



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

LOKALIZACE A JEJÍ VLIV NA DALŠÍ PROCESY V BEZDRÁTOVÉ SÍTI

LOCALIZATION AND ITS INFLUENCE ON OTHER PROCESSES IN A WIRELESS NETWORK

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. KAREL ZEMÁNEK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PATRIK MORÁVEK

BRNO 2010



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: Bc. Karel Zemánek

ID: 77738

Ročník: 2

Akademický rok: 2009/2010

NÁZEV TÉMATU:

Lokalizace a její vliv na další procesy v bezdrátové síti

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je vyšetřit přínosy lokalizace a její vliv na další procesy v bezdrátové senzorové síti. Práce bude obsahovat studium úvodní problematiky a vypracování přehledu procesů, které jsou lokalizací a znalostí pozice ovlivněny.

Dále bude práce demonstrovat vliv chyby lokalizace na geograficky zaměřené směrování a následnou energetickou spotřebu v síti pomocí simulací, které budou provedeny v simulačním nástroji Matlab.

DOPORUČENÁ LITERATURA:

[1] HOLGER, K., ANDREAS, W. Protocols and Architectures for Wireless Se Networks. John Wiley & Sons Inc., 2005. 481 s. ISBN 978-0-470-09510-2.

[2] FENG, Z., LEONIDAS, G. Wireless Sensor Networks: An Information Processing Approach. Morgan Kaufmann publishers Inc., 2004. 376 s. ISBN 1-55860-914-8.

Termín zadání: 29.1.2010

Termín odevzdání: 26.5.2010

Vedoucí práce: Ing. Patrik Morávek

prof. Ing. Kamil Vrba, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Tato diplomová práce se zabývá lokalizací a jejím vlivem na další procesy v bezdrátové síti. První kapitola je věnována studiu lokalizačních algoritmů v bezdrátových senzorových sítích. Druhá a třetí kapitola je věnována hierarchické agregaci a protokolu Greedy perimetr stateless rating (GPSR). Čtvrtá část je zaměřena na implementaci GPSR protokolu v simulačním nástroji MATLAB. Obsahuje popisy jednotlivých m-souborů, které jsou použity při simulaci. Pátá část se zabývá samotnou simulací. Poslední část je věnována prezentaci výsledků.

Abstract

The masters thesis concerns localization and its influence on other processes in a wireless network. The first part of the thesis is devoted to the study of localization algorithms in wireless sensor networks. The second and third part is devoted to the description of hierarchiacal aggragation and Greedy perimetr stateless rating (GPSR). The fourth part presents implementation of GPSR protokol into the MATLAB simulation tool. It contains the description of specific m-files, that are use for simulation. The fifth part deals with the simulation itself. And the final part presents simulation results.

Klíčová slova: Lokalizace, bezdrátové senzorové sítě, Greedy perimetr stateless routing (GPSR)

Key words : Localization in wireless sensor networ, Greedy perimetr stateless routing (GPSR) ,

ZEMÁNEK, K *Lokalizace a její vliv na další procesy v bezdrátových sítích: diplomová práce*. Brno: FEKT VUT v Brně, 2010. 46 stran. Vedoucí práce Ing. Patrik Morávek

Prohlášení

Prohlašuji, že svoji diplomovou práci na téma Lokalizace a její vliv na další procesy v bezdrátových sítích jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne

.....

Děkuji vedoucímu diplomové práce Ing. Patriku Morávkovi, za užitečnou pomoc a cenné rady při zpracování diplomové práce.

V Brně dne

SEZNAM TABULEK

Tabulka 1 Použité uzly při doručení paketu – původní rozložení	35
Tabulka 2 Použité uzly při doručení paketu – chyba 4%.....	37
Tabulka 3 Použité uzly při doručení paketu – chyba 7%.....	38
Tabulka 4 Použité uzly při doručení paketu – chyba 10%.....	40
Tabulka 5 Použité uzly při doručení paketu – chyba 14%.....	41

SEZNAM OBRÁZKŮ

Obr. 1.1 Princip metody ToF	13
Obr. 1.2 Princip metody APIT	15
Obr. 1.3 Zvyšování počtu skoků v síti	16
Obr. 2.1 Příklad greedy forwarding	17
Obr. 2.2 Uzel X se vzdálenými sousedy	19
Obr. 2.3 Pravidlo pravé ruky uplatněné uvnitř trojúhelníku	20
Obr. 3.1 Směrování do omezené oblasti	21
Obr. 3.2 Směrování ke všem uzlům	22
Obr. 3.3 Ukládání dat	24
Obr. 3.4 Doručení dotazu a odpovědi	25
Obr. 4.1 Síťová topologie	27
Obr. 4.2 Princip metody Planarized Graph	28
Obr. 4.3 Rotace kolem počátku	30
Obr. 4.4 Smyčka vzniklá při směrování	32
Obr. 5.1 Cesta paketu při původním rozložení uzlů	36
Obr. 5.2 Cesta paketu k cíli při chybě 4%	38
Obr. 5.3 Cesta paketu k cíli při chybě 7%	39
Obr. 5.4 Cesta paketu k cíli při chybě 10%	40
Obr. 5.5 Cesta paketu k cíli při chybě 14%	41
Obr. 5.6 Cesta paketu k cíli při chybě 21%	42
Obr. 5.7 Cesta paketu k cíli při chybě 28%	43

OBSAH

Úvod	10
1 Lokalizační techniky	12
1.1 Cetralizované lokalizační techniky	12
1.2 Decentralizované lokalizační techniky.....	12
1.2.1 Range-based.....	12
ToF.....	13
GPS	13
RADAR.....	14
AHLoS	14
CBR.....	14
1.2.2 Range-free lokalizační techniky	14
Lokální techniky	15
DV-Hop lokalizační techniky	15
2 Greedy perimetr stateless routing	17
2.1 Greedy forwarding	17
2.2 Pravidlo pravé ruky: Perimetry	18
3 Hierarchická agregace.....	21
3.1 Agregace dat.....	21
3.2 Hierarchická agregace dat	22
3.2.1 Shlukování a geografická hierarchie.....	23
3.2.2 Ukládání dat.....	23
3.2.3 Doručení dotazu a odpovědi	24
4 Návrh simulace.....	26
4.1 Vytvoření síťové topologie	26
4.2 Vytvoření matice sousedů	27
4.3 Realizace směrování.....	29
4.3.1 Realizace greedy směrování	29
4.3.2 Realizace perimetrického směrování	30
4.4 Chyby při realizaci směrování za použití protokolu GPSR	32
4.5 Změna polohy uzlů.....	33
5 Simulace	35
6 Závěrečné hodnocení simulace	44
Závěr	45
SEZNAM POUŽITÉ LITERATURY.....	47

Úvod

Tato diplomová práce se zabývá problematikou lokalizace a jejím vlivem na další procesy v bezdrátových sítích. Těmito sítěmi jsou myšleny především bezdrátové senzorové sítě. Tyto sítě se skládají z prostorově rozmístěných autonomních senzorů, které sledují různé fyzikální veličiny nebo vnější podmínky, jako například teplotu, hluk, vibrace, tlak, pohyb nebo imise. Vývojem těchto sítí se zpočátku zabývala výhradně armáda, a to hlavně z důvodu znát přesné údaje na bojišti. V dnešní době se již bezdrátové senzorové sítě rozšířily z vojenského do civilního sektoru. Můžeme se s nimi setkat při monitorování průmyslových výrobních procesů, v zařízeních sledujících lidské zdraví, při monitorování životního prostředí nebo například při řízení dopravy.

První část práce se věnuje teoretickému popisu problému lokalizace v bezdrátových senzorových sítích. V druhé kapitole je uveden přehled různých lokalizačních technik, jejich přesnost a energetické nároky. Přesnou polohu čidel potřebujeme znát jak pro potřebu přiřadit konkrétní hodnotě místo kde byla naměřena, tak i pro potřebu směrování toku dat skrze senzorovou síť. Právě při směrování paketů nám může vadit špatná lokalizace jednotlivých senzorů, neboť se může stát, že pro směrování dat nebude zvolena nejkratší a nejvýhodnější cesta. Síť se tedy zbytečně zatíží a zvýší se spotřeba energie jednotlivých senzorů.

Problematicke směrování v bezdrátových sítích se věnuji v kapitole 3, kde je popsán směrovací protokol Greedy perimetr stateless routing (GPSR), tento protokol pracuje se znalostí polohy jednotlivých senzorů. Další procesem závislým na znalosti polohy senzorů je hierarchické směrování a agregace dat. Tato technika je popsána v kapitole 4. Z těchto dvou procesů jsem si k simulaci zvolil směrovací protokol GPSR, neboť jsem usoudil, že právě na něj by mohlo mít chybné určení vlastní polohy jednotlivých uzlů výraznější vliv.

Popis vlastního návrh simulace je uveden v kapitole 5. V této kapitole se především věnuji realizaci vlastního protokolu. Jsou zde popsány jednotlivé funkce v m-souborech, které realizují vytváření vlastní testované topologie, počítání sousedů a nakonec vlastní algoritmus výběru vhodného uzlu ke skoku. Na závěr této kapitoly jsou

uvedeny problémy protokolu, jako je například zacyklování skoků mezi uzly při použití greedy forwardingu nebo skok do slepé ulice. Tedy skok na uzel, jenž má pouze jednoho souseda, od kterého mu právě paket přišel.

Kapitola 5 je věnována samotné simulaci. Je zde vygenerována topologie a vzorová cesta od počátečního uzlu k cíli. Následně je tato topologie pomocí funkce několikrát měněna. Uzly jsou posunovány oproti původní poloze o zadanou hodnotu. Výsledná topologie je následně testována. Opět se ze zvoleného počátečního uzlu směřuje paket k zadanému cíli. Výsledky jsou popsány pod příslušnými obrázky s cestami paketů

Poslední kapitola je věnována závěrům, zjištěným při simulaci. Zejména překvapivým výsledkům, jak moc je ovlivněno celkové směřování při prvních skocích prováděných greedy směřováním.

1 Lokalizační techniky

Lokalizace v bezdrátových sensorových sítích má smysl především u velkých sítí, čítajících stovky čidel. Pokud například sledujeme teplotu v budově, není problém instalovat například 10 čidel, u nichž známe přesně jejich pozici. Zde pro nás lokalizační algoritmy nemají význam. Pokud však chceme sledovat teplotu například v deštěném pralese a jednotlivé čidla budou rozmístěna pomocí letadla, nebudeme znát jejich přesné polohy. Právě tady přijdou na řadu vhodné algoritmy, které pomocí všech dostupných informací vypočítají polohu všech senzorů.

1.1 Centralizované lokalizační techniky

Centralizované lokalizační techniky zahrnují lokalizaci senzorů založených na základě vysílání dat v pořadí do centrálního uzlu, kde se vypočítá pozice jednotlivých senzorů. Vysílání časové sekvence dat skrze síť do centrálního uzlu je nákladné, neboť napájecí energie každého senzoru je omezená a dochází tak ke zkrácení životnosti senzoru. Další negativní prvek této metody je to, že vysílání všech prvků napříč sítí vede k zatížení sítě a dochází poté ke zpoždění důležitých dat.

