



Universitat de Girona
Escola Politècnica Superior

Projecte/Treball Final de Carrera

Estudi: Eng. Tècn. Informàtica de Sistemes. Pla 2001

Títol:

**Incorporació de mesures d'ample de banda disponible a
l'algorisme d'encaminament AntNet-Qos**

Document: Resum

Alumne: Xavier Vallejo López

Director/Tutor: Liliana María Carrillo Flórez

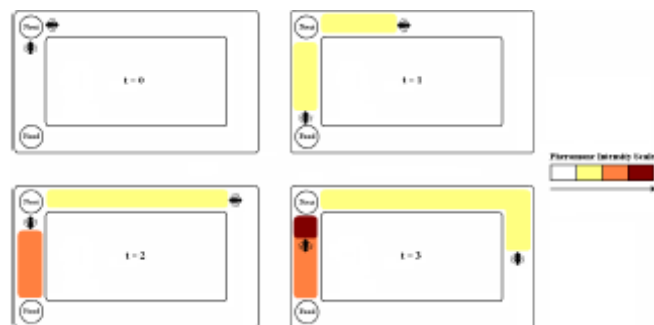
Departament: Electrònica, Informàtica i Automàtica

Àrea: Arquitectura i Tecnologia de Computadors

Convocatòria (mes/any): 09/06

Resum del projecte:

En el grup d'investigació BCDS de la Universitat de Girona, s'ha estat desenvolupant i experimentant un algorisme anomenat AntNet-QoS per gestionar diferents tipus de trànsit oferint Qualitat de Servei (QoS) en una xarxa de Diferenciació de Serveis (DiffServ). Per adaptar-se a les necessitats de la xarxa, implementa uns agents de control (paquets anomenats formigues) que l'analitzen, i actuen sobre els encaminadors (routers) variant les rutes del trànsit si és necessari. Els paquets de control s'anomenen formigues perquè es desenvolupa una metodologia semblant a la de les formigues reals, que consisteix en la capacitat d'anar a buscar menjar marcant els camins més curts possibles amb feromones.



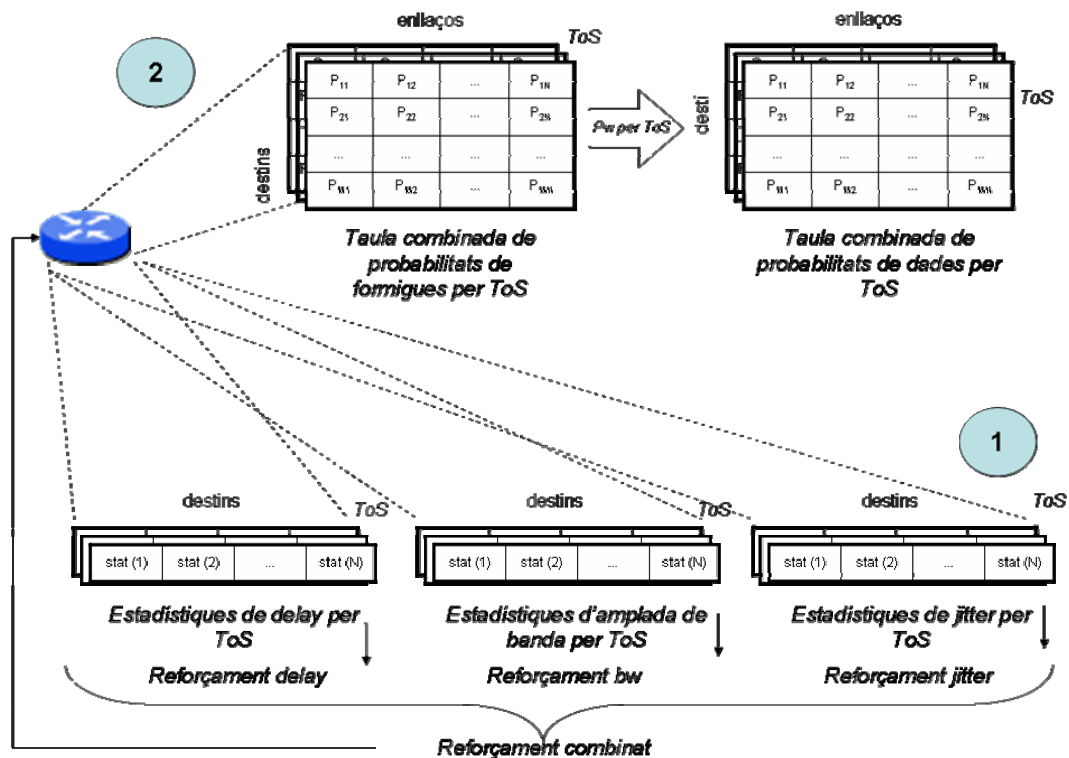
Imatge 1: El camí més curt es va actualitzant amb un color més fort (representa en la vida real a les feromones que van deixant pel camí) i significa que és un camí millor que l'altre

