Bezdrôtové senzorové siete

MICHAL PECÚCH

2010

Bezdrôtové senzorové siete

DIPLOMOVÁ PRÁCA

Michal Pecúch

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
KATEDRA INFORMATIKY

Študijný odbor: INFORMATIKA

Vedúci diplomovej práce:
Doc. RNDr. Rastislav Královič, PhD.

BRATISLAVA 2010

Abstrakt

PECÚCH, Michal: Bezdrôtové senzorové siete. [diplomová práca] – Fakulta matematiky, fyziky a informatiky Univerzita Komenského v Bratislave;
Katedra informatiky. – Školiteľ: doc. RNDr. Rastislav Královič, PhD. – Bratislava 2010. 73 strán.

Práca tvorí ucelené zhrnutie umožňujúce ľahkú orientáciu v súčasnom hardvéri a protokoloch využívaných v bezdrôtových senzorových sieťach. Protokoly rozdeľuje podľa zaužívaných sieťových vrstiev a podľa spoločných vlastností a kritérií. Práca by mala byť chápaná ako dizajnérska príručka pri návrhu a tvorbe nových sietí tohoto typu nakoľko táto problematika je pomerne rozsiahla a neustále sa rozvíja.

Kľúčové slová: bezdrôtové sieťe, distribuované algoritmy, grafové algoritmy, komunikácia, topológia

Ďakujem svojmu vedúcemu diplomovej práce, doc. RNDr. Rastislavovi Královičovi, PhD. za jeho rady a pripomienky pri tvorbe tejto práce.

Čestne prehlasujem, že túto diplomovú prácu som vypracoval samostatne len s použitím uvedenej literatúry.

V Bratislave 5 mája 2010 _____

Obsah

Zoznam obrázkov 8

Zoznam tabuliek 9

Zoznam skratiek 10

1 Úvod 11

2 Hardware 14

3 Fyzická vrstva 18

3.1 Modely šírenia sa signálu 18

3.2 Chybové modely 20

3.3 Modely citlivosti 21

4 Dátová vrstva 23

4.1 Podvrstva pre riadenie prístupu k médiu 23

4.1.1 Protokoly s pevným priradením prístupu ku kanálu 25

4.1.2 Protokoly s náhodným priradením prístupu ku kanálu 27

4.1.3 Protokoly s priradením prístupu ku kanálu na požiadanie 28

4.2 Podvrstva pre kontrolu logickej linky 29

5 Sieťová vrstva a routovacie protokoly 31

5.1 Protokoly s výmenou globálnych routovacích informácií 33

5.2 Protokoly s výmenou lokálnych routovacích informácií 34

6 Transportná vrstva 38

6.1 Protokoly pre posielanie jednotlivých paketov 39

6.2 Protokoly pre posielanie viacerých paketov 40

6.3 Protokoly riešiace problém zahltenia v BSS 41

7 Riadenie topológie 43

7.1 Protokoly riadiace vysielací výkon 45

7.1.1 Problém kritického vysielacieho výkonu 46

7.1.2 Problém priradenia vysielacieho výkonu 46

7.1.3 Algoritmy z výpočtovej geometrie 47

7.2 Distribuované protokoly pre riadenie topológie 47

7.2.1 Protokoly na princípe lokalizácie 47

7.2.2 Protokoly na princípe smeru vysielania 50
7.2.3 Protokoly na princípe počtu susedov 52
7.2.4 Protokoly na princípe existencie routovacích trás 55
7.3 Protokoly budujúce hierarchiu v sieti 55
7.3.1 Protokoly budujúce chrbticu v sieti 56
7.3.2 Protokoly vytvárajúce clustre 60
7.3.3 Adaptívne protokoly 61
7.4 Hybridné protokoly pre riadenie topológie 62
8 Prípadová štúdia 64
9 Záver 66
Použitá literatúra 67

Zoznam obrázkov

Obrazok 1: Schématická architektúra prvku BSS
Obrazok 2: Dvojlúčový povrchový model
Obrazok 3: Markov dvojstavový chybový model
Obrazok 4: Problém skrytého a nechráneného terminálu
Obrazok 5: Triedenie MAC protokolov
Obrazok 6: Protokoly pre detekciu a opravu chýb
Obrazok 7: Triedy routovacích protokolov
Obrázok 8: Protokoly transportnej vrstvy
Obrazok 9: Rozdelenie protokolov pre TC
Obrazok 10: DistRNG
Obrázok 11: Preposielací región a uzáver uzla
Obrazok 12: K-susedský graf a jeho symetrický nadgraf a podgraf

Zoznam tabuliek

Tabulka 1: Frekvencie ISM
Tabulka 2: Hodnoty α pre niektoré prostredia pre logaritmický model šírenia signálu.

Zoznam skratiek

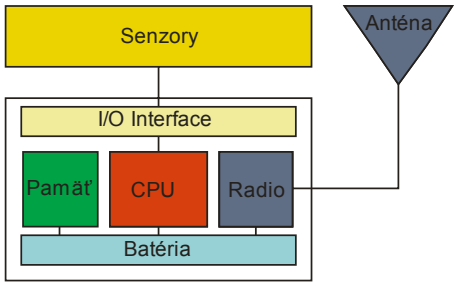
ARQ	– Automatic Repeat reQuest (Automatic Repeat Query)
BER	– Bit Error Rate
bps	– Bit Per Secound
BSS	– Bezdrôtové Senzorové Siete
CDMA	– Code-Division Multiple Access
CRC	– Cyclic Redundancy Check
CSMA/CD	– Carier Sence Multiple Access with Collision Detection
CTR	– Critical Transmission Range
CTS	– Clear To Send
EMST	– Euclidean Minimal Spaning Tree
FDMA	– Frequency-Division Multiple Access
FEC	– Forward Error Correction
ISM	– Industrial, Science and Medical radio bands
LLC	– Logical Link Control
MAC	– Medium Access Control
RA	– Range Assignment
RISC	– Reduced Instruction Set Somputer
RTS	– Request To Send
TC	– Topology Control
TDMA	– Time-Division Multiple Access

Kapitola 1
Úvod

Nedávno predstavený koncept bezdrôtových senzorových sieti (BSS) si za posledné roky získal veľkú pozornosť. Rozvoj v oblasti komunikačnej technológie a mikroelektroniky umožnil vznik veľmi zaujímavých riešení. Ako názov napovedá, ide o bezdrôtové siete typu ad-hoc, ktoré sa samé organizujú a ich úlohou je zbierať a prípadne vyhodnocovať informácie získané z prostredia na základe inštalovaných čidiel a senzorov, následne tieto informácie doručiť do konkrétneho miesta v sieti, kde sú ponúknuté na ďalšie spracovanie alebo vyhodnotenie. V dnešnej dobe sa začíname stretávať s BSS, ktorých prvky dokonca na základe informácií, ktoré zozbierali a boli vyhodnotene centrálnou jednotkou, dokážu vykonávať jednoduché úlohy, ako napríklad premiestniť sa alebo iným spôsobom reagovať. V súčasnej dobe sú prvky bezdrôtovej siete kvôli dôrazom kladeným na nízku cenu a nízku spotrebu energie pomerne obmedzené čo sa týka výpočtovej sily a pamäte. Pre ne vyvinuté algoritmy a protokoly sa preto musia zaoberať rôznymi obmedzeniami a hranicami a tvoria tak vlastnú doménu v oblasti komunikačných protokolov. Avšak ako sa zdá a súčasný vývoj tomu aj nasvedčuje, možnosti týchto sietí budú v budúcnosti omnoho väčšie. Dá sa predpokladať, že miniaturizáciou budú prvky čoraz menšie a zároveň čoraz výkonnejšie a teda bude možné využiť aj multimediálne aplikácie alebo širokopásmový prenos dát. Už dnes široké spektrum dostupných senzorov sa ďalej rozšíri a tým aj aplikačné využitie senzorových sieti bude omnoho rôznorodejšie. Ako príklad môže slúžiť len nedávno predstavený koncept pod názvom smartdust, pri ktorom jednotlivé uzly siete majú veľkosť zrniek prachu, sú schopné vznášať sa v prostredí a monitorovať kvalitu ovzdušia, teplotu alebo svetlosť prostredia.

Samotná architektúra jednotlivých uzlov je pomerne jednoduchá. Ide o prvky skladajúce sa z riadiacej jednotky, pamäte, rádiového transcevera, batérie a senzora väčšinou pripojeného k procesoru pomocou nejakého vstupno-výstupného konvertora alebo interfejsu. Ako procesor sa zväčša používa jednoduchý mikrokontroler s obmedzenou inštrukčnou sadou a malým počtom registrov. Je to z dôvodu menších rozmerov a menších nárokov na spotrebu energie. Často sa jedná o špeciálne navrhnuté RISC procesory z nízkou taktovacou

frekvenciou. Ako pamäť sa zväčša používa flash pamäť nakoľko je pomerne lacná, ma nízku spotrebu, vysokú odolnosť a životnosť. Tu môže byť uložené aj úplné programovanie mikroprocesora, čo nám ušetrí nutnosť ďalšej ROM pamäte. Ako batérie sa zväčša používajú dve batérie typu AA s dlhou životnosťou, avšak môžme sa stretnúť aj s akumulátormi či solárnymi článkami. Druhou najdôležitejšou súčasťou uzla je jeho senzorové vybavenie. Môže sa jednať o akékoľvek zariadenie generujúce informácie od GPS prijímača cez tepelné, svetelné či tlakové čidlá až po mikrokamery alebo mikrofóny. Najdôležitejšou súčasťou každého uzla je jeho komunikačne zariadenie. Jedná sa zväčša o modul, ktorý je schopný vysielať a prijímať v určitom frekvenčnom spektre. Takéto zariadenie sa nazýva transcever. Vo všeobecnosti sa v BSS používajú nízko výkonové transcevery v spektre pre priemysel, vedu a medicínu znamom aj pod skratkou ISM, ktoré nepotrebuje licencovanie telekomunikačným úradom. Väčšina transceverov dokáže fungovať v štyroch režimoch, pričom každý z týchto režimov ma iné energetické nároky. Sú to vysielanie, príjem, standby a spánok. Mnohí výrobcovia ponúkajú dokonca viacero úrovní jednotlivých režimov, kedy dochádza k deaktivácii alebo aktivácii len niektorých okruhov v rámci komunikačnej jednotky, alebo sa v určitých krokoch mení vysielací výkon či citlivosť prijímača. Samotné transcevery dokážu poskytovať už konkrétne komunikačné protokoly ako napríklad Bluetooth či ZigBee, alebo sa môžu k dátam správať transparentne ako napríklad rôzne rádiové modemy. Ďalej samotný transcever môže poskytovať rôzne frekvenčné filtre, konvertory medzi analógovým a digitálnym signálom, zosilňovače, mixéry a podobne. Rovnako rôzne transcevery sa líšia vo vzdialenosti, na ktorú dokážu komunikovať a šírke pásma, ktoré dokážu zabezpečiť pre komunikáciu. V súčasnosti existuje na trhu veľké množstvo komponentov. Záleží len od tvorcu siete a účelu, za akým má sieť fungovať a aké aplikácie má podporovať.



Obrázok 1 : Schematická architektúra prvku BSS

Sieťová architektúra BSS záleží predovšetkým na počte jednotlivých uzlov a ich hustote, ďalej na účele, za akým je samotná sieť prevádzkovaná. V prípade malých sieti nám zväčša stačí komunikácia uzlov priamo s riadiacou jednotkou, prípadne nejaké centralizované riadenie topológie. V prípade sietí v ktorých sa počty prvkov počítajú na tisíce či desaťtisíce a pokrytie na kilometre, centrálne riadenie sa stáva neefektívnym a nespoľahlivým. V takomto prípade potrebujeme distribuované riadiace protokoly, ktoré nám zabezpečia zorganizovanie siete a jej fungovanie a zaručia, že informácie zozbierané v jednotlivých uzloch budú doručené do vopred stanoveného miesta. Vzhľadom na už spomínané obmedzujúce vlastnosti jednotlivých uzlov to rozhodne nie je jednoduchá úloha. Preto musia byť protokoly jednoduché, no na druhej strane spoľahlivé do takej miery, aby sieť správne fungovala. V tomto smere vývoj napreduje snáď ešte rýchlejšie ako v oblasti vývoja hardvéru. Uskutočňuje sa mnoho konferencií, na ktorých sa stretávajú ľudia zaoberajúci sa senzorovými sieťami a predstavujú mnoho zaujímavých riešení a nápadov. Zdá sa, že v tejto oblasti je budúcnosť v špecializácii jednotlivých riešení a spoločný zostáva len najhlbší koncept fungovania jednotlivých protokolov, prípadne pôvodné akademické koncepty, z ktorých jednotlivé riešenia vznikajú.

Použitie BSS je v poslednej dobe pomerne časté. Svoje využitie našli v mnohých odvetviach a odboroch. Používajú sa napríklad na kontrolu zavlažovania na veľkých poliach. Ďalej armádne a bezpečnostné využitie na detekciu narušenia. V oblasti priemyslu a ekológie je to monitorovanie prítomnosti škodlivých prvkov v prostredí či na detekciu lavín alebo lesných požiarov. Existujú projekty na kontrolu prostredia v budovách so zameraním na kvalitu ovzdušia, teplotu či svetelnosť. Vznikajú mnohé projekty pre dopravu. Ako môžme vidieť, možností na využitie je veľmi veľa, preto nie je žiadnym prekvapením zvyšujúca sa popularita.

Kapitola 2
Hardvér

Pre bezdrôtové senzorové siete je možné použiť ľubovoľnú frekvenciu a zariadenia, ktoré ich používajú. Avšak zariadenia dostupné na trhu zväčša používajú frekvencie ISM, ktoré nie sú striktne kontrolované väčšinou telekomunikačných úradov. Obmedzenia sú zväčša len v šírke pásma a maximálnom vysielacom výkone, ktorý možno použiť bez licencie. Samozrejme pre rôzne aplikácie BSS je možné použiť aj rôzne frekvencie. V takomto prípade záleží od realizácie, lokálnej legislatívy nasadenia BSS a prostriedkov, ktoré sú k dispozícii. V tejto kapitole sa budeme zaoberať len štandardami pre ISM [58] a zariadeniami fungujúcimi v týchto frekvenciách. V súčasnosti je mnoho spoločností ponúkajúcich na trhu komunikačné moduly líšiace sa podporovanými protokolmi a množstvom ďalších vlastností, ktoré majú za úlohu získať stabilné miesto na trhu pre danú spoločnosť. Z tohoto konkurenčného boja vychádzajú zariadenia s pomerne nízkou cenou a výbornými vlastnosťami či už s ohľadom na spotrebu energie alebo vlastnosti používaného bezdrôtového kanála. Napríklad oprava chýb pri prenose, presné určenie chybovosti kanála a podobne. Tu je rozhodnutie na tvorcovi konkrétnej BSS, pre akú technológiu a pre ktorý konkrétny produkt sa vo svojom riešení rozhodne. Najčastejšie používané pásma sú 240 až 930 MHz pre RF zariadenia a 2400 až 2524 MHz pre technológie Wifi, Bluetooth a ďalšie im podobné.

Frekvenčný rozsah [Hz]	Stredová frekvencia
6.765–6.795 MHz	6.780 MHz
13.553–13.567 MHz	13.560 MHz
26.957–27.283 MHz	27.120 MHz
40.66–40.70 MHz	40.68 MHz
433.05–434.79 MHz	433.92 MHz
902–928 MHz	915 MHz
2.400–2.500 GHz	2.450 GHz
5.725–5.875 GHz	5.800 GHz
24–24.25 GHz	24.125 GHz
61–61.5 GHz	61.25 GHz

122–123 GHz	122.5 GHz
244–246 GHz	245 GHz

Tabuľka 1: Frekvencie ISM

Následne uvádzame niekoľko zariadení s niektorými ich vybranými vlastnosťami. Zariadenia a ich cena sú porovnané na webovom portáli [59]. Tu je možné nájsť prepojenie priamo na stránky ich výrobcov a k ich kompletným špecifikáciam. Ďalej štandardy pre Wifi, Bluetooth a Zigbee je možné nájsť na stranách IEEE dokumentácie [60]. Konkrétne sú to štandardy IEEE 802.11 a, b, g a n (WiFi), IEEE 802.15.1 (Bluetooth) a IEEE 802.15.4 (ZigBee/XBee).

HAC-UM96 [61] – Jedná sa o 10mW rádio modul pracujúci s frekvenciami 433MHz. Dosah do 500m v otvorenom priestore. Vie použiť až 32 oddelených dátových kanálov a dosiahnuteľná prenosová rýchlosť je 9600bps. Modul zaručuje transparentný prenos dát, takže môžu byt použité ľubovoľné komunikačné protokoly. Nízka spotreba energie pre vysielanie a príjem je menej ako 40 a 30 mA, v spiacom režime menej ako 20μA.

RFM12 a RFM22 [65] – Sú to dva produkty jednej spoločnosti líšiace sa programovateľnosťou a rozšírenou funkčnosťou. Pracujú v pásmach 240 až 930 MHz. Ide o transceverové moduly. Výrobca ponúka aj rozdelené verzie vysielač a prijímač. Jedná sa teda o transcever pre rádiový prenos. Dosiahnuteľná prenosová rýchlosť je až 256 kbps. Energetická spotreba pre vysielanie je 27mA pri vysielacom výkone 11dBm a 18,5mA pre príjem. V standby režime je spotreba okolo 0,3 μA Vysielací výkon je nastaviteľný od 8 do 17dBm. Citlivosť prijímača -118dBm. Transcever je schopný analógovej aj digitálnej indikácie sily prijímaného signálu. Moduly majú zabudované hodiny pre zobúdzanie sa zo spánku.

RN41 [62] - Ide o modul využívajúci štandard Bluetooth. Prenosová rýchlosť pri stabilnom neprerušovanom prenose je 240kbps. 3Mbps pri vysielaní v návaloch (burstoch). Spotreba energie 65mA pri vysielaní 10mA pri počúvaní. Pri spánku je minimálna spotreba 250 μA, modul však podporuje aj režim hlbokého spánku, v ktorom je spotreba minimálna. Tento modul podporuje väčšinu funkcií a protokolov vyvinutých pre Bluetooth. Využíva tri

frekvenčné pásma 2402, 2441 a 2480 MHz. Citlivosť prijímača je -86dBm a maximálny vysielací výkon je16dBm.

RN-131G [62] - Tento modul používa komunikačný štandard WiFi, je teda navrhnutý pre protokoly a komunikáciu na báze TCP/IP s plnou podporou väčšiny funkcií sieti tohoto typu. Spotreba energie je 210mA pre vysielanie, 40mA pre príjem a 4 μA v režime spánku. Modul podporuje šifrovanie WEP až WPA2. Ide v podstate o minimalizovaný štandardný modul pre WiFi známy z mobilných zariadení a počítačov s upravenými vlastnosťami týkajúcimi sa energetickej spotreby.

nRF2401A a nRF24L01+ [64] - Tieto moduly pracujú v pásme 2400MHz. Opäť sa jedná o dva moduly od jedného výrobcu líšiace sa predovšetkým v možnostiach úprav a vybavení. Dosiahnuteľná rýchlosť je 256kbps,1Mbps alebo 2Mbps. 2Mbps dosahuje len vylepšená verzia nRF24L01+. Moduly podporujú 250 kanálov pričom prepínací čas pri zmene kanála je menej ako 200 μs. Spotreba energie pre vysielanie a príjem je približne 9mA pri standby podľa režimu 26 alebo 320 μA a pri vypnutí 900nA. Citlivosť prijímača je až -94dBm

XBee a XTend [63] – Jedná sa o zariadenia pracujúce v pásme 900 alebo 2400MHz podľa štandardu ZigBee IEEE802.15.4 špeciálne navrhnutého pre siete typu BSS. Ide o štandard odvodený od štandardu Bluetooth, kde sa znížením prenosovej rýchlosti zvýšila vzdialenosť, v ktorej vedia medzi sebou tieto zariadenia komunikovať. V tejto časti uvedieme niekoľko modulov, ktoré sa od seba líšia vysielacími výkonmi. Výrobca spomínaných zariadení rozdelil svoje produkty do troch tried. Prvá trieda je pre vnútorné použitie, druhá pre vonkajšie použitie a tretia pre vonkajší prenos na veľké vzdialenosti. Všetky typy okrem najsilnejších sú v štyroch prevedeniach v závislosti od použitej antény, s integrovanou drôtovou anténou, integrovanou chipovou anténou, RPSMA alebo U.FL konektorom.

