

Algoritmi de rutare bazați pe informațiile de poziție ale nodurilor într-o rețea ad-hoc

Petruș Andrei
Universitatea "Politehnica" din București
București, Romania
Email: andrei.petrus@yahoo.com

Introducere – Folosirea pe scară largă a rețelelor complexe de senzori wireless, în mod particular pentru aplicații precum monitorizarea mediului înconjurător (apă, sol, agricultură etc.) impune ca nodurile componente ale rețelei să fie de dimensiuni foarte reduse, ușoare, flexibile și heterogene. Problema localizării, în esență determinarea poziției fizice a unui nod în rețea, este de actualitate și ridică destul de multe probleme; cu toate acestea, este un parametru crucial pentru majoritatea aplicațiilor amintite mai sus. Considerentele practice precum dimensiunea redusă, forma, costul și constrângerile energetice ale nodurilor exclud din start implementarea soluțiilor de localizare bazate exclusiv pe obținerea poziției prin GPS pentru toate nodurile rețelei.

În prezent rețele de senzori wireless, dezvoltate pentru a reliza sarcinile expuse anterior, pot conține sute sau chiar mii de noduri. Datorită proprietăților pe care trebuie să le respecte aceste rețele, precum consumul scăzut și eficientizat de energie sau întindere foarte mare dar și datorită nenumăratelor restricții de proiectare și a resurselor limitate de care dispun la nivel de nod, fac din studierea protocoalelor de rutare o temă interesantă.

Sunt foarte importante caracteristicile de stabilitate și de robustețe a rețelei, iar tratarea problemei

aparitiei buclelor de rutare este o adevărată provocare. De asemenea, este de dorit obținerea unei perioade cât mai mici de convergență a rețelei privind atât din perspectiva configurării inițiale a acesteia precum și în urma modificărilor de topologie ce pot apărea pe parcursul funcționării.

Un alt aspect important este viteza pe care protocolul de rutare o asigură transmiterii pachetelor ce conțin datele prelevate de la senzori, urmărindu-se în această direcție calcularea de rute optime din perspectiva timpul necesar unui pachet să ajungă la destinație (costul de transmisie), dar și la un consum cât mai mic de energie, adică implicit un număr cât mai mic de transmisii necesare.

Conceptul general. Protocoalele de rutare bazate pe localizare reprezintă o îmbunătățire a performanțelor într-o rețea prin introducerea informației de localizare în procesul de selecție al rutei.

GPSR – Greedy Perimeter Stateless Routing

Protocolul GPSR utilizează informația de localizare pentru reducerea costurilor și obținerea unei scalabilități mai bune atunci când distanța între terminalele mobile și dimensiunea rețelei crește. GPSR preia informația pozițională ca și metrică cheie în direcționarea pachetelor și utilizează o metodă simplă „greedy”.

LLR - Location-aware long-lived route selection

Protocolul LLR (Location-aware long-lived protocol) aparține familiei de protocoale de rutare on-demand. Protocolul utilizează informația de localizare urmărind minimizarea rutelor greșite și ca o consecință minimizarea utilizării procedurii de reconstrucție a rutelor.

DREAM - Distance routing algorithm for mobility

Algoritmul DREAM folosește informația de localizare cu scopul de a reduce informația de rutare. Protocolul combină abordările proactive și reactive, bazându-se pe actualizările terminalelor pentru propagarea informației de localizare și procesul de inundare la transmiterea pachetelor spre destinație.

LAR - Location-aided routing

Protocolul LAR este un protocol de rutare on-demand. Pentru a stabili o rută între sursă și destinație se bazează pe un proces de inundare pornit de sursă prin stabilirea unui pachet de cerere de rutare prin broadcast (RRQ Route Request). Pachetul este trimis de nodurile din rețea până când ajunge la destinație sau timpul de transmitere expiră în cazul în care ruta este calculată greșit.

Descriere generală a rețelelor de senzori wireless

Rețelele de senzori wireless sunt alcătuite din sisteme autonome dotate cu senzori răspândite spațial care interacționând între ele permițând monitorizarea diferitelor condiții fizice ale mediului precum temperatura, sunetul, vibrațiile, presinea, mișcarea și poluarea. Inițial ele au fost dezvoltate în scopuri militare, dar datorită costului redus, al

mobilității oferite și a posibilității de extindere practic nelimitate, rețelele au ajuns să fie folosite în multe domenii civile și industriale.

Pe lângă senzorii de care dispune, un nod al unei rețele de senzori este echipat cu un transceiver radio (sau un alt dispozitiv de transmisie), un microcontroler folosit la procesarea datelor la nivel de nod și de o sursă de energie (de cele mai multe ori o baterie).

Uzual rețeaua de senzori se constituie într-o structură ad-hoc de plasă (mesh), fiind folosit un protocol de rutare multi-hop pentru transmiterea datelor. Caracteristicile unei astfel de rețele sunt următoarele:

- Multi-hop: posibilitatea transmiterii datelor peer-to-peer (de la nod la nod) către nodul care are rolul de „Bază a rețelei”, conferind astfel o scalabilitate sporită;
- Auto-Configurare: rețeaua se poate configura fără intervenție umană;
- Auto-Regenerare: caracteristica rețelei de a accepta sau pierde noduri dinamic, fără a fi necesară o oprire sau o repornire a acestuia în ansamblu;
- Rutare Dinamică: nodurile rețelei sunt capabile să aleagă rutele de transmisie a datelor către bază în funcție de contextul actual al rețelei.

