in modo più efficiente le situazioni critiche ed inoltre l'implementazione completa del sistema in linguaggio Java, porteranno in futuro a sistemi ancor più efficienti, affidabili e probabilmente esportabili in ambiti diversi da quello militare.

6. Conclusioni

E' chiaro che la guerra come la conosciamo noi sta indiscutibilmente per cambiare, anche se le soluzioni proposte non sono ancora implementate nei vari eserciti, la tendenza è ormai delineata. Nel giro di pochi anni è quindi plausibile pensare che vedremo un campo di battaglia completamente diverso da quello che siamo abituati a vedere negli attuali telegiornali e reportage.

Se potessimo sorvolare la linea del fronte fra 20 anni da adesso davanti ai nostri occhi vedremmo un panorama degno dei migliori libri di fantascienza: mezzi radiocomandati o addirittura autonomi che solcano i cieli in lungo e in largo recuperando informazioni essenziali per i vertici militari, squadre di robot cingolati impegnati in attività di sminamento o di ricerca del nemico, soldati bardati come cavalieri medievali ad alto contenuto tecnologico sorretti da esoscheletri per potenziare le loro capacità fisiche, squadre di tecnici-soldato impegnati in guerre elettroniche nel tentativo di distruggere l'infrastruttura informatica del nemico e, chissà, forse anche veri robot che sostituiranno i soldati in un futuro sempre più vicino. Allo stato attuale quasi tutti i paesi sviluppati hanno in cantiere

progetti di questo tipo, l'analisi del sistema americano è dettata esclusivamente dal fatto che grazie agli ingenti stanziamenti di fondi e alla generale superiorità tecnologica dell'industria bellica statunitense, il progetto stesso è in fase di avanzato sviluppo e di test. Già dal 2000 la 4th Infantry Division è stata dotata di tecnologie per supportare il sistema FCS. La non esistenza di uno standard per questo tipo di tecnologie, specialmente a livello infrastrutturale, nemmeno in seno ad un'alleanza come la NATO, ci porta a considerazioni negative sulla capacità di integrazione futura degli eserciti alleati e alla loro capacità di operare contestualmente sugli stessi teatri bellici in missioni multiforme come quelle in corso attualmente nella maggior parte degli scenari internazionali.

Questa grande spinta alla modernizzazione in un campo così vasto come le forze armate sicuramente avrà delle ricadute nella nostra vita di tutti i giorni, come d'altronde è sempre stato, si pensi al sistema GPS studiato per scopi militari, ma che in pochi anni ha rivoluzionato il nostro modo di viaggiare. Le nuove tecnologie sviluppate per la digitalizzazione del campo di battaglia potrebbero essere utilizzate per numerosi scopi civili e sociali. Ad esempio la rete proposta per l'infrastruttura informatico-tattica sarebbe facilmente ed auspicabilmente implementabile per il controllo del traffico automobilistico e degli spostamenti sulle nostre strade creando un vero e proprio sistema di auto "intelligenti". Il protocollo di QoS potrebbe essere una base per rendere i contenuti di internet finalmente fruibili in modo rapido e efficace sui nostri palmari e cellulari. Gli agenti software, che sono ancora nella fase embrionale del loro sviluppo, potrebbero divenire utili per superare le difficoltà di gestione dell'enorme quantità di dati disponibili su internet, a causa della quale i motori di ricerca si trovano sempre più difficoltà nell'individuare i contenuti desiderati; viceversa si pensi all'efficienza di una ricerca con un agente software, capace di riconoscere il volto di un criminale, "sguinzagliato" sulle migliaia di database di videosorveglianza o sugli archivi delle polizie mondiali.

Riguardo all'applicazione puramente bellica di queste tecnologie abbiamo una grande paura e una piccola speranza. La nostra paura è che una guerra sempre più impersonale ed asettica, mediata da strumenti elettronici, porti i nostri governanti ad una minore riflessione sui rischi e sulle conseguenze di un'azione bellica contro altri popoli. La piccola speranza è che nel momento in cui la guerra appaia necessaria od inevitabile, tralasciando qualsiasi tipo di discussione sull'effettiva utilità della guerra stessa, questo balzo tecnologico possa almeno contribuire a diminuire il numero delle vittime del conflitto. Riteniamo inoltre che sistemi di questo tipo andranno ad aumentare sempre di più il divario di potere esistente fra paesi sviluppati e non, con il rischio che per alcune nazioni la guerra diventi soltanto un costo di mezzi e non più di uomini, con tutto quello che comporta una visione di questo tipo nei confronti di quei popoli che continueranno ad avere un esercito tradizionale.