- **XBee 1mW a 2mW** – Sú to najslabšie moduly určené pre vnútorné použitie s dosahom do 100m. Používajú pásmo 2400MHz, maximálna prenosová rýchlosť je 250kbps. Citlivosť prijímača je -97dBm a vysielací výkon 1 alebo 3dBm v silnejšej verzii. Spotreba energie je 35mA pri vysielaní, 38mA pre príjem a maximálne 1 μA

pri spánku. Moduly podporujú 128AES šifrovanie, 64-bitovú hardvérovú adresu a 16 kanálov.

- **XBee PRO 50mW a 60mW** – Sú to moduly so stredným výkonom pracujúce v pásme 2400MHz a dosahom do 1,5km. Maximálna prenosová rýchlosť zostáva 250kbps, rovnako ako pri predchádzajúcom zariadení. Citlivosť prijímača je -100dBm, vysielací výkon 18dBm v silnejšej verzii a 17 dBm v slabšej verzii. Spotreba energie 215mA pre vysielanie, 55mA pre príjem a menej ako 10 μA v spánku. Taktiež je tu podpora 128bitového AES šifrovania a 64bitovej hardvérovej adresy, avšak moduly vedia používať len 12 komunikačných kanálov.
- **XBee PRO 900** - Patrí do triedy najsilnejších modulov. Pracuje v pásme 900MHz a dosah modulu je okolo 10km. Maximálna prenosová rýchlosť je 156kbps, citlivosť prijímača je -100dBm a vysielací výkon 17dBm. Spotreba energie je 210mA pre vysielanie, 80mA pre príjem a 60 μA pre režim spánku. Tento modul má rovnaké vlastnosti ako predchádzajúce moduly a taktiež 12 použiteľných kanálov alebo možnosť využitia jediného širokého kanálu.
- **XBee PRO 900 XCS** – Tento produkt taktiež patrí medzi najsilnejšie moduly. Používa pásmo v rozmedzí 902 až 928MHz. Dosah prenosu je až 15km. Citlivosť prijímača je v tomto prípade -106dBm a maximálny vysielací výkon 20dBm. Energetická spotreba pri vysielaní je 265mA a 65mA pre príjem. V spánku je spotreba 45 μA. Tento modul nepodporuje AES šifrovanie a hardvérová adresa je zmenšená na 32bitov. Modul vie použiť len 7 kanálov v danom spektre.
- **XTend 900 1W** – Najsilnejší modul spomínaného výrobcu taktiež pracuje v pásme 900MHz. Pri vysielaní s vysokoziskovou smerovou anténou je dosah až 64km, s klasickou dipólovou anténou 22km. Prenosová rýchlosť je 10 alebo 125kbps, maximálna citlivosť prijímača je -110dBm a vysielací výkon sa pohybuje v rozmedzí 0 až 30 dBm. Spotreba energie pri maximálnom vysielacom výkone je 730mA, pre príjem je to 80mA a pre režim spánku 147 μA. Modul vie pracovať až s päťdesiatimi frekvenciami a podporuje 128 a 256bitové AES šifrovanie.

Spomenuli sme len niekoľko vybraných modulov, ktoré sú v súčasnosti dostupné na trhu. Kritériami ich výberu bola nízka cena, nízka spotreba energie a pásmo, pre aké sú určené.

Kapitola 3
Fyzická vrstva

Uzly bezdrôtových senzorových sieti pre komunikáciu používajú signály, ktoré cestujú medzi vysielačom a prijímačom v rádiových kanáloch. Signál, ktorý je odoslaný vysielačom na určitej energetickej úrovni, je tlmený prostredím a okolitým vysielaním. Platí tu, že signál je prijatý pokiaľ sila signálu pri prijatí je vyššia ako hraničná citlivosť prijímača. V prípade, že signál vyslaný jedným uzlom je prijatý iným, hovoríme, že medzi uzlami existuje rádiový kanál. Útlm kanála priamo závisí od vzdialenosti medzi vysielačom a prijímačom, frekvencie vysielaného signálu a šírky pásma využitého pre tento prenos. Rádiové kanály sú modelovateľné pomocou niekoľkých dôkladne popísaných modelov šírenia signálu. Vo všeobecnej literatúre k bezdrôtovým senzorovým sieťam sa najčastejšie vyskytujú tri hlavné modely [1]. Tieto slúžia na predpovedanie, či medzi dvoma uzlami existuje alebo neexistuje rádiový kanál.

3.1 Modely šírenia sa signálu

Model šírenia sa vo voľnom priestore (*The free space propagation model*)
Je to najjednoduchší model. Používa sa keď je medzi prijímačom a vysielačom priama viditeľnosť bez žiadnych prekážok alebo zábran. Prijatý výkon vo vzdialenosti *d* metrov od vysielača je určená Friisovou rovnicou tlmeného kanála:

$$P_{rx}(d) = \frac{P_{tx} \times G_{tx} \times G_{rx} \times \lambda^2}{(4\pi)^2 \times d^2 \times L} = C_f \times \frac{P_{tx}}{d^2}$$

v ktorej $P_{rx}(d)$ je sila signálu prijatého prijímačom vo vzdialenosti *d*. P_{tx} je sila vyslaného signálu. G_{tx} a G_{rx} sú zisky na výkone na strane vysielača a prijímača, λ je vlnová dĺžka v metroch, $L \geq 1$ zahŕňa stratu v elektrických obvodoch na oboch stranách. C_f je teda konštantou závislou na parametroch komunikujúcich uzlov. Z rovnice teda vyplýva, že útlm signálu je priamo závislý na druhej mocnine vzdialenosti, ktorú musí prekonať. Z rovnice je

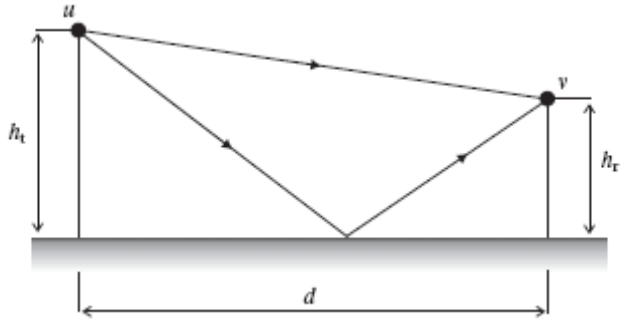
taktiež vidieť, že priestor pokrytia je kruh s polomerom $r = \sqrt{C_f \times P_{tx}}$ so stredom vo vysielači.

Dvojlúčový povrchový model (*The two-ray ground model*)

Tento model predpokladá omnoho realistickejší scenár, kedy prijímač príjme ako signál z vysielača aj jeho odraz od povrchu, ako vidieť na obrázku 2. Tento model je omnoho presnejší ako predchádzajúci model. Sila signálu prijatého prijímačom je vyjadrená nasledujúcou rovnicou:

$$P_{rx}(d) = \frac{P_{tx} \times G_{tx} \times G_{rx} \times h_t^2 \times h_r^2}{d^4} = C_t \times \frac{P_{tx}}{d^4}$$

Zmena oproti predchádzajúcej rovnici je h_t a h_r, ktoré sú výškami antén na strane vysielača prijímača a C_t je opäť konštanta závislá na vlastnostiach oboch uzlov. V tomto prípade je útlm signálu závislý od štvrtej mocniny vzdialenosti uzlov a priestor pokrytia je kruh s polomerom $r = \sqrt[4]{C_f \times P_{tx}}$.



Obrázok 2: Dvojlúčový povrchový model

Dráhový model logaritmickej vzdialenosti (*The log-distance path model*)

Tento model bol vytvorený predovšetkým na základe meraní, experimentov a reprezentovaní nameraných hodnôt. Ako taký by sa mal najlepšie blížiť realite. Tento model sa používa pri návrhoch sieti pre neobvykle prostredia ktoré sú bohaté na prekážky a miesta kde sa signál odráža. Formálne sa tento model vyjadruje rovnicou:

$$P_{rx}(d) \propto \frac{P_{tx}}{d^{\alpha}}$$

v ktorej premena α je závislá od podmienok prostredia a bola experimentálne určená pre viacero scenárov. V tabuľke je uvedených niekoľko hodnôt pre názornosť. Priestor pokrytia je kruh s polomerom $r = \sqrt[\alpha]{P_{tx}}$.

Prostredie	α
Voľný priestor	2
Zastavaná oblasť	2,7-3,5
Vo vnútri s priamou viditeľnosťou	1,6-1,8
Vo vnútri bez priamej viditeľnosti	4-6

Tabuľka 2: Hodnoty α pre niektoré prostredia pre logaritmický model.

Ďalšími dvoma zaujímavými oblasťami úzko súvisiacimi s fyzickou vrstvou sú chyby vzniknuté v komunikačných kanáloch a priestor citlivosti, ktorý nemusí byť zhodný s priestorom pokrytia.

3.2 Chybové modely

Chyby na linkách sa vyskytujú v rôznom počte a v rôznom časovom rozložení. Sú spôsobené prenosovým médiom a vonkajšími vplyvmi ako sú presluchy, rýchlosť odosielania a prijímania, prekážky a rušenie. Bezdrôtové siete sú považované za pomerne chybové a preto je potrebné počítať s týmito chybami v komunikačných protokoloch pre ne navrhnutých. Pri modelovaní a analýze BSS (Bezdrôtových senzorových sieti) sa môžme najčastejšie stretnúť s dvoma chybovými modelmi.

Model nezávislých chýb

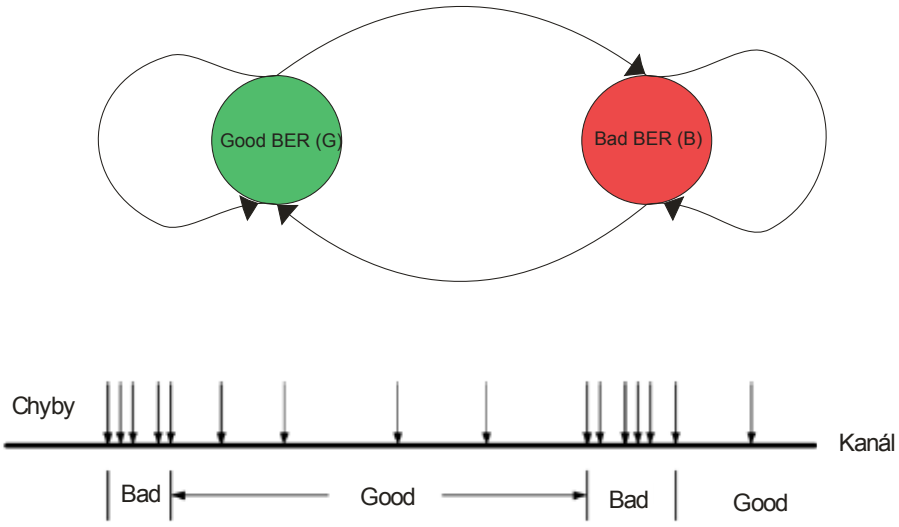
Tento model ako jeho názov napovedá, vychádza z predpokladu, že chyby sa v prenose vyskytujú náhodne. Nepredpokladá sa, že chyby sú na seba nejakým spôsobom naviazané alebo sa ovplyvňujú. V skutočnosti však vieme, že chyby v bezdrôtových linkách sa vyskytujú v návaloch a teda sú na sebe zvájomne závislé. Použitie tohto modelu býva aj

napriek tomu pomerne časté vďaka jeho jednoduchému využitiu pre matematické analýzy. Pravdepodobnosť P_e, že rámec veľkosti L bitov bude prijatý s chybou, je daný rovnicou:

$$P_e = 1 - (1 - p)^L$$

Kde p je úroveň bitovej chyby (BER) kanálu.

Markov dvojstavový chybový model



Obrázok 3: Markov dvojstavový chybový model

Ide o lepší model pre bezdrôtové kanály. Ako možno vidieť na obrázku 3, je znázorniteľný pomocou dvojstavového automatu. Keď je model v stave GOOD(G) predpokladá sa, že úroveň bitovej chyby je dobrá. V tomto stave zostáva model za normálnych okolností väčšinu svojho času. Pokiaľ sa úroveň bitovej chyby zvýši, model sa presunie do stavu BAD(B), v ktorom zotrváva pokiaľ sa vyskytujú chyby v kanáli. Podrobné informácie ako aj návod na výpočet pravdepodobnosti výskytu chýb sú v [2].

3.3 Modely citlivosti

Modely citlivosti sa zaoberajú priestorom citlivosti. Ide o oblasť, v ktorej keď sa niečo udeje, je tento jav zaznamenaný senzorom uzla danej oblasti. Najčastejšie sa môžme stretnúť

s dvoma pomerne jednoduchými modelmi. Tieto modely sa dajú prispôsobiť väčšine typov dnes používaných senzorov.

Binárny model citlivosti

Oblasť citlivosti v tomto prípade je kruh so stredom v senzore. Jav je senzorom detekovaný, pokiaľ bod jeho výskytu nie je ďalej ako je citlivostný dosah senzoru. Inými slovami, pokiaľ sa jav vyskytne vnútri kruhu o polomere citlivostného dosahu senzoru, je tento detekovaný s pravdepodobnosťou 1. Pokiaľ sa udeje mimo kruhu, tak s pravdepodobnosťou 0.

Pravdepodobnostný model citlivosti

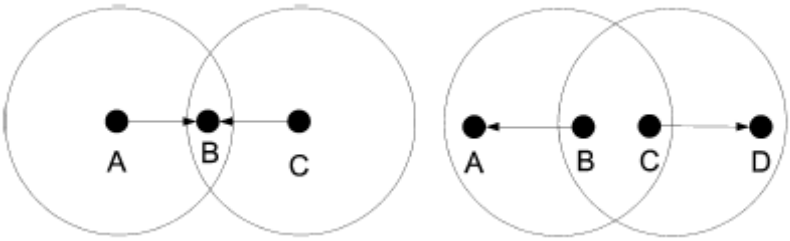
V tomto modeli je oblasť citlivosti rozdelená na tri časti. Oblasť, v ktorej je výskyt javu detekovaný s pravdepodobnosťou 1, ďalej oblasť v ktorej jav nie je detekovaný - teda je detekovaný s pravdepodobnosťou 0. Tretia oblasť je známa ako oblasť s nejasnou pravdepodobnosťou. Pravdepodobnosť detekovania v tejto oblasti exponenciálne klesá so vzdialenosťou od senzoru. Podrobnejšie informácie sa dajú nájsť v [3].

Kapitola 4
Dátová vrstva

V bezdrôtových senzorových sieťach (BSS) sa dátová vrstva skladá z dvoch podvrstiev. Sú to podvrstva pre riadenie prístupu k médiu (MAC) ktorá priamo komunikuje s fyzickou vrstvou a podvrstva pre kontrolu logickej linky (LLC), ktorá poskytuje služby vyšším vrstvám.

4.1 Podvrstva pre riadenie prístupu k médiu

Je to vrstva zodpovedná za poskytovanie prístupu k zdieľanému médiu, v tomto prípade bezdrôtovému komunikačnému kanálu. Stará sa o to, aby mal každý z uzlov v médiu dostatočnú rýchlosť a aby bol samotný kanál dostatočne využitý. V drôtových médiách je táto vrstva tvorená protokolmi pre predchádzanie kolíziam, prípadne ich odstraňovaniu. Známy je napríklad CSMA/CD protokol zahrnutý v štandarde IEEE 802.3. Aj keď v BSS je cieľ tejto vrstvy rovnaký ako pri drôtových sieťach, musia sa protokoly pre ne určené zaoberať omnoho špecifickejšími problémami. Napríklad spomínaný CDMA/CS má značné problémy s dvoma špecifickými problémami bezdrôtových sieti a to problém skrytého terminálu (*Hidden terminal problem*) a problém nechráneného terminálu (*Exposed terminal problem*).



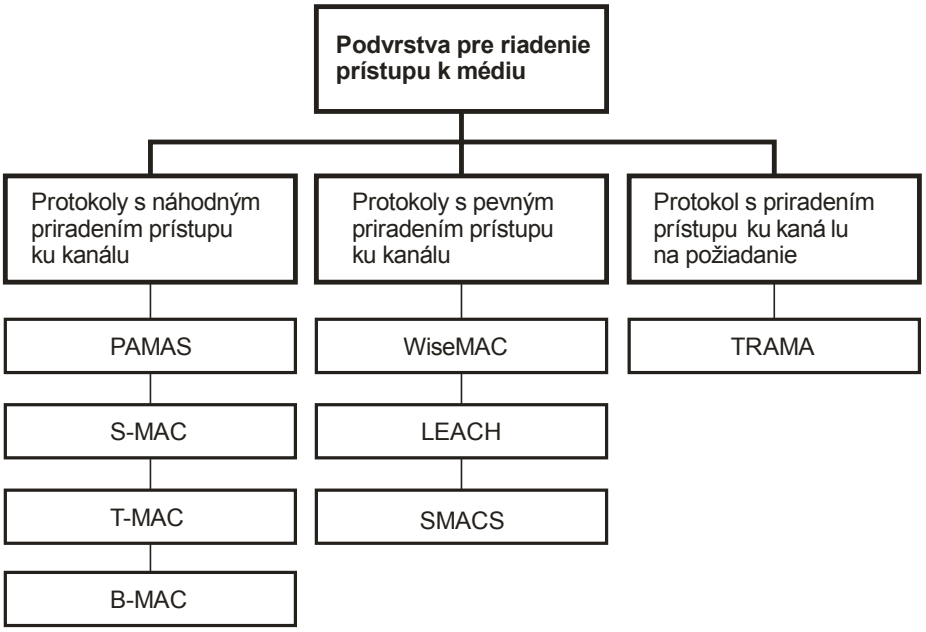
Obrázok 4: Problém skrytého a nechráneného terminálu

V prvom prípade sa uzly A aj C snažia v tom istom čase komunikovať s B bez toho aby vedeli, že sa o to isté pokúša aj druhý uzol. V tomto prípade na uzle B dôjde ku kolízii vysielaných správ. V druhom prípade pokiaľ by B vysielal správu pre A a C by sa snažil

poslať správu D, tak ju nebude možné odoslať, nakoľko komunikačný kanál je už využitý komunikáciou medzi A a B. Tieto problémy zväčša vznikajú pretože rozhodovacia zodpovednosť je na strane vysielača. Na vyriešenie tohto problému treba časť zodpovednosti presunúť aj na prijímač. BSS majú však aj určité špeciálne vlastnosti oproti všeobecným bezdrôtovým sieťam, ktoré by sa mali zohľadniť vždy pri návrhu komunikačných protokolov. BSS väčšinou používajú malé rámce, ktoré sú posielané len veľmi zriedka. Je teda minimálna šanca, že by niektorý uzol obsadil komunikačný kanál na príliš dlhú dobu a znemožnil tak komunikáciu ostatných. Ďalej dáta prenášané medzi uzlami sú dátovo orientované a len zriedkavo sú kladené požiadavky na oneskorenie či rýchly prístup ku komunikačnému kanálu. Takisto vyťaženie kanálu nehrá rolu nakoľko väčšina uzlov v sieti je väčšinu času neaktívnych. Problémy však nastávajú pri riešení otázky spotreby energie. Keďže väčšinu energie spotrebuje vysielanie a prijímanie dát, komunikačný protoko by sa mal vyvarovať určitých nie príliš žiadaných vlastností.

