

## Üzenetszórás az ad-hoc drótnélküli hálózatokban

Matkó Imre-Zoltán  
Babeş-Bolyai Tudományegyetem  
Matematika-informatika kar  
Informatika szak, IV. év  
Kolozsvár

e-mail:  
[mi50440@scs.ubbcluj.ro](mailto:mi50440@scs.ubbcluj.ro),  
[desertsm@gmail.com](mailto:desertsm@gmail.com)

Témavezető tanárok:  
Dr. Ionescu Klára,  
Dr. Robu Judit

## Tartalomjegyzék

0.Kivonat.....	3
1.Bevezetés.....	3
2.Az ad-hoc drótnélküli hálózatok.....	3
3.Drótnélküli üzenettovábbítás, forgalomirányítás.....	4
4.Üzenetszórás (broadcast).....	5
5.A binomiális fák alkalmazása az üzenetszórás útvonalválasztó módszereiben.....	6
5.1.A binomiális fák.....	6
5.2.Binomiális fák az útvonalválasztó (routing) módszerekben.....	7
5.3.Az üzenetszórás fa felépítése és karbantartása.....	8
6.Az algoritmusok tesztelése.....	9
7.Következtetések, jövőbeli munka.....	10
8.Könyvészet.....	11

## 0. Kivonat

A dolgozat célja bemutatni az ad-hoc drótnélküli hálózatok üzenetszórási módszereiben felmerülő problémákat és lehetőségeket, illetve összehasonlítani és fejleszteni néhány alkalmazott adatszerkezetet és algoritmust. Ezen hálózatok stabil működése nagyon hasznos lehet olyan körülmények között, ahol a hagyományos hálózatok nem elérhetőek illetve időben és térben gyorsan változik a hálózat topológiája, ezért egy központi adó-vevő beiktatása nem hatékony megoldás.

A binomiális fák alkalmazása hatékony módszernek bizonyult a multicast típusú üzenetek továbbítására, ezért ezen a téren érdemes további kutatásokat folytatni.

A legkézenfekvőbb megoldás az üzenetszórásra a feszítőfa építés, ami azonban NP teljes feladat. Erre léteznek jó heurisztikák és bár egy binomiális kupac nem biztosítja a legjobb heurisztikát, de más tulajdonságai hatékonyra teszik az alkalmazását a drótnélküli hálózatokban.

## 1. Bevezetés

Az ad-hoc drótnélküli hálózat hatékony csomagtovábbítását biztosító eljárás a hálózat tulajdonképpeni működését biztosító algoritmus és szoftver. Ezen hálózatok stabil működése nagyon hasznos lehet olyan körülmények között, ahol a hagyományos hálózatok nem elérhetőek illetve időben és térben gyorsan változik a hálózat topológiája, ezért egy központi adó-vevő beiktatása nem lehetséges.

A témában már számos dolgozat született, ezek eredményei alapján a binomiális fák alkalmazása hatékony módszernek bizonyult az üzenetek továbbítására, ezért ezen a téren érdemes további kutatásokat folytatni, hatékonyabb illetve bizonyos körülményekre optimalizált technikákat kifejleszteni, hiszen néhol a drótnélküli egységek energiatakarékossága a legfontosabb tényező, máshol a gyors adatátvitel vagy a megbízhatóság. Ezen írás célja megvizsgálni néhány eddigi eredményt és a különféle gyakorlati tapasztalatokat figyelembe véve fejleszteni az alkalmazott adatszerkezeteket és algoritmusokat.