1.2 Decentralizované lokalizační techniky

Decentralizované, nebo též distribuované lokalizační techniky jsou závislé na tom, že každý ze senzorů je schopný určit svou vlastní polohu pouze s minimální komunikací se sousedními senzory. Decentralizované techniky dělíme na dva typy: range-based a range-free.

1.2.1 Range-based

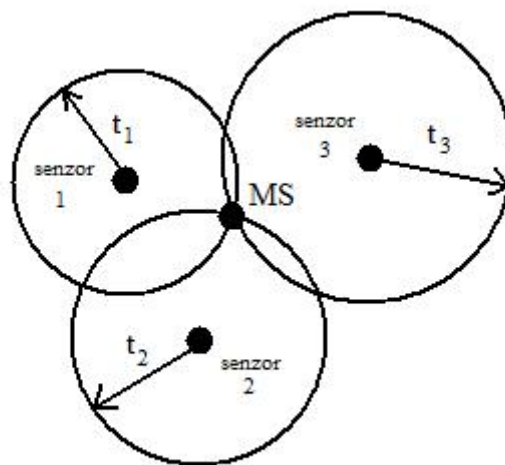
Tato technika je založena na nalezení polohy pomocí absolutního odhadu vzdálenosti nebo odhadu úhlu mezi dvěma sousedními senzory. Do této techniky

zahrnujeme spoustu metod jako například GPS, ToF (z anglického Time of flight), RADAR, AHLoS (Ad-Hoc Localization System), Calamari, CBR a další.

ToF

Tato metoda je založena na měření zpoždění šířícího se signálu. Senzor (na obrázku označen jako MS) vypočítá svou vzdálenost na základě časového zpoždění signálu vyslaného od sousedních senzorů, její princip je znázorněn na obrázku 1.1 [2]. Tento jednoduchý princip má však velkou nevýhodu. Jelikož je rychlost šířícího se signálu velmi vysoká (v podstatě se blíží rychlosti světla), musí být měření času velice přesné aby nedocházelo v lokalizaci k velkým chybám. Například přesnost lokalizace na 1 metr požaduje měření času s přesností 3,3 nanosekundy [2], což lze v bezdrátových senzorových sítích jen velice těžko dosáhnout.

Pokud bychom použili akustický signál, rychlost šíření by klesla přibližně na 333 m/s (dle teploty a vlhkosti vzduchu). V tomto případě by nám stačila přesnost měření času 1 ms na přesnost lokalizace v hodnotě 35 cm.



Obr. 1.1 Princip metody ToF

GPS

Metoda založená na principu sledování signálu ze satelitů na různých oběžných drahách kolem Země. Pro příjem signálu potřebuje přijímač přijímat signál minimálně ze tří satelitů. Systém je však koncipován tak, aby v kterémkoliv místě na Zemi byl možný příjem aspoň ze čtyř satelitů. Pokud má přijímač k dispozici více dostupných satelitů, zvyšuje to přesnost jeho lokalizace.

RADAR

Technika vyvinutá společností Microsoft Research group. Tento systém využívá rádiových frekvencí a primární použití této metody je pro sledování uživatelů uvnitř budovy. Metoda vychází z principů IEEE.11 WaveLAN bezdrátové sítě. Hlavní princip spočívá v sledování a zaznamenávání úrovně signálů od vysílací bodů. V podstatě vznikne rádiová mapa celé budovy a to ve všech jejích patrech. Na základě této mapy je přijímač schopen určit, dle získaných úrovní signálů, svou polohu. Mapa musí být neustále přepočítávána, neboť je nutné počítat s tím, že uživatelé se pohybují a s nimi se i mění i zdroje rušení (například mobilní telefony, notebooky apod.).

AHLoS

AHLoS je lokalizační systém v bezdrátových sensorových sítích. Je založen na souboru distribuovaný opakujících se algoritmů [3]. Základním kamenem techniky AHLoS je skupinka pár senzorů, u kterých přesně známe jejich polohu, a od nichž uplatňuje metodu ToF, jež je hlavní metodou pro tento systém.

CBR

U této metody zjišťuje senzor svou polohu na základě úrovně přijatého signálu nebo času, za jak dlouho přijal signál (metoda ToF).

Všechny range-based lokalizační techniky, jak jsem uvedl již dříve, jsou decentralizované. Pomocí nich se dá s velkou přesností určit poloha daného senzoru. Daň za tuto přesnost je potřeba dalšího externího hardwaru, který musíme k senzoru přidat. Každá součástka navíc stojí více peněz a spotřebovává další energii. Tím zkracuje dobu života senzoru, což je jeden z nejdůležitějších parametrů. Tyto okolnosti omezují použití těchto metod pro nízkoenergetické senzory.

1.2.2 Range-free lokalizační techniky

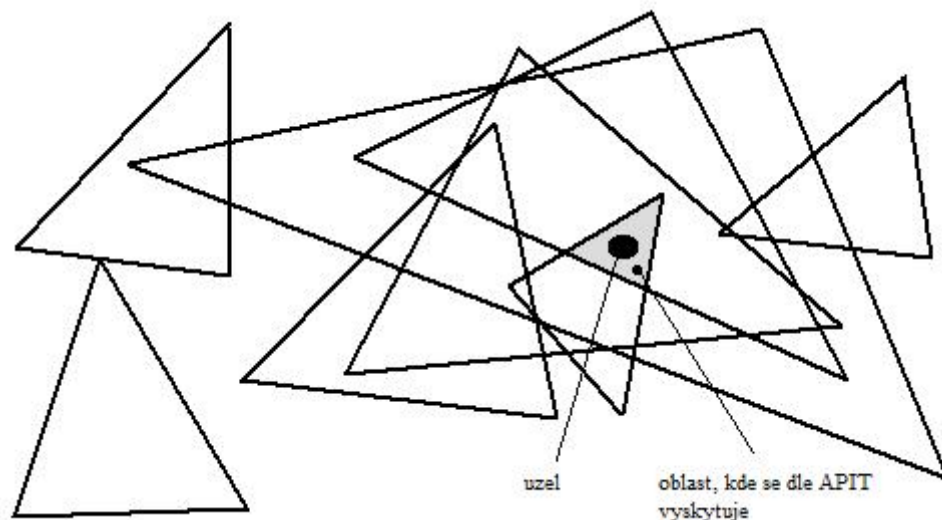
Výzkum ukázal i alternativní řešení lokalizačních problémů v bezdrátových sensorových sítích, které není závislé na vzdálenosti mezi uzly. Lokalizační techniky založené na tomto principu dovolují používat standardní rádiové moduly a základ lokalizace je závislý jen na obsahu přijatých zpráv. Z toho vyplývá, že tyto techniky

nepotřebují žádný přídavný hardware. To z nich dělá levnější a energeticky méně náročné řešení při lokalizaci senzorů v bezdrátových sensorových sítích.

Lokální techniky

Tyto techniky předpokládá, že přijímací stanice jsou „hloupé“, a proto každý senzor naslouchá a komunikuje s více přijímacími stanicemi zároveň. Příkladem je technika nazvaná Centroid (N. Bulusu, J. Heidemann, D. Estrin) pro velmi malé a levné zařízení nezávislé na GPS. Je založena na kruhovém šíření rádiového signálu a ve své podstatě je velice jednoduchá. Každý senzor určí svou polohu na základě měření a vypočítání pomyslného středu mezi všemi přijímacími stanicemi, které slyší.

Další technika se nazývá APIT [1]. Tato technika potřebuje heterogenní síť snímacích zařízení, ve které jsou v malém procentu zastoupeny tzv. anchor uzly (uzly opatřené GPS modulem). Při této metodě rozdělíme vždy prostor pomocí tří různých anchor uzlů na trojúhelníkové oblasti, abychom senzoru dovolili zúžit oblast, ve které se pravděpodobně nachází. Princip je naznačen na obrázku 1.2 [1].

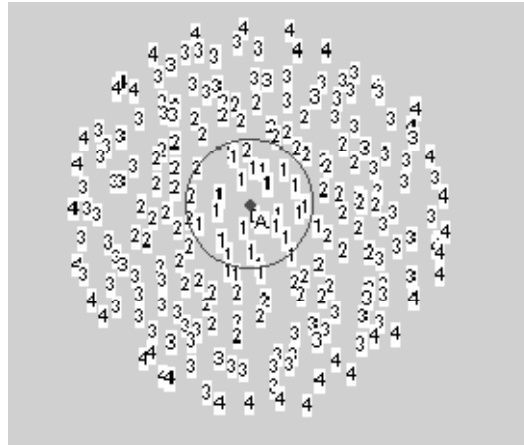


Obr. 1.2 Princip metody APIT

DV-Hop lokalizační techniky

Tyto techniky používají mechanismus, který je podobný klasickým distance-vector protokolům. Přijímací zařízení vyše směrem k senzorům zprávu, ve které je obsažena jeho poloha a čítač skoků je nastaven na jedna. Každý senzor přijme všechny

zprávy od všech přijímacích zařízení a ignoruje ty s největším počtem skoků. Tato zpráva se šíří celou sítí a postupně se zvyšuje počet skoků s tím, jak přeskakuje od jednoho senzoru k druhému. Tímto způsobem budou znát všechny senzory v síti nejkratší vzdálenost k nejbližšímu přijímacímu zařízení. Zvyšování přeskoků skrz síť pro jedno přijímací zařízení je naznačeno na obrázku 1.3 [1].



Obr. 1.3 Zvyšování počtu skoků v síti

Pro přepočítání skoků na skutečnou fyzickou vzdálenost stanoví systém průměrnou vzdálenost na jeden skok. Přijímací zařízení toto stanoví na základě získání pozice a počtu skoků od všech přijímacích zařízení v síti. Průměrná vzdálenost na jeden skok pro přijímací zařízení i je dle [1] dána vztahem 2.1

$$HopS = \frac{\sum \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}{\sum h_j} \quad (1.1)$$

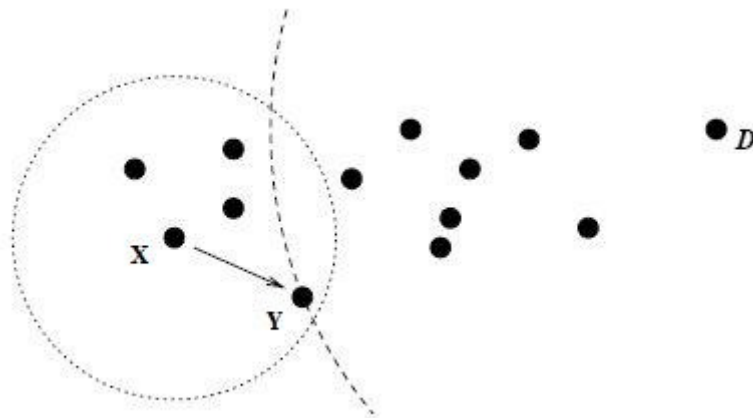
kde je (x_j, y_j) pozice přijímacího zařízení j a h_j je vzdálenost ve skocích od přijímacího zařízení j k zařízení i . Jakmile dojde k výpočtu, zařízení vyše tuto informaci k jednotlivým uzlům.