- Kolízie – malo by sa zabrániť vzniku kolízií, aby nedochádzalo k zbytočnému preposielaniu rámcov.
- Overhead – kontrolne správy a dlhé hlavičky by sa mali používať čo najmenej.
- Počúvanie zbytočnej komunikácie – počúvanie komunikácie neurčenej pre konkrétny uzol a dekódovanie správ by sa malo minimalizovať, aj keď pri zdieľanom médiu to nie je jednoduchá záležitosť.
- Počúvanie prázdneho kanála – počúvanie na kanáli, na ktorom nikto nevysiela. Zväčša sa rieši sporadickým vypínaním a zapínaním uzlov.
- Komplexnosti – spracovanie zbytočné zložitých protokolov či už správovo alebo výpočtovo je z dlhodobého hľadiska nevhodné. Preto sa odporúčajú čo najjednoduchšie protokoly.

Protokoly riadiace prístup k médiu sa dajú rozdeliť do troch základných tried podľa toho, akým spôsobom prideľujú zdieľaný kanál.



Obrázok 5: Triedenie MAC protokolov

4.1.1 Protokoly s pevným priradením prístupu ku kanálu

Protokol priradí každému uzlu pevne stanovenú časť zdieľaného kanálu bez toho, aby sa uvažovalo o tom či uzol daný kanál práve potrebuje alebo nie. Táto rodina protokolov zväčša potrebuje centrálne spracovanie nakoľko je veľmi ťažko distribuovane spracovateľná a teda pomerne nevhodná pre siete pozostávajúce z veľkého poctu uzlov. Lepšie výsledky však dostávame pokiaľ je sieť clustrovaná a pokiaľ šéf clustra ma prostriedky na spracovanie, výpočet a riadenie takéhoto priraďovania. Ďalej sa tieto protokoly ľahko integrujú spolu s procedúrami, ktoré dočasne vypínajú a zapínajú uzly siete (*sleep, wake-up*) keďže vieme určiť, kedy bude mat ktorý uzol svoje komunikačné okno. Pri týchto protokoloch dochádza ku kolíziám len zriedkavo. Do tejto triedy patria dobre známe metódy ako TDMA, FDMA a CDMA.

- **WiseMAC** – *Wireless sensor MAC protocol* [12]. Ide o centrálne riadený protokol ktorý používa vzorkovanie úvodov. Uzly vzorkujú médium periodicky v čase, kedy je vysielaný tento úvod. Centrálna jednotka sa stará o to, aby každý uzol počúval v čase, v ktorom budú vysielané preň určené dáta. To je zabezpečené informáciou o časovom

intervale v úvode. Pokiaľ uzol zistí, že médium je v jemu priradenom úseku použité, začne prijímať dáta až do momentu, kedy ich nepríjme všetky.

- **LEACH** – *Low energy adaptiv clustering hierarchy protocol* [13]. Tento protokol používa clustrovanie s tým rozdielom, že šéf clustra nie je iný ako ostatné uzly. Ide skôr o funkciu, ktorú si medzi sebou uzly predávajú, aby dochádzalo k dobrému rozloženiu spotreby energie vo všetkých uzloch clustra. Komunikácia medzi prvkami clustra je založená na TDMA MAC metóde pretože táto adresuje a rieši pomerne dosť problémov ako kolízie, problém skrytého a nechráneného terminálu a iné. Nevýhodou tohoto protokolu je, že nepodporuje komunikáciu medzi jednotlivými clustrami, ale len komunikáciu šéfov clustra s centrálnou jednotkou. Za týmto účelom používa komunikáciu založenú na CDMA, aby všetky clustre s centrálou vedeli komunikovať naraz. Je zrejmé, že agregácia dát z celého clustra môže byť v takomto prípade prospešná a môže šetriť energiu. Na druhej strane komunikácia s centrálnou jednotkou na väčšiu vzdialenosť a takisto spracovanie agregovaných dát je energeticky náročnejšie. Značnou nevýhodou je aj nutnosť vzájomnej komunikácie medzi centrálnou jednotkou a šéfom clustra na jeden skok, čo znižuje priestorové možnosti samotnej siete ako celku. Tento protokol je však často používaný ako základ pre mnohé iné a budeme sa mu ešte neskôr venovať.
- **SMACS** – *Self-organizing MAC for sensor networks* [14]. Protokol navrhnutý pre spustenie a organizovanie linkovej vrstvy založený na organizačnom routovaní pre využitie predovšetkým v mobilných BSS, kde sa topológia a rozloženie uzlov neustále mení. Je to distribuovaný protokol založený na objavovaní susedných uzlov a výmene vysielacích harmonogramov. Dva uzly, ktoré sa navzájom objavia, sa dohodnú na komunikácii v určitých časových intervaloch na určitej frekvencii z množiny využiteľných frekvencií. Nakoľko je tento výber náhodný, je malá pravdepodobnosť, že dôjde k využitiu rovnakej frekvencie rôznymi uzlami v jednej susednej oblasti. Tento protokol je teda bez kolízií a jeho základy sú v TDMA a CDMA metódach. Tento protokol sa však používa väčšinou len v prípade mobilných sietí, pretože je pomerne komplexný a výpočtovo náročný.

4.1.2 Protokoly s náhodným priradením prístupu ku kanálu

Sú najpružnejšie pre implementáciu, vedia bežať v plne distribuovanej forme a teda nepotrebujú centrálne spracovanie. Uzly sa snažia získať prístup ku kanálu vždy keď ho potrebujú. Môže dochádzať ku kolíziám a teda protokoly potrebujú nástroje na ich detekciu a zotavenie v prípade, že takéto situácie nastanú. Pomerne známymi protokolmi v tejto rodine sú Aloha protokol, ktorý patrí medzi najstaršie, alebo už spomínaný CSMA/CD pre pevné siete. Aloha protokol slúžil na posielanie dát satelitom. Protokol CSMA/CD MAC a protokoly od neho odvodené používajú pri komunikácii malé kontrolné správy. RTS žiadosť o posielanie je vygenerovaná v prípade, že uzol chce komunikovať. Následne mu ako odpoveď príde správa CTS alebo odosielanie povolené, ktoré znamenajú, že dátový prenos môže začať. Kolízii sa zabráni pretože uzly v okolí vysielača alebo prijímača zachytia RTS alebo CTS, čím majú zakázané vysielať po dobu dostatočne dlhú na to, aby nerušili prebiehajúci prenos. Tento spôsob takmer úplne rieši problém skrytého terminálu a kolízie môžu vznikať len v prípade, že dva uzly v tom istom čase vygenerujú RTS. Toto sa dá vyriešiť vložením náhodného intervalu pred opakovaním správy, pokiaľ CTS nebolo prijaté. Na rovnakom princípe, ale s mnohými vylepšeniami sú postavené protokoly MACA [4], ktorý do RTS-CTS komunikácie pridáva aj informáciu o dĺžke nasledujúceho prenosu, MACAW [5], ktorý prichádza s ďalšou kontrolnou potvrdzovacou správou ACK a FAMA [6].

- **PAMAS** – *Power aware Multi-access with Signaling protocol* [8]. Je to protokol odvodený od protokolu MACA s tým rozdielom, že využíva dva oddelené komunikačné kanály. Jeden pre posielanie signalizačných správ a druhý pre posielanie dát. Na základe signálnych sprav uzly vedia kto a ako dlho bude vysielať a teda sa na túto dobu môžu vypnúť. Nevýhodou je nutnosť druhého hardvérového vysielača, čo značne predražuje celé riešenie.
- **S-MAC** – *Sensor MAC protocol* [9]. Znižuje spotrebu energie vypínaním a zapínaním rádia na všetkých uzloch v rovnakom čaše. Pre tieto účely musí byť sieť čiastočne synchronizovaná. Za týmto účelom autori zaviedli nový typ správ, ktorý je broadcastovaný všetkým susedom. Ide o takzvaný SYNC rámec, ktorý je vysielaný periodicky a obsahuje krátkodobý časový harmonogram. Na základe týchto sprav si

vedia okolité uzly prispôsobiť svoj program podľa okamžitých požiadaviek. Počas času, kedy je rádio vypnuté, sú všetky správy pripravené na odoslanie, ukladané do pamäte a po zapnutí rádia a synchronizačnej fáze odoslané príslušným susedom.

- **T-MAC** – *Timeout-MAC protocol* [10]. Tento protokol je odvodený od predchádzajúceho s rozdielom, že sa snaží minimalizovať spotrebu energie predĺžením doby spánku. Umožňuje to adaptívne skrátenie aktívnej fázy čo znamená, že pokiaľ uzol nepredpokladá svoje ďalšie zapojenie do prebiehajúcej komunikácie, vypne sa až do začiatku ďalšej fázy. Toto môže spôsobiť prepočutie správy, ktorá mu bola poslaná neskôr.
- **B-MAC** – *Berkley media access control protocol* [11]. Ide o protokol založený na CSMA s množstvom zaujímavých mechanizmov. Prvým je spôsob efektívneho vyhýbania sa kolíziám, vzorkovaním komunikačného kanálu a určovaním úrovne šumu. Pokiaľ je úroveň šumu nižšia ako určitá hraničná hodnota, uzol predpokladá, že môže vysielať, nakoľko nikto iný kanál nepoužíva. Ďalším mechanizmom je vzorkovanie úvodov správ, na základe ktorých sa dá predísť zbytočnému prijímaniu správ danému uzlu neadresovaných. Ako taký je tento protokol veľmi dobre popísaný a má mnoho nastaviteľných a laditeľných vlastnosti, ako napríklad úroveň vysielacieho výkonu, spotreba energie, dĺžka odozvy a podobne.

4.1.3 Protokoly s priradením prístupu ku kanálu na požiadanie

Protokoly sú viac-menej odvodené od protokolov s pevne priradeným prístupom, riešia však otázku, čo s kanálom, ktorý je nevyužívaný. Centrálna entita tento zdroj ponúka postupne podľa poradia všetkým možným záujemcom. Kedže sa to deje organizovane a podľa stanovených priorít, nemalo by dochádzať ku kolíziám a ako taký je daný proces pomerne dobre predpovedateľný. Nevýhoda je, že potrebujeme opäť centrálne spracovanie priraďovania prístupu. Iný prístup v takomto prípade je rozdelenie časových intervalov na dve fázy, kedy si jednotlivé uzly rezervujú prístup ku kanálu a fázu, v ktorej sa podľa jednotlivých rezervácii prenášajú bloky dát. Samotné rezervácie v prvej fáze môžu využívať protokoly s pevným alebo náhodne priradeným prístupom ku kanálu. Po prijatí všetkých požiadaviek pre daný časový interval centrálna jednotka určí na základe precedentov a priorít, ktoré

časové úseky sú priradené jednotlivým uzlom v druhej fáze. Na tomto princípe je založený aj protokol PRMA [7].

- **TRAMA** – *Trafic-adaptive medium access protocol* [15]. Založený na TDMA umožňujúci dynamické prideľovanie časových intervalov. Protokol sa skladá z troch hlavných komponentov, je distribuovaný a prideľuje vysielacie časové intervaly len uzlom, ktoré majú informáciu, ktorú potrebujú predať ďalej. Komponentmi sú susedský protokol a protokol na výmenu harmonogramov, pomocou ktorých si uzly vymieňajú harmonogramy a informácie o súčasnom časovom intervale, identifikátory uzlov do vzdialenosti dvoch skokov a žiadosti o kanál. Tretím komponentom je algoritmus na adaptívnu voľbu, ktorý vyberá pre konkrétny časový interval komunikujúce uzly ako aj riadi periodické vypínanie a zapínanie uzlov.

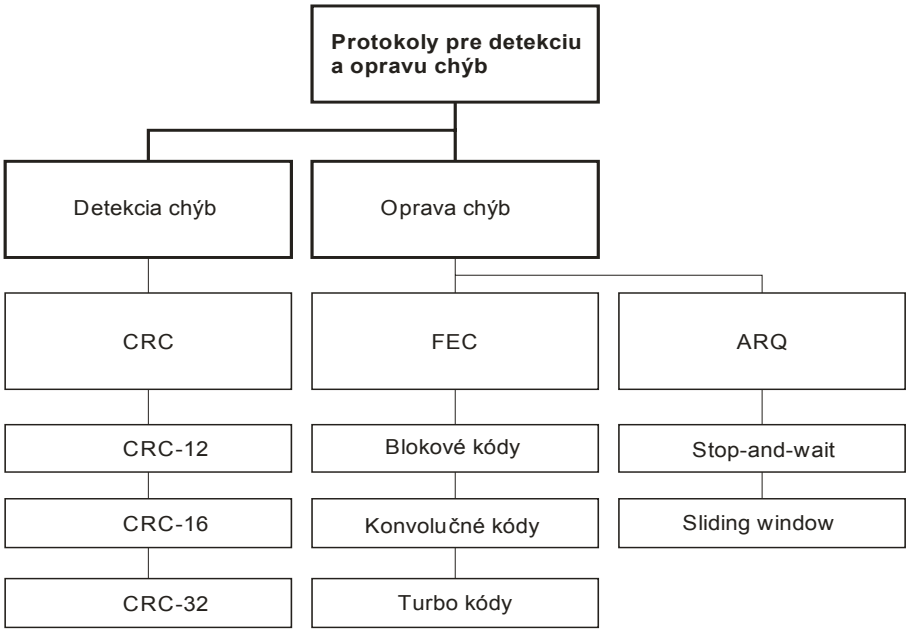
4.2 Podvrstva pre kontrolu logickej linky

Je nad MAC vrstvou a zabezpečuje spoľahlivosť a dobré využívanie komunikačného kanála. Poskytuje servis vrchnejším vrstvám bez ohľadu na aktuálnu kvalitu bezdrôtovej linky. Samotná vrstva sa stará o nasledujúce funkcie.

- Rámcovanie - v BSS je ideálne, aby rámce boli čo najmenšie, takže táto vrstva sa musí starať o fragmentáciu dát a ich zabalenie (enkapsuláciu) do vhodných rámcov a na druhej strane o ich rekonštrukciu.
- Riadenie a kontrola chýb – detekcia a oprava chýb spôsobených pri prenose.
- Riedenie toku dát – predchádzanie a riešenie problémov ako je rýchly vysielač - pomalý prijímač a zahltenie kanálu.
- Spravovanie kanálu – odhadovanie kvality kanálu, objavovanie a rozpoznávanie susedných uzlov a vytvorenie samotných komunikačných liniek medzi jednotlivými uzlami.

Pre oblasť riadenia a kontroly chyb sú vo všeobecnosti používane klasické protokoly známe z ostatných typov sieti s menšími či väčšími úpravami pre konkrétne využitie a na adresovanie konkrétnych problémov. Známe sú napríklad CRC, FEC, ARQ. Pre riadenie toku

napríklad sliding-windov protokol známy z protokolu TCP alebo protokol pre selektívne opakovanie známe skôr pre transportnú a aplikačnú vrstvu. Na obrázku je stručný prehľad týchto protokolov.



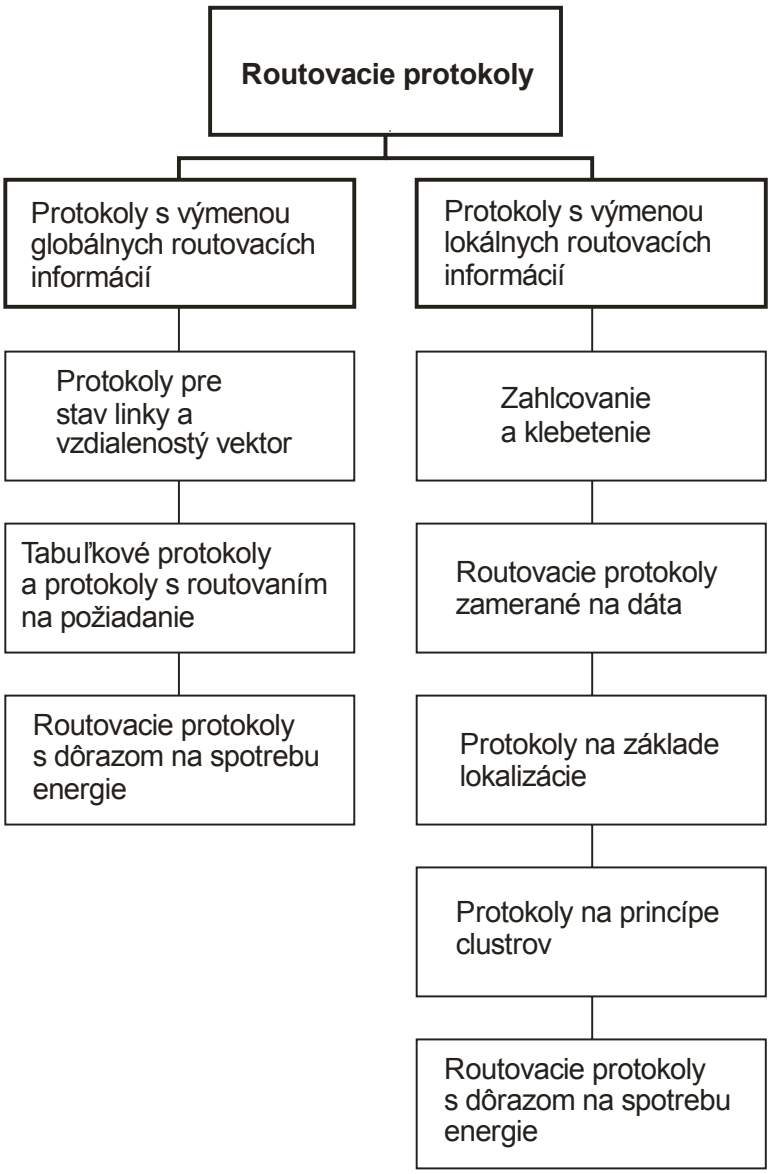
Obrázok 6: Protokoly pre detekciu a opravu chyb

Kapitola 5
Sieťová vrstva a routovacie protokoly

Protokoly tejto vrstvy majú za úlohu prenos informácie zo zdroja do cieľa. V prípade BSS je cieľom zväčša centrálna jednotka, ktorá zbiera informácie z celej siete, vyhodnotí ich a ponúkne správcovi siete. Často ide o predávanie informácie cez veľa skokov v sieti. O vyberanie a spravovanie najvýhodnejších trás pre prenos informácie sa starajú routovacie protokoly. Tie majú v BSS obzvlášť sťaženú funkciu vyplývajúcu z vlastností sietí ako je dôraz na spotrebu energie, časté zmeny topológie siete, prípadne zmena funkcie určitých uzlov v sieti. Väčšina routovacích algoritmov funguje na princípe vyhodnocovania informácií o sieti, ktoré si uzly predávajú. Tieto informácie sa dajú rozdeliť podľa veľkosti častí siete, ktorej sa týkajú, na globálne a lokálne. Informáciami sú napríklad dostupnosť susedných uzlov, možné cesty k centrálnej jednotke, informácie o zahltení časti kanála a takisto aj informácie o množstve energie v susedných uzloch. Routovací protokol si teda na základe týchto informácií buduje routovaciu tabuľku, na základe ktorej sa snaží vybrať najoptimálnejšiu trasu k cieľu. Táto tabuľka musí byť pravidelne updatovaná, aby odrážala každú zmenu v sieti. Čas, ktorý zaberie zmena routovacích tabuliek od zmeny v sieti sa nazýva čas konvergencie. V BSS sa dôraz pri tvorbe routovacích protokolov kladie hlavne na nasledujúce tri vlastnosti:

- Jednoduchosť – routovací algoritmus by mal byť jednoduchý, s čo najmenšou výpočtovou komplexnosťou. Optimalizačné algoritmy by mali byť čo najjednoduchšie s ohľadom na nízku výpočtovú kapacitu uzlov.
- Energetická efektívnosť – vysielanie routovacích informácií by malo byť znížené na minimum, keďže komunikácia je z pohľadu spotreby energie najdrahšia. Rovnako by sa mala vždy vybrať energeticky najoptimálnejšiu trasa.
- Škálovateľnosť – routovací protokol by mal byt škálovateľný čo sa týka počtu uzlov zapájajúcich sa do siete. Niektoré protokoly, ktoré vedia veľmi dobre fungovať pri malých sieťach, môžu pri väčšom nasadení spôsobiť zbytočné preposielanie informácií a tým aj spotrebu energie. Takisto do tejto kategórie patrí aj rozhodovanie o nutnosti lokálnych vs. globálnych routovacích informácií.