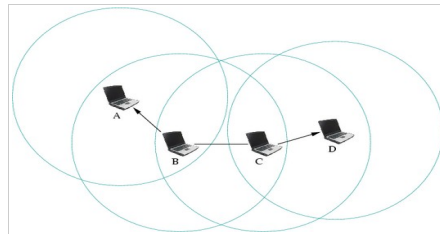
## 2. Az ad-hoc drótnélküli hálózatok

Egy mobil ad-hoc drótnélküli hálózat ( MANET – mobile ad-hoc network ) egy nem állandó kapcsolatban levő egységek hálózata, központi irányítás és előre meghatározott infrastruktúra nélkül működik. Minden csomópont rendelkezik egy bizonyos jelerősséggel és amennyiben közvetlenül nem ér el egy célcsoomópontot, mert az kívül esik az általa lefedett területen, akkor a közbeeső csomópontok fogják továbbítani a csomagokat, adatokat.

Az ad-hoc ideiglenes hálózatokra akár többféle architektúrájú egységek is csatlakozhatnak, amelyek kompatibilisek a 802.11 szabvánnyal. A hálózat rendelkezhet akár hálózati átjáróval ( gateway ), ami az internettel vagy egy kábeles LAN hálózattal is összekötheti a drótnélküli egységeket.

Az ilyen jellegű hálózatok nagyon hasznosak lehetnek olyan helyeken, ahol a telekommunikációs infrastruktúra hiányzik vagy sérült, például mozgó járművek kommunikációjában, különféle mentőegységeknél katasztrófák helyszínén, de számos egyéb civil és katonai alkalmazás elképzelhető.

A MANET hálózatoknál fontos a minél biztonságosabb és takarékosabb adattovábbítás, vagyis a rendelkezésre álló sáv szélesség optimális kihasználása és az energiatakarékosság. A csomópontok állandó mozgásban lehetnek, emiatt a hálózat topológiája folyamatosan változik, ehhez alkalmazkodó módszereket kell találni az üzenetek továbbítására, biztonságosan és gyorsan el kell dönteni, hogy egy csomópont elérhető-e vagy sem és optimális stratégiát kell választani az üzenettovábbításra [1].



1. ábra: ad-hoc hálózat

Az egyes csomópontok egyben router szerepet is betöltenek, ez kell biztosítsa a rendszer önszervező képességét illetve a 802.11 WiFi szabványú kommunikációs standard lehetővé teszi a legkülönbözőbb eszközök részt vételét a hálózatban.

Egy speciális MANET hálózat például a VANET (Vehicular Ad-Hoc Network ), amely járművek közötti hálózati kommunikációra van optimalizálva.

### 3. Drótnélküli üzenettovábbítás, forgalomirányítás

Az üzenettovábbítási módszerek a következő szempontokat kell figyelembe vegyék:

- minél kisebb méretű route tábla
- optimális út választás (leggyorsabb, biztonságosabb, megbízhatóbb, olcsóbb út)
- az új, kilépő vagy mozgó csomópontok megfelelő kezelése
- a hálózat minél kisebb terhelése

Az IEEE 802.11 szabványát használó módszerek a következő csoportokba oszthatóak [1] :

- I. egyszerű „flooding” - minden csomópont minden bejövő csomagot továbbküld, ami nem neki van címezve.
- II. valószínűség alapú – a hálózat állapotának függvényében minden csomóponthoz egy valószínűséget rendelünk, amely eldönti, hogy az adott csomópont továbbítsa-e vagy nem a bejövő csomagokat, ha a valószínűség 1, akkor az I. esetet kapjuk, illetve, ha a valószínűségeket véletlenszerűen adjuk meg, akkor egy véletlenített algoritmust kapunk.
- III. terület alapú – a csomópontok közötti becsült távolság alapján egy csomópont továbbítja a csomagokat, ha egy adott irányba megfelelő lefedettséggel rendelkezik.
- IV. szomszédsági – a csomópontok a szomszédjaikkal való kommunikáció alapján határozzák meg az üzenettovábbítási stratégiát, léteznek 1 illetve több szintű algoritmusok, aszerint, hogy milyen távolságú szomszédok kommunikálnak közvetlenül.

