

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Damir Arbula

**RASPODIJELJENI ALGORITAM  
ZA LOKALIZACIJU U  
NEUSIDRENOJ MREŽI  
ODREĐIVANJEM SMJERA  
DOLASKA SIGNALA**

MAGISTARSKI RAD

Zagreb, 2008

Magistarski rad je izrađen na Zavodu za automatiku i računalno inženjerstvo  
Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Mentor: Prof.dr.sc. Zdenko Kovačić

Magistarski rad ima 90 stranica  
Magistarski rad br.: 03-Af-1/2004-Z

**Povjerenstvo za ocjenu u sastavu:**

1. Prof. dr. sc. Stjepan Bogdan, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu – predsjednik
2. Prof. dr. sc. Zdenko Kovačić, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu – mentor
3. Prof. dr. sc. Mario Essert, Fakultet strojarstva i brodogradnje Sveučilišta u Zagrebu

**Povjerenstvo za obranu u sastavu:**

1. Prof. dr. sc. Stjepan Bogdan, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu – predsjednik
2. Prof. dr. sc. Zdenko Kovačić, Fakultet elektrotehnike i računarstva Sveučilišta u Zagrebu – mentor
3. Prof. dr. sc. Mario Essert, Fakultet strojarstva i brodogradnje Sveučilišta u Zagrebu

**Datum obrane:** 30. listopada 2008.

# Sadržaj

<b>1. UVOD</b>	1
1.1. Ad hoc mreže . . . . .	1
1.1.1. Bežične mreže osjetila . . . . .	2
1.1.2. Istraživački izazovi . . . . .	5
1.2. Motivacija . . . . .	7
1.3. Pregled rada . . . . .	9
<b>2. LOKALIZACIJA ČVOROVA</b>	10
2.1. Osnovna obilježja lokalizacije . . . . .	10
2.1.1. Centralizirani i raspodijeljeni algoritmi . . . . .	10
2.1.2. Relativna i apsolutna lokalizacija . . . . .	12
2.1.3. Izravne, iterativne i statističke metode . . . . .	12
2.2. Mjerne metode . . . . .	14
2.2.1. Mjerenje udaljenosti . . . . .	14
2.2.2. Mjerenje azimuta . . . . .	19
2.2.3. Mjerenje profila signala . . . . .	21
2.3. Izvori pogrešaka . . . . .	21
2.3.1. Pogreške mjerenja . . . . .	21
2.3.2. Pogreške zbog karakteristika mreže i čvorova . . . . .	22
2.4. Pregled relevantnih algoritama . . . . .	23
2.4.1. Lokalizacija korištenjem udaljenosti među čvorovima . . . . .	23
2.4.2. Ad hoc sustav pozicioniranja . . . . .	26
2.5. Definicija problema . . . . .	27
<b>3. TEORIJA KRUTOSTI</b>	28
3.1. Mreža – graf – formacija točaka . . . . .	28
3.2. Udaljenost kao ograničenje . . . . .	29
3.2.1. Problem nejednakosti između krutosti i globalne krutosti . . . . .	30
3.3. Azimut kao ograničenje . . . . .	32
3.3.1. Paralelna formacija točaka . . . . .	32
3.3.2. Problem globalne orijentacije . . . . .	33

---

3.4. Mrežne strukture . . . . .	34
3.5. Operacije nad mrežnim strukturama . . . . .	37
3.5.1. Proširivanje . . . . .	37
3.5.2. Spajanje podgrozdova . . . . .	37
<b>4. PRIJEDLOG METODE LOKALIZACIJE</b>	<b>40</b>
4.1. Metoda lokalizacije inicijalnih podgrozdova . . . . .	40
4.1.1. Iterativna triangulacija . . . . .	41
4.2. Robusna angulacija . . . . .	43
4.3. Put spajanja grozdova u globalni grozd . . . . .	44
4.3.1. Optimalno stablo . . . . .	44
4.3.2. Idealna težina bridova . . . . .	45
4.4. Spajanje grozdova . . . . .	49
4.4.1. Opis metoda spajanja . . . . .	49
4.4.2. Izbor podgrozdova i metode spajanja . . . . .	51
4.4.3. Međuspajanje . . . . .	52
<b>5. RASPODIJELJENI ALGORITAM</b>	<b>54</b>
5.1. Raspodijeljeno računalno okruženje . . . . .	55
5.1.1. Entiteti . . . . .	55
5.1.2. Komunikacija . . . . .	55
5.1.3. Ponašanje entiteta . . . . .	57
5.1.4. Pretpostavke – ograničenja primjene algoritma . . . . .	58
5.1.5. Metode projektiranja algoritma . . . . .	59
5.2. Analiza projektiranog algoritma . . . . .	60
5.2.1. Inicijalizacija algoritma . . . . .	60
5.2.2. Odabir vođe i konstrukcija korijenskog optimalnog stabla . . . . .	61
5.2.3. Mjerenje, izmjena podataka i lokalizacija . . . . .	66
5.2.4. Spajanje podgrozdova . . . . .	67
<b>6. REZULTATI</b>	<b>68</b>
6.1. Kriteriji evaluacije algoritma . . . . .	68
6.1.1. Korijen srednje kvadratne pogreške lokacije . . . . .	69
6.1.2. Cramér-Rao limit . . . . .	69
6.2. Utjecaj postavnih parametara . . . . .	71
6.2.1. Postavni parametri mreže . . . . .	72
6.2.2. Povezanost . . . . .	73
6.2.3. Veličina mreže . . . . .	74
6.3. Lokalizacija uz prisutnost mjernog šuma . . . . .	75
6.3.1. Usporedba različitih funkcija težine optimalnih stabala . . . . .	75
6.3.2. Usporedba sa centraliziranom verzijom algoritma . . . . .	76
6.3.3. Usporedba sa Cramér-Rao limitom . . . . .	76

---

<b>7. ZAKLJUČAK</b>	78
7.1. Planovi budućeg istraživanja. . . . .	79
<b>LITERATURA</b>	81
<b>PRILOZI</b>	85
Prilog 1. Robusna angulacija. . . . .	85
Prilog 2. Pseudokod algoritma Mega-merger. . . . .	87
Prilog 3. Pseudokod algoritma za lokalizaciju inicijalnih grozdova . . . . .	90

# 1. Uvod

Nakon što su se dugo vremena pripremali tehnološki, ali i sociološki temelji, danas se može reći kako pojam sveprisutnog računarstva (engl. *ubiquitous computing*) više ne označava nešto čemu se teži ili nešto što se tek treba realizirati, već je ono u svom punom opsegu prisutno u životu suvremenog čovjeka. Danas se od tehnologije, a pogotovo informacijske tehnologije, očekuje da bude dostupna doslovno na svakom mjestu i u svako vrijeme. Svjedoci smo vremena u kojem se teži biti dijelom globalne informacijske mreže bez obzira na vrijeme i prostor, a kako bi se ostvario cilj potrebno je razviti infrastrukturu koja prvenstveno mora zadovoljavati potrebe korisnika.

## 1.1. Ad hoc mreže

Osim trenutno prevladavajućih infrastrukturnih mreža poput interneta ili mreža za fiksnu i mobilnu telefoniju, javlja se velika potreba za takozvanim ad hoc mrežama u kojima infrastrukturu čine sami uređaji koji je i koriste. Takve mreže nazivaju se još i privremene zbog njihove prirode da mogu nastati bilo gdje i bilo kada i trajati samo onoliko koliko je to korisnicima potrebno. S obzirom na nepostojanje fiksne infrastrukture mogle bi se nazvati i jedinim pravim bežičnim mrežama.

Ideja koja stoji iza ad hoc mreža je komunikacijski sustav sastavljen od autonomnih uređaja bez podržavajuće fiksne infrastrukture ili nekog oblika središnje administracije. To je ujedno i veliki iskorak iz okvira tradicionalnih infrastrukturnih bežičnih mreža, koje su u osnovi fiksne, a koriste mogućnosti bežične komunikacije samo za proširenje dostupnosti, ograničenu mobilnost i sl. prema potpuno bežičnim sustavima.

Iako je u samom pojmu bežičnih komunikacija sadržana ideja o slobodi kretanja, mobilnosti i općenito ideja o odbacivanju ograničenja koje donosi konvencionalan medij poput žice, puno je čimbenika koji zahtijevaju potpunu kontrolu da bi ta ideja bila realizirana i u praksi. Ad hoc mreže, sa svim svojim prednostima, ali i problemima, upravo su na gornjoj margini trenutnog razvoja i dostignuća u bežičnim komunikacijama i kao takve predstavljaju oblik bežične komunikacije u svom trenutno najslabijem obliku.

Danas postoji stroga hijerarhija korisnika prema vremenu pristupa novim tehnolo-

gijama koja počinje od uskog kruga vojnih primjena za čiju implementaciju su potrebni veliki napori i financijska ulaganja. Zatim slijedi ključni dio u kojem svoju ulogu preuzima profesionalna upotreba u raznim granama ljudske djelatnosti u kojima se, kroz niz specifičnih primjena, tehnologija proširuje i prilagođava. U toj fazi karakterističan je razvoj proporcionalan financijskoj iskoristivosti i komercijalnom uspjehu. Naposljetku tehnologija se približava običnom korisniku, zbog veličine tržišta cijena pada, a skup aplikacija se dodatno širi i tehnologija se sa većim ili manjim uspjehom integrira u samo tkivo društva i postaje njegov neizostavan dio.

Tehnologija ad hoc bežičnih mreža slijedila je malo drugačiji put razvoja [1] pri čemu prve primjene ad hoc načina umrežavanja datiraju još iz 1972. godine i projekta PRNet (Packet Radio Network) pokrenutog u vojne svrhe od strane agencije DARPA (Defence Advance Research Projects Agency), a svrha mu je bilo iskorištavanje pogodnosti tehnologije komutacije paketa (engl. *packet switching*) kao što su dijeljenje odn. bolja iskoristivost frekvencijskog pojasa te spremi-i-prosljedi protokola za usmjeravanje prometa (engl. *routing protocol*).

Početak ovog desetljeća kroz eksponencijalno širenje WLAN i Bluetooth tehnologija ad hoc mreže ušle su i u široku primjenu u društvu. Pogodnosti koje nude običnom korisniku i mnoštvo najrazličitijih primjena potenciralo je njihov uspjeh na tom tržištu i omogućilo razvoj kroz simultani pad cijena i povećanje tržišta.

Ipak jedan korak u razvoju je preskočen, pa usprkos svim naporima koje ulažu i industrija i istraživačka zajednica kroz široki spektar istraživanja usmjerenih u to područje u svim aspektima ad hoc mreža, od temeljnih tehnologija do same primjene, činjenica je da za naprednu, pouzdanu i financijski opravdanu profesionalnu primjenu u raznim granama ljudske djelatnosti još uvijek postoje otvoreni problemi i pitanja koje je potrebno riješiti.

Aktivno istraživanje ad hoc mreža počinje godine 1995. unutar zajednice nazvane IETF (Internet Engineering Task Force) što je rezultiralo njihovim osnivanjem radne grupe koja se bavi standardizacijom protokola za usmjeravanje prometa sa funkcionalnošću podobnom za primjenu unutar statičkih i dinamičkih topologija mreže koje se javljaju zbog mobilnosti čvorova i ostalih dinamičkih čimbenika u takvim mrežama. Nešto kasnije osnovan je i Ad Hoc Wireless Networking / Computing Consortium sa ciljem objedinjavanja interesa i napora industrije i istraživača u primjeni ad hoc umrežavanje u širokom spektru aplikacija.