7. Bibliografia

- [1] Future Combat System. *Disponibile presso*
<http://www.globalsecurity.org/military/systems/ground/fcs.htm>.
- [2] <http://www.army.mil/fcs/systems.html>
- [3] K.Xu, X.Hong, M.Gerla . "An ad hoc network with mobile backbones*", *Computer Science Department at UCLA, Los Angeles, CA.*
- [4] www.army.mil/fcs/network.html.

- [5] P.F.Tsuchiya. "The landmark hierarchy: a new hierarchy for routing in very large networks". *Computer Communication Review*, vol. 18, No. 4. 1988.
- [6] http://it.wikipedia.org/wiki/Distance_Vector.
- [7] http://en.wikipedia.org/wiki/Donald_Knuth.
- [8] S.Banerjee, S.Khuller. "A clustering scheme for hierarchical control in multi-hop wireless networks". *IEEE Infocom 2001, Anchorage, Alaska*. April 2001.
- [9] C.R.Lin, M. Gerla. "Adaptive clustering for mobile networks". *IEEE Journal on Selected Areas in Communications*, Vol. 15, No. 7, Sep. 1997, pp. 1265-1275. 1997.
- [10] M.Gerla, J.T.Tsai. "Multicluster, mobile, multimedia radio network". *ACM-Baltzer Journal of Wireless Networks*, Vol.1, No.3, pp.255-65, 1995. 1995.
- [11] M.Takai, L.Bajaj, R.Ahuja, R.Bagrodia, M.Gerla. "GloMoSim: a scalable network simulation environment". *Technical report 990027, UCLA, Computer Science Department*. 1999.
- [12] QualNet. Simulatore di Reti. *Disponibile all'indirizzo <http://www.qualnet.com>*. June 2007.
- [13] G.Ahn, A.Campbell, A.Veres, L.Sun. "SWAN: Service differentiation in stateless wireless ad hoc networks". *Proceedings of IEEE INFOCOM 2002*, Jun. 2002.
- [14] S.Lee, G.Ahn, X.Zhang, A.Campbell. "INSIGNIA: An IP-based quality of service framework for mobile ad hoc networks". *Journal of Parallel and Distributed Computing (JPDC)*, 60(4), Apr. 2000.
- [15] P.Sinha, R.Sivakumar, V.Bharghavan. "CEDAR: a core-extraction distributed ad hoc routine algorithm". *Proceedings of IEEE INFOCOM 1999*, Aug. 1999.
- [16] S.Blake, D.Black, M.Carlson, E.Davies, Z.Wang, W.Weiss. "An architecture for differentiated services". *RFC 2475*, Dec. 1998.
- [17] P.Brenner. "A technical tutorial on the IEEE 802.11 protocol". *Disponibile all'indirizzo http://www.sss-mag.com/pdf/802_11tut.pdf*
- [18] S.T.Sheu, T.F.Sheu. "DBASE: A distributed bandwidth allocation/sharing/extension protocol for multimedia over IEEE 802.11 ad hoc wireless LAN". *Proceedings of IEEE INFOCOM 2001*, Apr. 2001.
- [19] G.Pei, M.Gerla, T.W.Chen. "Fisheye state routing in mobile ad hoc networks". *Proceedings of ICDCS 2000 workshops*, Apr. 2000.
- [20] D.M.Chiu, R.Jain. "Analysis of the increase and decrease algorithms for congestion avoidance in computer networks". *Computer Networks and ISDN Systems*, 17(1-14), 1989.

- [21] P.Maes, Agents that Reduce Work and Information Overload. *Communication of the ACM*, Vol. 37, No. 7, July 1994.
- [22] D.Horvat, D.Cvetkovic, V.Milutinovic, P.Kocovic, V.Kovacevic. Mobile Agents and Java Mobile Agents Toolkits. *Proceedings of the 33rd Hawaii International Conference on Systems Sciences*, 2000
- [23] IBM Aglets Workbench, Programming Mobile Agents in Java. *Disponibile all'indirizzo <http://www.trl.ibm.co.jp/aglets/whitepaper.htm>*.
- [24] General Magic Odissey. *Disponibile all'indirizzo <http://www.genmagic.com/agents>*