Na základe vlastností a samotného prístupu k predávaniu informácií možno routovacie protokoly rozdeliť do niekoľkých tried. Tieto triedy routovacích protokolov ešte ďalej delíme na dve skupiny podľa prístupu k lokálnym (max. uzly do vzdialenosti dvoch skokov) a globálnym (informácie o celej sieti) routovacím informáciám.



Obrázok 7: Triedy routovacích protokolov

5.1 Protokoly s výmenou globálnych routovacích informácií

Protokoly s výmenou globálnych routovacích informácií majú zväčša korene v protokoloch určených pre drôtové siete. Ako také sú v BSS použiteľné zväčša len pre menšie siete.

- **Protokoly pre stav linky a vzdialenostný vektor** (*link state and distance vector protocols*) – zakladajú sa zväčša na Dikstrovom a Bellman-fordovom algoritme. Vyberajú trasu podľa skokovej vzdialenosti a metriky na základe kvality linky. Z pevných sieti sú známe OSPF a RIP. Sú však pomerne zle využiteľné v BSS kvôli ich veľkej výpočtovej zložitosti a objeme dát, ktoré si uzly používajúce tieto algoritmy musia medzi sebou neustále vymieňať. Protokoly pre stav linky zaplavujú sieť lokálnymi informáciami a protokoly vzdialenostného vektoru zase zdieľajú globálne informácie len medzi lokálnymi susedmi.
- **Tabuľkové protokoly a protokoly s routovaním na požiadanie** (*Table driven and on-demand routing protocols*) – sú modifikáciou predchádzajúcej triedy pre bezdrôtové siete, kde sa ohľad berie na mobilitu uzlov a časté straty spojenia medzi jednotlivými uzlami. Ich základným princípom je posielanie pravidelných updatov routovacích tabuliek svojim susedom, čím je zabezpečené, že každý uzol má neustále pomerne presný obraz o stave celej siete. Nevýhodou je posielanie príliš veľkého množstva routovacích informácií. Tento overhead je priamo úmerný počtu uzlov v sieti a intervalu, v ktorom sú updaty posielané. Tabuľkové routovacie protokoly posielajú pravidelne súhrnné updaty svojich routovacích tabuliek. Používane sú napríklad DSDV (*Destinantion sequence distance vector*) a WRP (*Wireless routing protocol*) protokoly. Protokoly fungujúce na požiadanie na druhej strane si svoju routovaciu tabuľku vytvárajú až v prípade, že ju potrebujú. Pokiaľ uzol potrebuje odoslať informáciu nejakému inému uzlu, spustí proces objavovania routovacích trás v celej sieti. Pokiaľ niektorý z uzlov už takú trasu pozná, tak mu ju pošle. Z prijatých trás si uzol vyberie tú najlepšiu a potom ju použije. Ostatne uzly, ktoré túto informáciu taktiež zachytia, si ju ukladajú a upravujú pre ďalšie použitie. Tento prístup značne minimalizuje preposielanie routovacích informácií v sieti. Zo známejších protokolov

sem patria DSR (*Dynamic source routing*), AODV (*Ad-hoc on-demand distance vector*) a TORA (*Temporary ordered routing algorithm*)

- **Routovacie protokoly s dôrazom na spotrebu energie** (*Energy aware routing protocols*) – ich hlavným cieľom je zvýšiť životnosť siete. Používajú na to rôzne metriky pre vyhodnocovanie routovacích tabuliek. Napríklad spotreba energie na paket, rozdiel medzi úrovňou energie v jednotlivých uzloch a podobne. Protokol sa v takomto prípade snaží nájsť napríklad trasu cez uzly s najväčšou zásobou energie. Takýto prístup napríklad šetrí od komunikácie uzly s menšou energetickou zásobou a tak predlžuje už spomínanú životnosť celej siete. Vo väčšine prípadov sa jedná o modifikované protokoly odvodené od Dijkstrovho algoritmu.

5.2 Protokoly s výmenou lokálnych routovacích informácií

Protokoly s výmenou lokálnych routovacích informácií sú vo všeobecnosti vhodnejšie pre BSS. Väčšina týchto protokolov bola priamo navrhnutá pre spomínaný typ sietí.

- **Zahlcovanie a klebetenie** (*flooding and gossiping*) – Táto trieda obsahuje pomerne jednoduché algoritmy. Zaplavovanie je nekomplikovaná procedúra, kedy paket, ktorý má byť poslaný, je poslaný na všetky dostupné odchádzajúce linky. Toto je však veľmi neefektívne z pohľadu spotreby energie nakoľko dáta sa dostanú k uzlom, ktoré ich mať nepotrebujú. Na druhej strane je tento prístup veľmi spoľahlivý. Pre tieto účely sa využíva broadcastovacia funkcia vrstvy MAC. Zlepšenie voči spomínanému prístupu je možné nájsť pri metóde nazvanej klebetenie alebo kontrolované zaplavenie, ktoré sa od pôvodného protokolu líšia iba tým, že výstupné linky vyberajú náhodne alebo podľa nejakého pravdepodobnostného modelu, ktorý sa snaží o to, aby sa správa nedostala do časti siete, v ktorej neleží príjemca danej správy. Ďalším zlepšením je prístup nazývaný šírenie zvestí (*Rumour routing*), ktorý vychádza z predpokladu, že väčšina trás sa bude niekde pravdepodobne krížiť. Podrobnosti tohto protokolu je možné nájsť v [16].

- **Routovacie protokoly zamerané na dáta** (*Data centric routing*) – táto trieda sa vyznačuje zvláštnym prístupom k posielaným dátam. Routovanie a spracovanie paketov nie je na základe identifikátorov uzlov, ale na dátach samotných. Dáta sú v takomto prípade pri spracovaní agregované alebo komprimované, čo znižuje spotrebu energie pre komunikáciu. Za zmienku stoja dva protokoly v spomínanej triede a to SPIN a DD.

  SPIN (*Sensor protocols for information via negotiation*) [17]. Tento protokol používa tri typy správ na základe obsahu informácie, ktorú chce poslať. V prvom kroku je broadcastovaný paket, ktorý hovorí o type dát, ktoré sa snaží uzol poslať. Všetci záujemcovia spomedzi susedov o tieto dáta odpovedia žiadosťou o dáta, ktoré sú im následne poslané. Takto si každý uzol, ktorý ma záujem o daný typ dát, uloží ich kópiu. Proces sa potom celý opakuje až do momentu, kedy všetky uzly so záujmom o tieto dáta ich skutočne dostanú.

  DD (*Direct difusion*) [18]. Je to protokol pozostávajúci z troch fáz. V prvej fáze centrálna jednotka broadcastuje do siete informáciu alebo popis, o aké dáta má záujem. Táto informácia je predávaná medzi uzlami pričom si každý uzol značí ako kritéria dát, tak aj smerník na suseda, od ktorého túto informáciu dostal. Následne pokiaľ niektorý uzol získa informácie, ktoré odpovedajú kritériám, ktoré ma uložené, pošle ich smerom z ktorého mu prišla požiadavka. Táto informácia je potom medzi uzlami prenášaná na základe smerníkov, ktoré majú uložené až sa dostanú k centrálnej jednotke, ktorá celý proces iniciovala. V tretej fáze dochádza k optimalizácii, kedy sa centrálna jednotka snaží posilniť určité trasy priamym smerovaním požiadaviek, aby sa predišlo zbytočnému vytváraniu paralelných trás. Týmto spôsobom majú niektoré uzly niektoré svoje smerníky prioritizované a snažia sa využívať len tie.
- **Routovacie protokoly na základe lokalizácie** (*Location-based routing*) – táto trieda sa vyznačuje neobvyklým prístupom k routovaniu. Namiesto udržiavania si routovacích tabuliek si protokoly udržiavajú informáciu o lokalizácii susedov a zväčša centrálnej jednotky, ku ktorej sú smerované všetky zozbierané dáta. Nevýhodou je nutnosť lokalizačnej techniky ako je napríklad GPS alebo triangulácia. V prípade GPS narastajú nároky na cenu zariadenia, spotrebu energie a výpočet lokalizácie. Takisto predávanie si informácií o lokalizáciách nie je vždy jednoduché. Výhodou je napríklad

to, že nie sú nutné pravidelné updaty routovacích informácií a ich spracovanie. Samotné protokoly v tejto triede fungujú na princípe posielania informácie len tým svojím susedom, ktorí sú smerom k cieľu, pre ktorý je táto informácia určená alebo sa od tohto smeru odchyľujú čo najmenej. Zaujímavé protokoly v tejto triede sú napríklad DREAM a GPRS.

DREAM (*Distance routing effect algorithm for mobility protocol*) [19]. Tento protokol bol špeciálne navrhnutý pre mobilné siete. Každý uzol si drží tabuľku o všetkých prvkoch v sieti. V tejto tabuľke je pozícia vzhľadom na nejaký referenčný systém, čas kedy bol paket s takouto polohou prijatý a rýchlosť, ktorou sa uzol pohybuje. Na základe týchto informácií si vie uzol vypočítať približnú oblasť, do ktorej má byť správa smerovaná, v prípade, že sa snaží poslať informáciu práve tomuto druhému uzlu.

GPRS (*Greedy perimeter stateless routing*) [20]. Autori tohto protokolu sa snažia adresovať problém, kedy by informácia bola poslaná do časti siete, z ktorej sa už na základe smerovania nevie vrátiť alebo keď sa musí dostať cez veľkú prázdnu oblasť bez aktívnych uzlov. Pre tieto prípady protokol disponuje rozhodovacím pravidlom pravej alebo ľavej ruky. Nevýhodou však je, že tento protokol je použiteľný len pre siete reprezentovateľné planárnym grafom.

- **Routovanie protokoly s dôrazom na spotrebu energie** (*Energy-aware routing*) – cieľ je ten istý ako pri predchádzajúcej skupine. Spomenieme opäť dva protokoly z tejto triedy a to EAR a SELAR.

  EAR (*Energy aware routing for low energy ad-hoc sensor networks protocol*) [21]. Základom tohto protokolu je pravdepodobnostná funkcia, na základe ktorej sa vyberá trasa, ktorou bude paket poslaný. Na základe výpočtov tejto funkcie je vybudovaná routovacia tabuľka, ktorá na prvý pohľad nie je optimálna, ale kladie najnižšie energetické nároky na prenos paketu od zdroja k cieľu. Pre podrobnosti čitateľa odporúčame na uvedenú literatúru.

  SELAR (*Scalable energy-efficient location aided routing protocol*) [22]. Jeho základy sú v protokole DREAM s tým rozdielom, že prihliada na aktuálnu úroveň energie v uzloch pri výbere routovacej trasy. SELAR posiela paket na suseda s najväčšou zásobou energie v uhle 15% kde os uhla je smerovaná na cieľ paketu. Pokiaľ v tomto

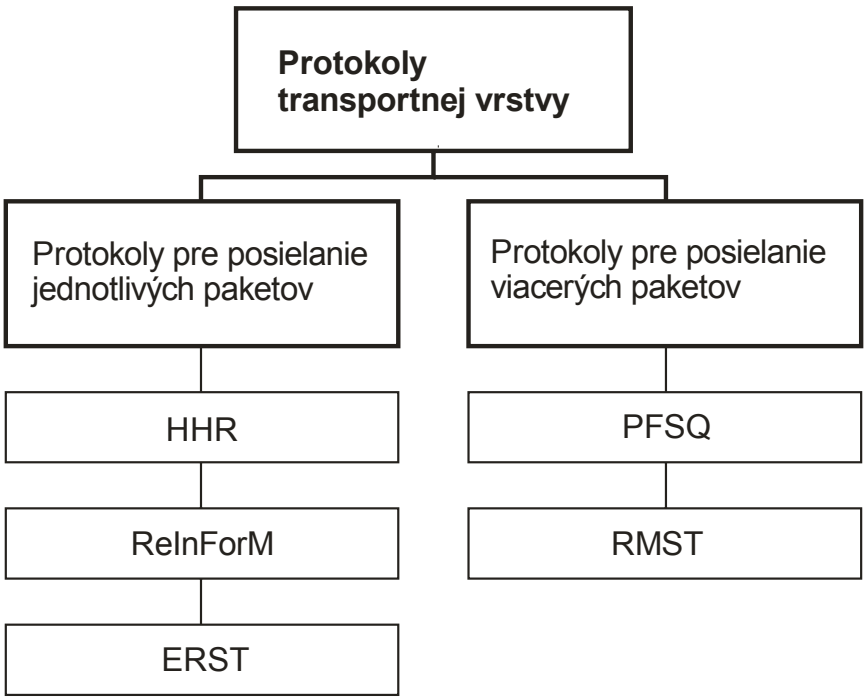
rozpätí nenájde žiadneho vhodného suseda, tento uhol inkrementálne zväčšuje. Nevýhodou tohto protokolu je, že negarantuje doručenie paketu do cieľa. Istou možnosťou by tu bolo zakomponovanie rozhodovacích pravidiel ako napríklad v protokole GPRS.

- **Routovacie protokoly na princípe clustrov** (*Cluster based routing*) - protokoly tejto triedy sa snažia znížiť komplexnosť routovania rozbitím siete na clustre. Dobrým príkladom je už spomínaný protokol LEACH (*Low energy adaptiv clustering hierarchy protocol*) [13]. Jeho nevýhodou je, že každý šéf clustra musí mať priamy dosah na centrálnu jednotku siete. Táto nevýhoda je riešená v protokole HEED (*Hybrid energy-efficient distributed clustering*) [23], kedy agregovaná informácia, ktorú posiela šéf clustra, je odroutovaná cez šéfov iných clustrov do centrály. Inými slovami šéfovia clustrov tvoria určitú podsieť alebo chrbticu siete.

Kapitola 6

Transportná vrstva

Tak ako aj v iných sieťach je táto vrstva zodpovedná za doručenie segmentov od zdroja k cieľu. Jej úlohou je riadiť tok dát ako aj riešiť problémy zahltenia a detekciu a opravu chýb. Svojím spôsobom je veľmi podobná LLC podvrstve dátovej vrstvy s tým rozdielom, že svoje funkcie zabezpečuje nie na úrovni prenosu dát medzi susednými uzlami, ale medzi koncovými uzlami celého prenosu. Väčšina vlastností a funkcií tejto vrstvy je veľmi dobre popísaná v dokumentácii k protokolom TCP a UDP používaných v IP sieťach. Pre účely BSS možno roztriediť protokoly tejto vrstvy podľa dvoch základných kritérií. Podľa miery spoľahlivosti, ktorú zabezpečujú a podľa toho, či potrebujeme transportovať jeden alebo viacero paketov, prípadne prúd paketov.



Obrázok 8: Protokoly transportnej vrstvy

6.1 Protokoly pre posielanie jednotlivých paketov

Vo väčšine BSS, kedy nám ide o doručenie správy o výskyte nejakej udalosti v časti siete do centrálnej jednotky, nepotrebujeme vysokú spoľahlivosť. Je pomerne dosť pravdepodobné, že pokiaľ bola nejaká udalosť zachytená jedným uzlom, bude zachytená aj uzlami v okolí a teda aspoň jedna z vygenerovaných udalostí sa dostane do miesta spracovania. Pre tento prípad by teda mal stačiť jednoduchý transportný protokol, ktorý sa príliš nezaoberá spoľahlivosťou a využíva pritom spoľahlivosť poskytnutú na úrovni dátovej vrstvy. Čiže neriešiť preposielanie a potvrdzovanie prenášanej informácie medzi zdrojom a cieľom, ale len medzi jednotlivými uzlami. Opäť uvádzame niekoľko konkrétnych protokolov.

- **HHR** (*Hop-by-Hop reliability protocol*) [24] je protokol založený na princípe viacnásobného posielania paketov medzi jednotlivými uzlami. Na základe požadovanej spoľahlivosti a odhadu kvality linky, ktorá ma byž použitá na prenos, je vypočítaný počet koľko krát ma byť paket poslaný susedovi, aby bolo zabezpečené jeho prijatie v centrálnej jednotke. Tento protokol nevyžaduje žiadne potvrdzovanie prijatia. V materiáloch k tomuto protokolu sú uvádzané ďalšie tri odvodené protokoly, kedy je využité potvrdzovanie a broadcastovanie a ich kombinácia. Potvrdzovanie slúži na zastavenie opakovaného posielania pokiaľ bol niektorý z paketov úspešne prijatý. V broadcastovej verzii namiesto posielania kópií jedinému susedovi opakovane broadcastuje tieto správy všetkým susedom.
- **ReInForM** *protocol* [25] je od tých istých autorov ako predchádzajúci protokol. Namiesto opakovaného preposielania paketov medzi susedmi sú kópie správy poslané viacerými cestami s pripojenou informáciou o stave siete do týchto paketov. V každom uzle, v ktorom je takýto paket prijatý na základe informácií, ktoré nesie a lokálneho stavu liniek, sú vybraté najvhodnejšie trasy, ktorými paket pokračuje. Nevýhodou tohto protokolu je, že algoritmus pre vyhodnocovanie trasy potrebuje znalosť vzdialenosti príslušného uzla od cieľa, ktorému je paket určený.
- **ERST** (*Event-to-Sink reliable transport protocol*) [26]. Ide o silne centralizovaný protokol, v ktorom je väčšina práce vykonaná centrálnou jednotkou, čo umožňuje

minimalizovať spracovanie v uzloch a tým zabezpečiť ich jednoduchosť. Tento protokol používa potvrdzovanie a preposielanie paketu medzi zdrojom a centrálnou jednotkou.

6.2 Protokoly pre posielanie viacerých paketov

Protokoly pre posielanie viacerých paketov sú v BSS používané veľmi zriedkavo, keďže tieto siete sú málokedy uspôsobené na posielanie prúdových dát ako je hlas alebo video. Existuje však jeden prípad, kedy sú tieto protokoly veľmi užitočné. Ide o preprogramovanie alebo zmenu softvérového vybavenia siete pokiaľ chceme zmeniť použitie siete, prestaviť hraničné hodnoty senzorov, zmeniť používané protokoly alebo upraviť softvérové chyby. V takomto prípade sú z centrálnej jednotky posielané dáta vo väčšom objeme a vyžadujeme vysokú spoľahlivosť pre ich doručenie. V tejto časti uvádzame dva protokoly.