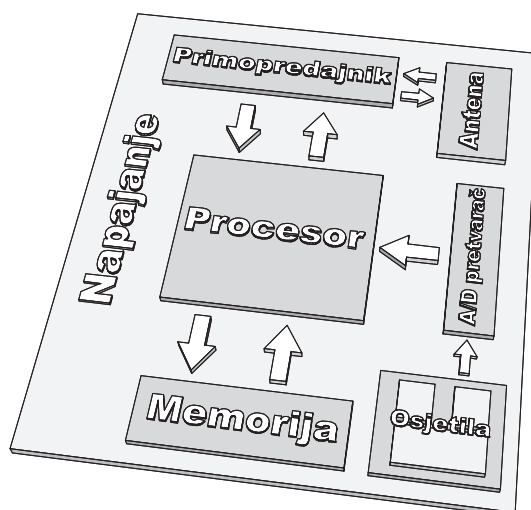
### 1.1.1. Bežične mreže osjetila

Jedna od primjena u kojima bi ad hoc umrežavanje moglo uskoro učiniti svoj veliki proboj kod svih skupina korisnika je primjena u raspodijeljenom mjerenju, nadgledanju i praćenju. Za tu primjenu su potrebna elektronička osjetila (engl. *sensors*) koja mogu detektirati i mjeriti razne parametre i pojave u okolišu u kojem se nalaze,



poput temperature, vlažnosti, zvuka, svjetla, prisutnosti kemikalija itd. Posebna vrsta ad hoc mreža koje su opremljene osjetilima naziva se bežična mreža osjetila (engl. *wireless sensor network* - WSN).

Tipični čvor bežične mreže osjetila sastoji se od procesora za obradu podataka koji može biti realiziran u raznim varijantama kao mikrokontroler, mikroprocesor, programirajući logički kontroler (FPGA) i sl. zatim od vanjske memorije, primopredajnika za komunikaciju sa ostalim čvorovima, skupa potrebnih osjetila zajedno sa analogno digitalnim pretvaračima i napajanja. Prikaz tipične arhitekture čvorova dan je na slici 1.1.



Slika 1.1. Tipična arhitektura čvorova.

Primjer suvremenog komercijalnog čvora je čvor Imote2<sup>®</sup> tvrtke Crossbow<sup>®</sup> [2] koji se sastoji od procesora niske potrošnje Marvell PXA271 sa mogućnošću regulacije potrošnje variranjem napona i frekvencije u širokom rasponu od 13 do 416 MHz. Isti čip uključuje i 256 kB SRAM, 32 MB SDRAM i 32 MB FLASH memorije i sadrži mnoštvo ulazno izlaznih sučelja za povezivanje sa različitim tipovima osjetila i uređajima poput kamere i primopredajnika. Dodatno sadrži i koprocesor sa instrukcijama za obradu digitalnih signala. U čvoru se nalazi radio primopredajnik Texas Instruments CC2420 koji radi u nelicenciranom ISM pojasu od 2.4 GHz koristeći IEEE 802.15.4 protokol za niže slojeve komunikacijskog modela uz brzinu prijenosa od 250 kbit/s i domet od 30 metara korištenjem integrirane antene. Čvor je veličine 36x48x9mm i težine 12 g bez napajanja koje je izvedeno korištenjem 3 standardne AAA baterije.

Potencijalni uspjeh ovih mreža temelji se na pristupu koji uvjetuje postojanje velikog broja malih, jeftinih, potrošivih i energetski nezahtjevnih čvorova raspoređenih na određenom području sa komunikacijskim, računalnim i osjetilnim mogućnostima. Streloviti razvoj komunikacijske i računalne tehnologije i trend minimizacije i integracije najrazličitijih tipova osjetila približava tipični čvor razini na kojoj je moguće

realizirati sve prednosti ovakvog pristupa.

Bežičnim mrežama osjetila pretpostavljen je niz teoretskih karakteristika koje ih u većoj ili manjoj mjeri izdvajaju od ostalih ad hoc mreža. One su:

- samoorganizirajuće mogućnosti
- komunikacija kratkog dometa i usmjeravanje prometa s više skokova
- veliki broj i gustoća jeftinih, potrošivih čvorova uz univerzalno prisutnu mogućnost suradnje
- otpornost na česte promjene topologije zbog gubitaka signala i zastoja u radu čvorova
- ograničene računalne i memorijske mogućnosti te vrlo ograničeni izvori energije.

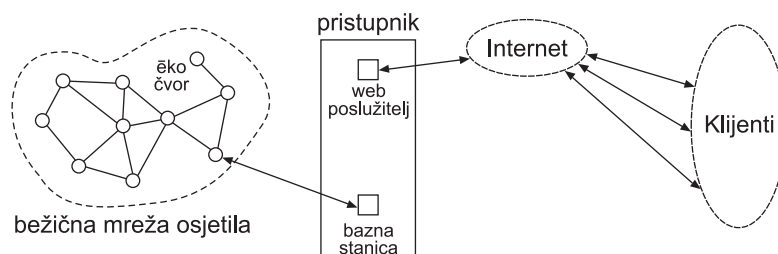
Ako bi se držali ovih pretpostavki i pokušali projektirati mrežu za određenu primjenu naposljetku bi naišli na velike praktične probleme. Primarni razlog tome je činjenica kako trenutno dostupna tehnologija još uvijek ne omogućava pretpostavljenu veličinu, cijenu i fleksibilnost čvorova, pa ih se stoga ne može smatrati ni jeftinima ni potrošivima. Ovakvo razilaženje između teorije i prakse prisutno je i u istraživanjima, s jedne strane teži se univerzalnom pristupu s navedenim pretpostavkama i naprednim tehnikama koje će biti u stanju zadovoljiti najširi spektar mogućih primjena. S druge strane postoje specifični praktični pristupi s malim brojem čvorova i što jednostavnijim algoritmima koji su orijentirani na pojedinu aplikaciju.

Jasno je da će do pojave mreža i čvorova koji mogu zadovoljiti navedene karakteristike i koji se mogu sagledati kroz univerzalan koncept rada, biti potreban prijelazni period. U tom periodu će se postepeno razvijati aplikacije koje iskorištavaju trenutno dobavljive čvorove, drugim riječima, manji broj većih i skupljih čvorova u kojima su primarni ciljevi praktična upotrebljivost, robusnost, jednostavnost korištenja i održavanja te dugi period autonomije. U ovom radu se pretpostavlja upotreba malog do srednjeg broja skupljih čvorova koji su orijentirani na sam zadatak ili manji skup ciljnih primjena te se ne pokušavaju zadovoljiti neke univerzalne teoretske pretpostavke.

Područja primjene bežičnih mreža osjetila su široka [3], od vojnih, preko ekoloških i biomedicinskih, ali i ona su kao i sama tehnologija još u razvoju i nedovoljno istražena. Neki od primjera inženjerske primjene su telemetrija u vozilima, robotiziranim industrijskim postrojenjima, kontrola klimatizacije u tzv. inteligentnim stanovima i uredima, praćenje dobara u proizvodno-prodajnom lancu, nadgledanje i analiza slabih točaka arhitektonskih struktura poput mostova i nebodera te u inteligentnim sustavima prijevoza. Zatim postoje primjene u biološkim istraživanjima, medicini, ekologiji te u inteligentnoj poljoprivredi i stočarstvu. Primjeri su seizmička

mjerenja, detekcija požara i poplava na velikim područjima, praćenje ugroženih životinjskih vrsta i njihovih staništa, kvalitete voda i širenje onečišćenja. U medicini se takve mreže mogu koristiti u kontinuiranom nadgledanju zdravstvenog stanja pacijenata na višoj razini, primjerice za praćenje infekcija i analizu krvi npr. detekciju tromboza. Naposljetku tu su i vojne primjene koje sežu od nadzora nad sredstvima, oružjem, ljudstvom i opremom, pa sve do aktivnog praćenja kretanja neprijateljskih postrojbi i zaštitu vitalnih objekata uz pomoć polja osjetila koji mogu detektirati razne vrste uljeza.

Jedna od reprezentativnih komercijalnih primjena je precizna agrikultura korištenjem *ěKo* sustava tvrtke Crossbow [2]. Sustav se sastoji od bežičnih čvorova sa osjetilima za vlažnost tla i temperaturu te pristupnika (engl. *gateway*) sa ugrađenim web poslužiteljem. Čvorovi dodatno podržavaju osjetila vlažnosti listova, solarnog zračenja, smjera i brzine vjetra, pritiska i protoka za sustav navodnjavanja. Napajanje je izvedeno uz pomoć punjivih baterija i solarnih ćelija te u nedostatku sunčevog svjetla omogućena je autonomija od 3 mjeseca. Ovakav pristup agrikulturi omogućava povećanje i dosljednost prinosa, višu kvalitetu proizvoda, smanjenje ulaznih troškova i naravno povećanje profitabilnosti. Funkcionalni izgled ove mreže uklapa se u okvire tipičnih bežičnih mreža osjetila za nadgledanje okoliša.

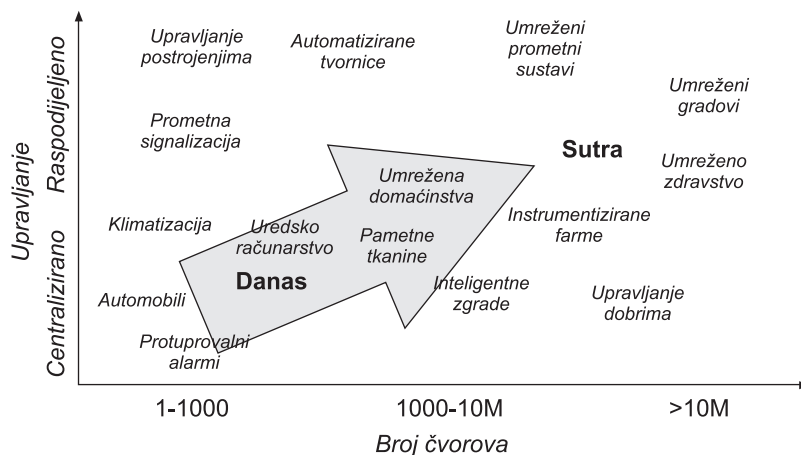


Slika 1.2. Topologija *ěKo* sustava.

Glavni dugoročni trendovi razvoja bežičnih mreža osjetila su povećanje broja čvorova i povećana raspodijeljenost takvih sustava. Kako se trend bude razvijao mogu se očekivati sve veće mogućnosti primjene od kojih su neke prikazane na odgovarajućem grafu na slici 1.3 [4]. S druge strane to će dovesti i do pojave mnoštva novih hardverskih i softverskih problema i izazova koji će zahtijevati snažan istraživački napor.