Aquest projecte final de carrera pretén investigar i experimentar una nova línia de desenvolupament d'algorismes dinàmics. A partir de l'algorisme AntNet-QoS s'han incorporat noves mesures (mesura de l'amplada de banda disponible i jitter), les quals combinant-les amb la mesura de delay (retard) ja feta servir, ens permet adaptar-nos millor a les condicions actuals del trànsit en la xarxa i als requeriments específics de qualitat (QoS) per part del trànsit.

Per poder realitzar les estimacions d'amplada de banda i de jitter, ens calen enviar trens de formigues cap a cert destí. Un tren de formigues consisteix en enviar una ràfega de formigues alhora. Observant en quins instants arriben tant en valors absoluts de temps, com en valors relatius entre elles, obtenim les dades necessàries per realitzar els càlculs de l'amplada de banda, el retard i la variació del retard.

Cada node de la xarxa disposa d'un **router** que ha de ser capaç de dirigir tot el trànsit que li arriba cap a les seves destinacions. En l'algorisme AntNet-QoS (i el d'aquest projecte), hi ha dues taules d'encaminament (una de formigues i una altra de dades), que a través de probabilitats seleccionen un destí. Aquesta probabilitat és directament proporcional a la qualitat del camí (per un camí que sigui curt, amb poc trànsit disposarà d'una bona probabilitat). Les formigues són les encarregades de modificar aquestes prioritats, augmentant-ne unes i disminuint-ne les altres. Per tant, gràcies a aquests agents de control, es pot calcular alguns paràmetres de la xarxa, i aquests valors es guarden en taules d'estadístiques que actuen a les taules d'encaminament.

Aquests càlculs es realitzen dins de cada router, per cada ToS i destí que es vol anar. A partir de les estadístiques recol·lectades, es calcula un valor anomenat reforçament, que serà el que determinarà si el camí és bo o no. Un cop s'obté el valor final a partir de les tres mesures (retard, amplada de banda i variació del retard), es procedeix a actualitzar les taules d'encaminament. El següent dibuix resumeix aquest procés.



Imatge 2: Esquema d'una estructura interna d'un router en un node

A més a més d'un router, cada node de la xarxa té el seu propi **generador de formigues**. Aquest mòdul té com a missió la generació de formigues i trens de formigues. És important fer aquesta distinció, ja que per la QoS que faci servir trens, aquesta només enviarà trens de formigues i no formigues normals. A l'enviar trens, les taules d'encaminament faran servir les tres mesures (delay, bandwidth i jitter). En canvi, si se li defineix sense tren, només enviarà formigues soltes, i les taules d'encaminament només es veuran afectades per les mesures de delay (com a l'AntNet-QoS).

El comportament tradicional de les formigues consisteix en el següent:

1. Les formigues es creen cada Δt de temps, i s'envien inicialment cap a destins aleatoris per tal d'anar explorant tota la xarxa en busca de bons camins i després s'envien més formigues cap als destins més probables del trànsit.
2. Cada formiga actua al mateix temps, conjuntament, amb les altres. Les formigues es comuniquen entre elles modificant les dades per on passen.
3. A cada salt de la formiga, aquesta calcula les dades i mesures fins a arribar al destí. Per arribar al destí, en cada node es decideix el pròxim salt.
4. Quan arriba al destí, es genera una formiga de tornada, que conté la ruta per on ha anat.
5. Aquesta formiga, **sí**, modifica les taules d'encaminament de cada router per on passa. Es modifiquen segons els resultats calculats sobre la qualitat del camí (les mesures).
6. En arribar al destí, la formiga de tornada (o sigui, en el node d'origen de la formiga d'anada) mor.