Intreacțiunea cu o astfel de rețea este realizată de obicei prin intermediul unei aplicații software. Astfel din acest punct de vedere, rețeaua poate fi privită ca și conlucrarea următoarelor trei nivele:

- Nivelul Client: oferă interfațarea grafică dintre rețea și utilizatorul uman, permițând acestuia să obțină și să interpreteze datele provenite de la senzori, precum și să transmită acestora diferite comenzi;
- Nivelul Server: reprezintă o entitate situată între rețea și utilizator și care este desemnată să

inmagazineze permanent sau provizoriu informațiile transmise de nodurile rețelei și să le transmit clientului în momentul în care acesta le solicită;

- Nivelul Nodurilor: este constituit din software-ul aflat pe nodurile rețelei și care gestionează preluarea și transmisia datelor la nivel de nod.

Taxonomia protocoalelor de rutare bazate pe informațiile de poziție

Protocoalele de rutare bazate pe localizare reprezintă o îmbunătățire a performanțelor într-o rețea prin introducerea informației de localizare în procesul de selecție al rutei.

GPSR – Greedy Perimeter Stateless Routing

Protocolul GPSR utilizează informația de localizare pentru reducerea costurilor și obținerea unei scalabilități mai bune atunci când distanța între terminalele mobile și dimensiunea rețelei crește. GPSR preia informația pozițională ca și metrică cheie în direcționarea pachetelor și utilizează următoarea metodă simplă „greedy”.

1. Fiecare pachet este marcat de către terminalul sursă cu cea mai recentă informație despre localizarea destinației.
2. Fiecare nod intermediar trimite pachetele către nodul vecin cel mai apropiat de destinația marcată în pachet.

Metoda „greedy” nu garantează că o anumită cale între sursă și destinație este determinată chiar dacă ea există, acest lucru este exemplificat în următoarea figură. Nodul x , după primirea unui pachet asociat destinației D de la sursa S , nu poate determina un nod vecin care să respecte metoda „greedy” de trimitere a pachetelor, deoarece ambii vecini y și z sunt mai departe decât x de destinația D .

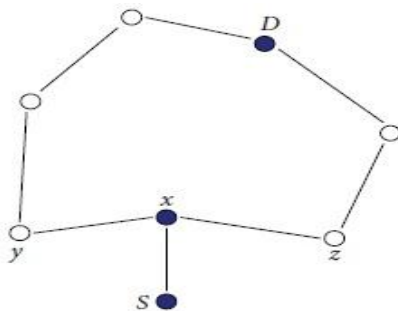


Figura 1: Destinație greșită – exemplu

Când metoda greedy nu se poate aplica, se aplică perimetru forwarding pentru a înlătura blocajul creat. Astfel se folosește regula mâinii drepte pentru trimiterea pachetelor. În (figura 2).

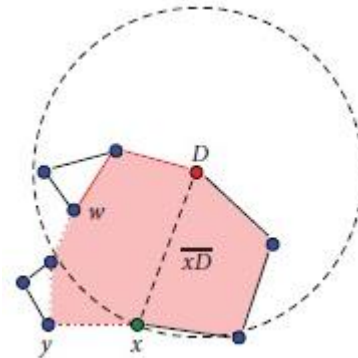


Figura 2: Perimeter forwarding

Presupunem că x este primul nod în care metoda greedy nu se poate aplica.

1. Nodul x înregistrează poziția opusă L_f a câmpului din pachet.
2. Nodul x folosește informația corespunzătoare poziției curente și cea a destinației D înregistrată în pachet pentru determinarea liniei $\overrightarrow{xD}$.
3. Nodul x în funcție de poziția nodurilor vecine, determină care este prima muchie în ordine inversă acelor de ceasornic față de linia $\overrightarrow{xD}$ (muchia $\overrightarrow{xy}$ în figura 2).
4. Nodul x trimite pachetele spre terminalul y .
5. Nodul y verifică dacă este mai apropiat de destinația D decât poziția înregistrată în câmpului L_f . Dacă este îndeplinită condiția nodul y schimbă cu metoda greedy, altfel se repetă procesul începând cu pasul 2.

În exemplul din figură pentru nodul x se aplică perimetru forwarding, iar pentru nodul w metoda greedy.

Location-aware long-lived route selection (LLR) în rețele ad hoc.

Protocolul LLR (Location-aware long-lived protocol) aparține familiei de protocoale de rutare on-demand. Protocolul utilizează informația de localizare urmărind minimizarea rutelor greșite și ca o consecință minimizarea utilizării procedurii de reconstrucție a rutelor.

În protocolul LLR, sursa S include două câmpuri de informație adițională în fiecare pachet generat: propria poziție (XS,YS) și zona de acoperire cu semnal radio Rs. Fiecare terminal A în conexiune fizică cu S, care primește pachetul folosește informația și propria poziție (XA,YA) și zona de transmisie RA pentru a evalua cifrele.