2 Greedy perimetr stateless routing

V této kapitole bude představen Greedy perimetr stateless routing (GPSR), směrovací protokol pro bezdrátové datagramové sítě, který využívá znalosti polohy jednotlivých uzlů v síti a na jejich základě směřuje jednotlivé pakety k cíli. GPSR protokol používá pro směrování metodu greedy forwarding, která vyžaduje po uzlu pouze znalost svých nejbližších sousedů v síťové topologii. Když paket dosáhne oblasti, kde greedy forwarding není možné použít, musí se využít algoritmus směrování do obvodu okolní oblasti (perimeter forwarding). Algoritmus GPSR se tedy skládá ze dvou metod pro směrování paketů: greedy forwarding (tato se používá, pokud je to možné) a perimeter forwarding, tato metoda se používá v oblastech, kde greedy forwarding nemůže být implementován [4].

2.1 Greedy forwarding

Každý paket je v místě svého vzniku cílovou lokalitou. Směřující uzel se po příjmu tohoto paketu rozhodne, jaký je nejoptimálnější směr dalšího skoku. Pokud uzel zná přesnou pozici svých rádiově dosažitelných sousedů, zvolí lokálně nejvýhodnější a nejkratší cestu pro další skok k uzlu nacházejícímu se nejbližší cíli. Doprava paketu k cíli spočívá tedy v postupných skocích, které jsou vždy u každého uzlu směřovány k následujícímu uzlu, jenž je geograficky nejbližší požadovanému cíli. Princip tohoto směrování vysvětlím na obrázku 2.1[4].



Obr. 2.1 Příklad greedy forwarding

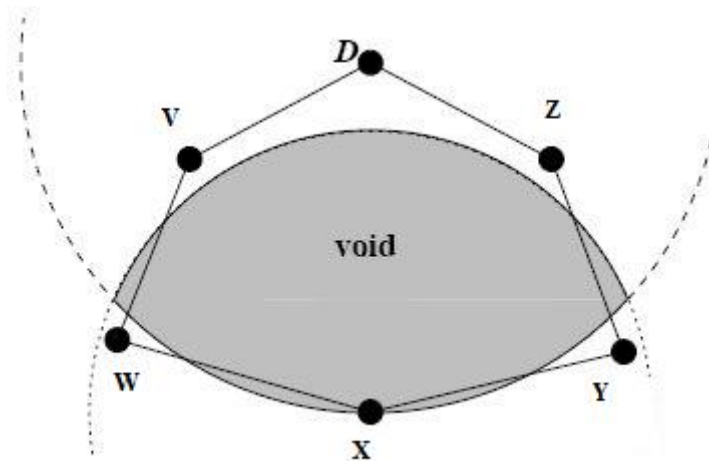
Uzel X obdržel paket určený pro uzel D. Čárkovaný kruh okolo uzlu X představuje jeho rádiový dosah, čárkovaná kruhová úseč procházející uzlem Y zase představuje vzdálenost od uzlu D. Uzel X bude směřovat paket na uzel Y, neboť jeho vzdálenost k uzlu D je ze všech uzlů dostupných pro X nejmenší. Tento směrovací proces se bude opakovat tak dlouho, dokud paket nedorazí až ke svému cíli, v tomto případě uzlu D.

Velká výhoda greedy forwardingu spočívá v jeho závislosti na znalostech pouze o směrování pro jeho nejbližší sousedy. V sítích, ve kterých má být multi-hop směrování užitečné, musí být počet rádiově dostupných sousedů pro daný uzel podstatně menší než celkový počet uzlů v celé síti.

Tato hlavní výhoda, spočívající v cestě pouze na sousední uzel, má však jednu obrovskou slabinu. Existují totiž topologie, v nichž jediná cesta k dosažení cíle je ta, při které se paket bude dočasně geometricky vzdalovat od cíle, to je znázorněno na obrázku 2. Pokud je tedy vzdálenost sousedních uzlů větší než vzdálenost uzlu odesílajícího paket, nemůže se daný uzel při použití greedy forwarding rozhodnout, ke kterému z uzlů paket poslat. V této situaci musí být tedy použit jiný mechanismus směrování paketů.

2.2 Pravidlo pravé ruky: Perimetry

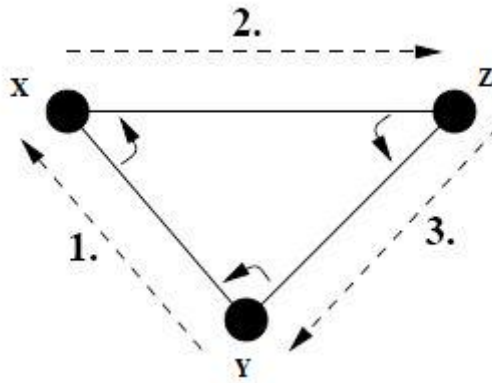
Důvod použití této metody je, jak jsem uvedl výše, nemožnost směrování pomocí greedy forwardingu k nejbližšímu uzlu, neboť každý dostupný uzel je dál než uzel, co vysílá paket. Tohle je znázorněno na obrázku 2.2 [4].



Obr. 2.2 Uzel X se vzdálenými sousedy

Na obrázku je znázorněn rádiový dosah uzlu X a jeho vzdálenost od cílového uzlu D. Uzel X hledá ve svém dosahu nějaký nejbližší uzel, přes který by mohl směřovat paket, ten však nenachází. Je zde pouze prázdné místo bez uzlů, v literatuře se označuje jako void (na obrázku vybarveno šedě). Pokud tedy existuje cesta mezi X a D, vede přes jiné uzly, neboť v šedé zóně se žádné uzly nenachází.

Proto zde použijeme známé pravidlo pravé ruky. Princip tohoto pravidla je naznačen na obrázku 2.3 [4]. Toto pravidlo nám říká, že pokud dorazí paket k uzlu x od y, bude další jeho skok směřován od uzlu x proti směru hodinových ručiček vůči cestě (x, y). Je známo, že pravidlo pravé ruky prochází vnitřní prostor v uzavřeném mnohoúhelníku podle směru hodinových ručiček. V našem případě tedy vzniknul trojúhelník, ohraničený hranami mezi uzly X, Y a Z. A dle pravidla pravé ruky ho projdeme v pořadí $Y \rightarrow X \rightarrow Z \rightarrow Y$. Pokud uplatníme stejné pravidlo na venkovní prostor, v našem případě na prostor mimo trojúhelník, bude paket postupovat proti směru hodinových ručiček.



Obr. 2.3 Pravidlo pravé ruky uplatněné uvnitř trojúhelníku

Tohoto kruhového cestování paketu se snažíme využít především kolem prázdných míst – voids. Například na obrázku 2.2 by paket při použití pravidla pravé ruky cestoval kolem naznačeného prázdného místa po trase $X \rightarrow W \rightarrow V \rightarrow D \rightarrow Z \rightarrow Y \rightarrow X$. Tuto posloupnost cestování paketu po hraně prázdných míst (angl. voids) dle pravidla pravé ruky nazýváme perimetr.

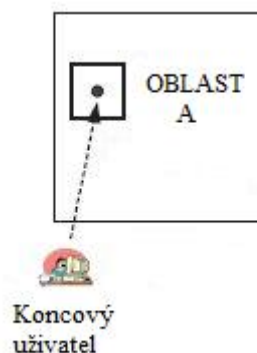
3 Hierarchická agregace

Bezdrátové sensorové sítě, které se z pravidla skládají ze stovek až tisíců uzlů, operují s různými druhy dat a jsou distribuovaně řízené. Každý uzel, jak jsem uvedl již dříve, má omezení v podobě výdrže napájecí zdroje (ve většině případů se jedná o baterii), rádiového dosahu, CPU, paměti a propustnosti. Pro efektivní sběr dat z mnoha uzlů bylo třeba navrhnout nové protokoly určené pro bezdrátové sensorové sítě. Spotřeba energie je hlavní problém v bezdrátových sensorových sítích. Největší podíl na spotřebě energie má rádiová komunikace uzlu, proto je efektivní seskupit data v síti, a tím snížit objem nadbytečně přenášených dat a také snížit vlastní režii v síti.

3.1 Agregace dat

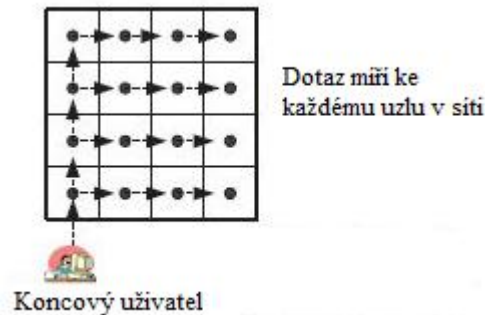
V sensorových sítích není datová komunikace přímo závislá na ID (identifikátoru) každého uzlu, ale záleží na datech přenášených v síti [6]. Když je dotaz od koncového uživatele adresován pouze do určité omezené oblasti, může být spotřeba energie pro datovou komunikaci omezena, jelikož nemusíme dotaz šířit na všechny uzly v síti.

Například pokud se koncový uživatel bude dotazovat na průměrnou teplotu v určité omezené oblasti nazvané například A, bude dotaz směřován pouze do této oblasti a nebude se šířit dále v síti. Toto je naznačeno na obrázku 3.1.



Obr. 3.1 Směrování do omezené oblasti

Na druhou stranu může být dotaz směřován do celé sítě, protože koncový uživatel nezná přesnou polohu odkud brát požadované data. Například po dotazu, ve které oblasti je teplota 20° Celsia, je nutné se ptát všech uzlů v síti a tudíž přeposlat tento dotaz do celé senzorové sítě, jak je demonstrováno na obrázku 3.2 [6].



Obr. 3.2 Směrování ke všem uzlům

Záplava dotazujících paketů se vyhýbá sbírání dat z celé sítě do centrálního serveru pomocí algoritmu GHT (z anglického názvu Geographical Hash Table). Centrální server GHT ukazuje agregaci dat s různými atributy pro některé uzly. Pokud některý uzel rozpozná nějaké nové události, doručí tyto data ke konkrétnímu uzlu, který je vybrán pomocí hash funkce. Koncový uživatel pak může pomocí stejné hash funkce sbírat data z těchto vybraných uzlů. Nicméně tyto data jsou uloženy do uzlů, aniž by byla brána v potaz celková topologie sítě. Proto se může při větším rozsahu sítě a užitím algoritmu GHT stát, že některé uzly budou doručovat data na zbytečně velké vzdálenosti. To má za následek růst nároků na komunikační režii. Zvýšení režie pro vybraný uzel může též nastat, pokud je objem přijímaných dat příliš veliký.