### 1.1.2. Istraživački izazovi

Jedan od najvažnijih istraživačkih problema u ad hoc mrežama, a koji je u ovom dijelu posebno istaknut jer ima utjecaj na sve druge aspekte projektiranja mreže, je **upravljanje energijom**. Izvor ovog problema leži u činjenici da su uređaji u ad hoc mrežama autonomni te da za svoj rad koriste baterije odn. ograničene i u većoj ili manjoj mjeri neobnovljive izvore energije. Zbog toga je nužno pronaći najpovoljnije strategije kako bi se potrošnja energije svela na minimum u okvirima normalnog rada



Slika 1.3. Trendovi u mrežama osjetila.

mreže. Dvije najvažnije činjenice koje se mogu iskoristiti pri analizi i rješavanju problema upravljanja energijom su [4]:

1. Energija potrebna za slanje jednog byte-a podataka je više redova veličine veća od energije potrebne za njegovu obradu
2. Poruka poslana u više manjih skokova s manjom snagom odašiljanja je energetski učinkovitija od one poslana u jednom skoku s većom snagom odašiljanja.

Osim upravljanja energijom mogu se izdvojiti sljedeća područja u kojima će biti potrebno riješiti mnoge probleme:

**Kvaliteta usluge** – potrebno je osigurati pouzdan i brz pristup podacima u mreži. Parametri koji utječu na kvalitetu usluge su propusnost mreže, brzina, kašnjenje i pouzdanost.

**Robusnost** – potrebno je razviti tehnike kojima će se osigurati rad mreže u slučaju zastoja čvorova ili prekida veza. Robusnost se osigurava kroz redundanciju, komunikaciju i zajedničku obradu podataka.

**Skalabilnost** – kako mreža osjetila može sadržavati veliki broj čvorova potrebno je osigurati što manji utjecaj veličine i gustoće mreže na performanse.

**Sigurnost** – pitanje sigurnosti u bežičnim mrežama izraženo je kroz samu prirodu odašiljanja podataka, a proteže se od mogućnosti pasivnog prisluškivanja do aktivnog glumljenja dijelova zaštićene mreže, mijenjanja, brisanja stvaranja lažnih poruka i sl. Takva narušena sigurnost velika je prepreka u komercijalnim primjenama bežičnih mreža. Razumijevanje načina napada na mrežu je prvi korak u izboru odgovarajućeg sigurnosnog rješenja. Danas postoje dva tipa mehanizma

za povećanje sigurnosti: zaštitni i otkrivajući. Zaštitni se temelje na kriptografiji po ključu, a najveći problem u tom pristupu predstavlja sigurnost distribucije ključa. Otkrivajućim mehanizmima zadatak je otkriti pokušaj prodiranja uljeza u mrežu.

**Implementacija** – najveći problem pri implementaciji mreža osjetila danas je njihova cijena koja je reda veličine 100 USD po čvoru, što je previše ako se sagledaju pretpostavke na kojima se temelji teorija rada tih mreža. Napredak u integriranju sklopova zasigurno vodi prema cilju, a to je mali, jeftini i potrošivi čvor, no ipak dvije komponente mogu se označiti kao ključne, a to su baterija i antena kod kojih je problem minijaturizacije veliki istraživački izazov.

**Raspodijeljena obrada signala** – mnogi zadaci zahtijevaju suradnju više čvorova, za koje je neophodno razviti protokole kojima će se provoditi učinkovita koordinacija, lokalna izmjena podataka i izgradnja hijerarhijske strukture. Jedan od takvih protokola opisan je u ovom radu.

**Reprogramirljivost** – tijekom rada mreže može se javiti potreba za reprogramiranjem čvorova mreže. Za sada ne postoji protokol kojim bi se takav zadatak obavio pouzdano.

**Usmjeravanje prometa** – zadatak protokola za usmjeravanje prometa je povezivanje dva čvorova preko virtualnih putova koji vode preko jednog ili više međučvorova. U ad hoc mrežama ovaj zadatak je dodatno otežan činjenicom da se topologija vrlo često mijenja, pa takvi protokoli moraju biti robusni i održavati mrežu uvijek povezanom. Dodatni zahtjev proizlazi iz problema učinkovitog upravljanja energijom pri čemu usmjeravanje prometa optimalnim putovima ima veliku ulogu. Mnogi postojeći protokoli nisu prikladni za tip prometa i zahtjeve kakvi postoje u bežičnim mrežama osjetila.

**Sinkronizacija i lokalizacija** – podaci o točnom vremenu i apsolutnoj ili relativnoj lokaciji čvorova mreže u određenim primjenama mogu biti jedni od najvažnijih podataka. Primjerice pri simultanom mjerenju i upravljanju vrijeme je od ključnog značaja dok podatak o točnoj lokaciji može dati potrebnu vrijednost sirovim podacima dobivenim od čvorova. I jedan i drugi problem su vrlo složeni i tema su mnogih istraživanja.

## 1.2. Motivacija

Opisivanjem značaja i potencijalnih područja primjene ad hoc bežičnih mreža i posebice bežičnih mreža osjetila zadani su okviri motivacije za provedeno istraživanje. U prethodnom odjeljku navedeni su mnogi problemi i izazovi koji traže rješenje kako bi

nova tehnologija mogla zaživjeti i realizirati svoj puni potencijal. Jedan od tih izazova je i lokalizacija koja predstavlja središnji problem ovog rada.

Ključna informacija kojom se u potpunosti iskorištava vrlo bitan prostorni kontekst mjerenja u bežičnim mrežama osjetila je lokacija čvorova. Primjerice, pri nadzoru određenog prostora mjerna informacija bez informacije o točnoj lokaciji mjerenja nema potrebnu vrijednost. Nadalje, u svim ostalim ad hoc mrežama lokacija čvorova može biti korištena kroz usluge koje mreža pruža poput kontrole pristupa, sustava naplate i različitih informacijskih sustava ili kroz protokole koji služe samoj mreži za što učinkovitiji rad poput geografskog usmjeravanja prometa.

Unutar mreže osjetila čvorovi mogu odrediti svoju lokaciju isključivo korištenjem i obradom određenih podataka. Do tih podataka čvorovi dolaze, u pravilu, mjerenjem različitih fizičkih veličina koja su podložna raznim pogreškama. Nadalje, povezanost čvorova u mreži može biti takva da s obzirom na izabrani modalitet mjerenja teoretski nije moguće odrediti jedinstvenu lokaciju pojedinih čvorova. Kako bi cijela mreža bila lokalizirana, s obzirom na neki koordinatni sustav, potrebno je da određeni broj čvorova pozna svoje lokacije u tom sustavu. Ti čvorovi se nazivaju sidra (engl. *anchors*). Naposljetku potrebno je posvetiti pažnju da lokalizacija ne bude energetska zahtjevna, odnosno da se računalno, a posebice komunikacijsko opterećenje, zadrži na što nižem nivou.

U ovom radu želi se projektirati raspodijeljeni algoritam koji će se koristiti mjerenjima kuta dolaska signala te će raditi u mreži koja ne pretpostavlja postojanje sidara (neusidrena mreža) niti čvorova koji poznaju svoju globalnu orijentaciju.

Kao što se može zaključiti algoritam lokalizacije mora biti takav da učinkovito rješava mnoge izazove i izbjegava potencijalne izvore pogrešaka, odnosno smanjuje njihov utjecaj.

Minimizacija utjecaja mjernog šuma je postignuta korištenjem potprostorne metode opisane u [5] za lokalizaciju prvih susjeda. Zatim je za međusobno spajanje tako lokaliziranih susjedstva korištena optimalna metoda sa aspekta najmanje kvadratne pogreške opisana u [6]. Potencijalna mogućnost pojave višestrukih rješenja analizirana je i riješena korištenjem teorije krutosti grafova opisane u [7]. Projektirani algoritam nalazi lokacije svih čvorova koji mogu biti jednoznačno lokalizirani unutar istog koordinatnog sustava, a čvorovi znaju kojem sustavu pripadaju i identitet ostalih čvorova u tom sustavu. Naposljetku algoritam je raspodijeljen kako bi se što više ujednačilo komunikacijsko i računalno opterećenje i time osigurala veća energetska i funkcionalna autonomija.

## 1.3. Pregled rada

U ovom radu raspodijeljeni algoritam predstavljen je kroz opis svih faza koje su vodile njegovoj realizaciji. Na kraju su predstavljeni i komentirani rezultati obavljenih ispitivanja i simulacija.

U 2. poglavlju dan je kratak pregled aktualnih metoda i algoritama za rješavanje problema lokalizacije ad hoc bežičnih mreža te je obavljena konačna definicija problema koji će se rješavati u ovom radu. Zatim je u 3. poglavlju revidirana teorija krutosti grafova, upotrijebljena za definiranje novih mrežnih struktura koje se koriste u 4. poglavlju kako bi se što lakše opisala korištena metoda lokalizacije. Metoda je pretočena u raspodijeljeni algoritam u 5. poglavlju, u kojem je kao preduvjet uspješne realizacije metode dan opis raspodijeljenog računalnog okruženja. Rad algoritma analiziran je kroz niz simulacija čiji rezultati su predstavljeni i komentirani u 6. poglavlju.

## 2. Lokalizacija čvorova

Kao što je već u uvodu napomenuto, mogućnost određivanja lokacije čvorova u bežičnim mrežama, a posebice u ad hoc i bežičnim mrežama osjetila je preduvjet kako za mnoge primjene tih mreža tako i za njen što kvalitetniji rad i potpuno iskorištavanje ograničenih resursa. Interes za istraživanja u području lokalizacije bežičnih mreža posljednjih godina su dosegla vrhunac. S obzirom na spomenute probleme i izazove u radu samih ad hoc mreža, mnoštvo parametara i razne implementacije i primjene, problem i pristup problemu lokalizacije je gotovo u svakom istraživanju definiran na jedinstven način. Mnogo je parametara koji definiraju problem. U ovom poglavlju pokušati će se u najkraćim crtama obuhvatiti i spomenuti sve važnije, a one najrelevantnije za ovaj rad detaljnije opisati.

Poglavlje započinje sa klasifikacijom metoda lokalizacije prema njihovima najvažnijim obilježjima, zatim se posebno obrađuje jedno od najvažnijih obilježja, a to su mjerne metode. Slijedi kratak pregled najvažnijih čimbenika koji utječu na kvalitetu lokalizacije i opis nekoliko algoritama koji su najbliži metodi lokalizacije koja se koristi u središnjem dijelu ovoga rada. Na kraju definiran je sam problem koji se želi riješiti i osnovna obilježja rješenja.

### 2.1. Osnovna obilježja lokalizacije

Problemi i algoritmi lokalizacije mogu se klasificirati na razne načine, a u sljedećim odjeljcima opisana su neka obilježja koja najviše utječu na izvedbu algoritma.

#### 2.1.1. Centralizirani i raspodijeljeni algoritmi

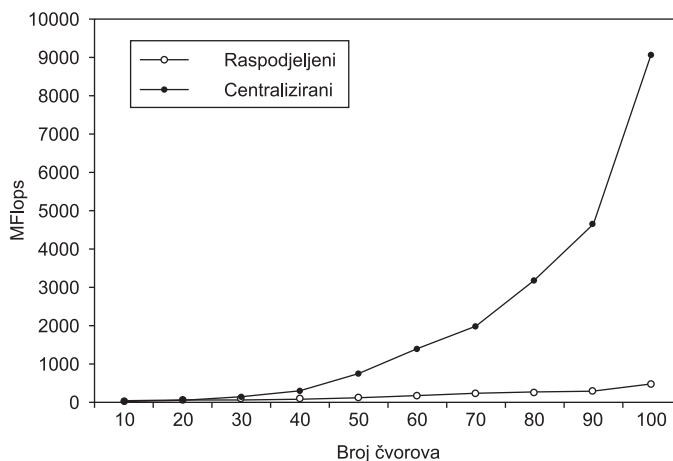
Centralizirani pristup podrazumijeva prenošenje svih potrebnih podataka u centralni čvor u kojem se obavlja estimacija lokacija ostalih čvorova. Ovakav pristup je izvedbeno prilično jednostavan, ali i osjetljiv s obzirom da ovisi o radu samo jednog čvora. S druge strane, raspodijeljenim algoritmima estimacija se izvodi na način da svi čvorovi mreže zajedničkom obradom i izmjenama podataka obavljaju samolokalizaciju.