1. Forward movement limit (FML) măsoară mișcarea relativă maximă între A și S care poate fi tolerată înainte ca distanța între A și S să fie mai mare decât zona de transmisie A, definită ca:

$$FML = R_A - \text{distance}(A, S) = R_A - \sqrt{(X_S - X_A)^2 + (Y_S - Y_A)^2} \quad (1)$$

2. Backward movement limit (BML) măsoară mișcarea relativă maximă între A și S care poate fi tolerată înainte ca distanța dintre A și S să fie mai mare decât zona de transmisie S, definită ca:

$$BML = R_S - \text{distance}(A, S) = R_S - \sqrt{(X_S - X_A)^2 + (Y_S - Y_A)^2} \quad (2)$$

FML și BML sunt folosite pentru determinarea limitei normalizate de mișcare NML (normalized movement limit), care asigură o medie pentru FML și BML (FML≠BML când RS≠RA).

$$NML = \frac{FML \cdot BML}{FML + BML} \quad (3)$$

NML este un indicator de stabilitate al legăturii: cu cât NML are o valoare mai mare cu atât este mai scăzută probabilitatea de pierdere a legăturii între terminale.

Distance routing algorithm for mobility DREAM

Algoritmul DREAM folosește informația de localizare cu scopul de a reduce informația de rutare. Protocolul combină abordările proactive și reactive, bazându-se pe actualizările terminalelor pentru propagarea informației de localizare și procesul de inundare la transmiterea pachetelor spre destinație.

Când terminalul sursă S începe procesul la $t=t_1$ presupunem că deține următoarea informație:

1. Propria poziție (XS,YS)
2. Poziția vecinilor alăturați
3. Poziția destinației D, (XD,YD) pentru un timp dat $t_0 < t_1$
4. Viteza maximă v spre destinație sau cel puțin funcția densitate de probabilitate a vitezei p(v)

Sursa pe baza informațiilor definește regiunea geografică în care se vor trimite pachetele rutate și determină vecinii din interiorul regiunii. Regiunea de trimitere a pachetelor este determinată de unghiul α . În funcție de informația vitezei de destinație sunt posibile două cazuri:

1. v cunoscut: x este distanța parcursă spre destinație în perioada t_1-t_0 , dată de $x=v \cdot (t_1-t_0)$ și având o distanță r între S și D rezultă:

$$\alpha = \arcsin \frac{x}{r} \quad (4)$$

2. p(v) cunoscut: pentru a determina regiunea incluzând D cu probabilitatea p corespunzătoare valorii α se determină astfel:

$$\int_{r \cdot \sin \alpha / (t_1 - t_0)}^{+\infty} p(v) dv \geq p \quad (5)$$

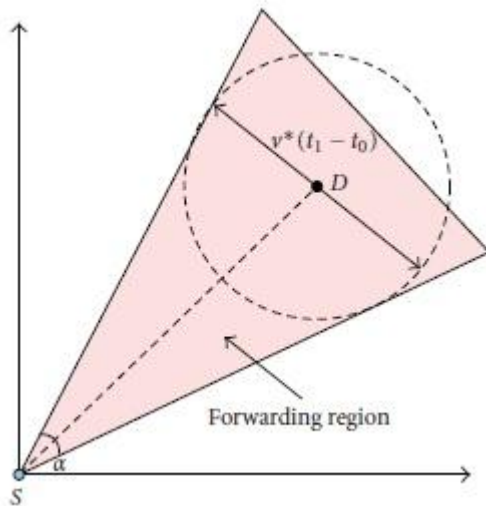


Figura 3: Determinarea regiunii de trimitere a pachetelor cu protocolul DREAM

Pentru α determinat, terminalul trimite pachetele marcând vecinii din cadrul regiunii. Fiecare terminal intermediar la recepția unui pachet evaluează dacă pachetul trebuie transmis mai departe printr-o regiune de transmitere nou corespunzătoare noului terminal, verificând dacă există vecini. Dacă nu se găsesc vecini, pachetul nu este transmis mai departe.

Procesul necesită informație pozițională legată de sursă, dar și de terminalele vecine și destinație.

Location-aided routing (LAR)

Protocolul LAR este un protocol de rutare on-demand. Pentru a stabili o rută între sursă și destinație se bazează pe un proces de inundare pornit de sursă prin stabilirea unui pachet de cerere de rutare prin broadcast (RRQ Route Request). Pachetul este trimis de nodurile din rețea până când ajunge la destinație sau timpul de transmitere expiră în cazul în care ruta este calculată greșit. Când destinația recepționează un pachet RRQ de la sursă răspunde cu un pachet

RRP (Route Reply Packet) transmis spre sursă, care apoi poate începe transmitia pachetelor de date. Un nod care determina o eroare de transmisie a pachetelor RRP și de date declanșează o alarmă prin transmitia pachetelor RRC (Route Reconstruction). Când un pachet RRC este transmis spre sursă și transmitia pachetelor de date este oprită, sursa fiind cea care decide dacă întrerupe conexiunea sau începe să caute o nouă cale spre destinație.