Nyilván az utóbbi két elv a legmegfelelőbb a hatékony algoritmusok kidolgozásához. Két csomópont távolsága alatt a leggyakrabban azt értjük, hogy hány közbeeső ponton át jut el egy csomag egyik pontból a másikba, vagyis ha például a távolság 3, akkor 3 lépésben fog egy üzenet

eljutni a célig. Tehát a csomagok általában ilyen utakat fognak bejárni, ezen útvonalak megválasztására, vagyis a forgalomirányításra, hatékony algoritmusokat kell alkalmazni.

Hagyományosan két csoportba oszthatjuk ezeket a forgalomirányító protokollokat: proaktív és reaktív protokoll. Az ún. reaktív protokollok által működtetett hálózatokban a csomópontok csak olyankor cserélnek egymás között a topológiára vonatkozó információt, mikor feltétlenül szükséges, mikor éppen vonalra kell tenni egy csomagot. Ilyenek például a DSR (Dynamic Source Routing, IETF RFC4728) és az AODV (Ad-hoc On-demand Distance Vector, IETF RFC3561) protokollok. Mikor egy kérés érkezik, akkor a reaktív protokoll egy útvonalválasztó mechanizmust indít be ami a célcsoomópont válaszcsoomagjának a kezdőpontba való érkezésével ér véget, ha létezik megfelelő út, vagy időtúllépés esetén a célt elérhetetlennek nyilvánítjuk. A proaktív protokollok esetén a csomópontok periodikusan, ha nincs hasznos adatforgalom, akkor is, információkat cserélnek egymás között, ezáltal egy új beérkező kérés esetén a csomópontok önmagukban, azonnal képesek eldönteni, hogy elérhető-e a célpont illetve merre küldjék a csomagot. Ilyen például az OLSR (Optimized Link State Routing Protocol, IETF RFC3626). Mindkét módszernek megvannak az előnyei és hátrányai, a reaktív protokollok igen hatékonyak a viszonylag kis hálózatok esetén és nem terhelik folyamatosan a hálózatot. A proaktív módszerek viszont a nagyobb hálózatok esetén lehetnek sokkal gyorsabbak, bár egy viszonylag folyamatosan terhelik a hálózatot a saját üzeneteikkel.

Egy másik szempont a protokollok osztályozására az, hogy milyen formában ismerik a hálózat topológiáját, az egyes csomópontoknak lehet globális teljes ismeretük vagy ismerhetik csak egy kisebb, lokális szeletét a hálózatnak. A DSR a hálózat minden egyes csomópontjáról gyűjt információkat és tárol el, ami alapján megépíti a route-tábláját, ez kevés számú csomópont esetén nagyon hatékony útválasztást tesz lehetővé. A másik csoport, amilyen az AODV protokoll, csupán lokális információkkal rendelkezik, az útvonalokról csupán annyit tudnak az egyes csomópontok, hogy azok milyen hosszúak (költségesek) és az aktuális csomóponton áthaladó utak esetén melyik a rögtön az aktuálist követő pont.

Az igen nagy hálózatok esetén ezeket a protokollokat kombinálni is lehet, kisebb csoportok egymás között kommunikálhatnak reaktív protokollal és rendelkezhetnek globális ismerettel a saját csoportjukról és egyes, ún. határ menti csomópontok pedig mind a saját csoportjukkal, mind a többi csoporttal tartják a kapcsolatot. A csoportok egymás között pedig proaktív módszereket használhatnak és nem szükséges a hálózat teljes ismerete. Egy erre az elvre épülő technika kidolgozása lehetne a legmegfelelőbb a MANET típusú hálózatok számára. Erre irányuló kutatást mutat be például a [2] cikk.

## 4. Üzenetszórás (broadcast)

Az üzenetszórás (broadcasting) kulcs elem, főleg a hálózati réteg szintű kommunikációkban. Egy, a hálózatról, globális ismeretre törekvő protokoll esetén nagyon gyakran szükséges bizonyos információkat minden egyes csomópontba eljuttatni. Erre különféle módszerek léteznek, amelyek azonban többnyire elég költségesek, ahogyan maguk az ilyen üzenetek is. Ezeket a csomagokat nevezi a szakirodalom „broadcast” csomagoknak, ezeknek a címzettje az IP hálózatokban mindig egy speciális egyezményes érték, ami mindig az aktuális alhálózatban megcímezhető legnagyobb érték, ami egy konkrét csomópont címe soha nem lehet. Az üzenetszórás egy speciális esete a „multicast”, amikor bár nem minden egyes csomóponthoz kell eljusson egy csomag, de több címzett van.