- **PFSQ** (*Pump slowly, fetch quickly protocol*) [27] protokol je založený na preposielaní informácií skok po skoku a negatívnych potvrdzovacích správach NACK. Teoretický predpoklad je, že v bezdrôtových sieťach sú linky chybové a preto pri multiskokovej komunikácii sa chyby kumulujú exponenciálnou mierou, takže tento protokol sa snaží všetky chyby odstrániť kým začne komunikácia na ďalšom skoku. Negatívne potvrdzovacie správy nás zase šetria pred posielaním veľkého množstva potvrdzovacích paketov. PSFQ ma tri fázy. V prvej fáze sú segmenty prenášaných dát posielané v pravidelných, veľmi pomalých intervaloch broadcastom všetkým susedom. Toto vysielanie je inicializované centrálnou jednotkou. Pokiaľ v určitom intervale niektorý uzol nedostane očakávaný segment, generuje NACK pre uzol, od ktorého mal tento segment prijať. Po obdŕžaní NACK tento uzol prepošle požadovanú informáciu a čaká na potvrdenie jej prijatia. Pre tento prípad si práve posielaný segment musí držať v pamäti až pokiaľ nie je úspešné odoslaný do celej ďalšej úrovne susedov, čím končí druhá fáza. V tretej fáze centrálna jednotka inicializuje generovanie reportu. Posiela túto inicializáciu opäť všetkým susedom, ktorí ju propagujú ďalej. Koncové uzly siete, ktoré už túto inicializáciu nemajú kam preposlať,

začínajú generovať report a posielajú ho späť smerom k centrálnej jednotke. Každý uzol, ktorý takýto report preposiela späť k nemu, pripojí svoju vlastnú časť, čím dochádza k agregácii reportov a je málo pravdepodobné že by došlo k zahlteniu správami niekde v blízkom okolí centrálnej jednotky, ktorá tieto reporty nakoniec spracuje.

- **RMTS** (*Reliable multi-segment transport protocol*) [28] vznikol ako protokol ktorý ma zabezpečiť spoľahlivosť pre vyššie spomenutý routovací protokol DD. Existujú dva varianty tohto protokolu. Prvý rieši stratu paketu na báze preposielania NACK medzi susednými uzlami, teda v rámci jednotlivých skokov. V tomto prípade sú segmenty posielaných dát uložené v jednotlivých uzloch. NACK cestuje po trase späť až do uzlu, ktorý má uložený tento stratený segment, ktorý je následne preposlaný. Druhý variant sa líši len tým, že všetky segmenty sú uložené len v centrálnej jednotke a teda pri strate segmentu/paketu pri prenose niekde v sieti je NACK preposlaný až do centrálnej jednotky, ktorá tento segment znova pošle. Takže stratégia je tu postavená skôr na preposielaní medzi koncovými bodmi komunikačnej trasy. Výber medzi týmito dvoma variantmi závisí na odhadovanej miere chybovosti liniek v sieť (BER). Pokiaľ je chybovosť väčšia, prvý variant dáva lepšie výsledky, pokiaľ nižšia a neočakáva sa veľa stratených paketov a tým pádom žiadostí o preposlanie, druhý variant je vhodnejší.

6.3 Protokoly riešiace problém zahltenia v BSS

Pri väčšine protokolov pre posielanie jednotlivých paketov sa s týmto problémom veľmi nepočíta. Je to na základe predpokladu, že jednotlivé linky sú využite minimálne a tým pádom je minimálna aj pravdepodobnosť zahltenia. Pri multipaketových protokoloch sa dá zahltenie očakávať niekde v blízkosti centrálnej jednotky, keďže toto je miesto, v ktorom je dátový prenos najčastejší a prenášaných dát najviac. Vo všeobecnosti nie je veľa transportných protokolov, ktoré by tento problém brali do úvahy, prípadne ho vo svojich algoritmoch riešili. Uvádzame tri bez bližších podrobností. Sú to už spomínaný ESRT [26], ďalej STCP protokol

[29] a CODA (*Congestion detection and avoidance protocol*) [30]. V prípade záujmu čitateľa odkazujeme na uvedenú literatúru.

Kapitola 7
Riadenie topológie

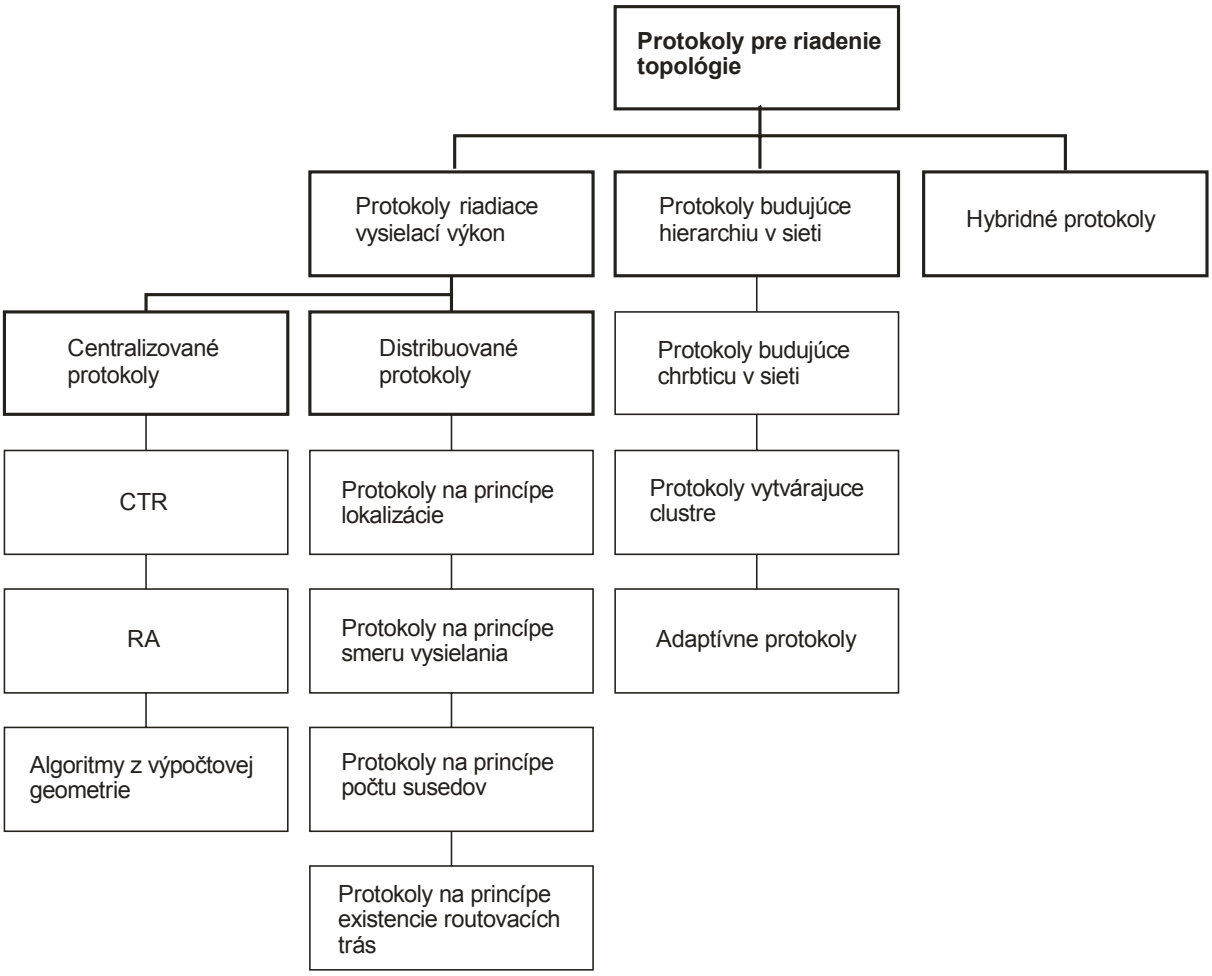
Riadenie topológie (TC) je proces pri ktorom nastavením jednotlivých uzlov meníme topológiu siete, pričom vyžadujeme, aby funkcia siete, jej vlastnosti a oblasť pokrytia zostali nezmenené, pričom sa zvýšila celková životnosť siete.

Jedná sa o protokoly špeciálne vyvinuté pre BSS na predlženie životnosti sieti. Tieto protokoly vykonávajú svoju funkciu na rozhraní dátovej a sieťovej vrstvy. Toto umiestnenie vychádza z nutnosti mať na jednej strane prístup k riadeniu jednotlivých liniek a objavovaniu susedov a na strane druhej spustiť proces na vytváranie a prepočítavanie routovacích tras kedykoľvek dôjde ku zmene. Vysvetlime si to na príklade, ktorý porovnáme s reprezentáciou na grafoch. Vieme, že vysielanie a prijímanie správ je energeticky najnáročnejšia činnosť, ktorú jednotlivé uzly vykonávajú a že energia spotrebovaná na prenos je exponenciálne závislá od vzdialenosti medzi uzlami. Preto z pohľadu spotreby energie je komunikácia na viac skokov s krátkymi vzdialenosťami medzi uzlami výhodnejšia ako vysielanie susedovi na veľkú vzdialenosť. Preto nastavením vhodnej vysielacej úrovne každého prvku sa síce zníži počet susedov, ale aj spotreba energie nutnej na komunikáciu a zároveň sa zníži aj úroveň šumu a interferencie. Stále však vysielací výkon a počet susedov každého uzla musí byť taký, aby celková sieť zostala komunikácie-schopná a nerozpadla sa na veľa izolovaných segmentov. Pri každej takejto zmene, kedy sa zmení množina dostupných susedov, treba spustiť proces, ktorý aktualizuje routovaciu tabuľku, aby nedochádzalo ku snahám komunikovať po trasách, ktoré už neexistujú. Pripadne aby boli zaznamenané a ohodnotené novo vzniknuté trasy. Takže z pohľadu grafov sa snažíme eliminovať dlhé hrany medzi vrcholmi spôsobom, aby graf zostal súvislý. Dobrým príkladom je graf, ktorý vznikne, keď všetky uzly vysielajú na plný výkon a minimálna kostra tohto grafu.

Iným prístupom k riadeniu topológie je rozbitie siete do určitých hierarchických štruktúr ako je napríklad budovanie chrbticovej siete alebo vytváranie clustrov, čo umožňuje lepšie spravovanie časti siete, výmeny funkcií jednotlivých uzlov alebo napríklad aj proces vypínania a budenia prvkov siete čo má tiež veľký dopad na životnosť celej siete.

Pri protokoloch pre riadenie topológie je potrebné si uvedomiť niekoľko faktov a vlastností, ktoré ovplyvňujú celú sieť a ich energetickú efektívnosť, ktorá je prvoradá.

- Distribuovanosť algoritmov. Je pomerne neefektívne a nepraktické používať centralizované protokoly hlavne pri rozsiahlejších sieťach, kde sa globálne informácie zle získavajú.
- Informácie o lokalizácii. Ako aj pri iných vrstvách, predražujú riešenie a sú pomerne náročné na zdroje. Pokiaľ nie sú nutnosťou, treba sa im vyhnúť.
- Lokálne informácie. Protokoly pre riadenie topológie by sa mali vedieť rozhodovať na základe lokálnych informácií.
- Redukovanie topológie. Pri redukovaní topológie musí zostať zachovaná celistvosť siete a takisto aj priestor, ktorý tato sieť pokrýva - či už citlivostne alebo komunikačne.
- Počet susedov. Čím je počet susedov menší, tým je menšia pravdepodobnosť kolízií a problémov so skrytým a nechráneným terminálom.
- Jednosmerné linky. Pri rôznych hodnotách vysielacích výkonov môžu vzniknúť jednosmerné linky. Nie všetky protokoly si s takýmto niečím vedia poradiť.
- Komplexnosť na úrovni správ. Riadenie topológie by nemalo vnášať do komunikácie príliš veľa kontrolných správ.
- Komplexnosť na úrovni algoritmov. Protokoly pre TC by mali byť jednoduché kvôli hardvérovým obmedzeniam sieťových uzlov.
- Rýchlosť. Samotný proces riadenia topológie by mal byť čo najrýchlejší.



Obrázok 9: delenie TC protokolov

7.1 Protokoly riadiace vysielací výkon

Protokoly pre riadenie topológie sa najčastejšie uberajú touto cestou. Medzi najstaršie a najrozpracovanejšie v tejto oblasti patria riešenia dvoch problémov. Oba spadajú pod centralizované riadenie topológie. Sú to problém kritického vysielacieho výkonu (CTR) a od neho odvodený problém priradenia vysielacieho výkonu (RA) [31, 32]. Ďalej spomenieme niekoľko algoritmov pochádzajúcich z výpočtovej geometrie a nakoniec distribuované protokoly využité pre riadenie topológie. V nasledujúcej časti budeme pomerne často zamieňať samotnú fyzickú sieť za graf ktorý ju reprezentuje. Z pohľadu topológie sa v podstate jedná o tú istú vec.

7.1.1 Problém kritického vysielacieho výkonu (CTR)

Ide o princíp, kedy sa snažíme nájsť taký najmenší vysielací výkon, ktorý keď si nastavia všetky uzly v sieti, tak vytvoria súvislú graf. Tento vysielací výkon sa nazýva kritický. Najjednoduchším prístupom k riešeniu tohto problému je nájsť najdlhšiu hranu k najbližším susedom pre všetky uzly a nastaviť vysielací výkon podľa nej. Problém nastáva pokiaľ sieť tvoria dva alebo viac súvislých komponentov, ktoré sú prepojené hranami dlhšími ako tá nájdená. Aby bolo možné predísť takémuto problému, potrebujeme poznať všetky dôležité hrany. Jednou z možností je nájsť pre takýto graf euklidovskú minimálnu kostru (EMST). Ide o strom tvorený podmnožinou najkratších hrán pôvodného grafu, ktorý pokrýva všetky vrcholy. Pokiaľ teda nastavíme vysielací výkon podľa najdlhšej hrany tohto stromu, máme zabezpečenú súvislosť celej siete. Veľkou nevýhodou tohto prístupu je, že pre výpočet EMST je potrebné poznať lokalizácie jednotlivých uzlov alebo aspoň ich aproximácie, ďalej centrálne spracovanie a naviac správová zložitosť algoritmov riešiacich tento problém je príliš veľká. Nakoľko ide o dlho skúmaný a dobre popísaný problém, vzniklo veľa teórií a riešení tohto problému. Častým prístupom je zavrhnutie úplnej súvislosti siete za predpokladu, že výsledný komponent bude pokrývať dostatočne veľkú oblasť. V takomto prípade nám stačí len hrubá aproximácia dĺžky najdlhšej hrany, napríklad vypočítateľná z hustoty siete alebo ako sa z pokusov ukázalo, zabezpečiť len aby všetky vrcholy mali nejaký minimálny stupeň. Vo veľkej väčšine prípadov sa však ukázalo, že výsledný vysielací výkon je veľmi blízky maximálnemu vysielaciemu výkonu.

7.1.2 Problém priradenia vysielacieho výkonu (RA)

Ide o rovnaký problém ako v predchádzajúcej časti s tým rozdielom, že sme upustili od podmienky, že všetky uzly v sieti používajú rovnaký vysielací výkon. Je dokázané, že riešenie tohto problému pre dvoj a troj-dimenzionálne siete spadá do triedy NP-ťažkých. Ďalej, ako už bolo skôr spomenuté, rôzne vysielacie výkony môžu viest k jednosmerným linkám. Pre tento prípad sa často dodáva podmienka, že len obojsmerné linky sú brané do úvahy a používané.

7.1.3 Algoritmy z výpočtovej geometrie

Tieto algoritmy potrebujú pre svoju činnosť poznať vzdialenosť medzi susedmi alebo ich relatívnu pozíciu, na čo je opäť nutná nejaká lokalizačná technika. Uvádzame ich tu len ako jednu z možností riešenia problému riadenia topológie.

- **RNG** (*Relativ neighbor graph*) [33]. Ide o algoritmus, ktorý z trojuholníka tvoreného vrcholom a jeho dvoma susedmi odoberie najdlhšiu hranu. Súvislosť grafu zostava zachovaná.
- **GG** (*Gabriel Graph*) [34]. Tento algoritmus spojí dva vrcholy práve vtedy, keď v kruhu, ktorého priemerom je hrana medzi týmito dvoma vrcholmi, neleží žiadny iný vrchol.
- **DT** (*Delaunay triangulation*) [35]. Je založená na spájaní všetkých susedov na základe Voronoivho diagramu. Tento diagram je geometrická konštrukcia popisujúca oblasť pokrytia a susednosť v grafe.

RNG a GG sú ako vidieť veľmi podobne. Uprednostňujú komunikáciu cez suseda skôr ako cez dlhé linky a oba majú základ v trojuholníkovej nerovnosti.

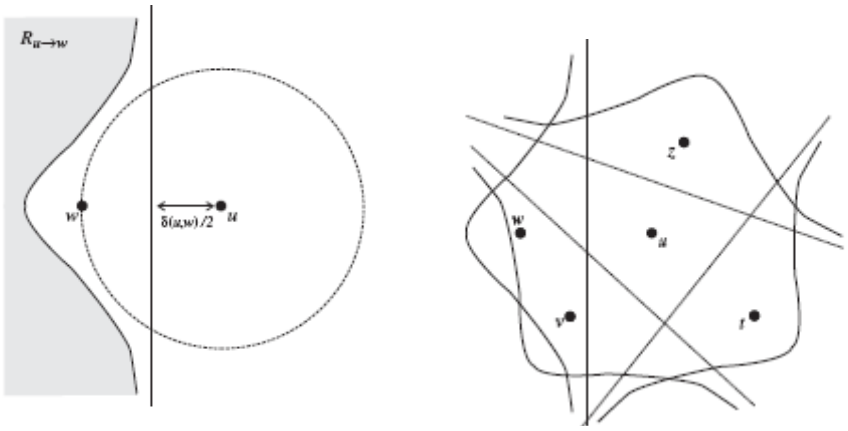
7.2 Distribuované protokoly pre riadenie topológie

V tejto časti budú popísané praktickejšie prístupy a protokoly ako v predchádzajúcich. Sú rozdelené do štyroch skupín. Protokoly fungujúce na princípe lokalizácie, smeru, počtu susedov a routovania.

7.2.1 Protokoly na princípe lokalizácie

- **R&M** (*Rudolph&Meng protocol*) [36]. Tento protokol využíva preposielací región (*relay region*) a uzáver uzla (*enclosure*). Ide o princíp, kedy namiesto posielania správy priamo do cieľa je použitý niektorý sused, ktorý správu prepošle. Toto

rozhodnutie je na základe spotreby energie a uzol si vypočítava oblasť, pre ktorú platí, že preposielanie cez určitého suseda je výhodnejšie ako priame posielanie správy vzdialenejším uzlom. Táto oblasť sa nazýva preposielací región. Pokiaľ si uzol vypočíta preposielacie regióny pre všetkých susedov vznikne oblasť, ktorá sa nazýva uzáver uzla. Sieť takto vzniknutá sa nazýva uzáverový graf. Samotný protokol beží v dvoch fázach. V prvej fáze si uzol postupne zvyšuje vysielací výkon až na maximálny a zapisuje si nájdených susedov - pokiaľ tí už nepatria do preposielacieho regiónu niektorého skôr nájdeného suseda. Takto si nakoniec vytvorí svoj uzáver. Na konci tejto fázy je utvorený uzáverový graf. V druhej fáze je využitý Bellman-Fordov algoritmus na nájdenie najkratšej cesty, ktorý vybuduje reverzný kostrový graf na energeticky najvýhodnejších linkách s koreňom v centrálnej jednotke.