Un dezavantaj al acestei metode este cantitate mare de resurse consumate în timpul căutării unei căi de transmisie. Protocolul LAR folosește informația de localizare la inițializarea conexiunii pentru a reduce numărul pachetelor de control, cât timp pentru transmiterea pachetelor de date nu este necesară transmiterea de informație de localizare.

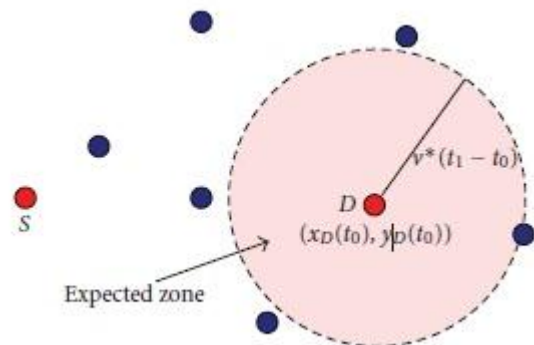


Figura 4: Zona de așteptare a pachetelor în cazul protocolului LAR

Informația necesară de localizare în cazul protocolului LAR:

- poziția destinației
- viteza poziția sursei
- maximă a terminalului

Informația este utilizată în timpul calculării rutei de trimitere a pachetelor. Se presupune că terminalul S este punctul de pornire în calcularea rutei spre destinația D la timpul $t=t_1$ și ultima informație

înregistrată în legătură cu destinația a fost recepționată la $t=t_0$. Pe baza estimărilor vitezei maxime v a terminalului D, S poate evalua distanța maximă spre D în funcție de ultima locație înregistrată. Această distanță este dată de $d_{MAX} = v \cdot (t_1 - t_0)$. Ca o consecință, poziția ocupată de D este în centrul regiunii circulare cu raza d_{MAX} centrată în punctul $(x_D(t_0), y_D(t_0))$ cunoscută sub denumirea de zonă de așteptare.

Zona de așteptare indică porțiunea din rețea la care trebuie să ajungă pachetele RRQ. Ideea principală a protocolului LAR este exploatarea acestei informații pentru a reduce cantitatea pachetelor RRQ care inundă rețeaua și permit transmiterea pachetelor doar în zona de așteptare conținută de destinație. Regiunea rețelei în care transmiterea pachetelor este permisă este numită zonă de cerere. Un terminal intermediar este permis pentru transmiterea pachetelor RRQ numai dacă este situat în zona de cerere definită de sursa de atașare a cererii.

Pentru definirea zonei de cerere trebuie să se țină cont de poziția sursei S și de zona de așteptare.

De la poziționarea GPS la UWB

Adoptarea tehnologiei UWB în locul GPS ca bază pentru determinarea poziției deschide noi uși aplicațiilor orientate pe localizare.

Deoarece UWB furnizează doar informație zonală, de fapt procesarea distribuită a măsurătorilor zonale este necesară pentru a construi o hartă a rețelei.

Algoritmul de poziționare SPA (Self-positioning algorithm) este compus din doi pași. Inițial fiecare nod din rețea încearcă să construiască un sistem de coordonate centrat pe el însuși și să determine poziția vecinilor din sistem.

În al doilea rând sistemele de coordonate centrate pe noduri converg către un sistem global de coordonate la nivelul întregii rețele.

Inițial fiecare nod N_i încearcă să construiască un sistem de coordonate prin:

- Detectarea vecinilor KN_i utilizând frecvența baliză
- Evaluând distanțele spre vecinii alăturați DN_i
- Transmitere broadcast DN_i și KN_i spre vecini.

Deoarece toate nodurile cunosc procedura precedentă, nodurile N_i cunosc distanțele spre toți vecinii alăturați, ID-ul vecinilor aflați la două noduri distanță și distanțele dintre vecinii alăturați și vecinii situați la două noduri distanță. Folosind această abordare, un sistem de coordonate la nivelul întregii rețele este realizat. Nodurile care nu își pot realiza singure un sistem de coordonate își pot obține poziția în sistemul de coordonate la nivel rețea dacă sunt în zona a trei noduri care și-au obținut sistemul de coordonate.

Analiza eficienței energetice și costul pentru algoritmi de rutare

Rutarea cu consum redus de putere este un subiect important în distribuirea rețelelor cu senzori și ad hoc. Majoritatea protocoalelor pentru monitorizarea puterii se bazează pe metode de rutare cu consum eficient de putere sau pe exploatarea informației adiționale – cum este localizarea terminalelor în rețea. În ambele cazuri, scopul final este găsirea căilor dintre sursă și destinație, care minimizează consumul de putere.

Rutele bazate pe monitorizarea puterii au în vedere:

- puterea disipată pentru transmisie
- suma dintre puterile transmise și primită
- puterea reziduală din fiecare nod.

Rezultatele simulărilor arată totuși că minimizarea puterii transmise nu duce neaparat la un timp de viață mai lung pentru fiecare terminal din rețea. Din contră, utilizarea metodei de rutare cu monitorizarea puterii duce la creșterea numărului de hopuri, rezultând un număr mai mare de terminale pentru fiecare comunicație.