Usporedba ova dva pristupa može biti učinjena s aspekta točnosti, računalnih zahtjeva i energetske zahtjeva. Kada se govori o energetskim zahtjevima, isključivo se promatra količina komunikacija potrebna za obavljanje zadatka. Također je bitno naglasiti, kako je raspodijeljeno rješenje što se tiče izvedbe, uvijek teže realizirati od centraliziranog rješenja zbog složenog odnosa lokalnog i globalnog ponašanja algoritma. Raspodijeljeni algoritam koji je lokalno optimalan, može se ponašati loše na globalnoj razini, a postoji i mogućnost propagacije grešaka.

Što se tiče energetskih zahtjeva, odnosno količine komunikacija kod centraliziranih algoritama, najveći udio služi za prosljeđivanje mjernih podataka u centralni čvor. Pri tome, čvorovi koji su blizu centralnog čvora predstavljaju uska grla i trpe najveće energetske i komunikacijske opterećenje te je njihova autonomija najugroženija. Računalni zahtjevi centraliziranog algoritma svode se na zahtjeve prema centralnom čvoru koji mora biti u mogućnosti adekvatno odgovoriti na njih, dok kod raspodijeljenih algoritama opterećenje snose svi čvorovi, u većoj ili manjoj mjeri podjednako.

Ako se uspoređuje točnost lokalizacije, onda su centralizirani algoritmi u prednosti, ali centralni čvor mora biti u mogućnosti podnijeti i značajna povećanja računalnog opterećenja za povećani broj čvorova. Na slici 2.1 je prikazana usporedba ukupnog računalnog opterećenja centralizirane i raspodijeljene verzije jednog algoritma lokalizacije [8], iz koje se vidi kako centralizirana verzija puno lošije reagira na povećanje broja čvorova koje treba lokalizirati od raspodijeljene. S druge strane, raspodijeljeni se algoritmi često oslanjaju samo na podskup podataka koja su na raspolaganju centraliziranom algoritmu, pa u biti čine tek aproksimaciju rada centralizirane verzije.



**Slika 2.1.** Usporedba ukupnog računalnog opterećenja za raspodijeljenu i centraliziranu verziju algoritma lokalizacije [8].

Raspodijeljeni algoritmi mogu se podijeliti u ovisnosti o tome dali se koriste čvorovima s poznatom lokacijom – sidrima. Oni koji ih koriste funkcioniraju tako što mjere odnos prema njima (azimut, udaljenost), a onda koriste te podatke kako bi odredili svoju poziciju i u nekim slučajevima i sami postali sidra za ostale čvorove. Skupina

raspodijeljenih algoritama koja djeluje u neusidrenim mrežama temelji se na optimizaciji lokacija širinom cijele mreže. Prva podgrupa pokušava početne grubo estimirane lokacije dodatno optimizirati u skladu s izmjerenim podacima, a druga podgrupa se koristi spajanjem inicijalno lokaliziranih preklapajućih dijelova mreže u jedan globalni koordinatni sustav.

Sve ove karakteristike, a i ispitivanja u praksi [9] dovela su do zaključaka da je raspodijeljena verzija robusnija, energetski učinkovitija i bolje odgovara na promjene u topologiji. Nabrojane prednosti čine raspodijeljene algoritme poželjnima u rješavanju problema lokalizacije te je iz tog razloga algoritam projektiran u sklopu istraživanja ovog rada, raspodijeljen.

### 2.1.2. Relativna i apsolutna lokalizacija

Ova klasifikacija ima izvorište u načinu na koji se definira lokacija. Lokacija u pravilu označava poziciju objekta u prostoru u odnosu na određenu referencu koja može biti zadana na različite načine. Relativna lokalizacija estimira pozicije čvorova ukoliko nema referentnih točaka izvan same mreže, odnosno estimirane koordinate čvorova  $(x, y)$  su valjane samo unutar relativnog koordinatnog sustava mreže ili grupe čvorova u mreži. Apsolutna lokalizacija podrazumijeva postojanje sidara koji znaju svoju lokaciju u odnosu na neku referencu izvan mreže, primjerice geografsku širinu i dužinu ili koordinatni sustav prostorije i sl.

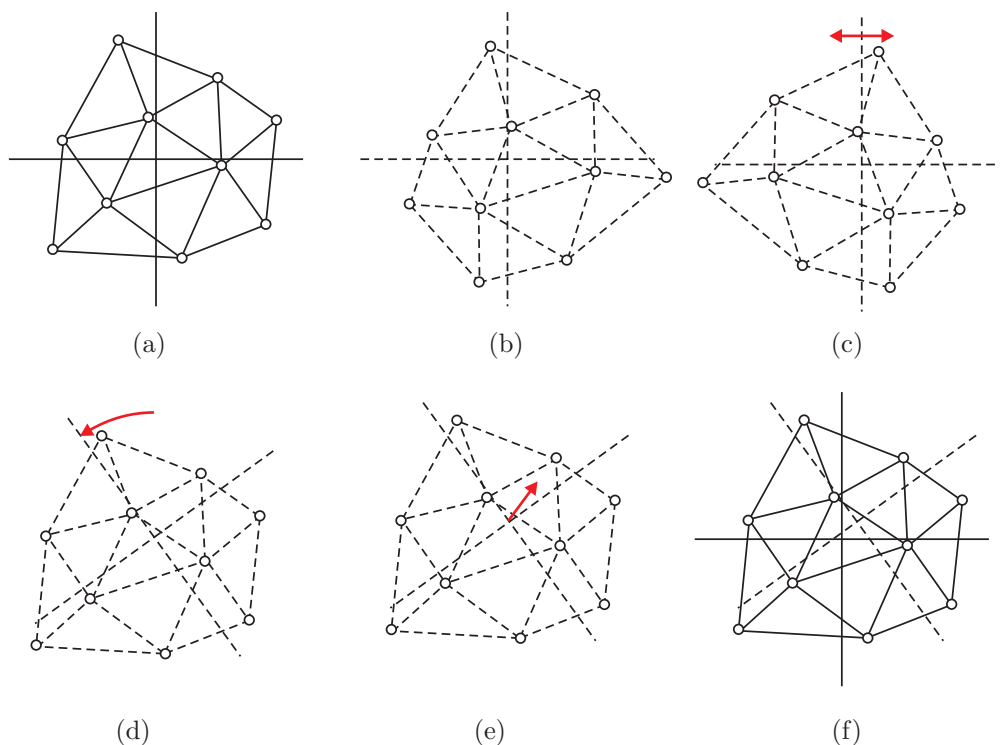
Koordinatni sustav u mreži sa mjerenjem udaljenosti može se dovesti u vezu sa apsolutnim koordinatnim sustavom jednostavnom translacijom, rotacijom i ukoliko je potrebno refleksijom kao što je to prikazano na slici 2.2. Skaliranje u ovom slučaju nije potrebno, jer izmjerene udaljenosti između čvorova predstavljaju stvarne udaljenosti.

Minimalan broj sidara u dvodimenzionalnoj mreži u kojoj se lokacije određuju mjerenjem udaljenosti je 3, dok za mrežu u kojoj se mjere azimuti među čvorovima taj broj je 2. Razlog je nemogućnost refleksije u odnosu na os koja prolazi kroz dva sidra. Više o tome u 3. poglavlju u kojem se detaljnije obrađuje teorija krutosti grafova i krute transformacije grafova.

### 2.1.3. Izravne, iterativne i statističke metode

Do estimiranih lokacija može se doći *izravno*, nizom računskih operacija koje rezultiraju konačnim rješenjem. *Iterativne* metode započinju obično izravnim proračunom inicijalnih lokacija, te se zatim optimiziranjem složenih nelinearnih težinskih funkcija u određenom broju iteracija pokušava povećati kvaliteta estimacije. *Statističkim* metodama modeli mjerenja uključuju se u konačni izračun rješenja koje je u tom slučaju za svaki čvor dano u obliku funkcije gustoće vjerojatnosti.

Računalni zahtjevi algoritma vrlo su bitna stavka u bežičnim mrežama zbog ve-



**Slika 2.2.** Apsolutno lokalizirana mreža (a), relativno lokalizirana mreža i proizvoljni koordinatni sustav (b) koji refleksijom (c), rotacijom i (d) translacijom (e) dovodi se do njegove apsolutne lokacije (f).

likog broja čvorova i njihovih ograničenih energetske i računalnih mogućnosti. Kod iterativnih metoda, velike pogreške u inicijalnim lokacijama, veliki broj čvorova, pa time i dimenzija prostora u kojem se nalaze rješenja može dovesti do konvergiranja u lokalne minimume. Iako izravne metode nemaju takvih problema i računalno, a posebice komunikacijski i energetski, su manje zahtjevni, rezultati su im očekivano slabiji od iterativnih metoda.

Algoritmi koji se temelje na statističkim metodama imaju dodatnu mogućnost da u proces estimacije lokacije izravno uračunaju statistički model mjerne metode. Klasične statističke metode su estimacija s maksimalnom vjerojatnošću (engl. *maximum likelihood* - *ML*), maksimum *a posteriori* estimacija (engl. *maximum a posteriori*) i Monte Carlo lokalizacija (MCL) koja se temelji na upotrebi čestičnih filtara (engl. *particle filter*).

## 2.2. Mjerne metode

Jedno od najvažnijih obilježja, koje čini osnovu svih metoda lokalizacije je mjerenje fizičkih odnosa među čvorovima, pa je iz tog razloga izdvojeno u ovo posebno poglavlje. Mjerenja se obavljaju različitim tehnikama koristeći poznate karakteristike propagacije radiofrekvencijskih, optičkih i akustičkih signala. Najčešće se mjeri jačina, vrijeme i smjer dolaska signala. S obzirom na odabranu mjernu tehniku moguće je odrediti različite modalitete odnosa među čvorovima, a to su udaljenost i azimut susjednih čvorova te profil signala.

### 2.2.1. Mjerenje udaljenosti

Udaljenost među čvorovima u komunikacijskom dometu određuje se mjerenjem vremena dolaska signala (engl. *time of arrival* – *TOA*) ili mjerenjem jačine signala (engl. *received signal strength* – *RSS*). U prvom slučaju koristi se poznata brzina propagacije u mediju, a u drugom slučaju poznata atenuacija signala u prostoru. Nekoliko najčešće korištenih tehnika opisano je u sljedećim odjeljcima.