Obrázok 10 : Preposielací región a uzáver uzla

- **LMST** (*Local minimum spanning tree protocol*) [37]. Princípom tohto protokolu je, že každý uzol si na množine svojich susedov vybuduje lokálny minimálny kostrový graf, ktoré spolu tvoria kostrový graf celej siete. Protokol má tri povinné fázy a jednu voliteľnú fázu, ktorá rieši problém jednosmerných liniek. Prvou fázou je výmena informácií. Každý uzol posiela svoju lokalizáciu a ID všetkým svojim susedom na maximálnom vysielacom výkone. V druhej fáze dochádza k budovaniu lokálnych kostrových grafov. Toto prebieha pomocou Primovho algoritmu na dátach, ktoré uzol prijal od všetkých svojich susedov. Váhou pre výber liniek do kostrového grafu je vzdialenosť uzlov vypočítaná z lokalizácii. Po tomto kroku teda každý uzol pozná kostrový graf medzi všetkými jeho susedmi. V ďalšom kroku je vybudovaná globálna

topológia spôsobom, že medzi dvoma susednými uzlami existuje linka práve vtedy, keď táto linka je súčasťou aj lokálnej minimálnej kostry. V treťom kroku je nastavený vysielací výkon tak, aby uzol mohol komunikovať so všetkými svojimi susedmi v globálnej topológii. Toto sa dá zabezpečiť napríklad meraním sily prijímaného signálu od jednotlivých susedov z prvej fázy, kedy bol použitý maximálny výkon a nastavenie si vlastného výkonu na základe výpočtu z propagačného modelu. Štvrtá nepovinná fáza sa zaoberá odstránením jednosmerných liniek nakoľko takéto linky by nemali byť v globálnej topológii. Existujú dva spôsoby. Prvým je zvýšenie vysielacieho výkonu pri detekcii jednosmernej linky tak, aby sa linka stala obojsmernou, čo zvýši počet liniek v lokálnej topológii. Druhým je vyhodenie týchto liniek a zabránenie v ich využívaní pri posielaní správ. Táto fáza by mala nastať ako kontrolná fáza po fáze tretej.

- **FLSS** (*Fault-tolerant local spanning subgraph protocol*) [38] je takmer rovnaký ako LMST protokol. Jediná zmena je v druhej fáze, kedy namiesto budovania lokálnej verzie minimálnej kostry je budovaný kostrový podgraf, ktorý zachováva k-súvislosť. Ostatne fázy sú rovnaké. Ako vidieť výsledný globálny graf je k-súvislý. Premenná *k* je pevne definovane malé číslo zvyčajne 2 alebo 3. Tento protokol minimalizuje vysielací výkon najlepšie spomedzi ostatných a teda sa javí ako energeticky najvýhodnejší.
- **GAF** (*Geograpical adaptive fidelity protocol*) [39]. Tento protokol využíva lokalizačnú techniku na zistenie hustoty siete a existencie násobných tras. Na základe týchto informácií sa rozhoduje o vypínaní alebo zapínaní niektorých uzlov siete. Protokol rozdelí oblasť pokrytia na určité bunky a zabezpečuje, aby sa každý aktívny uzol vedel vždy priamo spojiť s aktívnymi uzlami susedných buniek, takže uzly na základe svojej lokalizácie vedia, do ktorej bunky patria a na základe určitých kritérií, ako je napríklad zostatková energia v uzle sa dohodnú, ktoré zostanú aktívne a ktoré sa vypnú a na ako dlho. Toto sa periodicky opakuje až do konca životnosti siete.

7.2.2 Protokoly na princípe smeru vysielania

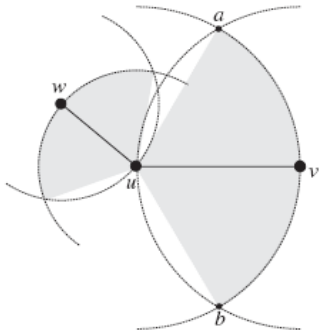
Pri týchto protokoloch sa predpokladá, že uzly vedia aspoň približne určiť smer a vzdialenosť, z ktorého správa prišla. Zistenie smeru sa dá zabezpečiť napríklad použitím smerových antén a vzdialenosti odvodením od sily prijatého signálu alebo oneskorenia pri prenose.

- **YG** (*Yao Graph protocol*) [40]. Ide o protokol s dvoma fázami. V prvej je pôvodný graf redukovaný na podgraf obsahujúci minimálnu kostru. V druhej sú odstránené ostatné prebytočné hrany až zostane čistá minimálna kostra tohto grafu. Redukovaný graf, ktorý vznikne v prvej fáze, sa nazýva Yaov graf. Predpokladom je, že každý uzol vie vysielať určité lúče signálu, ktoré vždy pokrývajú len určitý kruhový výsek či sektor priestoru okolo tohto vrcholu. A že týchto kužeľov, ktoré pokrývajú cele okolie uzla, je aspoň 6. Toto číslo bolo schválne zvolené, aby uhol každého kužeľa bol 60 stupňov, čo zaručí že každý vrchol bude mat maximálne 6 susedov a vzdialenosť dvoch susedov ležiacich v jenom kuželi nebude väčšia ako vzdialenosť toho vzdialenejšieho od vysielača. Uzol takto postupne preskúma susedov vo všetkých svojich sektoroch a vždy si zaznačí toho ktorý je v každom sektore najbližšie. Z každého sektoru je vybraný teda prave jeden sused ktorý sa dalej zapaja do komunikacie s centrálnym uzlom. V druhej fáze sa centralizovane na základe informácií o smere a vzdialenosti susedných uzlov vypočíta minimálna kostra. Tento protokol však nevie zaručiť súvislosť grafu, nakoľko nevie zabezpečiť, že vysielací výkon vybratého uzla z kužeľa je dostatočný na komunikáciu s vysielajúcim centrálnym uzlom. Tento nedostatok je riešený v niektorých jeho vylepšeniach.
- **CBTC** (*Cone-based topology control protocol*) [41]. Rovnako ako pri predchádzajúcom protokole je priestor okolo uzla rozdelený na sektory s rovnakým uhlom. Protokol potom v každom sektore postupne zvyšuje vysielací výkon až narazí na susedný uzol. Takto uzol nadviaže spojenie so susedom z každého sektoru. Opäť sa tento protokol skladá z dvoch fáz. V prvej sa zisťuje potrebný minimálny vysielací výkon na komunikáciu so susedmi vo všetkých sektoroch. V druhej fáze sa vyhodnocuje energetická efektívnosť liniek a tie nevhodné sa z konečnej topológie vypúšťajú. Celý proces funguje nasledovne. Uzol pošle majákovú správu na určitom

výkone, v ktorej je uvedený identifikátor tohto uzla a úroveň jeho vysielacieho výkonu. Túto správu dostanú všetci susedia, ktorí sú dosiahnuteľní na tomto vysielacom výkone. Po prijatí tejto správy odpovedajú potvrdzujúcou správou kde je identifikátor odpovedajúceho uzla, identifikátor pôvodcu majákovej správy a použitý vysielací výkon. Tieto informácie sú použité aj v druhej fáze. Na základe potvrdzujúcich správ si uzol zaznačí susedov dosiahnuteľných daným vysielacím výkonom a približný smer, ktorým ležia. Tu sú používané už spomenuté smerové antény a algoritmy na určenie smeru. Pokiaľ už uzol prijal informácie od všetkých susedov pre daný výkon, spúšťa algoritmus, ktorý vyhodnotí či v každom sektore leží aspoň jeden zo susedov. Pokiaľ nie, celý proces sa opakuje so zvýšeným výkonom až kým podmienky nájdenia susedov v každom smere nie sú splnené alebo je dosiahnutý plný vysielací výkon. Uzly, ktoré nedokázali nájsť susedov v každom sektore, sa označia ako hraničné a svoj výkon znížia tak, aby mali susedov v čo najväčšom počte sektorov. V optimalizovaných verziách tohto protokolu sa napríklad pre komunikáciu s konkrétnymi susedmi používa minimálny vysielací výkon na ich dosiahnutie zistený v tejto fáze. Aj tento protokol produkuje jednosmerné linky, s ktorými sa podľa potreby treba vysporiadať buď ich odstránením alebo zobojsmernením. Druha fáza je optimalizačná. Na základe vysielacích výkonov sa odhadnú vzdialenosti medzi uzlami. Potom sa použijú tie, ktoré sú najkratšie a pritom stále spĺňajú požiadavku, aby každý uzol mal nadviazané spojenie so susedným uzlom v každom sektore.

Existuje viacero variant tohto protokolu, napríklad jeden zabezpečuje k-súvislosť.

- **DistRNG** (*Distributed Relative Neighbor Graph protocol*) [42]. Ide o distribuovanú verziu RNG spomínaného v časti o výpočtovej geometrii. Ideou je odstrániť najdlhšiu hranu trojuholníka tvoreného troma susednými uzlami. Uzol *u* zvyšuje svoj vysielací výkon až nájde ďalšieho suseda *v*. Tohto suseda si pridá do množiny susedov a oblasť pokrytia pridá k oblastiam pokrytia už predtým nájdených susedov. Potom skontroluje či výsledná oblasť pokrytia zahŕňa celú jeho okolnú oblasť teda pre dvojrozmernú topológiu uhol 2π. Za oblasť pokrytia sa berie oblasť prieniku pomyseľných kruhov so stredmi vo vrcholoch *u* a *v* a polomerom vzdialenosti *uv*. Zvyšovanie výkonu a hľadanie susedov v nepokrytých oblastiach pokračuje až do momentu, kedy je pokrytá celá okolitá oblasť vrcholu *u* alebo sa dosiahne plného vysielacieho výkonu.



Obrazok 11 : DistRNG

- **Di-ATC** (*Angular topology control with directional antennas protocol*) [43]. Podobne ako pri DistRing si pri tomto protokole každý uzol drží len určitú podmnožinu svojich susedov. Týchto susedov vyberá tak, aby mal pokrytú celú svoju okolitú oblasť. Druhou podmienkou je, že veľkosť tejto podmnožiny je obmedzená konkrétnou, nie príliš veľkou konštantou *d*. Uzol *u* si pridá nového suseda *w* iba vtedy, ak je splnená aspoň jedna z nasledujúcich troch podmienok.
    1. *u* ešte nemá *d* vybratých susedov
    2. *u* ešte nemá suseda v oblasti 360/(d+1) stupňov s osou prechádzajúcou uzlom *w*
    3. pokiaľ už *u* má v danej oblasti suseda *z,* ale vrchol *w* má nadviazaných viac susedských vzťahov, tak suseda *z* vymení za *w*.

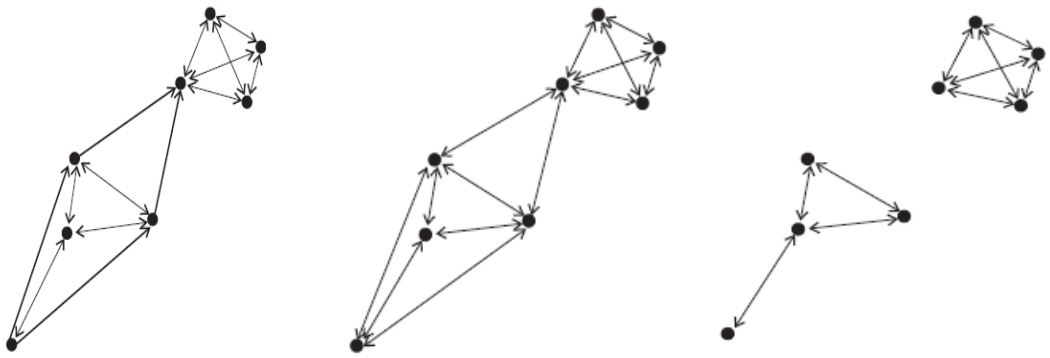
  Susedné uzly pravidelne oznamujú svoju pozíciu, aby si ostatné uzly vedeli vypočítať ich vzdialenosť a smer. Pokiaľ sa stratí komunikácia uzla *u* s niektorým svojim susedom, po nejakom čase je tento nahradený iným, ktorý najlepšie spĺňa spomenuté kritériá a umožní mu pokryť opäť celú svoju oblasť.

7.2.3 Protokoly na princípe počtu susedov

Protokoly postavené na tomto princípe ako jedinú informáciu pre proces riadenia topológie používajú množinu svojich dosiahnuteľných susedov a ich identifikátory, zoradenú podľa odhadovanej vzdialenosti napríklad podľa úrovne vysielacieho výkonu. Z pohľadu hardvérového vybavenia sú teda najjednoduchšie. Tvorcovia týchto protokolov vychádzajú

z predpokladu, že existuje taká premenná *k,* ktorá určuje minimálny počet susedov aký musí mat každý uzol, aby bol celý graf súvislý. Za tým, ako určiť vhodne *k* je pomerne veľa teórie a experimentov, ktoré sa týmto problémom zaoberajú. Pre pochopenie výsledkov, ktoré tu budú neskôr prezentované, sa pokúsime aspoň čiastočne do tejto teórie ponoriť.

K-susedský graf je taký orientovaný graf, v ktorom je každý uzol spojený práve s *k* svojimi najbližšími susedmi orientovanými hranami.



Obrázok 12: K-susedský graf a jeho symetricky nadgraf a podgraf

Ako vidieť, v takomto grafe môže existovať pomerne veľa jednosmerných hrán. Nakoľko sa pri stavbe topológii takýmto hranám snažíme vyhýbať, existujú dva spôsoby riešenia. Pri prvom vytvoríme z K-susedského grafu nadgraf zobojsmernením liniek, inými slovami zvýšením vysielacieho výkonu uzlov. V druhom riešení vytvoríme podgraf odstránením všetkých jednosmerných liniek. Takýto graf však môže byť nesúvislý.

Problém K-susedskej súvislosti rieši otázku aké je minimálne *k* také že K-susedský graf množiny vrcholov bude silne súvislý. V tomto prípade treba uvažovať aj oba varianty grafu s obojsmernými hranami.

Algoritmy, ktoré riešia tento problém priamo na BSS, buď vyžadujú veľa globálnych informácií a centrálne spracovanie alebo veľa správ v distribuovanej forme, kedy po pridaní ďalšej množiny hrán do siete treba overiť či nová sieť je súvislá alebo nie. Preto je vhodnejšie do protokolu definovať presnú hodnotu *k*. Teoretická hodnota pre najhorší prípad je $k=n-1$ kde *n* je počet vrcholov v sieti. Toto nie je v praxi dosiahnuteľné buď z dôvodu rozsiahlosti siete, alebo kvôli nedostatočnému vysielaciemu dosahu. V sieťach, v ktorých to dosiahnuteľné je, toto *k* zodpovedá nastaveniu maximálneho vysielacieho výkonu, čo zase nespadá medzi dobré riešenia pre riadenie topológie. Neskôr v [44] bolo ukázané, že pri

normálnom rozložení uzlov v priestore pre dosiahnutie súvislosti grafu stačí je hodnota k $\Theta(\log n)$. Experimentálne sa zase dokázalo, že pre husté grafy s veľkým počtom vrcholov číslo k konverguje k 6. Pri tejto hodnote najväčší komponent grafu obsahuje takmer všetky vrcholy.

- **KNeight** *protocol* [45]. Jedná sa o distribuovanú implementáciu výpočtu podgrafu K-susedského grafu s obojsmernými linkami na základe odhadnutých vzdialeností uzlov. Na začiatku každý uzol vyšle maximálnym vysielacím výkonom svoj identifikátor. Každý uzol, ktorý tento signál príjme, si ho značí ako suseda spolu s odhadovanou vzdialenosťou (napríklad odvodenou od sily prijatého signálu). Z týchto susedov si potom vyberie k najbližších. Tento zoznam opäť pošle maximálnym vysielacím výkonom všetkým svojim susedom. Na základe výmeny týchto informácií uzly vedia rozoznať jednosmerné linky a nepoužívať ich. Po tomto kroku si každý uzol nastaví svoj vysielací výkon tak, aby vedel komunikovať s najvzdialenejším zo svojich vybratých susedov. Existuje ešte nepovinná optimalizačná fáza, ktorá je podobná ako v CBTC a odstraňuje hranu medzi dvoma vrcholmi v prípade že existuje vrchol, ktorý je susedom oboch vrcholov, teda hrany s týmto vrcholom od oboch patria do výslednej topológie a komunikácia cezeň je energeticky výhodnejšia. Iná verzia tohto protokolu nazývaná KNeightLev používa namiesto odhadu vzdialenosti postupné zvyšovanie vysielacieho výkonu a nájdených susedov si značí do určitých skupín podľa výkonu, ktorý je potrebný na komunikáciu s nimi. Druhá zmena je, že svojich k susedov, s ktorými bude komunikovať, vyberá až po tom ako má overenú obojsmernosť príslušných liniek.
- **XTC** *protocol* [46]. Je to protokol odvodený od KNeight s tým rozdielom, že rieši súvislosť grafu, ktorá v pôvodnom protokole nie je zaručená. Počiatočná fáza je totožná s predchádzajúcim protokolom. Rozdiel nastáva pri usporiadaní množiny objavených susedov. Tento protokol dané usporiadanie robí na základe kvality liniek namiesto odhadovanej vzdialenosti. Toto usporiadanie opäť rozpošle všetkým susedom a príjme ich usporiadania. Vyhodnocovanie, či konkrétna linka bude alebo nebude patriť do výslednej topológie, prebieha na základe porovnávania zoznamov. Povedzme, že uzol u zvažuje linku s vrcholom v. Pokiaľ existuje vrchol w taký, že existujú linky s vyššou kvalitou medzi u a w a zároveň aj medzi w a v táto linka do

výslednej topológie zahrnutá nie je. Pokiaľ aspoň jedna takáto linka neexistuje, je linka medzi *u* a *v* do výslednej topológie zahrnutá. Ďalším rozdielom medzi týmito dvoma protokolmi je, že zakiaľ každý uzol *v* Kneight ma maximálne *k* susedov vo výslednej topológii. XTC takéto obmedzenie nemá a v najhoršom prípade môže vyhodnotiť linky so všetkými susedmi ako vhodné do výslednej topológie.

7.2.4 Protokoly na princípe existencie routovacích trás

Tieto protokoly riešia otázku súvislosti grafu na základe záznamov v routovacej tabuľke. Dá sa predpokladať, že pokiaľ v routovacej tabuľke existuje záznam trasy do každého iného uzla siete, je táto sieť súvislá.

- **COMPOW** (*Common power protocol*) [47]. Tento protokol môže fungovať s ľubovoľným routovacím protokolom. Jeho úlohou je zistiť minimálny vysielací výkon pre všetky uzly taký, aby routovacia tabuľka zostala nezmenená od úplnej. Jeho fungovanie je pomerne jednoduché. V prvom kroku si všetky uzly nastavia maximálny vysielací výkon a spustí sa routovací algoritmus, ktorý vybuduje úplnú routovaciu tabuľku. Táto tabuľka je uložená do pamäti. Následne si všetky uzly nastavia minimálny vysielací výkon a opäť sa spustí routovací protokol, ktorý vypracuje novú routovaciu tabuľku. Tá je porovnaná s tabuľkou získanou v prvom kroku. Postup sa opakuje so zvyšovaním vysielacieho výkonu až do momentu, kedy je výsledná routovacia tabuľka zhodná s pôvodnou tabuľkou.