În concluzie, această tehnică oferă un consum de putere rezonabil între terminale, crescând totuși timpul operațional din rețea înainte ca primul terminal să rămână fără energie.

Pentru cazul particular a UWB, fiecărei căi îi este asociat un cost de comunicație. Costul unei căi este suma tuturor legăturilor care o compun. Funcția de cost este exprimată sub forma unei sume:

$$C(x,y)=\delta \cdot C_0 \cdot d^{\alpha} + C_1 \cdot R \cdot d^{\alpha}$$

Primul termen al sumei arată costul de sincronizare necesar realizării unei legături noi.

- Dacă există deja o legătură activă între două noduri, $\delta = 0$ și nu există niciun cost de sincronizare.

- Dacă nu există legătura activă între cele două noduri, $\delta = 1$ și se adaugă costul de sincronizare.

Al doilea termen al sumei ia în considerare costul pentru transmiterea datelor și depinde de rata R a datelor cerute. Ambii termeni depind de consumul de putere și prin urmare de distanța d dintre două noduri. Parametrul α arată caracteristica de propagare a canalului și are de obicei o valoare cuprinsă între 2 și 4. Constantele C_0 și C_1 exprimă mărimile sincronizării și a transmisiei.

Spre deosebire de metoda de rutare tradițională, strategia bazată pe economisirea puterii, duce la căi de comunicație cu multe hopuri între terminale, crescând performanța rețelei prin reducerea puterii emise și a nivelelor de interferențe.

O formă îmbunătățită a funcției cost este prezentată mai jos:

$$C(x, y) = C(\text{putere}) + C(\text{sinc}) + C(\text{interferențe}) + C(\text{calitate}) + C(\text{întârziere})$$

Unde primii doi termeni, $C(\text{putere})$ și $C(\text{sinc})$, care arată puterea și sincronizarea, reprezintă cei doi termeni din relația definită anterior.

TESTE DE PERFORMANȚĂ

Testearea protocolului DREAM, precum și compararea acestuia cu protocolul LAR s-a realizat folosind patru dispozitive de tip IRIS și două plăci de interfațare de tip MIB520. Dintre acestea, pe unul dintre modulele IRIS, montat pe o placă MIB520, a fost încărcată aplicația XSniffer dezvoltată de către firma Crossbow pentru a permite monitorizarea traficului de pachete din rețea.

Unul din nodurile rămase a fost încărcat cu aplicația Designated Router Base și a îndeplinit rolul de nod Bază al rețelei, el fiind conectat prin intermediul aplicației XServe la componenta DREAM Router a protocolului DREAM. Celelalte două noduri rămase au fost poziționate în așa manieră încât să formeze o rețea multi-hop (o pereche copil – părinte).

Pe nodurile rețelei, atât în cazul folosirii DREAM cât și în cel al folosirii LAR a fost încărcată aplicația TinyOS, NetCountToLeds, care utilizează un timer intern de tipul TIMER_REPEAT pentru a incrementa o variabilă count, care numără evenimentele Timer.fired() declanșate de expirarea perioadei acestuia. De asemenea a mai fost folosită și interfața LedsC pentru a permite afișarea valorii variabilei count prin intermediul led-urilor cu care modulul IRIS este dotat. Aplicația folosește pe rând unul din cele două protocoale de rutare pentru a transmite valorile variabilelor count de pe cele două noduri, către nodul Bază, având setată o perioadă de transmise spre acesta de 20 secunde.

În plus trebuie menționat că intervalele Route Update Interval (RUI) al LAR și Interval de Puls (IdP) al DREAM au fost setate la 4 secunde. DREAM a fost configurat astfel încât să aibă o comportare optimă dat fiind diametrul scăzut al rețelei (diametru 2): durata stării Discovery a fost setată la 8 IdP (realizându-se astfel și o concordanță

cu LAR), iar perioada necesară strângerii de informații pentru estimarea legăturii dintre noduri a fost setată la 4 IdP.

Timpul de convergență

Timpul de convergență reprezintă intervalul de timp necesar pentru formarea unei topologii logice stabile a rețelei de senzori wireless. Este calculat ca fiind diferența dintre momentul transmiterii primului pachet de configurare al protocolului de rutare și momentul transmiterii primului pachet de date de către unul din nodurile rețelei.

În cazul folosirii protocolului DREAM, pe baza rezultatelor preluate de la Xsniffer (Figura 5) a fost calculată următoarea valoare a indicatorului Timp de Convergență:

Considerând:

t_1 = momentul transmiterii primului pachet

Discovery Beacon de către nodul cu id-ul 1899,

t_2 = momentul transmiterii primului pachet de către nodul 1899;

$TC = t_2 - t_1 = 1585,875 - 1511,781 = 74,094$ (secunde)

Folosind protocolul LAR s-a obținut următorul rezultat (Figura 6):

Considerând:

t_1 = momentul transmiterii primului pachet Route

Update de către nodul cu id-ul 13,

t_2 = momentul transmiterii primului pachet de către nodul 13.