#### Vrijeme dolaska signala

Mjerenje vremena dolaska signala svodi se na određivanje vremena propagacije signala u jednom smjeru, od čvora  $i$  do čvora  $j$  ili vremena obilaska (engl. *roundtrip time*) signala, od čvora  $i$  do čvora  $j$  i natrag. U prvom slučaju prikazanom na slici 2.3(a) početak i kraj intervala mjere različiti čvorovi, pa je preduvjet točnog mjerenja točna sinkronizacija lokalnih satova čvora  $i$  i čvora  $j$ , što je sam po sebi složen zadatak koji zahtjeva vrlo točne satove i mehanizme sinhronizacije, s obzirom na veliku brzinu i vrlo mala vremena propagacije radio signala.

Puno pogodniji način mjerenja vremena propagacije je mjerenje kružnog obilaska kako je prikazano na slici 2.3(b) pri čemu se vrijeme mjeri u istom, polaznom čvoru, pa nije potrebna sinkronizacija. U ovom slučaju jedina nepoznanica je vrijeme  $t_d$  potrebno za prijem, obradu i ponovno odašiljanje signala od strane čvora  $j$  koje se mora oduzeti od ukupnog vremena. To vrijeme polazni čvor ili pozna je prethodnom kalibracijom ili ga čvor  $j$  šalje u poruci.

Metoda kojom se mjeri vrijeme propagacije u jednom smjeru, koja ne zahtjeva sinkronizaciju satova, prikazana je na slici 2.3(c), a temelji se na razlici brzina propagacije radio signala i ultrazvučnog signala [10]. Simultanim slanjem radio i ultrazvučnog signala čvora  $i$  određišni čvor  $j$  može izmjeriti razliku vremena dolaska tih dvaju signala, a zatim poznavanjem brzina propagacije odrediti udaljenost. Ova metoda daje izuzetno točne rezultate, ali je uvjetovana opremljenošću čvorova ultrazvučnim primopredajnicima što povećava cijenu, veličinu i energetske zahtjeve samog

čvora. Također postoji problem višestaznog širenja ultrazvučnog signala pogotovo u zatvorenim prostorima.

Određivanje trenutka dolaska signala izvodi se poopćenom kros-korelacijom (engl. *generalized cross-correlation*) između primljenog signala  $s_i$  i poznatog signala koji se odašilje  $s_j$ .

$$\rho_{i,j}(\tau) = \frac{1}{T} \int_0^T s_i(t) s_j(t - \tau) dt. \quad (2.1)$$

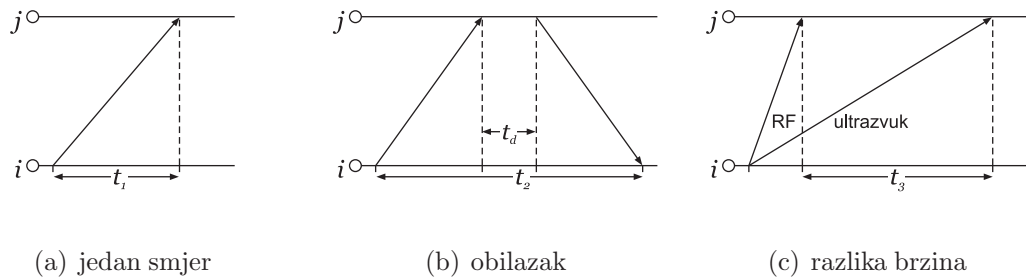
Poopćena verzija kros-korelacije koristi spektar snage signala i šuma kako bi pojačala komponente sa velikim signalom i malim šumom i prigušila one sa velikim šumom.

Analizom (Cramér Rao) donjeg limita za varijancu estimacije vremena propagacije dobiva se sljedeći izraz:

$$\text{var}(TOA) \geq \frac{1}{8\pi^2 B T_s F_c^2 \text{SNR}} \quad (2.2)$$

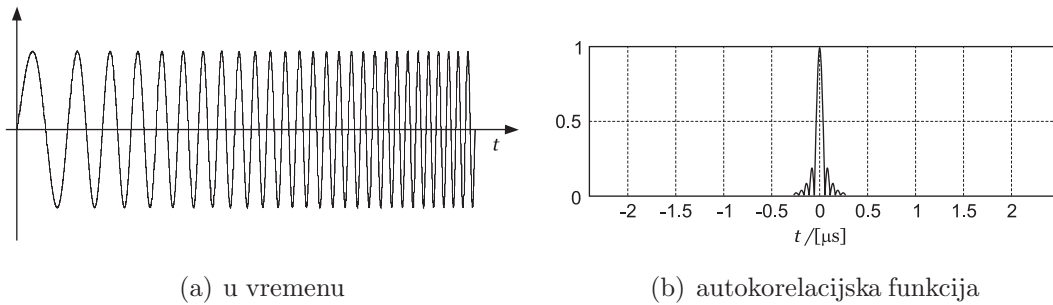
gdje je  $T_s$  duljina trajanja signala,  $B$  širina frekvencijskog pojasa signala (engl. *bandwidth*) koja mora biti puno manja od centralne frekvencije  $F_c$ , a SNR označava odnos signala i šuma. Iz dobivenog rezultata može se zaključiti koji parametri signala i na koji način utječu na smanjenje varijance estimacije vremena propagacije.

Primjer uspješne implementacije mjerenja udaljenosti uz pomoć određivanja vremena obilaska signala je sustav nanoLOC, tvrtke Nanotron technologies [11]. Osnovicu sustava sačinjava integrirani primopredajnik nanoLOC TRX (NA5TR1), koji ima mogućnost komunikacije i vrlo preciznog određivanja međusobne udaljenosti od istovrsnih primopredajnika. Pri tome se koristi *chirp* modulacijom sa proširenim spektrom (engl. *Chirp Spread Spectrum – CSS*). CSS modulacija se temelji na *chirp* signalima – širokopojasnim sinusoidalnim signalima čija frekvencija se mijenja linearno u vremenu, kao što je prikazano na slici 2.4(a). Karakteristike modulacija proširenog spektra su: otpornost na smetnje i višestazno širenje signala, veliki domet (nekoliko stotina metara), velika brzina prijenosa podataka (125 kbps – 2 Mbps) i malo vrijeme



**Slika 2.3.** Mjerenje vremena propagacije.

latencije. S aspekta mjerenja udaljenosti najvažnije svojstvo proizlazi iz pogodnog oblika autokorelacijske funkcije chirp signala koja je prikazana na slici 2.4(b) Takav oblik omogućava precizno određivanje vremena dolaska signala, a posljedično tome i udaljenosti koja za nanoLOC sustav iznosi manje od 1 metra.



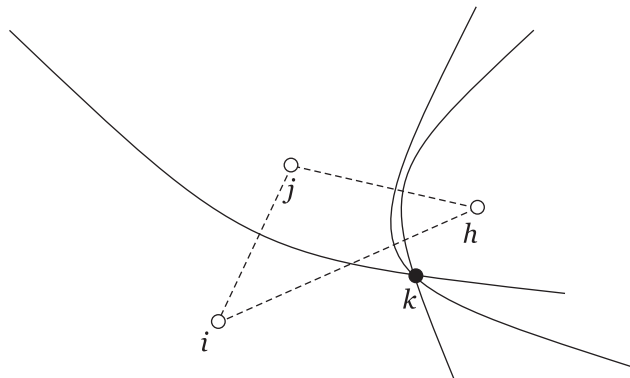
Slika 2.4. Chirp signal.

### Razlika u vremenu dolaska signala

Metoda mjerenja razlike u vremenu dolaska signala (engl. *time difference of arrival – TDOA*), omogućava izravno određivanje lokacije čvora  $k$ . Ako su čvorovi  $i$  i  $j$  vremenski sinkronizirani, odredili su vremena  $t_i$  i  $t_j$  u kojima je isti signal stigao do njih te poznaju svoje lokacije tada vrijedi sljedeća jednakost:

$$\Delta t_{ij} \triangleq t_i - t_j = \frac{1}{c} (\|\mathbf{r}_i - \mathbf{r}_k\| - \|\mathbf{r}_j - \mathbf{r}_k\|), \quad i \neq j \quad (2.3)$$

gdje su  $\mathbf{r}_i$ ,  $\mathbf{r}_j$  i  $\mathbf{r}_k$  lokacije čvorova  $i$ ,  $j$  i  $k$ , a  $\|\cdot\|$  označava euklidsku udaljenost. Za svaki par čvorova, jednadžba (2.3) opisuje po jednu hiperbolu, a lokacija čvora  $k$  nalazi se na njihovom sjecištu kako je prikazano na slici 2.5 [12].



Slika 2.5. Određivanje lokacije na sjecištu hiperbola.

Problem nepreciznih mjerenja može se riješiti upotrebom estimatora s maksimalnom vjerojatnošću ili korištenjem metode kojom se hiperbole aproksimiraju asimptotama.

### Jačina primljenog signala

Jedan od parametara signala koji najizrazitije mijenja svoju vrijednost sa udaljenošću je njegova jačina. Odnos primljene snage (engl. *received signal strength* – *RSS*)  $P_r(d)$  i odašlane snage signala  $P_t$  ovisan je o valnoj duljini  $\lambda$ , dobicima antena odašiljača i prijammnika  $G_t$  i  $G_r$  te o udaljenosti prijammnika i odašiljača  $d$  prema sljedećoj formuli:

$$\frac{P_r(d)}{P_t} = \frac{G_t G_r \lambda^2}{(4\pi)^2 d^2}. \quad (2.4)$$

Jednadžba (2.4) vrijedi u slobodnom prostoru i kao takva je idealizirana. Zbog vrlo složenog međudjelovanja okoline i signala kroz mnogobrojne refleksije, ogibe i raspršenja, određivanje udaljenosti podložno je velikim sustavnim pogreškama koje se ne mogu anulirati velikim brojem mjerenja. Uz veliku mjernu nesigurnost, dodatni problem predstavlja povećanje mjerne pogreške sa udaljenošću zbog slabljenja signala odnosno zbog sve većeg utjecaja šuma.

Prednost ovakvog pristupa je njegova mala cijena, jer svaki čvor već posjeduje odašiljač, prijammnik i mogućnost mjerenja jačine signala.

Ova metoda mjerenja udaljenosti ispitana je u ad hoc lokalizacijskom sustavu (AH-LoS), opisanom u [9], gdje je izvršena i izravna usporedba sa TOA metodom. Pokazalo se kako je greška kod RSS metode za dva reda veličine veća, ali i kako su hardverski i energetski zahtjevi puno manji, slijedom čega su autori zaključili kako bi se ove dvije metode mogle nadopunjavati u ovisnosti o gustoći mreže.

### Detekcija signala - mjerenje broja komunikacijskih skokova

Najjednostavnija metoda mjerenja udaljenosti jačinom signala je detekcija signala, odnosno određivanje da li je jačina signala veća od praga prijammnika. Ovakvo mjerenje može se nazvati i 1-bitno mjerenje jačine signala. Na taj način može se dobiti jedino indikacija susjedstva i blizine (engl. *proximity*) dvaju čvorova, odnosno može se zaključiti sa velikom vjerojatnošću da je njihova udaljenost manja od maksimalnog komunikacijskog dometa  $R$ . U gušćim mrežama s velikim brojem čvorova, pokazalo se da je moguće, koristeći samo broj komunikacijskih skokova (engl. *hop-count*), dobiti predodžbu o relativnoj udaljenosti čvorova u mreži.