3.2 Hierarchická agregace dat

Předpokládáme následující :

- každý senzor zná svou pozici a své sousední uzly
- každý senzor zná rozsah celé fyzické sítě

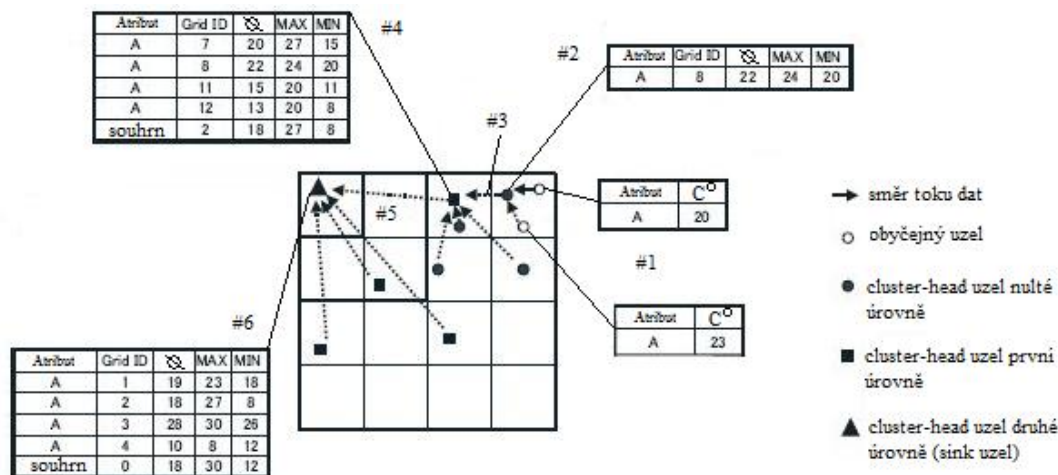
- každý senzor má společnou hashovací funkci na zvolení velícího uzlu (angl. cluster-head)

3.2.1 Shlukování a geografická hierarchie

Celá oblast, kde se vyskytují uzly, je opakovaně rozdělena na čtyři mřížky a hierarchicky uspořádaná tak, že vytváří mřížky různých úrovní. Pomocí hash funkce, kterou má každý uzel a každý koncový uživatel, se zvolí v každé mřížce tzv. cluster-head uzel.

3.2.2 Ukládání dat

Zde popíšeme, jak posílat a skladovat data od každého uzlu v jeho cluster-head uzlu. Data každého uzlu, představující atribut (angl. attribute) a hodnotu (value), jsou shromážděna do cluster-head uzlu v každé mřížce na nejnižší úrovni. To je naznačeno na obrázku 3.3 [6], a označeno jako #1. Poté jsou v tomto cluster-head uzlu spočítány společné hodnoty pro všechny uzly, jako například průměrná hodnota, nebo minimum a maximum pro danou mřížku. Toto je na obrázku označeno jako #2. Cluster-head uzel v každé mřížce je opatřen svým identifikačním číslem (ID), jež udává jeho geografickou polohu a kterým se prezentuje u cluster-head uzlu vyšší úrovně. Zde na obrázku je uvedeno jako Grid ID a celá operace je označena #3. Cluster-head uzel vyšší úrovně udržuje agregované informace všech čtyř mřížek nižší úrovně a opět počítá jejich průměrné hodnoty a agreguje data. Na obrázku označeno #4. Tato operace se stále opakuje (#5), dokud nejsou data z celé sítě shromážděna a agregována do tzv. sink uzlu (#6), který je domovským uzlem dané sítě. Každý cluster-head má udržet pouze informace od čtyř mřížek nižší úrovně, takže se tím vyhneme velkým režijním požadavkům na konkrétní uzel. Je též možné dynamicky měnit frekvenci registrace údajů na všech úrovních v závislosti na poptávce dané aplikace.

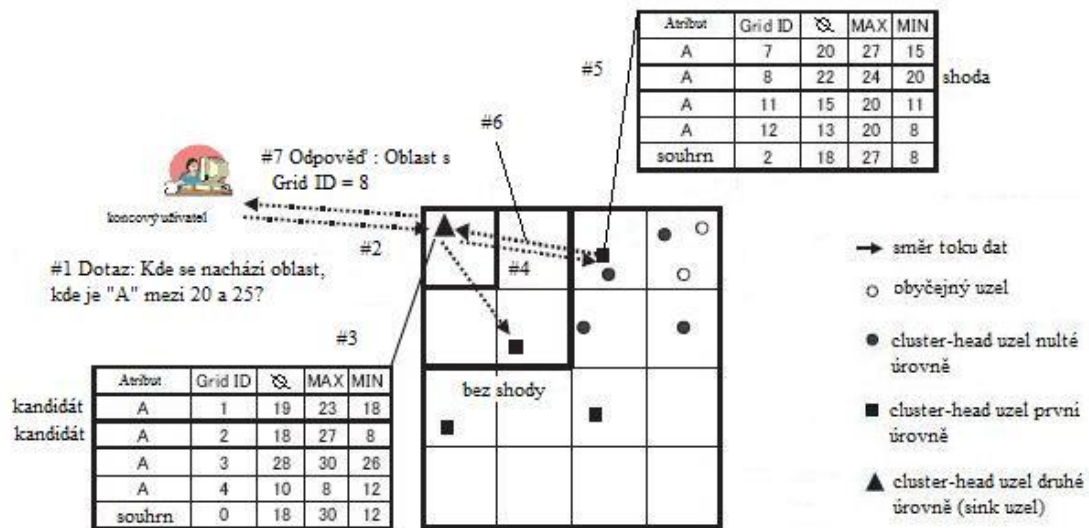


Obr. 3.3 Ukládání dat

3.2.3 Doručení dotazu a odpovědi

V této části popíšeme, jak se dostane dotaz od koncového uživatele do sítě a jak na tento dotaz každý cluster-head uzel reaguje. Jak je naznačeno na obrázku 3.4 [6], dotaz od koncového uživatele (#1), ve které oblasti je hodnota atributu A mezi 20 a 25, je doručen sink uzlu (#2). Ten zkontroluje, jestli v seznamu agregovaných informací nenajde uspokojivou odpověď na požadovaný dotaz (#3). Pokud sink uzel nenajde data, která přesně odpovídají požadavku, je tento paket s dotazem rozmnožen a je z něj vytvořeno pár multicastových paketů. Tyto pakety jsou pak směřovány na cluster-head uzly, které mohou mít na nižší úrovni data, jež odpovídají požadovanému dotazu (#4). Cluster-head uzel v mřížce nižší úrovně obdobně zopakuje vyhledávání na shodu s požadovaným dotazem a pokud neuspěje a nenajde přesnou shodu, rozešle pakety na cluster-head uzly nižší úrovně. Když je objevena oblast (Grid ID), která přesně odpovídá vznesenému dotazu (#5), odpoví cluster-head uzel uzlu na vyšší úrovni prostřednictvím zpětného průchodu (#6). Nakonec sink uzel navrátí koncovému uživateli požadovaný identifikátor oblasti (Grid ID), ve které jsou odpovídající hodnoty, na které se koncový uživatel dotazoval (#7).

Pokud tedy koncový uživatel vyhledává určité data, je metoda, kdy je oblast rozdělena na čtvercovou stromovou hierarchii, méně náročná na režii a také na pohyb dat v síti, než kdyby se dotaz šířil celou sítí uzel od uzlu.



Obr. 3.4 Doručení dotazu a odpovědi

4 Návrh simulace

Návrh simulace se věnuje vlivu lokalizace na geografické směrování protokolu GPSR a ovlivnění energetické spotřeby, zejména počtem uzlů zapojených do přenosu.

Při studii hierarchické agregace jsem usoudil, že zde chyba v lokalizaci nebude mít nějaký větší důsledek. Může se stát, že bude uzel chybou v lokalizaci zařazen do jiné mřížky a pod jiný cluster-head uzel. Nebo se cluster-head uzlem stane uzel mimo pomyslnou mříž a data se budou posílat o trochu dále, než bylo původně v plánu. Ale jinak by zde neměla chyba v lokalizaci jednotlivých uzlů více projevit. Z tohoto důvodu jsem se při simulaci zaměřil na směrovací protokol GPSR, neboť právě zde nastane vlivem špatné lokalizace zapojení více uzlů do směrovacího procesu.

Simulaci provádím v simulačním nástroji Matlab. Celá simulace je rozdělena na několik dílčích částí, které jsou realizovány jednotlivými vytvořenými funkcemi. Každá funkce je realizována v jednotlivých m-souborech. Následující kapitoly budou popisovat jednotlivé části, jejich vstupy, výstupy a vnitřní popis funkcí.

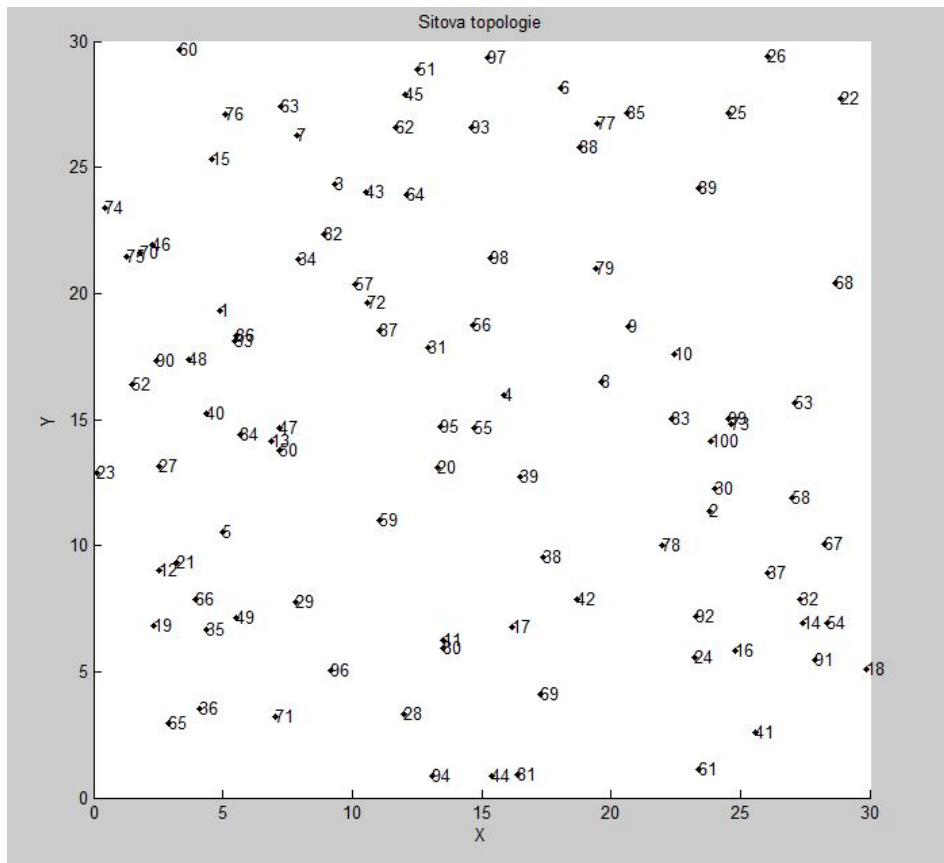
4.1 Vytvoření síťové topologie

První funkce, pojmenovaná `my_topo`, má za účel vytvoření topologie sítě. Tato funkce má tři vstupy:

- počet uzlů (ve funkci označen jako `maxn`)
- šířka pole, na kterém se mají senzory vyskytovat (ve funkci označena jako `maxx`)
- délka pole, na kterém se mají senzory vyskytovat (ve funkci označena jako `maxy`)

Na základě těchto vstupů se vygeneruje pole o zadaných rozměrech, na kterém se vyskytují jednotlivé uzly. Výskyt těchto uzlů je náhodným jevem, o který se stará příkaz `rand` v simulačním nástroji Matlab. Tento příkaz generuje náhodné číslíčky v intervalu mezi 0 a 1. Tyto hodnoty pak vynásobím patřičným rozsahem, zadaným ze

vstupních hodnot, a získám tak matici náhodných čísel. Ta je použita pro vytvoření jednotlivých uzlů, resp. jejich souřadnic. Výstupem této funkce je tedy matice uzlů, jejich čísla jsou uloženy v prvním řádku, souřadnice ve druhém resp. třetím řádku. Nakonec tato funkce vykreslí vygenerovanou topologii. Na obrázku 4.1 je zobrazena topologie s polem o velikosti 30 krát 30 a počtem 100 uzlů.