A l'incorporar la possibilitat de generar trens de formigues, sorgeixen nous paràmetres (tamany dels paquets, nombre de formigues del tren...) que han estat creats des de zero. Els routers també s'han d'adaptar a aquest canvi i poder gestionar-ho.

Hem parlat de formigues d'anada i formigues de tornada. El seu comportament és diferent, ja que no es gestionen per igual. Les formigues d'anada agafen les dades, i les calculen. En canvi, les formigues de tornada s'encarreguen d'actualitzar els nodes per on han passat les d'anada. Qui decideix per on passa per arribar a un destí, són les formigues d'anada (es poden veure com formigues exploradores).

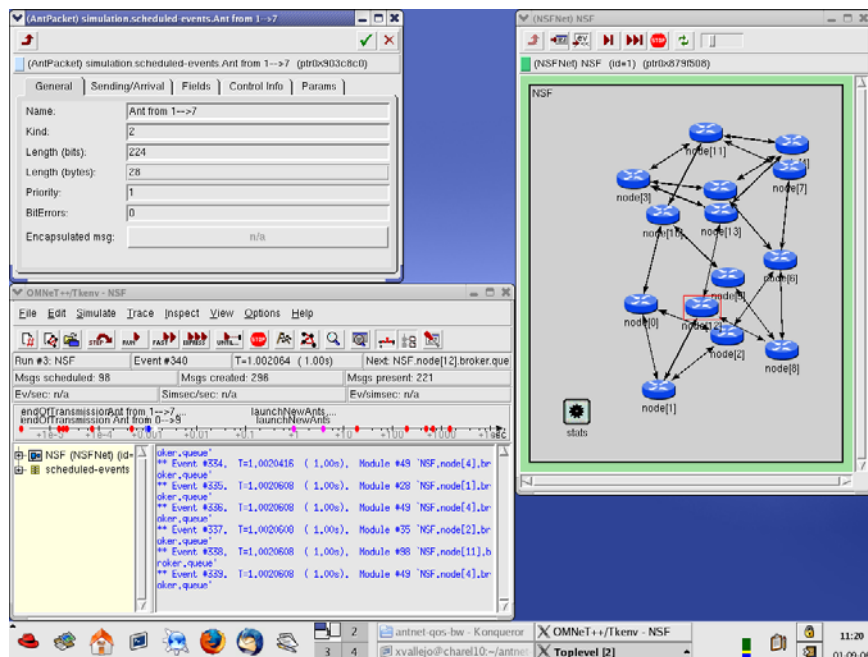
Pot semblar fàcil realitzar el canvi només aplicant les noves mesures, però en realitat sorgeixen noves complicacions. En destaquem una en concret, que són els bucles (cicles) que pot realitzar una formiga. En l'AntNet-QoS normal, si es detectava que una formiga feia un cicle, aquest s'eliminava de la seva ruta i la formiga de tornada no el feia. Funcionava bé perquè només es realitzaven mesures de delay.

En canvi, amb la incorporació de mesures d'amplada de banda i variació del retard, aquesta estratègia no funciona, perquè si s'eliminen els cicles, llavors les mesures són errònies.

Una mesura adoptada ha estat que no s'eliminin els cicles que faci una formiga d'un tren de formigues (en les dades sí els eliminem ja que no es fan mesures, i les formigues que no formen el tren, no mesuren bw ni jitter).