Ako se sa  $h_{ij}$  označi broj komunikacijskih skokova između čvorova  $i$  i  $j$ , tada se može zaključiti da je udaljenost čvorova  $d_{ij} < h_{ij}R$ . Nadalje, ukoliko je poznat i očekivani broj susjeda po čvoru  $n_s$ , tada se može izvesti i bolja procjena udaljenosti

jednog komunikacijskog skoka [3]:

$$d_{skok} = R \left( 1 + e^{-n_s} - \int_{-1}^1 e^{-(n_s/\pi) \arccos t - t\sqrt{1-t^2}} dt \right). \quad (2.5)$$

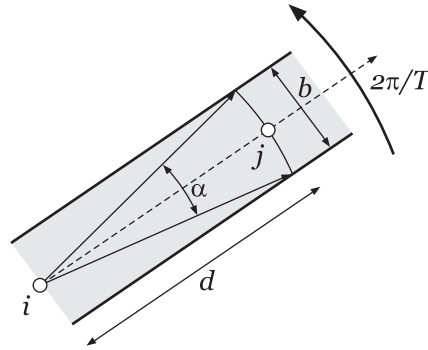
Aproksimacija duljine skoka (2.5) pokazala se korisnom za vrijednosti očekivanog broja susjeda po čvoru između 5 i 15.

Ovom metodom se može sa sigurnošću odrediti čvorove koji su u fizičkoj blizini drugih čvorova, ali obrnuto ne vrijedi, odnosno nepostojanje veze između dva čvora ne mora uvijek značiti i njihovu fizičku udaljenost, jer vezu može onemogućiti primjerice fizička prepreka između dva čvora.

Kako se samim mjerenjem pristaje na prosječnu pogrešku veličine polovice komunikacijskog dometa, moguće je zaključiti da se ovom metodom može koristiti samo u slučajevima kada nije potrebna velika preciznost lokalizacije.

### Metoda svjetionika

Jedna od nestandardnih metoda mjerenja udaljenosti opisana u [13] naziva se metoda svjetionika. Izvorišni čvor  $i$ , emitira rotirajuću svjetlosnu zraku konstantnom kutnom brzinom  $\omega$  prema slici 2.6.



Slika 2.6. Princip mjerenja udaljenosti uz pomoć metode svjetionika.

Odredišni čvor  $j$ , mjerenjem vremena  $t_1$ , u kojem detektira svjetlosnu zraku čvora  $i$ , te mjerenjem vremena  $T$  između dvije detekcije, može odrediti linearnu i kutnu brzinu zrake. Iz tih dvaju brzina se polumjer, odnosno udaljenost među čvorovima, određuje prema sljedećoj jednadžbi:

$$d \approx \frac{b}{2 \sin(\alpha/2)} = \frac{b}{2 \sin(\pi t_1/T)} \quad (2.6)$$

Praktični problem predstavlja zahtjev da zraka bude paralelna odnosno da ima konstantnu širinu  $b$ . Čak i malo odstupanje od tog uvjeta može na većim udaljenostima dovesti do velikih pogrešaka. Ipak, pokazalo se da se te pogreške mogu uspješno

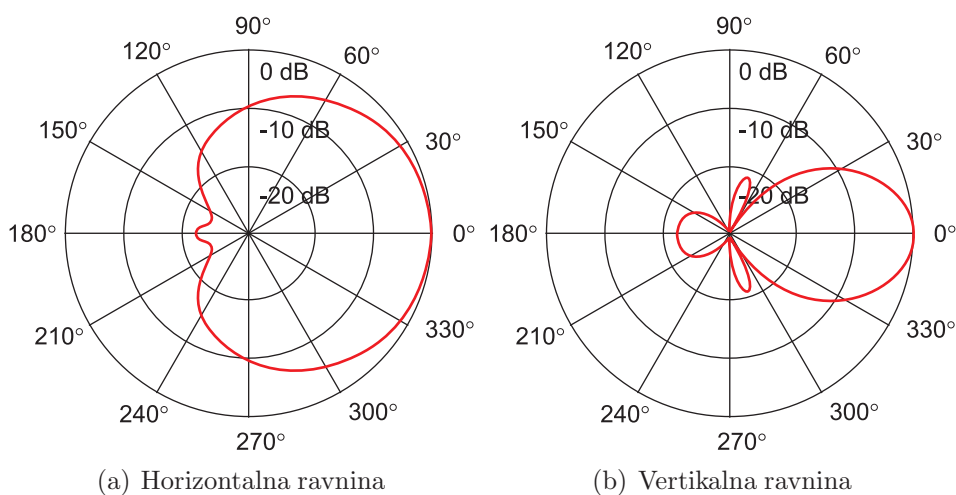


izbjeci analizom njihovih uzroka, kalibracijom i mjerenjem realnih parametara zrake. Prednost ovakvog pristupa je vrlo mala dimenzija i jednostavnost osjetila koji detektira zraku, ali zato izvorišni čvor mora biti većih dimenzija.

### 2.2.2. Mjerenje azimuta

Metode mjerenja kuta dolaska signala, odnosno azimuta susjednog čvora, mogu se podijeliti na dva pristupa – pristup mjerenja razlike u jačini signala i u fazi signala.

Jedna od osnovnih metoda prvog pristupa temelji se na dijagramu zračenja usmjerene antene, kojeg se može oblikovati elektronički ili samom konstrukcijom antene. Primjer dijagrama zračenja jedne usmjerene antene dan je na slici 2.7, a predstavlja grafički prikaz relativnog dobitka antene u odnosu na smjer izvora signala. Fizičkim ili elektroničkim rotiranjem dijagrama antene, mijenja se dobitak  $G_r$  za signal koji dolazi iz određenog smjera, pa se prema jednadžbi (2.4), mijenja i jačina signala  $P_r$ . Kada se detektira najviša razina signala prema trenutnom položaju dijagrama zračenja, može se zaključiti smjer dolaska signala. Ova metoda nije pouzdana, jer prijatelj ne može razlikovati promjenu jačine signala zbog rotacije i dobitka od one koja nastaje zbog promjene jačine odašiljanja ili zbog promjena u prostoru. Problem se može riješiti uvođenjem još jedne antene s neusmjerenim dijagramom zračenja, koja služi za kalibraciju, pa se više ne prati apsolutna jačina signala već omjer jačina signala između ove dvije antene.



Slika 2.7. Dijagram zračenja usmjerene antene.

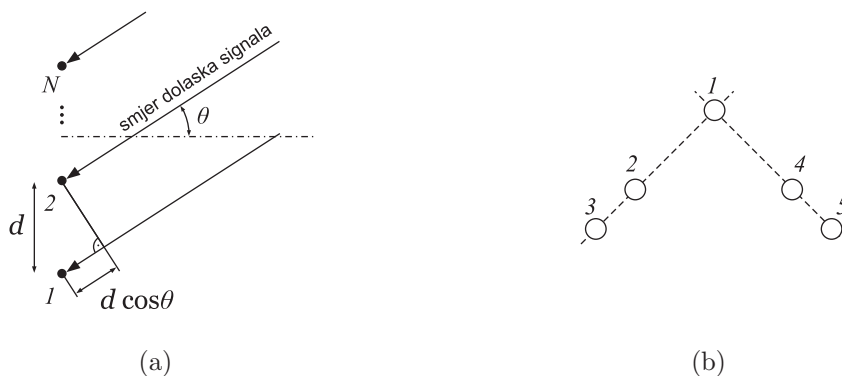
Još naprednija metoda uključuje više usmjerenih antena, usporedbom jačina signala sa pojedinih antena može se zaključiti o smjeru dolaska signala. Tipične pogreške za sustav s četiri antene su  $10^\circ - 15^\circ$ , za sustav sa šest antena  $5^\circ$  i sa osam antena  $2^\circ$ .

Metode koje se koriste mjerenjem faznih razlika RF signala zahtijevaju niz neusm-

jerenih antena koje su međusobno razmaknute na udaljenost  $d$  reda veličine valne duljine signala. Jedan takav antenski niz od  $N$  antena prikazan je na slici 2.8(a). Fazna razlika između susjednih antena, zbog razlike u duljini puta između pojedine antene i udaljenog odašiljača, ovisi o kutu dolaska signala, a može se prikazati jednadžbom

$$\varphi = 2\pi \frac{d \cos \theta}{\lambda} \quad (2.7)$$

prema kojoj je moguće odrediti i kut dolaska signala  $\theta$ . Metoda slična opisanoj, a koja koristi fazne razlike ultrazvučnih signala obrađena je u [14]. Kako bi se riješila dvoznačnost preslikavanja fazne razlike u kut  $\theta$ , u tom radu korišten je posebno oblikovan niz mikrofona prikazan na slici 2.8(b), te su postignuti vrlo dobri rezultati – pogreška mjerenja kuta bila je u okviru  $3^\circ$ .



**Slika 2.8.** Određivanje kuta iz fazne razlike primljenog signala.

Mjerenje smjera dolaska signala zahtjeva posebnu opremu, poput usmjerenih antena ili antenskog niza, što povećava cijenu čvora i njegovu veličinu. Nadalje, kako se mjerenje kuta temelji na propagaciji signala izravnom putanjom, vrlo je važno razdvojiti utjecaj višestazne propagacije. Ipak, kako će se kasnije u ovom radu ustanoviti, mjerenje azimuta ima i svojih prednosti, budući da je manja povezanost čvorova kako bi sustav imao jedinstveno rješenje problema lokalizacije. Također, smanjenje troškova proizvodnje odašiljača u području milimetarskih valnih dužina, može smanjiti veličinu potrebnog niza, a povećano gušenje u tom frekvencijskom području može pomoći ukloniti problem višestazne propagacije.

Osim navedene dvije tehnike, postoje i posebna rješenja za određivanje azimuta, poput onog opisanog u [15] gdje su sidra opremljena sinkroniziranim rotirajućim usmjerenim antenama, a obični čvorovi mjere razliku u vremenima u kojima detektiraju signal sa pojedinih sidara.

### 2.2.3. Mjerenje profila signala

Mjerenje profila signala odnosi se na mjerenje jačine, smjera ili vremena dolaska signala čvorova koji znaju svoju lokaciju (sidara). Usporedbom trenutno izmjerenih vrijednosti (naravno mnoge od njih se ne mogu izmjeriti zbog udaljenosti sidara ili raznih prepreka), sa vrijednostima u modelu prostora, može se donijeti zaključak o lokaciji čvora.

Samo mjerenje nije zahtjevno, ali je preduvjet lokalizacije postojanje modela prostora. Model se sastoji od niza lokaliziranih mjernih točaka u kojima se mjere profili, dok se profili na ostalim lokacijama dobivaju interpolacijom. Pojedini model ovisi o položaju sidara i okolini te je, ukoliko dođe do bilo kakve promjene, potrebno izraditi novi model, što povećava cijenu ove metode.

## 2.3. Izvori pogrešaka

Problem lokalizacije ad hoc bežičnih mreža zahtjeva složene metode rješavanja. Svaka od njih neminovno sadrži niz problema koji u konačnici mogu uzrokovati veće ili manje pogreške. Pogreške se mogu svrstati u dvije skupine – slučajne vremenski promjenjive i sustavne koje nastaju zbog utjecaja okoliša na propagaciju.