Obr. 4.1 Síťová topologie

Dalším krokem při vytváření simulace bylo vytvoření matice sousedů. Neboť, jak jsem uvedl výše v teoretické části, protokol GPSR vyžaduje, aby každý uzel znal své sousedy ve svém rádiovém dosahu.

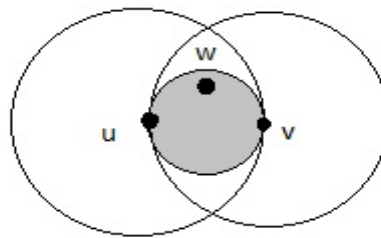
4.2 Vytvoření matice sousedů

Matici sousedů vytváří funkce `my_distance`. Jejím vstupem je matice uzlů vygenerována předchozí funkcí a rádiový dosah uzlů, ve funkci označen jako proměnná `range`. Vytváření matice je poměrně jednoduché. Jedná se o dva cykly, které postupně

počítají vzdálenosti mezi jednotlivými uzly a porovnávají, jestli jsou větší nebo rovny se zadaným rozsahem. Tímto způsobem vznikne matice o n řádcích, kde n je počet zadaných uzlů, a m sloupcích, kde m je počet sousedů uzlu s nejvíce sousedy.

Problém zde nastal až při samotném testování protokolu. Vzhledem k velkému počtu možných skoků a více cest mezi uzly se dost často stávalo, že se protokol v hustěji obsazených místech zacyklil. Tento problém bylo nutné řešit, v literatuře [4] je popsána technika řešení nazvaná Planarized Graphs.

Princip této metody spočívá ve vyloučení uzlu ze směrovacích tabulek (resp. tabulek sousedů). Uzel, jenž leží mezi dvěma dalšími uzly a jehož vzdálenost od nich je menší než poloviční vzdálenost mezi těmito uzly, je vyloučen z tabulky sousedů a pro další směrování se nepoužívá. Princip této metody je naznačen na obrázku 4.2.



Obr. 4.2 Princip metody Planarized Graph

Na obrázku jsou znázorněny 3 uzly, označené jako u , v a w . Mezi všemi třemi existuje cesta, neboť jsou v rádiovém dosahu. Pro nás je ale žádoucí vynechat mezilehlý uzel. Ať už kvůli eliminaci smyček, nebo z důvodu omezení počtu skoků. Proto jsem tuhle metodu zařadil hned do funkce, která vyhledává sousedy a nazval ji `my_distance1`. Pro zpětné porovnání jsou oba m -soubory s těmito funkcemi na příloženém CD.

Princip eliminace spočívá ve vyhledávání těchto uzlů mezi svými sousedy. Prvně vygenerovaná matice sousedů se zpětně prohledává a jsou z ní tyto uzly odstraněny. Prvním krokem je určení středu mezi uzlem a jeho sousedem. Následně se prohledává, zda se v poloviční vzdálenosti, než je vzdálenost těchto dvou uzlů, nenachází další ze sousedů. Pokud ano, tak je z této matice vynechán.

Výstupem těchto dvou funkcí je v obou případech matice obsahující v jednotlivých řádcích čísla sousedů. Číslo řádku odpovídá číslu uzlu, pro který jsou

daní sousedi dostupní. Pokud nemá uzel dostatek sousedů, doplní se zbylé sloupce hodnotou 0. Tohoto faktu využijeme při realizaci vlastního směrování.

4.3 Realizace směrování

Nejsložitější a nejdůležitější částí simulace je funkce realizující vlastní směrování. Realizována je funkcí nazvanou `my_routing`. Funkce má 6 vstupních parametrů, jsou to:

- matice uzlů, vygenerována funkcí `my_topo`
- matice sousedů, vygenerována funkcí `my_distance`
- šířka pole, stejná hodnota, jenž se zadávala ve funkci `my_topo` (ve funkci označena jako `maxx`)
- délka pole, stejná hodnota, jenž se zadávala ve funkci `my_topo` (ve funkci označena jako `maxy`)
- číslo uzlu, ze kterého chceme začít směrování (ve funkci označeno jako `start`)
- číslo uzlu příjemce (ve funkci označeno jako `destination`)

Funkce je logicky rozdělena na dvě části, první částí je klasické greedy směrování a druhou částí je putování po perimetru. Obě části jsou součástí cyklu `while`, s podmínkou opakování, dokud není dosaženo cílového uzlu.

4.3.1 Realizace greedy směrování

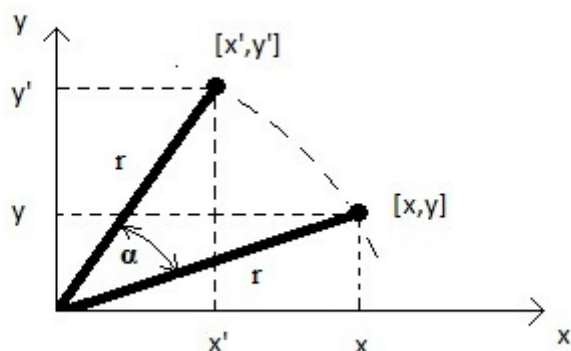
Tato část kódu má na starost nalezení uzlu z matice sousedů, který je nejbližší cílovému uzlu (pokud takový existuje). Princip spočívá v procházení řádku matice se sousedy uzlu, na kterém se právě nacházíme, a postupném počítání vzdálenosti těchto uzlů od cílového uzlu. Po výpočtu je porovnáváno, zda je soused blíže než aktuální uzel a případně navržen jako kandidátský uzel ke skoku. Pokud není žádný soused zvolen jako vhodný kandidát, zůstane číslo kandidáta rovno nule. To je signál pro to, aby funkce začala hledat souseda pomocí perimetru.

4.3.2 Realizace perimetrického směřování

Jak je uvedeno výše v teorii protokolu GPSR, pokud uzel nemůže najít souseda, jehož vzdálenost k cílovému uzlu je menší než vzdálenost aktuálního uzlu, musí obejít toto prázdné místo protisměru hodinových ručiček (viz obrázek 3.2). Musí tedy ze svého pohledu směrem k cílovému uzlu zvolit souseda, který je nejvíce vpravo. Tato ve své podstatě zdánlivě jednoduchá podmínka ovšem způsobila značné komplikace. K cíli se totiž můžeme pohybovat zleva doprava, shora dolů nebo naopak. Uzel mezi sousedy, který se nachází nejvíce vpravo, bude tedy pokaždé jiný. V literatuře totiž není uvedeno, jakým způsobem se tento výběr vhodného uzlu ke skoku pomocí protokolu GPSR provádí. Proto jsem při návrhu simulace musel přijít s vlastním řešením tohoto problému. Jako jediná fungující varianta, která byla schopna vybrat vhodný uzel ke skoku, se ukázala varianta s transformací souřadnic.

Jedná se o euklidovskou transformaci. Je to nejpoužívanější lineární transformace při zpracování obrazu. Tato transformace zahrnuje translaci (posunutí) a rotaci, obě zachovávají rozměry a úhly obrazu (objektů v něm obsažených) [5]. Při řešení problému s výběrem vhodného uzlu jsem použil rotaci.

Rotace bodu v rovině kolem počátku souřadného systému je naznačena na obrázku 4.3 [5].



Obr. 4.3 Rotace kolem počátku

Z vyznačeného obrázku s použitím goniometrických funkcí a původních souřadnic bodu je možné napsat:

$$\cos\beta = \frac{x}{r}$$

$$\cos\beta = \frac{y}{r} \quad (4.1)$$

Podobně, za použití základních goniometrických vzorců, platí pro souřadnice bodu otočeného kolem počátku o úhel α :

$$x' = r \cdot \cos(\alpha + \beta) = r(\cos \alpha \cdot \cos \beta - \sin \alpha \cdot \sin \beta)$$

$$y' = r \cdot \sin(\alpha + \beta) = r(\sin \alpha \cdot \cos \beta + \cos \alpha \cdot \sin \beta) \quad (4.2)$$

Dosazením rovnice 4.1 do rovnice 4.2 dostaneme výsledné souřadnice bodu, jenž je otočen kolem počátku o úhel α . Výpočet jeho souřadnic je uveden v rovnici 4.3 [5].

$$x' = x \cdot \cos \alpha - y \cdot \sin \alpha$$

$$y' = x \cdot \sin \alpha + y \cdot \cos \alpha \quad (4.3)$$

Tento výsledný vzorec je použit v algoritmu, který vybírá vhodný uzel při skoku po perimetru.

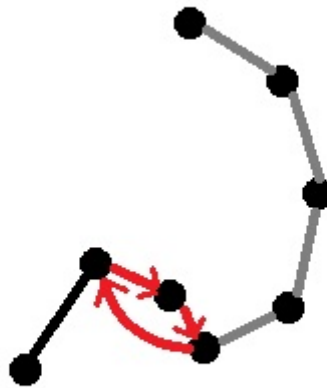
Jako první je spočítán úhel, který svírá vektor aktuální uzel - cílový uzel a y-osa souřadného systému. Tento úhel představuje úhel α , tedy úhel o který se bude otáčet souřadný systém. Aktuální uzel, na kterém se nacházíme, zvolíme jako počátek nového souřadného systému, má tedy nové souřadnice [0,0]. Proto je v první řadě třeba odečíst od souřadnic sousedních uzlů souřadnice aktuálního uzlu. Poté, co jsou od souřadnic sousedních uzlů odečteny souřadnice aktuálního uzlu, probíhá samotné přepočítávání souřadnic sousedních uzlů dle vzorce 4.3. Tyto sousední uzly a jejich nové souřadnice jsou uloženy v proměnné `new_sousedi`.

Poslední procedurou je kontrola, v jakém kvadrantu se sousední uzly s novými souřadnicemi nacházejí. To je realizováno cyklem, ve kterém se nachází podmínky, zda jsou souřadnice větší či menší než nula. Pokud se v jednom kvadrantu nachází více uzlů, rozhoduje o jejich výběru úhel, který svírají s novou osou ypsilon. V tomto případě je kandidátem zvolen uzel, jehož vektor (aktuální uzel – kandidátský uzel) svírá s osou ypsilon nejmenší úhel.

Výstupem funkce `my_routing` je matice s čísly a souřadnicemi uzlů, které byly do komunikace zařazeny. Zároveň se vykreslí topologie s červeně vyznačenou cestou jednotlivých skoků.

4.4 Chyby při realizaci směrování za použití protokolu GPSR

Největším problémem, který po realizaci protokolu nastal, byl problém s cyklickým skákáním mezi jednotlivými uzly. Tento problém nastal tehdy, když začal paket putovat po perimetru. První skok vždy proběhl v pořádku. Druhý skok díky podmínce, která zabraňuje skoku zpět na uzel, odkud paket přišel, proběhnul taky v pořádku. Problém nastal při třetím skoku, kdy protokol začne v sousedech hledat uzel nejbližší k cíli a vybere si uzel, odkud perimetrické směrování začalo, princip je naznačen na obrázku 4.4. Černě je naznačen poslední skok před perimetrickým směrováním, červeně je vyznačena smyčka, do níž se protokol dostane, šedě cesta, jakou by se měl paket správně vydat.