A hálózatokat rendszerint valamilyen gráf szerkezettel ábrázoljuk, leginkább fákkal, ebből

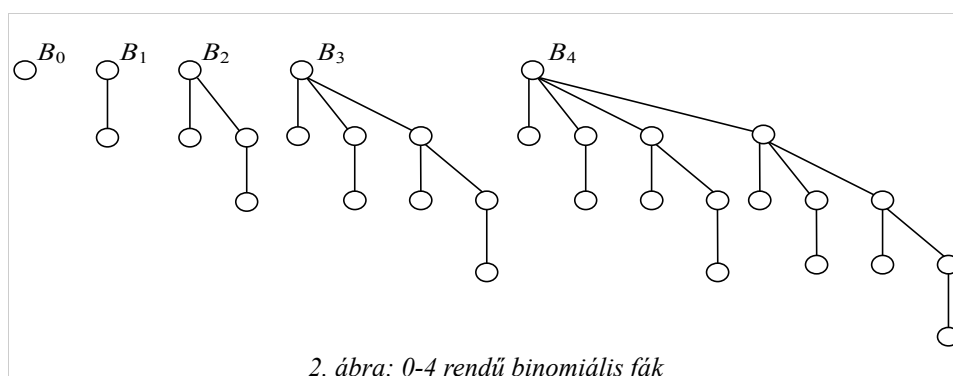
adódóan a leghatékonyabb forgalomirányítás az üzenetszórásra az optimális feszítőfára épül. Mivel a gyakorlatban az optimális feszítőfa meghatározása NP teljes feladat, közelítő megoldásokat, heurisztikus algoritmusok használunk. Ezen algoritmusok hatásfokát a forgalomirányításban alkalmazva nagyon jónak tekinthetjük, ha az optimális hatékonyság 70%-át elérik [3]. Általánosan jó megoldást nehéz találni, azonban bizonyos körülményekhez, hálózati topológiákhoz igazodva lehetséges jól teljesítő módszereket kidolgozni. Az ilyen jellegű forgalomirányításra használt fákat üzenetszórási fának (broadcast tree) nevezzük. Minden csomópont a beérkező broadcast csomagokat a fa élei által mutatott szomszédok felé továbbítja. Az üzenet az indulási pontból ezáltal minimális költséggel juthat el minden más pontba, a feszítőfa éleit követve. A fák fontos tulajdonsága itt, hogy aciklikusak, ezért megoldható az üzenet-redundancia probléma is, ami a drótnélküli hálózatban jelentős, mivel nem csak a közvetlen címzettek kapnak meg egy üzenetet, hanem minden csomópont, amelyhez eljut a rádiójel.

A hagyományos hálózatokra kidolgozott üzenetszórási fák nem viselik jól a gyors topológia változásokat, ezeket gyakran újra kell építeni a kisebb változások miatt is, ezért ezek alkalmazását a drótnélküli hálózatokban sokáig elvetették. Új lehetőségeket kínálnak viszont az újabb, dinamikusabb adatszerkezetek, amelyekkel fákat ábrázolhatunk. A hagyományos feszítőfák jó tulajdonságait és a dinamikus, gyorsan végrehajtható műveleteket ötvözik a binomiális fák. Ezeket főként osztott alkalmazások használták, de az utóbbi években számos figyelemreméltó eredmény született ezek alkalmazására a drótnélküli számítógép-hálózatok útvonalválasztó módszereiben, ezt mutatjuk be a következő fejezetekben.