7.3 Protokoly budujúce hierarchiu v sieti

Pri riadení topológie budovaním komunikačnej hierarchie medzi uzlami nám ide o šetrenie prostriedkov znížením preposielania duplicitných správ, agregáciou dát a posunutím spracovania problémov na určitú podmnožinu uzlov, čím sa dá dosiahnuť odľahčenie spracovania v okolitých, prípadne možnosť ich úplného vypnutia na nejakú dobu. Ďalším efektom takéhoto riešenia je zjednodušenie výslednej topológie. Nevýhodou sa tu stáva väčšia spotreba energie na vybranej podmnožine uzlov, ktorých zodpovednosť bola zvýšená. Preto je

vhodné uvažovať aj mechanizmy, ktoré budú tieto zodpovednosti cyklicky rotovať medzi jednotlivými uzlami, aby dochádzalo k rovnomernej spotrebe energie v uzloch. Tieto protokoly sa dajú rozdeliť do troch tried alebo podskupín na protokoly budujúce chrbticu v sieti, protokoly vytvárajúce clustre a špeciálna trieda adaptívnych protokolov.

7.3.1 Protokoly budujúce chrbticu v sieti

Úlohou týchto protokolov je nájsť takú podmnožinu uzlov, ktoré zabezpečia súvislosť grafu a komunikačné pokrytie celej oblasti, v ktorej je sieť rozprestrená. Takže uzly tejto podmnožiny vedia medzi sebou komunikovať a všetky uzly mimo tejto podmnožiny majú v svojom dosahu aspoň jeden z týchto uzlov. Na riešenie tohto problému sa zväčša používajú algoritmy príbuzné tým z matematiky alebo geometrie, ktoré riešia nájdenie spojenej dominujúcej množiny. Ideálne by bolo nájdenie minimálnej spojenej dominujúcej množiny, ale tento problém tak ako slabší problém nájdenia minimálnej dominujúcej množiny patri medzi NP-ťažké a teda sa musíme uspokojiť s ich lepšími či horšími aproximáciami. Sú známe mnohé prístupy k riešeniu tohto problémom. Medzi mnohými spomeňme napríklad budovanie stromov na grafoch alebo hľadanie minimálnej nezávislej množiny. Mnohé protokoly sú založené hlavne na týchto dvoch prístupoch.

- **A3** *protocol* [48]. Je postavený na postupnom budovaní stromu na uzloch siete. Tento strom je potom považovaný za chrbticu siete. Pre výber optimálnych liniek používa špeciálnu metriku založenú na rovnici

$$M(x,y) = W_E \times \frac{E_x}{E_{\max}} + W_D \times \left(\frac{RSSI_Y}{RSSI^*} \right)$$

  Kde x je uzol oslovený uzlom y. W_E je váha zbytkovej energie, E_x je aktuálna energia v uzle x a $E_{\max}$ je maximálna energia, ktorú tento uzol mohol mať. W_D je váha vzdialenosti komunikujúcich uzlov, $RSSI_Y$ je prijatá úroveň signálu vyslaného y a $RSSI^*$ je minimálny prijatý výkon potrebný na nadviazanie spojenia. Hodnota priradená touto metrikou je v intervale 0 až 1. Samotný protokol je inicializovaný v nejakom konkrétnom uzle, zväčša je to centrálna jednotka, nakoľko je vhodné, aby

ležala v koreni tohto stromu. Sú použité 4 druhy správ, uvítacia správa, správa rozpoznania rodiča, správa rozpoznania potomka a uspávacia správa. Uzol inicializujúci komunikáciu rozosiela uvítaciu správu. Všetci jeho susedia, ktorí tuto správu dostanú a ešte nie sú v strome zaradení, odpovedajú správou rozoznania rodiča, tento uzol si značia za svojho rodiča a na iné uvítacie správy už neodpovedajú. V tejto správe je aj ich vyhodnotenie metriky. Rodič čaká určitú dobu kým sa mu vyzbierajú všetky odpovede. Potom si týchto susedov utriedi podľa metrík, ktoré prijal. Takto utriedený zoznam posiela všetkým svojim potomkom v správe rozpoznania potomka. Tí si nastavia odpočítavanie podľa svojej pozície na danom zozname. Čím má uzol priradenú horšiu metriku, tým si nastaví dlhší čas. Po tomto čase posiela každý uzol uspávaciu správu. Pokiaľ mu ešte čas nevypršal a on dostal uspávaciu správu od svojho suseda, do ďalšieho budovania topológie sa už nezapája. Toto zabezpečí, že uzly s najlepšou metrikou, ktoré sa navzájom nevidia, sa zapoja do budujúceho sa stromu. Uzol, ktorý pošle uspávajúcu správu, sa stáva súčasťou stromu a celý proces opakuje ako otec. Pokiaľ uzol neprijme žiadnu správu rozpoznania rodiča na svoju uvítaciu správu, do budujúceho stromu sa nezapája, nakoľko je okrajovým uzlom, alebo všetky jeho susedne uzly už sú pokryté. Ide o jednoduchý energeticky efektívny protokol s lineárnou správovou zložitosťou (4n), čo nám umožňuje napríklad cyklické spúšťanie tohto protokolu pre optimalizáciu spotreby energie v celej sieti.

- **EECDS** (*Energy efficient connected dominating set protocol*) [49]. Ide o protokol, ktorý má dve fázy a používa farbenie a váhu každého uzla. Váha je v tomto prípade istá metrika získaná z energetickej zásoby uzla a počtu susedov, s ktorými vie komunikovať. V prvej fáze sa nájde minimálna nezávislá množina, v druhej sa táto množina pomocou ďalších uzlov prepojí do chrbtice. Opäť celý proces začína v centrálnej jednotke. Všetky uzly sú na začiatku označené ako biele. Centrálna jednotka začínajúca celý proces sa označí za čiernu a posiela čiernu správu. Všetky uzly, ktoré dostanú čiernu správu sa označia ako šedé a pošlú svojím susedom šedú správu. Biele uzly, ktoré dostanú šedú správu vedia, že by sa mohli stať čiernymi a teda nastáva proces, kedy uzly súťažia o túto farbu. Toto súťaženie prebieha na základe správ, ktoré tieto uzly generujú. Obsahom tejto správy je stav uzlu a jeho váha. Tato výmena trvá len určitý pevne stanovený čas. Pokiaľ v tomto čase uzol

nedostane čiernu správu a po tomto čase vyhodnotí svoju váhu ako najvyššiu možnú spomedzi váh prijatých od susedov, označí sa na čierno. Inak zostáva označený na bielo. Celý proces sa opakuje až sú všetky uzly zafarbené na šedo alebo čierno. Čierne uzly teda predstavujú minimálnu nezávislú množinu. Druha fáza je založená na podobnom princípe ako prvá. Čierny uzol generuje svojim sivým susedom požiadavku. Tí po prijatí spúšťajú opäť proces porovnávania svojich váh. Uzol s najvyššou váhou sa zapája do chrbtice a generuje svojim susedom modrú správu, aby ich o tomto stave informoval. Týmto spôsobom vznikne spojená chrbtica. Dôkaz, že táto chrbtica naozaj vznikne a je spojená, je možne nájsť v uvedenej literatúre. Tento protokol má tiež lineárnu správovú zložitosť *O(n)* neberú sa tu však do úvahy kolízie vzniknuté pri vysielaní sprav počas súťaženia.

- **D2** (*Distance-2 coloring algorithm protocol*) [50]. Tento distribuovaný protokol funguje na princípe farbenia vrcholov a ma tri fázy. V prvej fáze je použitý D2 farbiaci algoritmus na zafarbenie všetkých vrcholov. Tento algoritmus pridelí farby vrcholom tak, aby žiadny uzol nemal suseda rovnakej farby do vzdialenosti dvoch skokov. Týmto je zabezpečené, že dva uzly rovnakej farby vedia vysielať správy paralelne bez kolízií. Počas tejto fázy si uzly posielajú správy, menia svoju farbu a rozhodujú sa, či aktuálna farba bude trvalou alebo nie. Ako ukazujú autori protokolu, po tejto fáze je zabezpečené, že všetky uzly sú zafarbené a žiadny vrchol nemá rovnakú farbu ako niektorý jeho sused do vzdialenosti dvoch skokov, čo bolo podmienkou. V druhej fáze sa buduje Minimálna nezávislá množina vo viacerých krokoch. Všetky farby použité pri farbení sú očíslované. V každom kroku sú do vybranej množiny pridané tie uzly príslušnej, práve spracovanej farby, pre ktoré platí, že žiadny ich sused ešte v tejto množine nie je. Pokiaľ túto podmienku splnia, sú zaradené do množiny vybraných a informáciu o zmene tohto stavu pošlú všetkým svojim susedom. V ďalšom kroku sa spracováva ďalšia farba až kým protokol neprejde všetky použité farby. V tretej fáze sa vrcholy vo vybranej množine spájajú nasledovným spôsobom: pomocou správ si zisťujú prehľad o svojich susedoch do vzdialenosti troch skokov. Na základe týchto informácií potom vyberajú optimálne linky na prepojenie všetkých susedných vrcholov z množiny zostrojenej v druhej fáze. Správová zložitosť tohto algoritmu je $\mathrm{O}\left(n \log^2 n\right)$.

- **SPAN** *protocol* [51]. Ide o protokol, ktorý používa uzly označujúce sa ako koordinátori. Uzol označí sám seba za koordinátora pokiaľ medzi svojimi susedmi nájde dva vrcholy, ktoré vedia spolu komunikovať len cez neho a nie je v okolí žiadny koordinátor, ktorý by túto úlohu mohol zastať. Množina všetkých koordinátorov tvorí chrbticu tejto siete. Ďalej tento protokol obsahuje rutinu, na základe ktorej sa funkcia koordinátora medzi uzlami pravidelne mení pokiaľ je to možné. Koordinátor pravidelne komunikuje so svojimi susedmi a overuje či uzly nevedia komunikovať priamo alebo v okolí nevznikol iný koordinátor, ktorý by tuúo úlohu mohol prevziať. Svojej úlohy sa môže teda vzdať len v prípade, že všetci jeho susedia vedia komunikovať medzi sebou navzájom bez jeho pomoci. Správová zložitosť tohto algoritmu je *O(n)*.
- **CDS-Rule-K** *protocol* [52]. Snahou tohto protokolu je vybudovať spojenú dominujúcu množinu uzlov a potom túto množinu postupne zmenšovať na základe jedného z troch pravidiel. V prvej fáze uzly posielajú uvítacie správy všetkým svojim susedom, aby ich identifikovali. Potom si medzi sebou vymieňajú zoznamy svojich susedov. Keď niektorý uzol zozbiera zoznamy od všetkých svojich susedov, tieto zoznamy spojí do jedného a sprienikuje so svojím. Pokiaľ výsledný počet uzlov, ktoré ležia v tomto prieniku je menší ako počet uzlov, ktoré má vo vlastnom zozname susedov, vyhlási sa tento uzol za člena dominujúcej množiny. V druhej fáze sú niektoré uzly z tejto množiny vyhodené. To je na základe niektorého z už spomenutých troch pravidiel. Pri každom odznačení uzla z dominujúcej množiny musí byť táto skutočnosť ohlásená ostatným, aby si mohli updatnuť svoje informácie.
    1. Označený uzol sa môže odznačiť pokiaľ sú všetci jeho susedia pokrytí iným označeným vrcholom.
    2. Označený uzol sa môže odznačit pokiaľ sú všetci jeho susedia pokrytí dvoma navzájom spojenými označenými uzlami.
    3. Je generalizáciou predchádzajúcich dvoch pravidiel pre *k* označených navzájom spojených uzlov. Teda označený uzol sa môže odznačiť pokiaľ množina jeho susedov je pokrytá množinou *k* navzájom prepojených označených uzlov.

Tento proces odznačovania je pomerne náročný na správy oznamujúce zmenu situácie. Výsledná správová zložitosť tohto protokolu je $O(n^2)$.

7.3.2 Protokoly vytvárajúce clustre

Tieto protokoly riadia topológiu siete rozdeľovaním uzlov do určitých skupín na základe určitých kritérií. Skupiny nazývame clustre. V rámci každého clustra je zvolený uzol, ktorý preberá úlohu šéfa clustra. Medzi jeho povinnosti patrí riadenie topológie vo vnútri clustra, agregácia dát a komunikácia s centrálnou jednotkou alebo inými šéfmi clustrov. Táto úloha je pomerne dosť energeticky náročná a teda protokol by mal mať v sebe zakomponovanú metódu ako vhodne presúvať túto funkciu medzi uzlami v jednotlivých clustroch tak, aby celková spotreba energie bola čo najnižšia a životnosť a funkčnosť siete čo najvyššia. V tejto kategórii uvedieme dva protokoly, ktorými sme sa zoberali už na úrovni sieťovej a dátovej vrstvy. Sú to LEACH a HEED.

- **LEACH** (*Low energy adaptive clustering hierarchy protocol*) [13] riadi topológiu v pravidelných intervaloch. Každý interval má dve fázy. V prvej sú zvolení šéfovia clustrov spomedzi všetkých uzlov. Tým je zabezpečené, že v každom intervale je táto funkcia presúvaná a teda spotreba energie jednotlivých uzlov je rovnomerne rozdelená. Voľba šéfa je založená na náhode a výpočte na základe zbytkovej energie uzla, čísla aktuálneho časového intervalu a počtu šéfov clustrov, ktorých chceme mať v sieti. Takže uzol, ktorý už raz šéfom bol, sa ním stať môže až potom ako boli šéfmi aj všetci ostatní. V momente, kedy voľba šéfov úspešne skončila, začnú šéfovia vysielať oznamovací signál. Toto je druha fáza. Ostatné uzly sa pripoja do clustra so šéfom, od ktorého prijmú najsilnejší oznamovací signál. Celý tento proces sa opakuje v každom časovom intervale. Už spomínanou nevýhodou tohto protokolu je, že šéfovia clustrov vedia komunikovať len priamo s centrálnou jednotkou. Výhodou je, že dáta, ktoré posielajú šéfovia clustrov do centrály, sú agregované. Druhou výhodou je, že v rámci riadenia topológie šéf clustra prideľuje uzlom clustra časový harmonogram, na základe ktorého môžu so šéfom komunikovať a teda v čase mimo tieto pridelene rámce sa môžu prepnúť do úsporného režimu.

- **HEED** (*Hybrid energy-efficient distributed clustering protocol*) [23] je takmer totožný s protokolom LEACH až na to, že namiesto priamej komunikácie s centrálnou jednotkou môže byť použitý ľubovoľný routovací algoritmus sieťovej vrstvy bežiaci na množine šéfov clustrov a tvoriaci tak chrbticu pre prenos agregovaných dát až k centrále. Tým sú zrušene obmedzenia na veľkosť siete alebo jej priestorové rozprestrenie a energetické náklady na komunikáciu chrbticových uzlov s centrálou sa značne znižujú.

7.3.3 Adaptívne protokoly

Táto trieda protokolov má tú vlastnosť, že o vybudovanú topológiu sa stará spôsobom, že ju neustále vylaďuje podľa podmienok a udalostí, ktoré sa menia.

- **ASCENT** (*Adaptiv self-configuring sensor networks topologies protocol*) [53] je protokol založený na meraní kvality liniek. Pokiaľ nejaký uzol zistí vysokú chybovosť, osloví okolité uzly, aby sa zapojili do topológie a pomohli s preposielaním dát. Uzly pracujúce s týmto protokolom môžu byť v štyroch stavoch. Sú to aktívny stav, pasívny stav, testovací stav a spánok. Na začiatku je len niekoľko uzlov v aktívnom stave, ostatné sú v stave pasívnom. V prípade, že aktívne uzly zistia vysokú chybovosť pri komunikácii s niektorými uzlami alebo stratia s nimi spojenie úplne, posielajú svojim susedom žiadosť o zapojenie sa do topológie. Uzol, ktorý takúto žiadosť dostane, sa môže rozhodnúť, že sa pokúsi do topológie zapojiť. Prepne sa do testovacej fázy, v ktorej sa snaží zistiť, či jeho zapojenie do topológie bude mať kladný výsledný efekt alebo nie. Pokiaľ jeho zapojenie zlepší komunikáciu alebo zníži stratovosť v sieti, prepne sa do aktívneho stavu. Pokiaľ nie, vracia sa do stavu pasívneho. Z pasívneho stavu uzol môže na základe svojho rozhodnutia prejsť do spiaceho stavu na nejakú dobu. Počas tejto doby vypína svoje rádio a nie je teda schopný zachytiť žiadne vyzvanie o zapojenie sa do topológie. Po uplynutí doby spánku sa uzol vracia do pasívneho stavu, kedy čaká či bude vyzvaný, aby sa zapojil do topológie alebo sa rozhodne zase zaspať. Aktívne uzly zostávajú aktívnymi až do konca doby ich životnosti.

- **MPCP** (*Minimum power configuration protocol protocol*) [54]. Ide o protokol využívajúci distribuovanú formu Bellman-Fordovho algoritmu pre nachádzanie najkratších ciest. Ako metrika je tu braná do úvahy skoková vzdialenosť komunikujúcich uzlov, ďalej operačný stav uzlov a kvalita ich liniek. Obdobným protokolom je MASP (*Minimum active subnet protocol*), ktorý v metrike neberie do úvahy kvalitu liniek, čo zmenšuje nároky na routovaciu tabuľku a jej spracovanie. Oba protokoly adaptívne menia topológiu a routovacie trasy zväčša len smerom k centrálnej jednotke podľa aktivizácie uzlov v sieti. Predpokladá sa tu, že uzly prechádzajú medzi aktívnym a spiacim stavom.

7.4 Hybridné protokoly pre riadenie topológie

Snahou protokolov tejto triedy je optimálne skĺbiť predchádzajúce dva prístupy, a teda vybudovať komunikačnú hierarchiu a zároveň optimalizovať vysielací výkon uzlov.

- **CLUSTERPOW** *protocol* [55]. Tento protokol je odvodený od protokolu COMPOW, ktorý sme už spomenuli. Protokol je postavený na tom, že každý uzol má niekoľko nastaviteľných hladín vysielacieho výkonu. Toto záleží predovšetkým od hardvérového vybavenia uzla. Každý uzol si buduje pre každú úroveň vysielacieho výkonu oddelenú routovaciu tabuľku uzlov, s ktorými sa vie pri použití tohto vysielacieho výkonu v sieti spojiť. Takže uzly s nastaveným určitým výkonom tvoria v rámci siete izolované clustre. Zvyšovaním výkonu sa tieto clustre rozširujú a spájajú. Výsledná routovacia tabuľka je zostrojená z routovacích tabuliek pre všetky úrovne spôsobom, ktorým k určitému uzlu je vybraná trasa z routovacej tabuľky s najnižším možným vysielacím výkonom, ktorá ho obsahuje. Samotné posielanie správy prebieha nasledovne. Uzol *u* chce uzlu *v* poslať správu. Vo svojej routovacej tabuľke nájde príslušnú trasu s uvedenou úrovňou vysielacieho výkonu. Túto trasu použije. Ako si túto správu uzly medzi sebou predávajú podľa normálneho správania routovacieho protokolu, vyberajú trasu zo svojej routovacej tabuľky. V určitom momente správa prepadne do trasy, ktorú má niektorý uzol označenú s nižšou vysielacou úrovňou.

Tento uzol si zníži vysielaciu úroveň na príslušnú a pokračuje. Týmto spôsobom je správa nakoniec uzlu *v* predaná na niektorej zo spodných vysielacích hladín.