Obr. 4.4 Smyčka vzniklá při směrování

V literatuře [4] popisují při simulaci protokolu v simulačním prostředí NS2 podobné problémy. Na jejich vyřešení vynalezli metodu Planarized Graph (popsána v kapitole 3.2). Tato metoda měla za úkol vyřadit některé cesty k uzlům v oblastech,

kteře jsou hustě obsazené uzly. Mělo se dosáhnout toho, aby v dosahu jednoho uzlu v jednom směru byl maximálně jeden uzel. I po realizaci tohoto řešení měl protokol stále problémy s doručením. V literatuře [4] si dopomohli dodatečnou informací v hlavičce přepravovaného paketu. Tuto možnost simulační nástroj MATLAB nenabízí. Místo toho jsem použil jednoduchou metodu, kdy před určením dalšího skoku kontroluji, zda paket přes tento uzel již nepřecházel. Tato podmínka se nachází jak u výběru uzlu při greedy směrování tak i při výběru uzlů v jednotlivých kvadrantech perimetrického směrování.

Dalším problémem byl skok do slepé uličky. Tedy do míst, kam vedla pouze jedna a tatáž cesta tam i zpět, to je znázorněno na obrázku 4.4. Už z definice protokolu, kterou jsem uvedl výše, vyplývá, že uzel nesmí poslat paket zpět na uzel, od kterého právě paket přijal. V tomto případě sem však tuto podmínku ignoroval a poslat paket zpět na uzel, od kterého přišel. Zároveň jsem tento slepý uzel odstranil ze sousedů uzlu, který ho tam poslal. Takže v dalším cyklu směrování už si aktuální uzel zvolí jinou cestu, než tu se slepým uzlem. Ve většině případů při doladování programu se směrování ze slepé uličky vydalo dobrým směrem, ale v případech, kdy tento slepý uzel byl nejbližším a zároveň jediným zdánlivým můstek mezi oblastmi uzlů, paket nedorazil. To bylo způsobeno rozložením uzlů a jejich rozsahem, který byl nevhodně zvolen.

Tato skutečnost vede k dalšímu problému, který může nastat. A tím je nedoručení paketu a následný pád nebo zacyklení programu. Je tedy nutné vhodně volit dosah uzlů, dle velikosti zadaného pole a počtu uzlů. Už z vygenerované topologie je poznat, že mezi uzly existují díry, které není možné překonat. Při neexistenci cesty mezi skupinou uzlů, skončil paket vždy na uzlu, který je v dané skupině uzlů nejvíce vpravo. Tudíž ve většině případů na pravém okraji zadaného pole. Další problém nastává, je-li slepá ulice tvořena několika po sobě jdoucími uzly. S touto situací si protokol neumí taktéž poradit.

4.5 Změna polohy uzlů

Poslední funkcí použitou pro simulaci je funkce na změnu polohy jednotlivých uzlů. Vstupem pro tuto funkci je matice uzlů vygenerována funkcí `my_topo`, hodnota

posunutí v procentech a opět rozměry generovaného pole. Základem je funkce rand v simulačním programu MATLAB. Ta generuje náhodné čísla v rozmezí od nuly do jedné dle normálního rozložení pravděpodobnosti. Od těchto čísel odečteme hodnotu 0,5 a to z důvodu, aby se uzly pohnuly ve všech čtyřech směrech a ne jen v kladných hodnotách směrem v ose x a y. Tyto hodnoty se vynásobí koeficientem počítaným z rozměrů pole a zadané chyby v procentech. Nakonec jsou přičteny k hodnotám uzlů vygenerovaných na začátku simulace. Topologie se znovu vykreslí, některé uzly však vlivem posunutí končí mimo námi předem vyznačené pole. Tato skutečnost nám ale pro simulaci nevádí.

Hodnota, zadávaná v procentech, představuje posun uzlu v obou směrech souřadného systému, tedy hodnotu, o kterou se posune jak v ose x tak v ose y. Například hodnota 7% znamená maximální posun uzlu o 7% v ose x a o stejnou velikost v ose y. Výsledný maximální posun, který může nastat je tedy $7 \cdot \sqrt{2}$, což představuje hodnotu cca 10%.

Uzel se tedy může pohnout v kladném i záporném směru v obou osách. Velikost vychýlení oproti původní pozici je dána více faktory. Především zadanou velikostí posunu v procentech, dále náhodným rozdělením pravděpodobnosti pomocí funkce randn simulačního nástroje MATLAB a taktéž velikostí zadaného pole.

5 Simulace

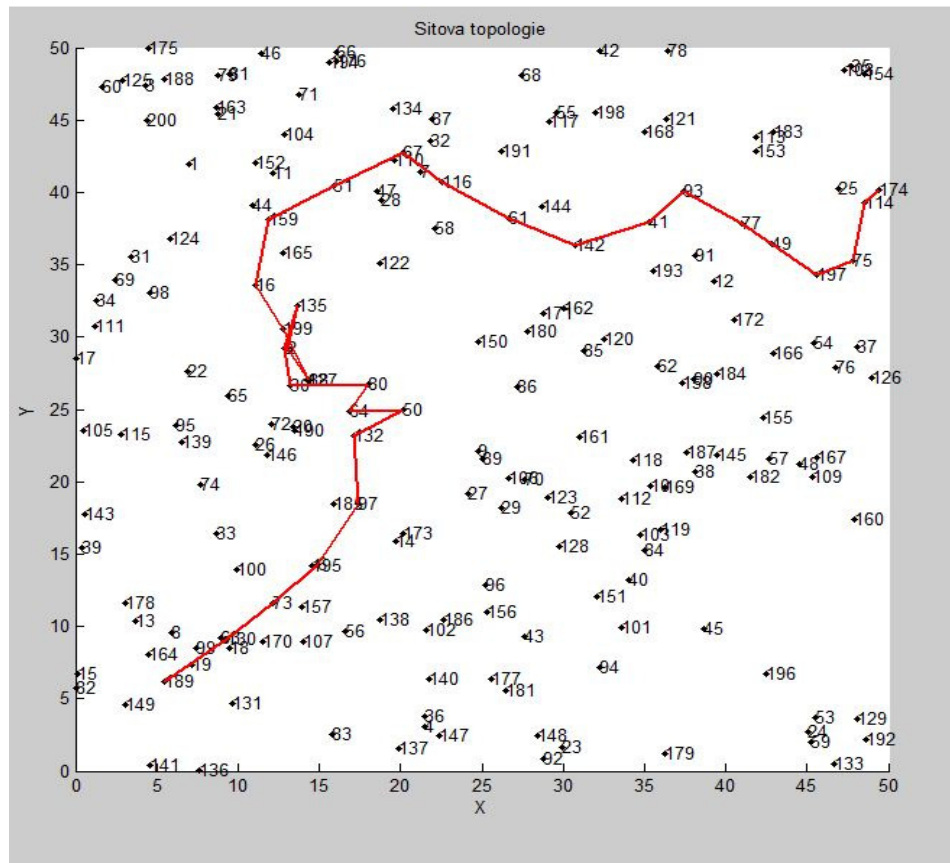
Při simulaci vlivu špatného určení polohy jednotlivých senzorů na směrovací protokol GPSR byly použity funkce, které jsou popsány v kapitole 4. Pro první simulaci byla zvolena situace s rozměrem pole 50 krát 50 metrů a počtem 200 senzorů. Tyto hodnoty sloužily jako vstupy pro funkci `my_topo`. V příkazovém okně simulačního nástroje MATLAB bylo nutné vytvořit si proměnnou a přiřadit jí výstup této funkce. To se děje napsáním dané proměnné, za ní rovná se a jméno volané funkce, která má v závorkách uvedeny vstupní hodnoty. Při zobrazení dané topologie byl dosah jednotlivých senzorů v rámci simulace zvolen 5 metrů. A to zejména z důvodu existence více cest přes toto pole. Dalším krokem bylo zavolání funkce `my_distance`, která měla jako vstupní parametry uvedeny matici uzlů a zvolený dosah pro tyto uzly. Při volání této funkce se postupovalo stejně jako při vytváření topologie, tj. vytvoření proměnné, do níž byla uložena matice sousedů. Po těchto krocích bylo již vše připraveno pro samotné směrování. Byla známa poloha jednotlivých uzlů, jejich nejbližší sousedi ve zvoleném dosahu a byla zobrazena vygenerovaná topologie, z níž se dal vybrat startovací a cílový uzel.

Pro následnou simulaci byly náhodně zvoleny dva uzly tak, aby se při doručování paketu prošlo téměř celé pole. Po zvolení výchozího uzlu s číslem 189 a cílového uzlu s číslem 174 se již mohla volat samotná funkce vykonávající směrování, tedy funkce `my_routing` s parametry uvedenými v popisu v kapitole 4.3. Tato funkce, do níž vstupuje matice původních uzlů, generuje trasu, po níž by se paket vydal, kdyby uzly znaly naprosto přesně svou pozici. Cesta paketu od výchozího uzlu k cíli je zobrazena na obrázku 5.1.

1	2	3	4	5	6	7	8	9	10
189	130	73	6	97	132	50	64	80	30
11	12	13	14	15	16	17	18	19	20
5	135	2	88	199	16	159	51	67	116
21	22	23	24	25	26	27	28	29	30
61	142	41	93	77	49	197	75	114	174

Tabulka 1 Použité uzly při doručení paketu – původní rozložení

Seznam uzlů, přes které paket k cíli přecházel, je uveden v tabulce 1. V tmavější části je pořadí skoků (výchozí uzel má číslo jedna) a ve světlejší části jsou čísla jednotlivých uzlů, které se účastnily komunikace.



Obr. 5.1 Cesta paketu při původním rozložení uzlů

Na obrázku 5.1 je názorně vidět chování protokolu GPSR, tedy kdy používá greedy směrování a kdy perimetrické směrování. Greedy směrování je použito ve většině případů, perimetrické směrování musí využít jako první uzel číslo 50, následně po něm číslo 80. Funkce výběru uzlu, který je nejvíce vpravo, jde krásně vidět u uzlů číslo 135 a 2. Oba totiž ze svých sousedů volí vzhledem k cíli uzly nejvíce napravo a tak se zdánlivě chovají iracionálně a posílají paket zpět. Uzel číslo 88 však již musí volit jiné sousedy, neboť uzly pro něj nejvíce napravo již paket přenášely, a tak jsou z výběru kandidátů vyřazeni. Uzel číslo 88 vybírá ze svých sousedů uzly, které se ještě přenosu neúčastnily, a tím nasměruje paket správným směrem. Do komunikace bylo zapojeno 30

uzlů, a to včetně výchozího a cílového. Toto zdánlivě vysoké číslo je dáno vzdáleností těchto dvou uzlů i zadaným dosahem pro všechny uzly.