### 2.3.1. Pogreške mjerenja

Prva velika skupina su pogreške nastale zbog problema pri mjerenju odnosa među čvorovima. Neke od njih su već spomenute, a u ovom potpoglavlju ih se detaljnije klasificira i opisuje.

#### Složena propagacija signala

Pri mjerenju jačine radio signala višestazno širenje i gušenje uzrokuje velike varijacije u dobivenim rezultatima, a primjerice raspršenje signala u blizini prijarnika može vrlo negativno utjecati na određivanje smjera dolaska signala čak i ako postoji vidljivost između odašiljača i prijarnika. Kod određivanja vremena propagacije, estimatori koji se temelje na kros korelaciji mogu biti pod utjecajem višestaznog gušenja, što može dovesti do pomaka u određivanju vremena dolaska signala.

Ako između prijarnika i odašiljača ne postoji optička vidljivost, greške u mjerenju kuta dolaska i vremena dolaska mogu biti izuzetno velike zbog činjenice da prijarnik ne može razaznati reflektirani signal od izravnog.

### **Interferencija odašiljača**

Do pogrešaka može doći zbog maskiranja signala slabijih odašiljača onim jačima. Također kod ultrazvučnih sustava problem može nastati zbog interferencije među signalima sa susjednih sidara.

### **Kalibracija primopredajnika**

U sustavima koji se temelje na mjerenju jačine primljenog signala, problemi mogu nastati zbog upotrebe jeftinih nekalibriranih primopredajnika kojima emitirana snaga može značajno varirati iako se odašiljač radi sa istom nominalnom snagom. Također, kod mjerenja jačine rezultati mogu varirati za isti nivo snage na ulazu prijmnika.

### **Varijacije brzine propagacije signala**

Brzina propagacija signala, posebno akustičnih, varira u ovisnosti o nekim parametrima medija, poput temperature, vlažnosti itd., a veličina pogreške ovisi o dužini puta.

## **2.3.2. Pogreške zbog karakteristika mreže i čvorova**

### **Ograničeni resursi**

Ideja je da čvorovi bežične mreže osjetila budu što manji i što jeftiniji. Taj zahtjev donosi mnoge prednosti, poput mogućnosti da se na nekom prostoru rasporede veliki broj čvorova i obavljaju zadaću robusno i sa velikim redundancijama. Isti zahtjev uvjetuje i mnoga ograničenja. Najvažnije je ograničena autonomija, koja proizlazi iz činjenice da su čvorovi napajani baterijama. To znači ograničenje u svim zadacima koje čvor obavlja: mjerenje, komunikacija, obrada podataka i sl. Nadalje, veličinom čvora i cijenom ograničene su računalne i memorijske mogućnosti čvorova. U ovom slučaju, za razliku od energetske ograničenosti, može se očekivati veliki napredak s obzirom da tehnologija omogućava sve veću integraciju i manju potrošnju potrebnih sklopova.

Ograničeni resursi uvjetuju jednostavnost metoda lokalizacije. Prvenstveno se to odnosi na odabir metoda mjerenja, ali i na količinu komunikacija. Sve to uvelike utječe na kvalitetu estimacije lokacija.

### **Gustoća čvorova**

Mnoge metode lokalizacije ovisne su o gustoći čvorova. Ako se veze između čvorova predstave kao fizička ograničenja kojima su definirane i ograničene moguće lokacije

čvorova, onda su gušće odnosno povezanije i time lokacijski definirane mreže puno pogodnije za estimiranje tih lokacija. Osim gustoće na povezanost utječe i komunikacijski domet. Odnos između ove tri veličine opisan je u poglavlju 6. Još jedan bitan parametar kod mreža koje sadrže sidra i koje je potrebno apsolutno lokalizirati, je udio sidara i njihov položaj unutar mreže.

### Nekonveksne topologije

Točnost lokalizacije uvjetovana je i relativnim položajem čvorova koji obavljaju lokalizaciju. Nepovoljan položaj čvorova dovodi do velikih pogrešaka kao što je opisano u dijelu 4.1.1. Topologije u kojima se pojavljuje veliki broj čvorova koji je povezan sa ostalim čvorovima na nepovoljan način nazivaju se nekonveksne topologije. Uzrok tome je što, u nekonveksnim topologijama za veliki broj čvorova, susjedi ne tvore konveksnu formaciju, koja je preduvjet za kvalitetnu i jednoznačnu lokalizaciju.

## 2.4. Pregled relevantnih algoritama

Standardna metoda lokalizacije za mnogobrojne primjene je korištenje sustava globalne lokalizacije (Global Positioning System – GPS), koji se trenutno sastoji od 31 orbitirajućih satelita u 6 različitih ravnina na udaljenosti od 20230 km od zemljine površine. Kako bi ova metoda lokalizacije radila, prijamnik GPS signala mora biti u dometu signala sa najmanje 4 satelita te složenom, energetske zahtjevnom obradom signala i proračunima, prijamnik izvodi trilateraciju i vremensku sinkronizaciju. Kako su tipični čvorovi u bežičnim mrežama osjetila mali, jeftini, energetske ograničeni, te mogu biti raspoređeni u prostorima u kojima nema GPS signala, ova metoda lokalizacije nije pogodna.

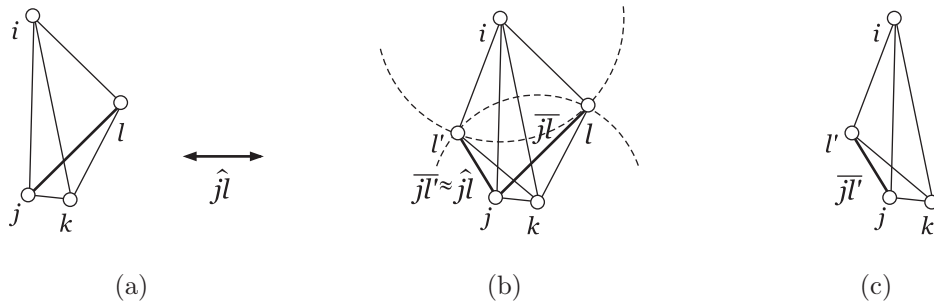
Unazad nekoliko godina pojavilo se mnoštvo različitih metoda lokalizacije bežičnih mreža osjetila. Zbog niza parametara čini se kako svaka od njih ima poseban pristup te ih je vrlo teško izravno uspoređivati. Mogu se klasificirati na više načina od kojih su neki opisani u potpoglavlju 2.1. U nastavku je detaljnije opisana nekolicina metoda najbližnjih projektiranoj metodi koja čini osnovicu ovog magistarskog rada.

### 2.4.1. Robusna raspodijeljena lokalizacija korištenjem udaljenosti među čvorovima

Ova metoda [16] temelji se na korištenju mjerenja udaljenosti među čvorovima. S obzirom da su mjerenja nepouzdana, odnosno da se mogu modelirati kao slučajne varijable s većom ili manjom varijancom, lokacije koje se dobivaju analizom mjerenja mogu biti višeznačne. To znači da su estimirane lokacije podložne velikim strukturnim

pogreškama, zbog smanjene krutosti pripadajućih grafova. Središnji problem koji se nastoji riješiti je kako lokalizirati čvorove na način da se sa što većom sigurnošću dobivene lokacije mogu smatrati jedinstvenima. Rješenje koje je predloženo temelji se na korištenju robusnih četverokuta (engl. *robust quadrilaterals*), odnosno formacija koje sa velikom vjerojatnošću jamče jedinstveno rješenje.

Ako se uzme za primjer robusni četverokut na slici 2.9(a), potrebno je odrediti koji uvjet mora biti zadovoljen kako bi uz prisustvo mjernog šuma sa velikom vjerojatnošću mogli utvrditi da se neće pojaviti više strukturno različitih rješenja problema lokalizacije. Primjerice, ako se za četverokut sa slike, zbog pogreške u mjerenju, dužina  $\overline{j\bar{l}}$  zamijeni sa izmjerenom vrijednošću  $\hat{j\bar{l}}$ , tada strukturno pogrešno rješenje  $ijkl'$  na slici 2.9(c) bolje zadovoljava uvjete postavljene mjerenjem nego pravo rješenje  $ijkl$ .



**Slika 2.9.** Robusni četverokut – zbog mjernog šuma četverokut se može strukturno pogrešno lokalizirati.

Pokazalo se da vjerojatnost strukturno pogrešne lokalizacije ovisi o varijanci mjernog šuma, najkraćoj stranici  $b$  i najmanjem kutu trokuta  $\theta$  od kojih se sastoji robusni četverokut. U primjeru sa slike, to je stranica  $jk$  i kut  $\angle jlk$  u trokutu  $\triangle jkl$ . Minimalna pogreška mjerenja, koja može uzrokovati pogrešnu lokalizaciju, iznosi  $d_{err} = b \sin^2 \theta$ .

Ukoliko je izmjerena udaljenost  $X$  zadana kao slučajna varijabla sa normalnom distribucijom, standardnom devijacijom  $\sigma$  i srednjom vrijednošću jednako stvarnoj udaljenosti  $d$ , tada za topološki najgori mogući slučaj (simetrična topologija), vjerojatnost pogrešne lokalizacije iznosi

$$P(X > d + d_{err}) = \Phi\left(\frac{d_{err}}{\sigma}\right) \quad (2.8)$$

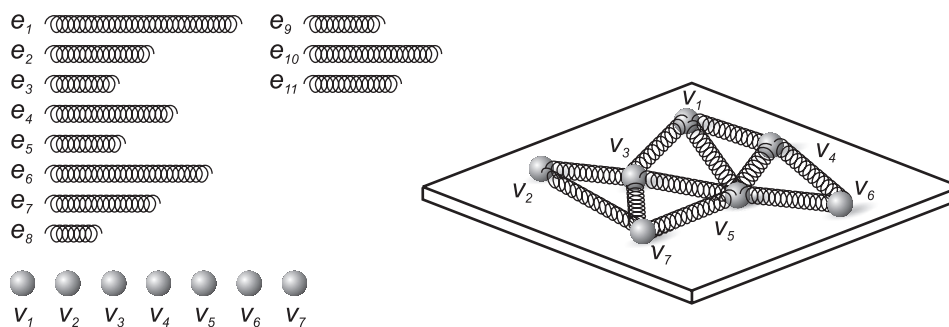
gdje je  $\Phi(x)$  funkcija koja integrira funkciju gustoće vjerojatnosti za normalnu distribuciju  $\mathcal{N}(0, 1)$  od  $x$  do beskonačnosti. Prema tome, ako je poznata standardna devijacija mjernog šuma, može se precizno odrediti parametar  $d_{min} = d_{err}$ , takav da sa točno određenom vjerojatnošću neće doći do većih pogrešaka lokalizacije uslijed nejedinstvenosti rješenja. Pri tome četverokuti se smatraju robusnima i koriste se samo ako je zadovoljen uvjet

$$b \sin^2 \theta > d_{min}. \quad (2.9)$$

## Optimizacija inicijalnih lokacija

Nakon što se lokaliziraju skupine čvorova koje tvore takve formacije pristupa se opcionalnoj optimizaciji lokacija čvorova koristeći Newton-Raphsonovu metodu ili metodu relaksacije virtualnog sustava masa i opruga [17]. Bitno je napomenuti kako je optimizacija komunikacijski zahtjevan proces te je njeno korištenje potrebno ispitati i opravdati.