- **TMPO** (*Topology management by priority ordering protocol*) [56]. Ide o protokol budujúci dominujúcu množinu na základe clustrov. V prvom kroku sú zvolení šéfovia clustrov na základe ich priority. Uzol sa stáva šéfom clustra na základe najvyššej priority spomedzi svojich susedov vzdialených maximálne jeden skok alebo priorít susedov suseda jeden skok vzdialeného. Táto priorita sa skladá z identifikátorov všetkých susedov a vôle šéfovať, čo je funkcia vypočítaná na základe zbytkovej energie a pohyblivosti uzla. Na základe týchto priorít každý uzol vie určiť, ktorý zo susedov je šéfom clustra. Táto funkcia je opäť postupne rotovaná medzi uzlami v sieti. Autori protokolu dokázali, že množina šéfov clustrov je dominujúca množina a žiadni dvaja susední šéfovia od seba nie sú vzdialení viac ako tri skoky. V nasledujúcom kroku dochádza k pribratiu ďalších uzlov do tejto dominujúcej množiny, aby táto bola spojená. Najskôr protokol spomedzi susedov vyberá také uzly, aby s ich pomocou boli šéfovia clustrov vzdialení maximálne jeden skok. Tieto uzly sú vyberané na základe najvyššej priority a najkratšej cesty medzi dvoma šéfmi clustrov. Tieto uzly sú tiež zahrnuté do dominujúcej množiny a označujú sa ako dvere. Následne sú vybrané ďalšie uzly, takzvané brány, ktoré spoja buď šéfov clustrov navzájom alebo šéfov clustrov s dverami iných šéfov. Tu je využité, že maximálna vzdialenosť medzi susednými šéfmi je tri skoky. Tieto brány sú tiež pridané do dominujúcej množiny, ktorá je teraz spojená a tvorí chrbticu siete.
- **TECA** (*Topology and energy control algorithm protocol*) [57] je protokol podobný ako TMPO. Taktiež začína výberom nezávislej množiny na základe voľby šéfov clustrov. V tomto prípade je však ako priorita braná len zostatková energia uzlov. Keď už sú šéfovia clustrov známi, ostatné uzly počúvajú kontrolné pakety, ktoré títo šéfovia posielajú, aby vedeli odhadnúť, či dokážu spojiť dvoch alebo viac šéfov. Pokiaľ áno, protokol vypočíta lokálny minimálny kostrový graf pre prepojenie šéfov, na základe kvality liniek, zbytkovej energie uzlov a stratovosti. Týmto spôsobom sú vybrané najvhodnejšie uzly na prepojenie šéfov clustrov a spojenie celej topológie.

Kapitola 8
Prípadová štúdia

Táto kapitola je skôr akousi úvahou o konkrétnom type BSS a jej prípadnom dopade na spoločnosť. Ukážeme si, ako by vhodne zvolený nízko nákladový hardvér, vhodné programovanie a široké aplikovanie mohli zmeniť určité aspekty fungovania súčasnej spoločnosti.

Ako vysielací modul si vyberieme XTend 900 s výkonom 1W a „všesmerovou“ dipólovou anténou. K nemu pridáme GPS modul, napríklad SUP500F 10Hz GPS prijímač a nejaký ARM mikrokontróler, napríklad LPC2138 a niekoľko drobností, ktoré nám tieto veci umožnia spolu prepojiť. Dodáme ešte nejaký 5V LiPol akumulátor s vysokou životnosťou a veľkým počtom nabíjacích cyklov. Tieto veci pri veľkoobjemovom nákupe predstavujú hodnotu cca 200Euro. Dostali sme však uzol siete, ktorý po vhodnom naprogramovaní dokáže posielať svoju lokalizáciu do centrálnej jednotky alebo určitú dobu uskladňovať svoje alebo prijaté lokalizácie. Ako protokoly vhodné pre takýto typ siete by mohli byť napríklad protokoly odvodené od protokolu SMACS a GPRS. Teraz si predstavme že by takýmto uzlom bolo vybavené každé vozidlo. Samozrejme, jedná sa o vec, ktorá by musela byť v takomto prípade podporená lokálnou alebo globálnou legislatívou. Napojením na elektrický rozvod vozidla vieme zaručiť pravidelné dobíjanie batérie a teda bezproblémové fungovanie počas pomerne dlhej doby. Ďalej dostatočným vysielacím výkonom vieme zabrániť výpadku jednotlivých uzlov zo siete. Takisto nás to zbaví potreby protokolov pre šetrenie energie alebo riadenie topológie. V prípade, že uzol zo siete vypadne, lokalizačné informácie sú uskladnené až do momentu opätovného zapojenia uzla do siete. V prípade, že sa GPS prijímač ocitne v tienenom prostredí ako je napríklad garáž, pre konkrétny časový interval bude použitá posledná známa lokalizácia. Vytvorili sme teda sieť, ktorá v krátkych časových intervaloch posiela lokalizácie všetkých vozidiel do centrály. Existuje veľa prípadov, kedy by sa takéto informácie dali využiť. Prvými možnosťami sú informácie napríklad pri haváriach a dopravných nehodách, kedy by bolo zbytočné z miesta nehody ujsť nakoľko táto informácia by bola už dávno uložená v databanke. V ďalšom prípade by takéto informácie mohli slúžiť na projektovanie nových a zmenu súčasných ciest a križovatiek na základe štatistických

údajov o pohybe vozidiel, ich počte, dobách v zápche alebo kolóne a ich priemernej rýchlosti. Naviac spoplatnenie využitia ciest alebo elektronické mýto, kedy by na základe informácií o použití konkrétnych ciest bola určená presná a adekvátna suma. Pokiaľ by bol tento systém prepojený na rádio v aute, dá sa takáto sieť využiť na prenos informácii o zápchach alebo nehodách, ktoré bránia v jednoduchom prechode na trase, po ktorej sa vozidlo pohybuje. Ako je na tomto prípade vidieť, pomerne jednoduchý model siete poskytuje informácie využiteľné vo viacerých odvetviach. Možnosti využitia BSS, aj keď sú zatiaľ len v počiatkoch, sú veľké. V súčasnej dobe sa na trh dostávajú zariadenia so stále s neustále sa zvyšujúcimi možnosťami a spoľahlivosťou a s čoraz nižšou spotrebou a cenou. Rovnako počet odvetví, v ktorých sú takéto siete využiteľné, sa vďaka rozpätiu použiteľných a dostupných senzorov neustále zvyšuje. Preto si myslíme, že budúcnosť bezdrôtových sieti je veľká a hranice ich možnosti a využitia ešte veľmi ďaleko.

Kapitola 9
Záver

V našej práci sa nám podľa našich cieľov podarilo vytvoriť dizajnérskú príručku pre tvorbu bezdrôtových senzorových sieti. V jednotlivých kapitolách sme si predstavili najzaujímavejšie a najznámejšie protokoly pre jednotlivé sieťové vrstvy. Rovnako sme kládli dôraz na klady a zápory jednotlivých riešení ako aj vlastnosti, ktoré by jednotlivé riešenia mali brať do úvahy alebo sa ich vyvarovať. V prvej kapitole sa nachádza stručný prehľad rôznych rádiových komponentov využiteľných pre tento typ sietí spolu so štandardmi, na ktorých sú postavené a ich základnými vlastnosťami. V nasledujúcich kapitolách sú spomenuté jednotlivé protokoly porovnávané medzi sebou a rozdelené do tried na základe ich spoločných vlastností alebo predpokladov. Jedná sa o viac či menej známe protokoly používané v súčasnosti aj s ich prípadnými vylepšeniami či odporúčaniami, ktorým smerom je najvhodnejšie ďalšie úpravy smerovať.

Veríme, že naša práca pomôže pri tvorbe nových riešení v danom obore, alebo aspoň uľahčí prácu pri oboznamovaní sa v tejto rozsiahlej tematike pred hlbším ponorením sa do konkrétnych problémov.

Ďalším možným rozšírení našej práce by bolo vytvorenie podrobného katalógu protokolov, postupov a nástrojov pre tvorbu BSS a jeho pravidelné úpravy a pridávanie najnovších poznatkov v danej tematike.

Použitá literatúra

[1] Rappaport, T.S.:Wireless Communications: Principles and Practice. Prentice Hall, New York (2002)

[2] Tranter, W.H., Shanmugan, K.S., Rappaport, T.S., Kosbar, K.L.: Principles of Communication Systems Simulation with Wireless Applications. Prentice Hall, New York (2004)

[3] 147. Zou, Y., Chakrabarty, K.: Sensor deployment and target utilization based on virtual forces. In: Proceedings of IEEE INFOCOM (2003)

[4] Karn, P.: A new channel access method for packet radio. In: Proceedings of the ARRL/CEEL Amateur Radio 9th Computer Networking Conference, pp. 134–140 (1990)

[5] Bharghavan, V., Demers, A., Shenker, S., Zhang, L.: MACAW: a media access protocol for wireless LAN's. In: Proceedings of ACM SIGCOMM, pp. 212–225 (1994)

[6] Fullmer, C.L., Garcia-Luna-Aceves, J.J.: Floor acquisition multiple access (FAMA) for packet-radio networks. In: Proceedings of ACM SIGCOMM, pp. 262–273 (1995)

[7] Goodman, D.J., Valenzuela, R.A., Gayliard, K.T., Ramamurthy, B.: Packet reservation multiple access for local wireless communications. IEEE Trans. Commun. 37(8), 885–890 (1989)

[8] Singh, S., Raghavendra, C.S.: PAMAS – power aware multi-access protocol with signaling for ad hoc networks. ACM Comput. Commun. Rev. 28(3), 5–26 (1998)

[9] Ye, W., Heidemann, J., Estrin, D.: An energy-efficient MAC protocol for wireless sensor networks. In: Proceedings of INFOCOM, pp. 1567–1676 (2002)

[10] van Dam, T., Langendoen, K.: An adaptive energy-efficient MAC protocol for wireless sensor networks. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, pp. 171–180 (2003)

[11] Polastre, J., Hill, J., Culler, D.: Versatile low power media access for wireless sensor networks. In: Proceedings of Second ACM Conference on Embedded Networked Sensor Systems (2004)

[12] Clementi, A., Penna, P., Silvestri, R.: Hardness results for the power range assignment problem in packet radio networks. In: Proceedings of 3rd International Workshop on Approximation Algorithms for Combinatorial Optimization Problems: Randomization, Approximation, and Combinatorial Algorithms and Techniques, pp. 197–208 (1999)

[13] Heinzelman, W., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: Proceedings of the 33rd International Conference on System Sciences (HICSS), pp. 1–10 (2000)

[14] Sohrabi, K., Gao, J., Ailawadhi, V., Pottie, G.J.: Protocols for self-organization of a wireless sensor network. IEEE Pers. Commun. 7(5), 16–27 (2000)

[15] Rajendran, V., Obraczka, K., Garcia-Luna-Aceves, J.J.: Energy-efficient, collision-free medium access control for wireless sensor networks. Wirel. Netw. 12(1), 63–78 (2006)

[16] Braginsky, D., Estrin, D.: Rumor routing algorithm for sensor networks. In: Proceedings of First ACM International Workshop on Wireless Sensor Networks and Applications, pp. 22–31 (2002)

[17] Kulik, J., Rabiner, W., Balakrishnan, H.: Adaptive protocols for information dissemination in wireless sensor networks. In: Proceedings of Fifth ACM/IEEE Mobicom Conference, pp. 174–185 (1999)

[18] Intanagonwiwat, C., Govindan, R., Estrin, D.: Directed diffusion: a scalable and robust communication paradigm for sensor networks. In: Proceedings of Sixth Annual International Conference on Mobile Computing and Networking, pp. 56–67 (2000)

[19] Basagni, S., Chlamtac, I., Syrotiuk, V.R., Woodward, B.A.: A distance routing effect algorithm for mobility (DREAM). In: Proceedings of Fourth ACM/IEEE Mobicom Conference, pp. 76–84 (1998)

[20] Karp, B., Kung, H.T.: Greedy perimeter stateless routing for wireless networks. In: Proceedings of ACM/IEEE Mobicom, pp. 243–254 (2000)

[21] Shah, R., Rabaey, J.: Energy aware routing for low energy ad hoc sensor networks. In: Proceedings of IEEE Wireless Communications and Networking Conference (WCNC) (2002)

[22] Lukachan, G., Labrador, M.A.: SELAR: scalable energy-efficient location aided routing protocol for wireless sensor networks. In: Proceedings of IEEE LCN Workshop on Wireless Local Networks, pp. 694–695 (2004)

[23] Younis, O., Fahmy, S.: HEED: a hybryd, energy-efficient, distributed clustering approach for ad hoc sensor networks. IEEE Trans. Mobile Comput. 3(4), 366–379 (2004)

[24] Deb, B., Bhatnagar, S., Nath, B.: Information assurance in sensor networks. In: Proceedings of the 2nd ACM Workshop on Wireless Sensor Networks (2003)

[25] Deb, B., Bhatnagar, S., Nath, B.: ReInForM: reliable information forwarding using multiple paths in sensor networks. In: Proceedings of IEEE LCN, pp. 406–414 (2003)

[26] Sankarasubramaniam, Y., Akan, O., Akyildiz, I.F.: ESRT: event-to-sink reliable transport in wireless sensor networks. In: Proceedings of ACM MobiHoc, pp. 177–188 (2003)

[27] Wan, C.Y., Campbell, A., Krishnamurthy, L.: PSFQ: a reliable transport protocol for wireless sensor networks. In: Proceedings of 1st ACM International Workshop on Wireless Sensor Networks and Applications, pp. 1–11 (2002)

[28] Stann, F., Heidemann, J.: RMST: reliable data transport in sensor networks. In: Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, pp. 102–112 (2003)

[29] Iyer, Y.G., Gandham, S., Venkatesan, S.: STCP: a generic transport layer protocol for wireless sensor networks. In: Proceedings of IEEE ICCCN Conference (2005)

[30] Wan, C.Y., Eisenman, S.B., Campbell, A.T.: CODA: congestion detection and avoidance in sensor networks. In: Proceedings of the ACM Conference on Embedded Networked Sensor Systems (2003)

[31] Santi, P.: Topology Control in Wireless Ad Hoc and Sensor Networks. Wiley, New York (2005)

[32] Labrador, M.A., Wightman, P.M.: Topology Control in Wireless Sensor Nettworks. Springer, Berlin (2009)

[33] Toussaint, G.: The relative neighborhood graph of a finite planar set. Pattern Recognit. 12, 261–268 (1980)

[34] Gabriel, K.R., Sokal, R.R.: A new statistical approach to geographic variation analysis. Syst. Zool. 18, 259–270 (1969)

[35] Li, X.Y., Calinescu, G., Wan, P.J.: Distributed construction of a planar spanner and routing for ad hoc wireless networks. In: Proceedings of IEEE INFOCOM, pp. 1268–1277 (2002)

[36] Rodoplu, V., Meng, T.H.: Minimum energy mobile wireless networks. IEEE J. Sel. Areas Commun. 17(8), 1333–1344 (1999)

[37] Li, N., Hou, J.C., Sha, L.: Design and analysis of an MST-based topology control algorithm. In: Proceedings of IEEE INFOCOM, pp. 1702–1712 (2003)

[38] Li, N., Hou, J.C.: FLSS: a fault-tolerant topology control algorithm for wireless networks.In: Proceedings of ACM Mobicom, pp. 275–286 (2004)

[39] Xu, Y., Heidemann, J., Estrin, D.: Geography-informed energy conservation for ad hoc routing. In: Proceedings of the 7th Annual International Conference on Mobile Computing and Networking, pp. 70–84 (2001)

[40] Yao, A.C.: On constructing minimum spanning trees in K-dimensional spaces and related problems. J. Comput. Soc. Ind. Appl. Math. 11(4), 721–736 (1982)

[41] Li, L., Halpern, J.Y., Bahl, P., Wang, Y.M., Wattenhofer, R.: A cone-based distributed topology-control algorithm for wireless multi-hop networks. IEEE/ACM Trans. Netw. 13(1), 147–159 (2005)

[42] Borbash, S.A., Jennings, E.H.: Distributed topology control algorithm for multihop wireless networks. In: Proceedings of the IEEE International Joint Conference on Neural Networks, pp. 355–360 (2002)

[43] Gelal, E., Jakllari, G., Young, N., Krishnamurthy, S.V.: An integrated scheme for fullydirectional neighbor discovery and topology management in mobile ad hoc networks. In : Proceedings of IEEE International Conference on Mobile Ad Hoc and Sensor Systems, pp. 139–149 (2006)

[44] Xue, F., Kumar, P.R.: The number of neighbors needed for connectivity of wireless networks. Wirel. Netw. 10(2), 169–181 (2004)

[45] Blough, D.M., Leoncini, M., Resta, G., Santi, P.: The K-Neigh protocol for symmetric topologycontrol in ad hoc networks. In: Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp. 141–152 (2003)

[46] Wattenhofer, R., Zollinger, A.: XTC: a practical topology control algorithm for ad-hoc networks. In: Proceedings of the 18th International Parallel and Distributed Processing Symposium (2004)

[47] Narayanaswamy, S., Kawadia, V., Sreenivas, R., Kumar, P.: Power control in ad hoc networks: theory, architecture, algorithm and implementation of the COWPOW protocol. In: Proceedings of the European Wireless Conference, pp. 156–162 (2002)

[48] Wightman, P., Labrador, M.A.: A3: a topology control algorithm for wireless sensor networks. In: Proceedings of IEEE Globecom (2008)

[50] Parthasarathy, S., Gandhi, R.: Fast distributed well connected dominating sets for ad hoc networks. Technical Report CS-TR-4559, University of Maryland (2004).

[51] Chen, B., Jamieson, K., Balakrishnan, H., Morris, R.: Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks.Wirel. Netw. 8(5), 481–494 (2002)

[52] Wu, J., Dai, F.: An extended localized algorithm for connected dominating set formation in ad hoc wireless networks. IEEE Trans. Parallel Distributed Syst. 15(10), 908–920 (2004)

[53] Cerpa, A., Estrin, D.: ASCENT: adaptive self-configuring sensor networks topologies. IEEE Trans. Mobile Comput. 3(3), 272–285 (2004)

[54] Xing, G., Lu, C., Zhang, Y., Huang, Q., Pless, R.: Minimum power configuration for wireless communication in sensor networks. ACM Trans. Sens. Netw. 3(2) (2007)

[55] Kawadia, V., Kumar, P.: Power control and clustering in ad hoc networks. In: Proceedings of INFOCOM, pp. 459–469 (2003)

[56] Bao, L., Garcia-Luna-Aceves, J.J.: Topology management in ad hoc networks. In: Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp. 129–140 (2003)

[57] Busse, M., Haenselmann, T., Effelsberg,W.: TECA: a topology and energy control algorithm for wireless sensor networks. In: Proceedings of the 9th ACM International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems, pp. 317–321 (2006)

[58] http://en.wikipedia.org/wiki/ISM_band

[59] http://www.sparkfun.com Časť stránok eshopu s wireless komponentmi

[60] http://standards.ieee.org/ IEEE 802.11 (802.11 a,b,g a n) a IEEE802.15 (802.15.1 a 802.15.4)

[61] Haccom http://www.rf-module-china.com/ Dokumentácia k modul HAC-UM96

[62] http://www.rovingnetworks.com Dokumentácia k modulom RN41a RN131G

[63] http://www.digi.com/ Dokumentácia k modulom nRF2401A a NRF24L01+

[64] http://www.nordicsemi.com/ Dokumentácia k modulom XBee a XTend.

[65] http://www.hoperf.com/ Dokumentácia k modulom RFM12 a RFM22