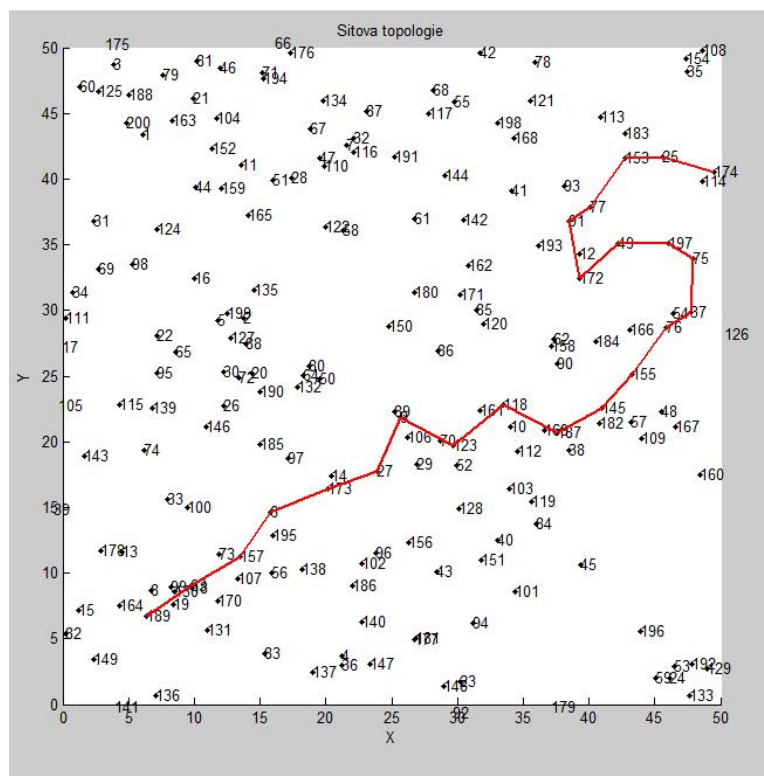
Další krok představoval posunování polohy uzlů mimo svoji původní pozici. Tento posun simuluje chybné určení vlastní polohy uzlu, kdy uzel sám o sobě prohlásí, že se nachází na souřadnicích, jenž si vypočítal a které mohou být naprosto odlišné od fyzické polohy uzlu. Tento posun se děje pomocí funkce `new_node`. Vstupy této funkce jsou matice uzlů vygenerované na začátku simulace, odchylka v procentech a rozměry pole. Tyto rozměry slouží pro přepočítání nové matice uzlů a vykreslení nové topologie s posunutými uzly. Princip funkce je uveden v kapitole 4.5.

Po vytvoření nové matice uzlů, která měla odchylku 3% odpovídající absolutní maximální hodnotě posunutí uzlu o 4,24 %, se volala opět funkce jako v prvním případě, tedy funkce pro výpočet sousedních uzlů. Jen s tím rozdílem, že místo matice s původními uzly se do této funkce vstupovalo s maticí s novými uzly. Poté již bylo vše připraveno na další volání funkce pro směřování. Tentokrát s novou maticí uzlů a novými sousedy. Výstupem byl další graf se směřováním a čísla uzlů, použitých při cestě paketu k cíli. Cesta paketu k cíli je zobrazena na obrázku 5.2, seznam uzlů zapojených do komunikace je zobrazen v tabulce 2.

1	2	3	4	5	6	7	8	9	10
189	63	157	6	173	27	9	123	118	187
11	12	13	14	15	16	17	18	19	20
145	155	76	37	75	197	49	172	91	77
21	22	23							
153	25	174							

Tabulka 2 Použité uzly při doručení paketu – chyba 4%.

Z tabulky 2 je již patrná výrazná změna cesty oproti původní topologii, kdy byly uzly přesně na svých místech. V tmavější části je pořadí skoků (výchozí uzel má číslo jedna) a ve světlejší části jsou čísla jednotlivých uzlů, které se účastnily komunikace. Do komunikace zapojilo výrazně méně uzlů než ve výchozím stavu. S cílovým a výchozím uzlem se komunikace účastnilo 22 uzlů.



Obr. 5.2 Cesta paketu k cíli při chybě 4%

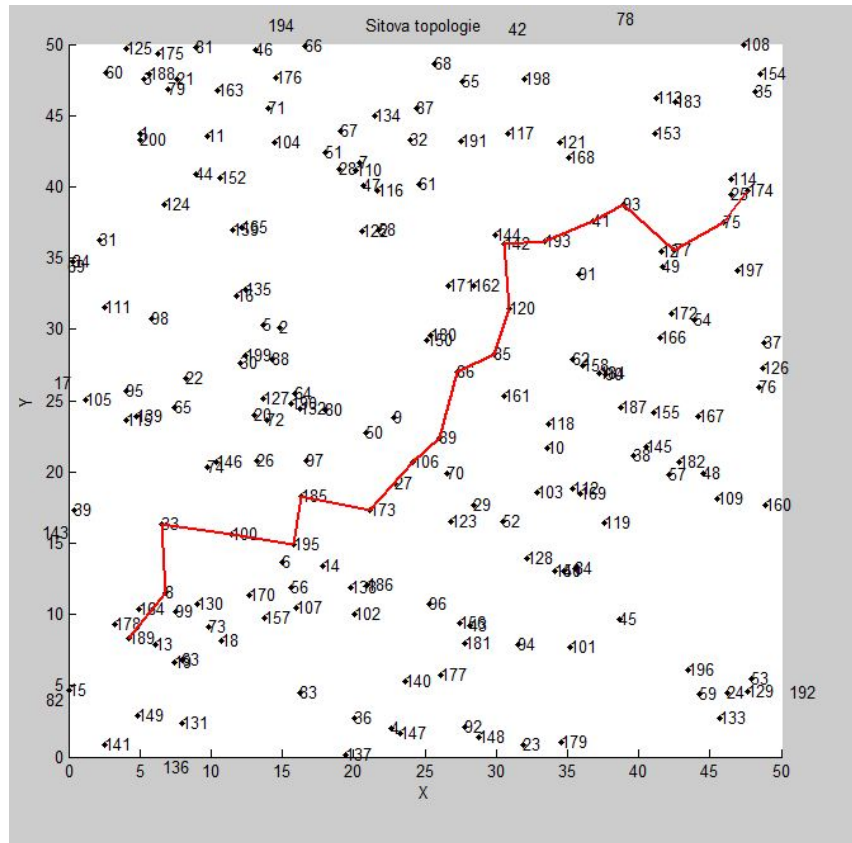
Z obrázku 5.2 je patrné, že již při výběru prvních uzlů, přes které se skáče, dochází k výraznému ovlivnění cesty paketu k cíli. V původní topologii došlo při prvotních skocích k nasměrování paketu na „horní“ cestu. Kdežto zde se volila cesta spíše přes spodní část uzlů. To má za následek menší počet skoků k cíli a celkově kratší cestu.

Další testovanou topologií byla topologie sestavená z uzlů, jež byly oproti původní topologii posunuty o 5%. To odpovídá maximální hodnotě posunutí uzlu o 7% oproti původní hodnotě. Opět byly volány stejné funkce jako v předchozím případě. Tentokrát se vstupními hodnotami odpovídajícími tomuto posunutí. Seznam uzlů, které se účastnily komunikace, je uveden v tabulce 3. Výsledná topologie s vyznačenou cestou paketu k cíli je uvedena na obrázku 5.3.

1	2	3	4	5	6	7	8	9	10
189	8	33	100	195	185	173	106	89	86
11	12	13	14	15	16	17	18	19	
85	120	142	193	41	93	77	75	174	

Tabulka 3 Použité uzly při doručení paketu – chyba 7%.

Z tabulky 3 je patrné, že komunikace se zúčastnilo ještě méně uzlů než v předchozím případě. V tomto případě to bylo včetně výchozího a cílového jen 19 uzlů.



Obr. 5.3 Cesta paketu k cíli při chybě 7%.

Z obrázku 5.3 je opět patrný vliv greedy směrování. Volba uzlů v tomto případě je téměř ideální. Opět je zvolena výhodnější cesta přes spodní uzly. Jediný skok mimo je na uzel číslo 33. Pokud by jeho okolní uzly byly blíže, zvolila by se pro směrování cesta přes horní uzly, tak jako při první simulaci, a výsledná trasa by byla podstatně delší. Zde se opět ukázalo, jaký vliv má výběr prvních skoků, kde se uplatňuje greedy směrování.

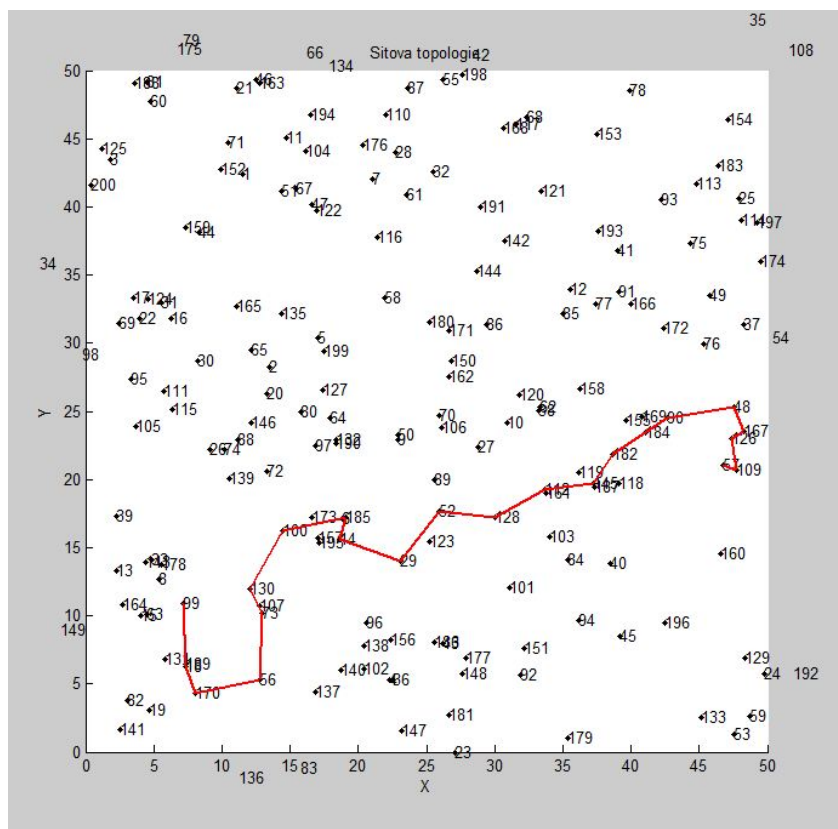
Při dalším kroku byla zvolena pro novou matici uzlů hodnota 7%. To odpovídalo maximálnímu posunu uzlu o cca 10% oproti původní poloze. Cesta paketu k cíli je zobrazena na obrázku 5.4, seznam uzlů zapojených do komunikace je zobrazen v tabulce 4.

Další simulace byla provedena pro posunutí o 10%, což odpovídá maximálnímu posunu cca 15%. Při volání směrovací funkce však došlo k chybě, paket nebyl doručen. Cesta paketu do slepé uličky je zobrazena na obrázku 5.5, seznam uzlů zapojených do komunikace je zobrazen v tabulce 5.

1	2	3	4	5	6	7	8	9	10
189	99	18	170	56	73	130	100	185	14
11	12	13	14	15	16	17	18	19	20
29	52	128	112	145	182	90	48	167	126
21	22	23							
109	57	109							

Tabulka 5 Použité uzly při doručení paketu – chyba 14%.