A gyakorlatban az üzenetszórás nagyon széles körben használják, úgy a kábeles, mint a drótnélküli hálózatokban, mobil hálózatokban, osztott és párhuzamos rendszerekben (főleg lineáris algebra algoritmusok), de még a p2p (peer to peer) hálózatokban a keresési műveletek is gyakorlatilag üzenetszórást takarnak, mivel egy adott üzenet ideális esetben minden egyes csomópontba el kell jusson.

## 5. A binomiális fák alkalmazása az üzenetszórás útvonalválasztó módszereiben

### 5.1. A binomiális fák



Egy 0-ad rendű binomiális fa 1 csomópontból áll, egy  $k > 0$  rendű pedig két  $k-1$ -ed rendű binomiális fából, amelyek közül az egyik gyökere a másik legjobboldalibb (vagy legbaloldalibb) csomópontja. Egy  $k$ -ad rendű binomiális fának  $2^k$  számú csomópontja van. A fenti  $B_0$

$B_1 B_2 B_3 B_4$  fák binomiális fák.

A  $k$ - rendű fa magassága mindig pontosan  $k$  és a fa  $l$ -edik szintjén  $C_k^l$  számú csomópont található. Amennyiben a kupac tulajdonságnak megfelelően töltjük fel a fát, egy binomiális kupacot kapunk, amely adatszerkezetnek ebben az alkalmazásban számos előnye van:

- a binomiális kupacot egy **binomiális fák tartalmazó láncolt listával ábrázolhatjuk**, az egyes fák gyökerei közé éleket téve, úgy, hogy a fák sorrendjét a listában ezeknek a rendje adja, ez igen gyors feldolgozást tesz lehetővé

- a binomiális kupac az  $n$  szám bináris ábrázolásában szereplő 1-es biteknek megfelelően tartalmaz egy-egy binomiális fát, ahol a fa rendjét a megfelelő bit helye adja

- a legtöbb művelet konstans illetve logaritmikus időben végrehajtható

- két kupacot  $O(n)$  idő alatt egyesíthetünk

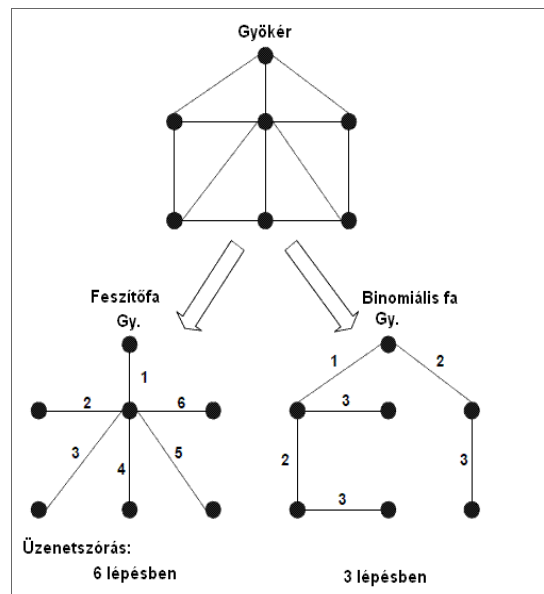
- a keresés algoritmusai lineáris

## 5.2. Binomiális fák az útvonalválasztó (routing) módszerekben

A binomiális fák előnyös alkalmazni az útvonalválasztó algoritmusokban, mivel a szerkezetük lehetővé teszi a gyors keresést és a csomópontok gyors hozzáadását és törlését, ami elsődleges szempont egy ad-hoc drótnélküli hálózat esetén. Két csomópont között van kapcsolat, ha a címkéjük bináris ábrázolása pontosan egy pozícióban különbözik, tehát például az 1. szinten levő csomópontok címkéjében 1 darab 1-es fordul elő a 0.. $k$  pozíciók egyikén, ahol  $k$  a fa rendje. Az útkeresésnél így egy olyan algoritmust kell alkalmazni, amelyik minden lépésben a csomópontok megfelelő bitjei alapján határozza meg a kapcsolatokat.