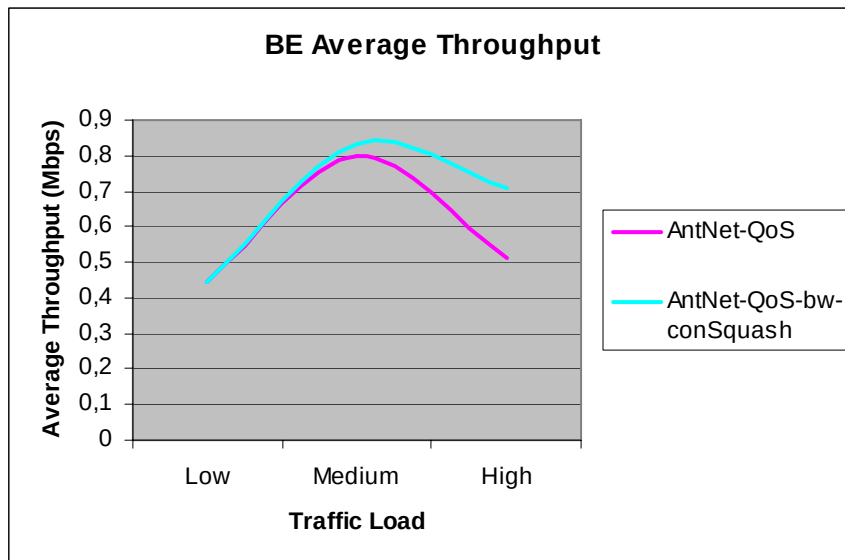
En principi, encara que la formiga realitzi un camí més llarg, aquest es farà poques vegades ja que el resultat que ha de donar és el d'un mal camí, i per tant es reforçarà un camí diferent, i probablement més curt.

Totes les millores que s'han implementat en l'algorisme AntNet-QoS s'han realitzat amb el simulador OMNeT++, que fou utilitzat anteriorment. L'OMNeT++ és un simulador d'events discrets, el qual és **gratuït** i de **codi lliure**, que compta amb un gran suport d'usuaris. Disposa de moltes utilitats que facilita la programació de models de xarxes, i altres usos. Es pot fer funcionar en gairebé tots els Sistemes Operatius i arquitectures, ja que es basa en un entorn UNIX per fer-lo funcionar.

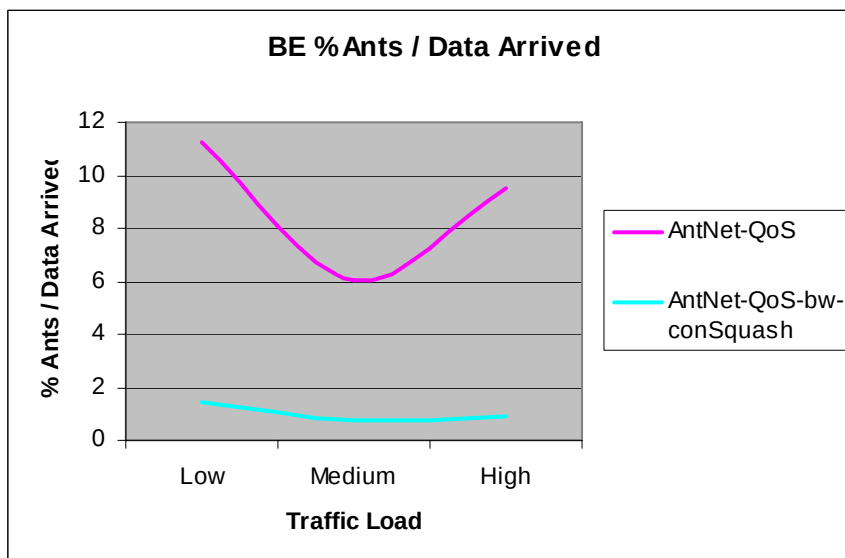


Imatge 3: L'algorisme AntNet-QoS en un entorn Linux en funcionament

Exposant alguns resultats obtinguts de la comparació entre l'algorisme original AntNet-QoS i el nou algorisme amb les mesures d'amplada de banda i variació del retard, tal i com es pot observar en les següents gràfiques, el nou algorisme aconsegueix millorar notablement el rendiment i, molt important, sense augmentar el consum d'overhead (temps d'execució que no forma part de l'usuari).



Imatge 4: En una classe de QoS BE (Best Effort), a mesura que augmentem la càrrega de trànsit, l'AntNet-QoS amb les noves mesures, obté un millor resultat en obtenir més velocitat.



Imatge 5: La quantitat de formigues necessàries per gestionar el trànsit, ha disminuït notablement gràcies a la millor gestió del nou algorisme.