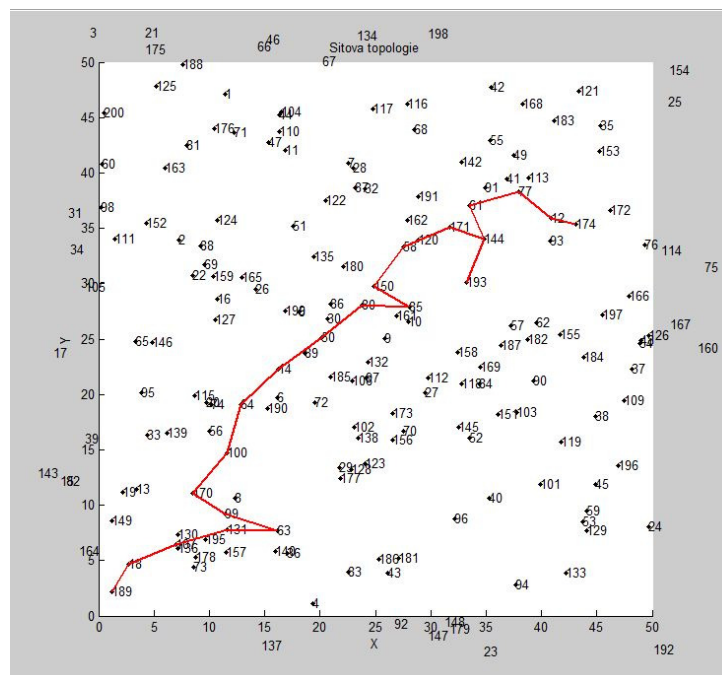
Tabulka 5 je zde pouze pro orientaci, kudy byl paket směřován. Vypovídací hodnotu, proč nebyl paket doručen, má zejména obrázek 5.5.



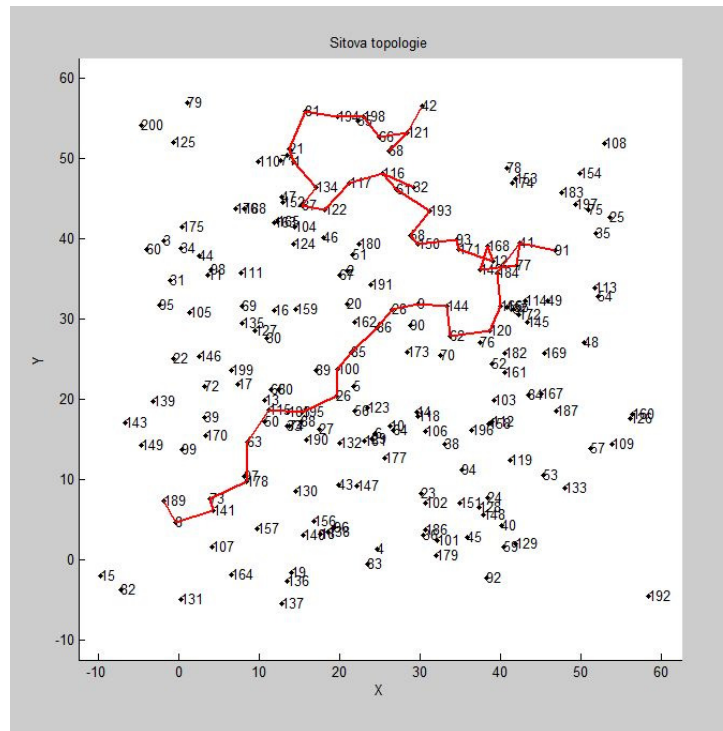
Obr. 5.5 Cesta paketu k cíli při chybě 14%.

Z obrázku 5.5 je patrné, že paket musel hned od počátku putovat po perimetru, aby obešel prázdné místo mezi uzly s číslem 99 a 30. Po úspěšném obkroužení prázdného místa přešel paket opět do módu greedy směřování. Kritické místo bylo na uzlu 52. Protože jediná cesta k cíli vede přes uzel 89. Ale jelikož při greedy směřování rozhoduje vzdálenost, byl zvolen pro další skok uzel s číslem 128. Tato cesta však paket zavedla do slepé uličky, která začíná na uzlu číslo 90. Na tomto příkladu se opět ukázalo, že hlavní vliv má greedy směřování před směřováním po perimetru.

Poslední simulace jsem provedl pro chyby 15% (maximální odchylka od původní pozice 21%) a pro 20% (maximální odchylka od původní pozice 28%). Při obou těchto simulacích již nedošlo k doručení paketu. Na obrázku 5.6 je cesta paketu při chybě 15% a na obrázku 5.7 při chybě 20%. Oba jsou zde umístěny pro demonstraci chování směrovacího protokolu GPSR, neboť obsahují skoky do slepých ulic, složených jen z jednoho uzlu. S těmi si vytvořený kód umí poradit. Na obrázku 5.6 je touto slepou ulicí uzel číslo 193. Na obou simulacích jde též vidět, že se paket přiblížil co nejbližší k cíli a jen díky neexistujícímu spojení, neboť se uzly nacházely daleko od sebe, nemohl být doručen. Pro obě simulace zde není umístěna tabulka s uzly použitými ke komunikaci.



Obr. 5.6 Cesta paketu k cíli při chybě 21%.



Obr. 5.7 Cesta paketu k cíli při chybě 28%.

6 Závěrečné hodnocení simulace

Především ze simulací, kde byla změněna poloha uzlů jen nepatrně, vyplývá velká závislost směrovacího protokolu Greedy perimetr stateless routing (GPSR) na změně polohy uzlů. Prvotní předpoklad před simulací byl takový, že změna polohy uzlu se projeví zřejmě nejvýrazněji na vybírání vhodného uzlu při perimetrickém směrování a chybná lokalizace se projeví několika skoky navíc. Simulace však prokázaly, že největší vliv na výslednou cestu paketu celou bezdrátovou sensorovou sítí mají první skoky prováděné greedy směrováním. Právě tyhle skoky rozhodují, zda bude použita nejkratší možná cesta, nebo zda se paket dostane k cíli zbytečnou oklikou. Jak již naznačila simulace, při níž byla poloha uzlů vůči prvotní topologii změněna jen o necelých 5%, směrovací protokol GPSR je závislý i na drobné změně (nebo špatném určení) polohy uzlu. Při směrování s původními polohami uzlů se paket vlivem greedy směrování dostal zbytečně na horní uzly, kde pak směrovací protokol složitě hledal cílový uzel. Hlavně se projevil vliv chyby při prvních skocích, které mnohdy můžou paket zavést na zbytečné okliky nebo dokonce do slepých míst.

Další poznatek ze simulací byl opět vliv greedy směrování při jediném skoku na výslednou cestu paketu celou sítí. Zejména pokud existuje jediná možná cesta k cíli, kterou se lze vydat. Ale při využití greedy směrování je pro následující skok vybrán uzel, který je blíže k cíli, a ne uzel přes který vede jediné možné spojení. Tady je další místo, kde může špatná lokalizace velice výrazně promluvit do celého procesu směrování. Pokud by se však nějakou zpětnou vazbou dala ověřit vhodná cesta a ta byla následně implementována například do hlavičky paketu, nemusela by mít chyba v lokalizaci v tomto případě takové katastrofální důsledky.

Závěr

V úvodu této práce byl uveden přehled základních lokalizačních technik. Tyto techniky mají vliv na celkovou přesnost určení polohy senzoru. Od polohy senzoru se odvíjí spousta dalších navazujících procesů.

Se znalostí polohy jednotlivých senzorů pracuje směrovací protokol Greedy perimetr stateless routing (GPSR), jehož princip byl uveden v kapitole 2. Dalším procesem, který vyžaduje znalost polohy jednotlivých uzlů, je hierarchická agregace dat. Její princip je naznačen v kapitole 3. Na základě teoretických znalostí jsem usoudil, že pro návrh simulace bude vhodné pracovat se směrovacím protokolem GPSR místo s hierarchickou agregací dat. Neboť byl předpoklad, že u tohoto protokolu se více projeví chyba lokalizace a to zejména při perimetrickém směrování.

V kapitole 4 jsem popsal návrh simulace. Jedná se především o popis jednotlivých funkcí, které jsou při simulaci zapotřebí. Jako první je uveden popis funkce vytvářející topologii a vlastní polohu jednotlivých uzlů. Dále popis funkce, která vypočítává dostupné sousedy. Neboť protokol GPSR vyžaduje pro svou funkci od jednotlivých uzlů znalost dostupných sousedů. Nejdůležitější z popsaných funkcí, které jsem vytvořil, je funkce `my_routing` realizující směrování. Tato funkce implementuje algoritmus popsaný v literatuře [4]. V uvedené literatuře však nebyly popsány důležité mechanismy při výběru vhodného uzlu při perimetrickém směrování. Proto jsem v této kapitole uvedl i svůj návrh řešení tohoto problému. Ke konci kapitoly jsem též uvedl problémy, na které jsem narazil během testování a simulace směrovacího protokolu GPSR. Především na problém vzniku smyček při směrování paketu a problém slepých ulic.

V kapitole 5 jsem uvedl simulaci vlivu znalosti polohy na směrovací protokol GPSR. Je zde uvedena vzorová testovací topologie, na níž je provedeno směrování z náhodně zvolených uzlů, které se nacházejí na opačné straně pole. Jsou zde uvedeny postupně výsledky simulací pro posunutí uzlů o 4, 7, 10, 14, 21 a 28% s následným popisem jednotlivých výsledků. U každé simulace je uvedena tabulka uzlů, které jsou zapojeny do komunikace.

V poslední kapitole jsem shrnul poznatky ze simulace. Především velkou závislost směrovacího protokolu GPSR i na nepatrnou změnu polohy. Především pak skutečnost, že největší vliv na směrovací proces má greedy směrování při výběru prvních uzlů, přes které posílá paket k cíli. Tyto první skoky totiž ovlivní celou cestu paketu sítě a v konečném důsledku mohou znamenat použití většího počtu uzlů, než by bylo potřeba při správném prvotním nasměrování. Dalším důležitým poznatkem bylo zjištění, že i když existuje cesta k danému cíli, díky greedy směrování ji nemusí protokol najít, místo toho se vydá na nejbližší uzel k cíli a tím se dostat do slepé ulice.

SEZNAM POUŽITÉ LITERATURY

- [1] ABDELZAHER, T.F., BLUM, B.M., HE, T., HUANG, C., STANKOVIC, J.A. Range-free localization schemes for large scale sensor networks. In *International Conference on Mobile Computing and Networking (MobiCom)*, 2003.
- [2] ALHMIEDAT, T.A., YANG, S.H. A Survey: Localization and Tracking Mobile Targets through Wireless Sensors Network. In *The Eighth Annual PostGraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting (PGNET)*, 2007.
- [3] CHRAIBI, Y. *Localization in wireless sensor networks*. Master's thesis, Royal Institute of Technology, Švédsko, 2005.
- [4] KARP, B., KUNG, H. T. GPRS: Greedy Perimeter Stateless Routing for Wireless Network, In *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, Boston, MA, 2000.
- [5] ŘÍHA, K. *Pokročilé zpracování obrazu*. Brno: Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav Telekomunikací, 2007.
- [6] SEKINE, M., SEZAKI, K. Hierarchical aggregation of distributed data for sensor network, In *TENCON 2004 IEEE Region 10 Conference*, Chiang Mai, Thajsko, 2004. ISBN 0-7803-8560-8