Amennyiben figyelembe vesszük a csomópontok jelerősségét, akkor ez alapján is csoportosíthatjuk a csomópontokat, ezáltal csökkenthető vagy kiküszöbölhető az üzenet-redundancia problémája és nőni fog a rendszer stabilitása. Ez a csoportosítás rendszerint nagyobb számítási kapacitást igényel, ami viszont az energia fogyasztásnál jár többlettel így befolyásolva a hordozható egységek akkumulátorának az élettartamát.

A főbb lépései az alkalmazásnak a broadcast fa megépítése, ez alapján a csomagok irányítása és a fa karbantartása. A fa megépítése viszonylag egyszerű, ha a protokoll, amelyet broadcast képességgel egészítünk ki, globális információkra támaszkodik a két pont közötti



3. ábra: A feszítőfa és a binomiális fa teljesítményének az összehasonlítása egy 7 csomópontból álló, irányított dipól antennákkal rendelkező drótnélküli hálózatban

útvonalválasztásban, ilyen a DSR protokoll. A csomópontok egy-egy sorszámot kapnak, amelyet megfeleltethetünk a hálózati csatoló fizikai címének. Mikor a hálózatról már rendelkezünk valamilyen szintű globális információval, tehát ismert vagy kiszámítható bármely két csomópont közti távolság, akkor, ez alapján, kerül megépítésre a broadcast fa. Érdeemes valamilyen optimalizált algoritmust használni, mint például a Balanced-Path algoritmus [4], amely képes optimalizálni egy naiv algoritlussal felépített fát. Ezek után elméletileg a legoptimálisabb lenne a topológia minden változása esetén újra építeni a fát, ami azonban megengedhetetlenül költséges, ezért dinamikus, gyors módszereket kell találni a változások, a be és kilépő csomópontok kezelésére [5].

### 5.3. Az üzenetszórás fa felépítése és karbantartása

A különféle fa-építő és optimalizáló algoritmusok kísérleti összehasonlítása [4] azt mutatja, hogy a Balanced-Path algoritmus a leghatékonyabb, az összkommunikáció eloszlását tekintve.

A módszer rövid leírása:

*balanced-path ()*  
*minden n csúcsra*  
*p a legtöbb leszármazottal rendelkező csúcs*  
*x = új kiválasztott gyereke p-nek*  
*x helyének frissítése a fában*

A többi algoritmus gyakran generál egy-egy túl költséges útvonalat a fában, míg ez, ahogyan a neve is mutatja, az utak kiegyensúlyozására törekszik. A módszer úgy rendezi át a fát,

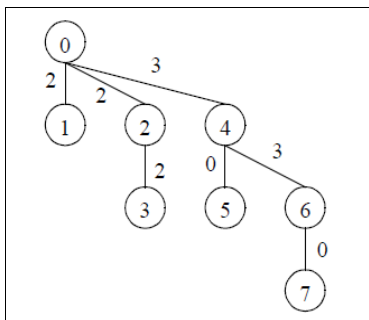
	0	1	2	3	4	5	6	7
0	0	2	2	0	3	3	0	0
1	2	0	0	2	5	5	2	2
2	2	0	0	2	5	5	2	2
3	0	2	2	0	3	3	0	0
4	3	5	5	3	0	0	3	3
5	3	5	5	3	0	0	3	3
6	0	2	2	0	3	3	0	0
7	0	2	2	0	3	3	0	0

4. ábra: Szomszédsági mátrix egy 7 csomópontból álló hálózatra (a 0 távolság közvetlen kapcsolatot jelöl)

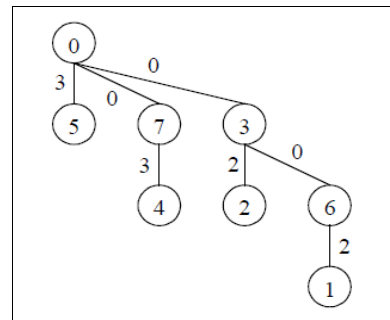
hogy előbb a legtöbb leszármazottal rendelkező csomópontok gyerekeiből a legközelebb esőeket teszi az adott csomópont helyére. Ez kiegyensúlyozza az utakat, mivel a leghosszabbakhoz nyilván azon csomópontok tartoznak, amelyek sok leszármazottal rendelkeznek.