$TC = t_2 - t_1 = 493,203 - 469,314 = 23,889$ (secunde)

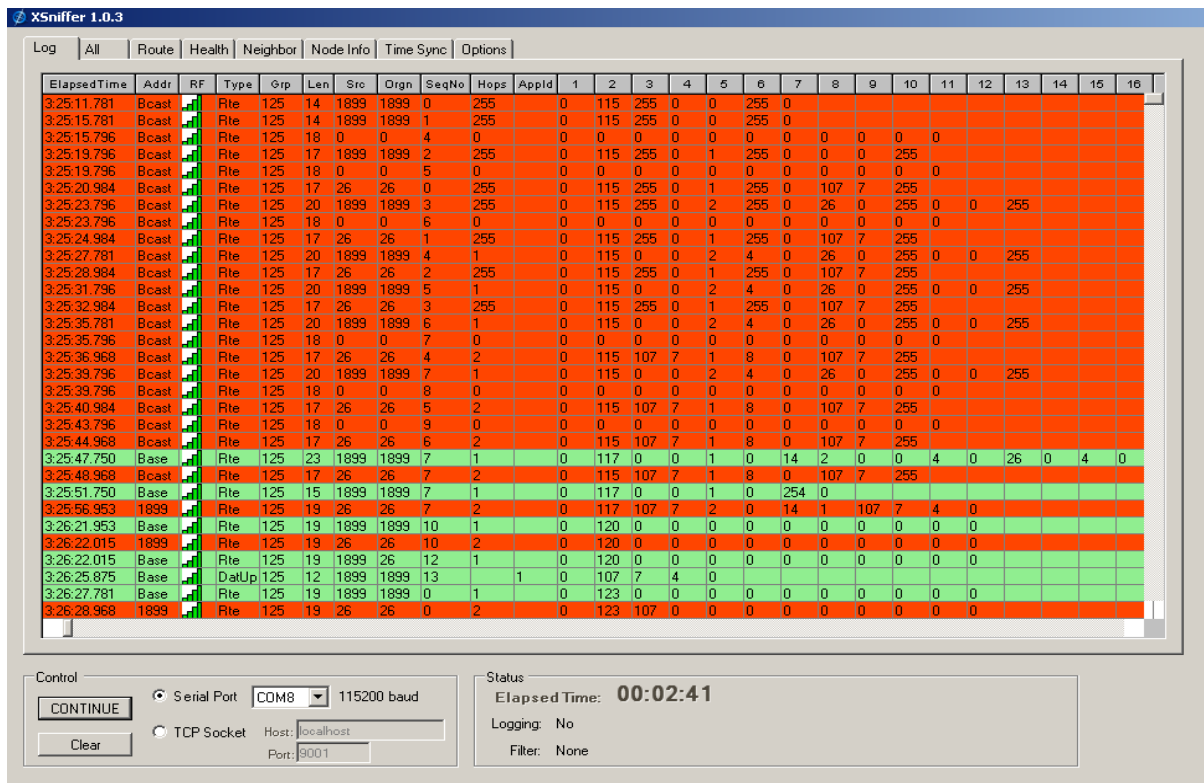


Figura 5: XSniffer – stările DREAM Discover și Advertise

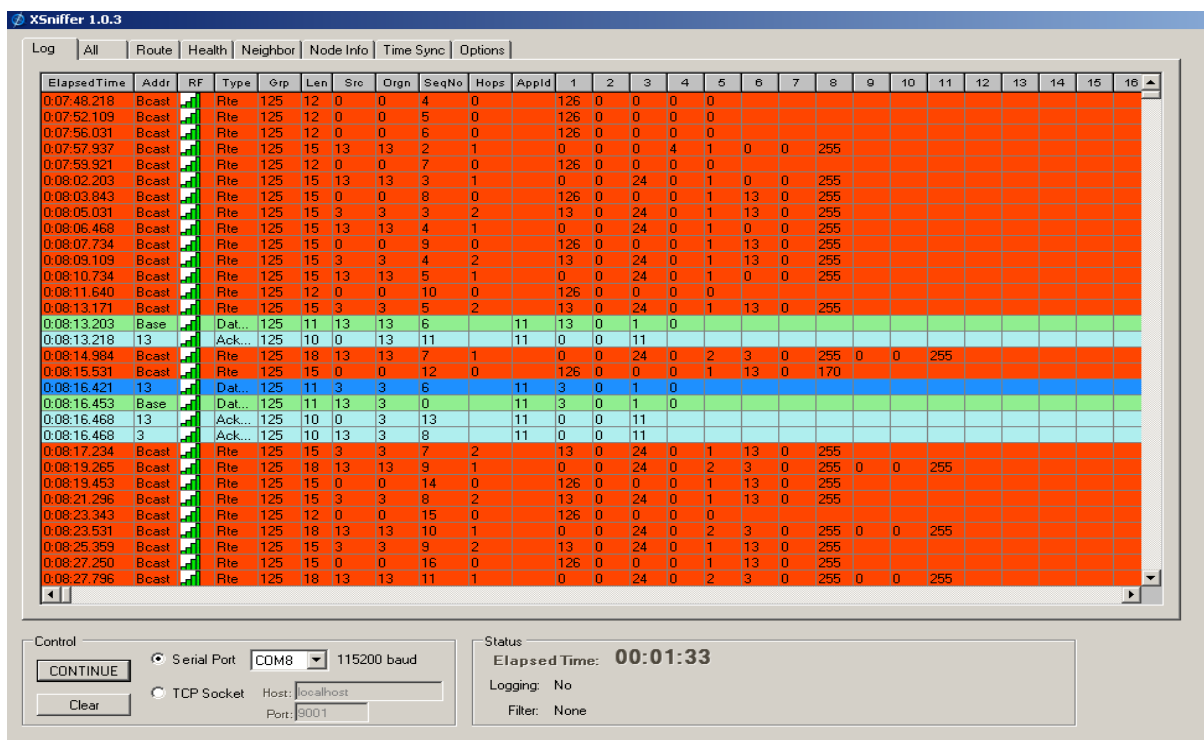


Figura 6: XSniffer – configurarea rețelei LAR

Viteza de rutare

Acest indicator de calitate a fost calculat în cazul ambelor protocoale, pe baza rezultatelor oferite de XSniffer (Figura 7, Figura 8), făcându-se o medie a vitezelor de rutare folosind informațiile oferite de către ambele noduri ale rețelei:

de LAR fiind optime la nivel local (pe vecinătatea nodului).

Cazul DREAM:

Interval transmisie aplicație (lap) = 20 (secunde);

Nodul 1899:

$$VR_{1899} = t_2 - t_1 - Iap = 1606.406 - 1585.875 - 20 = 0,531(\text{secunde})$$

,

Nodul 26:

$$VR_{26} = t_2 - t_1 - Iap = 1615.609 - 1595,093 - 20 = 0,516(\text{secunde})$$

,

$$VR = \frac{VR_{1899} + VR_{26}}{2} = 0,5235$$

Cazul LAR:

Interval transmisie aplicație (lap) = 20 (secunde);

Nodul 13:

$$VR_{13} = t_2 - t_1 - Iap = 513.718 - 493.203 - 20 = 0,515(\text{secunde})$$

,

Nodul 3:

$$VR_3 = t_2 - t_1 - Iap = 5156.990 - 496.453 - 20 = 0,537(\text{secunde})$$

,

$$VR = \frac{VR_{13} + VR_3}{2} = 0,5260$$

Din rezultatele obținute se observă faptul că pentru o rețea de diametru redus (în acest caz 2), ambele protocoale de rutare oferă aproximativ aceleași viteze de rutare. Totuși DREAM are un mic avantaj, care tinde să crească vertiginos pe măsura ce diametrul rețelei crește deoarece rutele calculate de acesta sunt optime, în schimb rutele calculate