A fenti ábrán látható hálózatban létezik egy 6 és egy 4 hosszú út is, amelyeken még sokat lehet javítani a fa kiegyensúlyozásával. A Balanced-path eljárást alkalmazva sokat javul fa egyensúlya, amint az alábbi ábrán is látszik, már csak 3 a legköltségesebb út hossza.





5. ábra: A hálózathoz naív algoritmussal megépített binomiális broadcast fa



6. ábra: A Balanced-Path eljárás jelentősen javította az utak költségeit

A fák további karbantartásához hatékony algoritmusok kellenek a csomópontok hozzáadására, törlésére és az utak költségeire vonatkozó új információk frissítésére. A módszerek hatékonyságát a fa költségével szokták mérni, ami a leghosszabb út költségét jelenti, illetve egy kiegyensúlyozottabb lehetőség az összes él, kapcsolat, költségeinek a szummázása. Ezen adatok újraszámítása  $O(n)$  időben lehetséges, mivel elégséges az előző értékre és az aktuális változásokra alapozni.

A különböző megközelítéseket használó algoritmusokkal (Family Swapping Algorithm, Path Swapping Algorithm, Leaf Swapping Algorithm, Position Swapping Algorithm) végzett szimulációk [6] azt igazolták, hogy átlagosan a legjobb teljesítményt a Path Swapping algoritmus nyújtotta, ezért a továbbiakban ezzel fogunk foglalkozni. Ez az algoritmus legrosszabb futási ideje (elvégzett lépések száma) is logaritmikus és csak a csomópontok számától függ. A módszer lényege, hogy ha két szomszédos csúcsot helyet cserél és megnő a költség (legyen „a” a szülő és „b” a leszármazott csomópont), akkor az algoritmus megpróbálja az a-t egy megfelelőbb helyre mozgatni a gyökér felé vagy a b-t, a leghosszabb úton, a levelek felé mozgatva. A gyökér csomópont soha nem kerülhet módosításra. Mivel ez az eljárás csupán egyetlen út mentén végez műveleteket, a többihez viszonyítva ez jár a legkevesebb számítási költséggel, ezért érdemes alkalmazni.

## 6. Az algoritmusok tesztelése

Jist/Swans - <http://jist.ece.cornell.edu/>

Mivel az ad-hoc hálózati protokollokat nehézkes a gyakorlatban tesztelni, megfelelő topológiájú hálózati felállásokat kialakítani és elegendő tesztet végezni, ezért az algoritmusokat szimulált környezetben lehetséges megfelelően fejleszteni. Ebben nyújtanak segítséget a JiST / SWANS rendszerek ( Java in Simulation Time / Scalable Wireless Ad-hoc Network Simulator ), amelyek lehetővé teszi a gyakorlatban alkalmazott bináris kódok futtatását szimulált, előre beállított környezetben. A Jist (Java In Simulation Time) egy járában írt diszkrét-esemény szimulációs motor, amire ráépítették a SWANS (Scalable Wireless Ad hoc Network Simulator) rendszert, amely az ad hoc drótnélküli hálózatok szimulációjához biztosít egyszerű és hatékony felületet.

A többi széles körben használt szimulátorokhoz képest (pl. ns2, GloMoSim) nagy előny, hogy egyszerűen kezelhető, könnyen megtanulható, optimalizálták a memóriahasználatot és a hálózati kommunikációt a jáva socket osztályára építették. Mivel a szimulátor felüldefiniálja a jáva socket osztályt, ezért az egyes csomópontokon futtatott programok, ha szükséges, akár a külső, „valós”, nem szimulált hálózatot is elérhetik, illetve a szimulált környezetben minden változtatás nélkül futtathatóak a gyakorlatban is működő programok, jelentősen megkönnyítve ezáltal úgy a

hálózati réteg programjainak, pl. hálózati protokollok, mint az osztott alkalmazásoknak a tesztelését. A szimuláció akár nagy számú csomópontok mellett is lehetséges egy asztali számítógépen, mivel a régebbi nagy szimulátor rendszerekkel ellentétben még több ezer csomópont mellett is igen kis memória és számítási kapacitásra van szükség. A SWANS akár egy millió csomópont tevékenységeit is képes szimulálni egy *2.0 GHz-es uni-processzor számítógépen 2GB RAM* memóriával [7] [8] .

## **7. Következtetések, jövőbeli munka**

A binomiális fák alkalmazásának előnyei különösen a kábeles hálózatokban voltak érzékelhetőek, azonban ha a sokkal energiatakarékosabb és jobb teljesítményű irányított drótnélküli antennákat használjuk a drótnélküli hálózatunkban, akkor az ehhez adaptált változat jól fog teljesíteni a drótnélküli környezetben is. További előnye, amit a jövőben érdemes lesz megvizsgálni, hogy különféle topológiák esetén is hatékonyan alkalmazható. Egyértelmű, hogy a hálózat adatszerkezetének a dinamikus frissítése, módosítása a legjobb megoldás az újraépítéssel szemben, ezért ebben az irányban érdemes további vizsgálatokat folytatni.

Hasznos lenne több ilyen algoritmust a jövőben a Jist/Swans rendszer számára implementálni, mivel a kísérletek, szimulációk többségét rendre igen nagy teljesítményű munkaállomásokon illetve szuperszámítógépeken végzik. Ezáltal az ilyen szimulációk sokkal szélesebb körben elvégezhetővé válnának, ami felgyorsíthatná az új eredmények megjelenését a témakörben.

## 8. Könyvészet

[1] Broadcasting Methods in Mobile Ad Hoc Networks: An Overview, Demetres Kouvatsos and Is-Haka Mkwawa - THIRD INTERNATIONAL WORKING CONFERENCE, PERFORMANCE MODELLING AND EVALUATION OF HETEROGENEOUS NETWORKS West Yorkshire, U.K., Monday 18th - Wednesday 20th July, 2005

[2] Multi-Channel Opportunistic Routing in Multi-Hop Wireless Networks, ANATOLIJ ZUBOW, MATHIAS KURTH and JENS-PETER REDLICH - International Conference On Communications And Mobile Computing, Vancouver, British Columbia, Canada, 2006

[3] Broadcast Trees for Heterogeneous Platforms, Olivier Beaumont, Loris Marchal, Yves Robert, In *International Parallel and Distributed Processing Symposium IPDPS'2005*, 2005. IEEE Computer Society Press.

[4] Improving Binomial Trees for Broadcasting in Local Networks of Workstations, Silvia M. Figueira, VECPAR'02, Porto, Portugal, June 2002.

[5] Heap-uri binomiale, Mihai Scorțaru - *Gazeta de informatică*, 14/7 – noiembrie 2004, p.28-33

[6] Dynamically Adaptive Binomial Trees for Broadcasting in Heterogeneous Networks of Workstations, Silvia M. Figueira and Christine Mendes, VECPAR 2004, 6th International Conference, Valencia, Spain, June 28-30, 2004, Revised Selected and Invited Papers. *Lecture Notes in Computer Science 3402 Springer 2005, ISBN 3-540-25424-2*, pag. 480-495

[7] JiST– Java in Simulation Time - User Guide, Rimon Barr, March 19, 2004, <http://jist.ece.cornell.edu/>

[8] SWANS– Scalable Wireless Ad hoc Network Simulator - User Guide, Rimon Barr, March 19, 2004, <http://jist.ece.cornell.edu/>