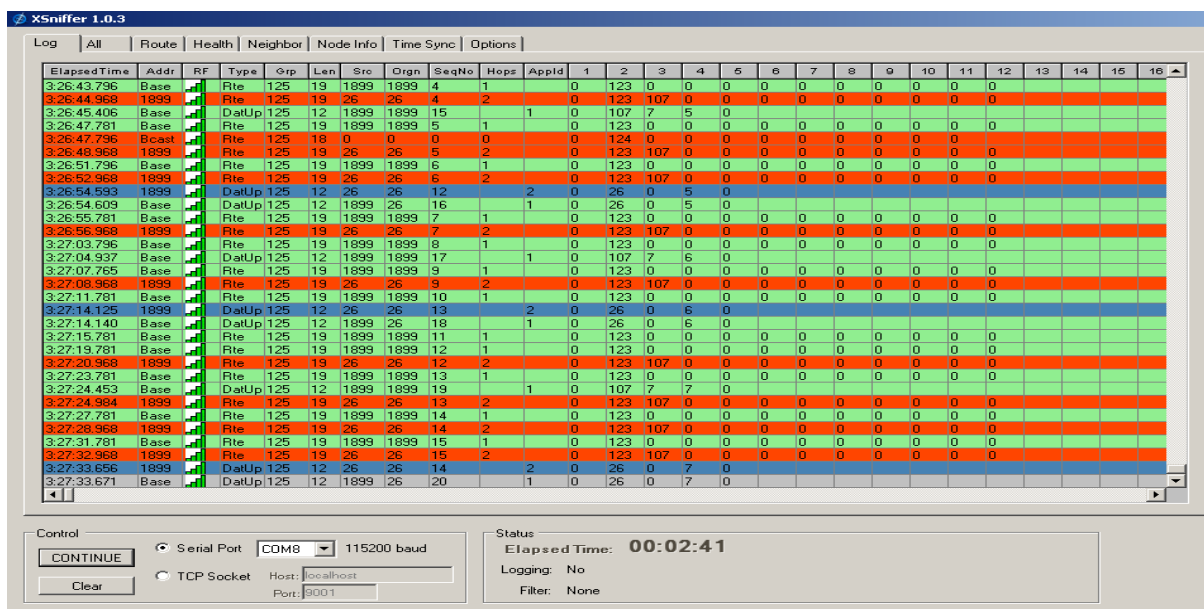


Figura 7: XSniffer – pachete de date transmise folosind DREAM

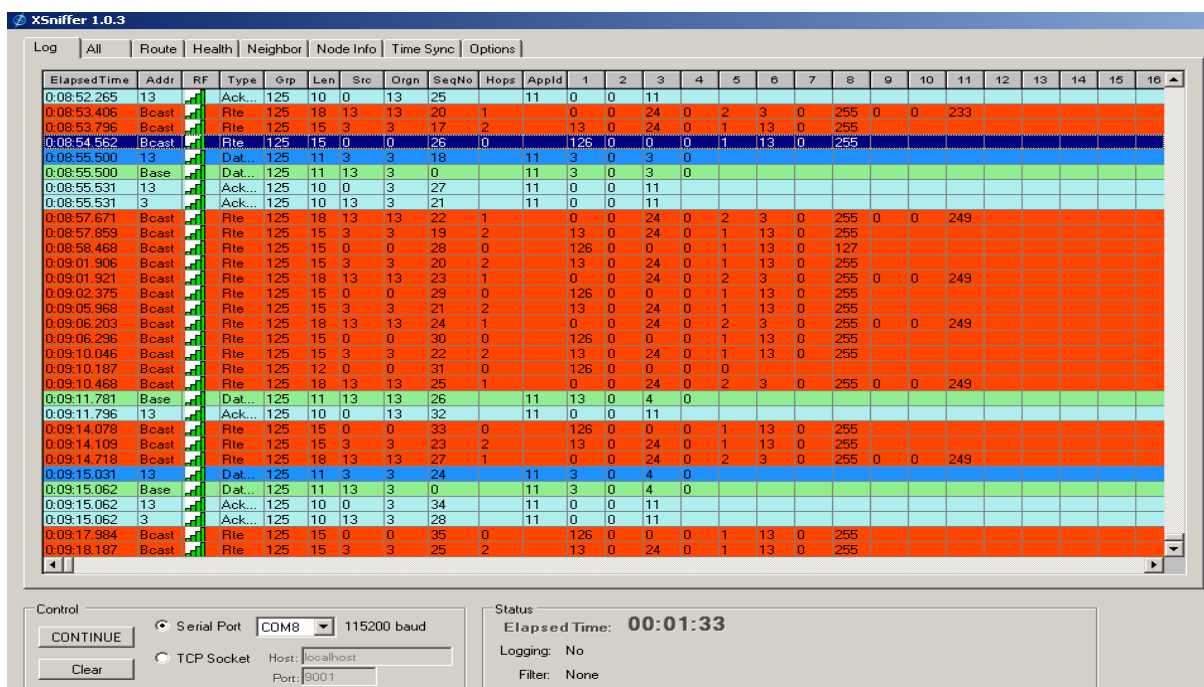


Figura 8: XSniffer – pachete de date transmise folosind LAR

Bibliografie

- [1] P. Bahl and V. N. Padmanabhan. Radar: An in-building RF-based user location and tracking system. In *Proc. IEEE INFOCOMM*, pages 775-784, 2000.
- [2] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications*, 7(5):28-34, 2000. Special Issue on Smart Spaces and Environments.
- [3] S. Capkun, M. Hamdi, and J.-P. Hubaux. GPS-free positioning in mobile ad-hoc networks. In *Proc. of 34th HICSS*, volume 9, page 9008, 2001.
- [4] B. Kusy, M. Maroti, G. Balogh, P. V. Olgyesi, J. Sallai, A. Nadas, A. Ledeczki, and L. Meertens. Node density independent localization. In *Proc. of IPSN*, pages 441-448, 2006.
- [5] K. Pister, L. Doherty, and L. El Ghaoui. Convex optimization methods for sensor node position estimation. In *Proc. of IEEE INFOCOMM*, pages 1655-1663, 2001.
- [6] R. Nagpal, H. E. Shrobe, and J. Bachrach. Organizing a global coordinate system from local information on an ad hoc sensor network. In *Proc. of IPSN '03*, pages 333-348, 2003.
- [7] D. Niculescu and B. Nath. SpotON: An indoor 3-d location sensing technology based on RF signal strength. Technical Report Report #200002-02, Department of CSE, University of Washington, Feb. 2000.
- [8] D. Niculescu and B. Nath. Ad hoc positioning system (APS). In *Proc. of GLOBECOM*, volume 5, pages 2926-2931, 2001.
- [9] D. Niculescu and B. Nath. Ad hoc positioning system (APS) using aoa. In *Proc. of IEEE INFOCOMM*, pages 1734-1743, 2003.
- [10] A. Savvides, C. C. Han, and M. B. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proc. of MobiCom*, pages 166-179, 2001.
- [11] A. Savvides, H. Park, and M. B. Srivastava. The n-hop multilateration primitive for node localization problems. *Mobile Networks and Applications*, 8(4):443-451, 2003.
- [12] A. Vargus. The OMNeT++ discrete event simulation system. In *Proc. of ESM*, pages 319-324, 2001.
- [13] Improved GPSR routing algorithm and its performance analysis. Software Engineering and Service Sciences (ICSESS), 2010 IEEE International Conference