Na slici 2.10 prikazan je takav sustav u kojem mase predstavljaju čvorove na svojim inicijalnim (neoptimiranim) lokacijama, a opruge su umetnute između odgovarajućih čvorova koji su izmjerili međusobnu udaljenost. Duljina opruge u opuštenom stanju odgovara toj izmjerenoj udaljenosti.



Slika 2.10. Optimizacija uz pomoć virtualnih sila.

Posljedica takve definicije sustava je pojava sila koje vode sustav u stanje minimalne potencijalne energije, odnosno lokacije čvorova vode u njihove optimizirane lokacije. Virtualne sile koje djeluju na čvorove mogu se opisati jednadžbom

$$\vec{F}_{i,j} = \vec{v}_{i,j}(\hat{d}_{i,j} - r_{i,j}) \quad (2.10)$$

gdje je  $\hat{d}_{i,j}$  udaljenost između estimiranih lokacija čvorova  $i$  i  $j$ ,  $r_{i,j}$  izmjerena udaljenost, a  $\vec{v}$  jedinični vektor usmjeren od čvora  $i$  prema čvoru  $j$ . Iterativnim računanjem sila, odgovarajućim pomakom u smjeru rezultatne sile te dijeljenjem nove lokacije sa susjednim čvorovima, postiže se stanje minimalne potencijalne energije. Veličina pomaka određuje se empirijski, a autori navode kako se najbolji rezultati postižu za vrijednost pomaka koji je obrnuto proporcionalan broju susjednih čvorova.

Naposljetku, robusne strukture se raspodijeljenim algoritmom međusobno spajaju, koristeći lokacije zajedničkih čvorova kako bi se izvršila relativna lokalizacija cijele mreže. Dobiveni rezultati pokazuju kako se upotrebom robusnih četverokuta izbjegavaju pojave velikih pogrešaka te njihove propagacije. Posljedica je potreba za velikom povezanošću čvorova, kako bi što veći postotak čvorova bio lokaliziran.

### 2.4.2. Ad hoc sustav pozicioniranja

Ad hoc sustav pozicioniranja (engl. *Ad hoc Positioning System – APS*) [18] metoda je estimacije lokacija čvorova koja se temelji na mjerenju udaljenosti od čvorova koji poznaju svoju lokaciju – sidara. Metoda se može koristiti sa tri različite tehnike određivanja udaljenosti i to uz pomoć broja komunikacijskih skokova – *DV-hop*, jačine signala – *DV-distance* i stvarne udaljenosti *Euclidean*.

U *DV-hop* metodi čvorovi prosljeđuju informaciju o lokaciji sidara i simultano određuju broj komunikacijskih skokova do svakog od njih. Uz pomoć poznatih lokacija i broja komunikacijskih skokova sidra mogu izračunati prosječnu duljinu skoka. Ovaj podatak se zatim kontroliranim preplavlivanjem šalje svim čvorovima koji, na temelju poznatog broja komunikacijskih skokova i prosječne duljine skoka dobivene od najbližeg sidra, mogu izvršiti multilateraciju i odrediti svoju lokaciju.

Kod ostale dvije tehnike mjerenja sličnim postupcima se može dobiti estimacija lokacije. *DV-hop* mjerenje se pokazalo najučinkovitije što se tiče komunikacijskih zahtjeva, a za veliku gustoću čvorova i udio sidara može se postići lokalizacija unutar 1/3 komunikacijskog dometa. Problem kod APS-a predstavlja propagacija pogreške mjerenja i velika ovisnost o broju i položaju sidara. Optimizacijom lokacija uz pomoć virtualnih sila, može se dodatno poboljšati preciznost za 20% – 30%.

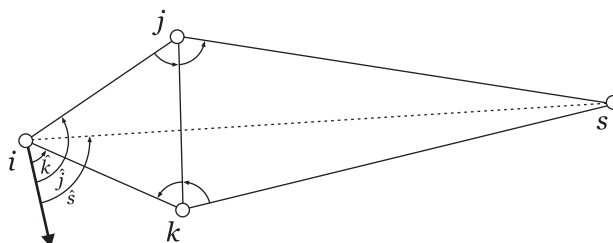
#### APS korištenjem mjerenja kuta dolaska signala

APS metoda, odnosno propagacija mjerenja od sidara i naknadna lokalizacija može se uspješno primijeniti i ako se propagira azimut (engl. *bearing*) sidara te se obavlja naknadna multiangulacija [19]. Uvjet da se pojedini čvor uspješno lokalizira je poznavanje lokacija od barem tri sidra i njihovih azimuta. Ukoliko neki od čvorova (primjerice sama sidra) ili svi čvorovi posjeduju kompas, čvor može odrediti i azimut od strane samog sidra (engl. *radial*) te se u tom slučaju može postići veća preciznost lokalizacije.

Princip propagacije azimuta sidra  $s$  do čvora  $i$  preko čvorova  $j$  i  $k$  prikazan je na slici 2.11. Čvor  $i$  poznaje azimute prema čvorovima  $j$  i  $k$  koji poznaju azimute prema čvorovima  $s$  i  $i$  te međusobno. Poznavajući azimute prema dva čvora može se izračunati kut između njih, pa slijedi da su poznati svi kutovi trokuta  $\triangle ijk$  i  $\triangle jks$ . Prema tome moguće je izračunati i kut  $\widehat{kis}$  koji omogućava određivanje azimuta prema sidru  $s$  kao  $\hat{s} = \hat{k} + \widehat{kis}$ . Dalje se dobiveni azimut može koristiti za čvorove koji su još udaljeniji od sidra, a ukoliko postoji više putova kojima se može propagirati azimut, koristi se onaj sa manjim brojem komunikacijskih koraka.

Jedan od nedostataka metode je taj da se mjerne pogreške pri određivanju azimuta čvorova propagiraju, odnosno mjerna nesigurnost se povećava kako raste broj komunikacijskih koraka potreban da čvor izračuna azimut prema sidru. Metoda djelomično ublažava ovaj problem, na način da izbjegava korištenje trokuta sa ostrim kutovima





Slika 2.11. Propagacija azimuta[19].

budući da je mjerna pogreška neovisna o iznosu kuta, pa je kod malih kutova utjecaj pogreške najizraženiji. Nadalje, kako se konačna estimacija lokacije izvodi koristeći trojke sidara, moguće je dobiti više različitih rješenja. Objedinjavanje rješenja obavlja se na način da se računa centroida svih točaka, a zatim se detektiraju i odbacuju rješenja koja su najudaljenija te se računa nova centroida. Naposljetku, propagacija greške se može ublažiti ograničavanjem širenja azimuta tako da čvorovi koriste samo komunikacijski najbliža sidra.

Rezultati simulacija pokazali su da je za standardnu devijaciju mjerenja od  $10^\circ$ , prosječnu povezanost 10.5 i udio sidara od 20%, srednja pogreška lokalizacije 0.6 komunikacijskog dometa.

## 2.5. Definicija problema

Iz navedenog proizlazi da su raspodijeljeni algoritmi koji koriste azimute trenutno ograničeni na njihovu propagaciju od sidara prema običnim čvorovima. Ovakav pristup, osim što jako ovisi o gustoći čvorova, udjelu sidara u mreži i njihovoj lokaciji, ne iskorištava na pravi način sve izmjerene azimute. Algoritmi koji iskorištavaju sve izmjerene azimute su centralizirani i osjetljivi na povećanje broja čvorova. U ovom radu nastoji se popuniti taj segment lokalizacijskih algoritama, koristeći se početnim idejama predloženim u [20, 21], a koje do sada nisu bile implementirane, simulirane niti su analizirani rezultati algoritama koji bi se temeljili na njima u slučaju realnih pogrešaka mjerenja azimuta. Središnji dio ovog rada objavljen je u [22].

Rezimirajući sve navedeno, problem glasi: *ako je zadan skup čvorova s nepoznatom lokacijom i orijentacijom te mehanizam po kojem čvorovi mogu izmjeriti, s većom ili manjom preciznošću, azimut svojih susjeda, potrebno je raspodijeljenim algoritmom odrediti koordinate čvorova koje zadovoljavaju tako izmjerene azimute.*

## 3. Teorija krutosti – preduvjet uspješne lokalizacije

Za uspješnu lokalizaciju čvorova u mreži, potrebno je odrediti njihove međusobne odnose. Ti odnosi i načini na koji se određuju definirani su i analizirani u prethodnom poglavlju. Svaka izmjerena veličina predstavlja dodatnu informaciju u opisu izgleda mreže i kao takva ograničava broj mogućih rješenja. Važno pitanje je da li je tih informacija dovoljan broj, kako bi broj mogućih rješenja bio sveden na jedno. Uz prethodno navedenu definiciju problema, odgovor na postavljeno pitanje je - *nije dovoljan*. Naime, ukoliko je mreža definirana kao neusidrena, tada nijedan čvor nema informaciju o pravoj lokaciji te je broj mogućih rješenja beskonačan. Na primjer, uz nepromijenjene udaljenosti između čvorova, cijela mreža se može translirati, rotirati i reflektirati.

Prema iznesenim zapažanjima treba ovom problemu prići sustavno što je u ovom poglavlju napravljeno koristeći se teorijom krutosti grafova (engl. *graph rigidity theory*).

Teorija krutosti grafova primijenjena na lokalizaciju mreža obrađena je u recentnim radovima. Primjerice u [23] opisani su grafovi i formacije točaka temeljeni na ograničenjima udaljenosti. Zatim je teorija krutosti upotrijebljena za ispitivanje uvjeta jedinstvenosti rješenja. U [24], problem lokalizacije proširen je na mreže s ograničenjima azimuta te su uvjeti jedinstvenosti rješenja obrađeni korištenjem svojstva paralelne krutosti. Za spajanje krutih formacija u nove krute formacije koristiti će se rješenja iz [25], gdje su izneseni dovoljni uvjeti za spajanje krutih formacija temeljenih na ograničenju udaljenosti. Koristeći rezultate tih radova, kratko revidiranih u narednim stranicama, kreirane su nove mrežne strukture, a koristeći rezultate iz [7], definiraju se i novi uvjeti za spajanje paralelno krutih formacija u ravnini.

### 3.1. Mreža – graf – formacija točaka

Teorija krutosti grafova, kao što joj ime i govori, reprezentira mrežu kao graf. Kao takva koristi se za analizu dovoljnih uvjeta za lokalizaciju mreže i izračunavanje

računske složenosti tog zadatka.

Za početak definiran je graf mreže. Ako se mreža označi sa  $\mathbf{N}$ , njen graf je definiran kao  $\mathbb{G}_{\mathbf{N}} = \{\mathcal{V}, \mathcal{E}_{\mathbf{N}}\}$ , u kojem vrhovi  $\mathcal{V} = \{1, 2, \dots, n\}$  predstavljaju čvorove mreže, a bridovi  $\mathcal{E}_{\mathbf{N}}$  su definirani kao skup uređenih parova čvorova  $(i, j)$  koji mogu izmjeriti međusoban odnos. U ovom radu, ova definicija susjeda, izjednačava se sa pretpostavkom da su susjedni čvorovi oni koji mogu ostvariti dvostranu komunikaciju, što je, s obzirom na metode određivanja odnosa, sasvim primjereno.

Graf mreže opisuje njenu topologiju, ali za rješavanje problema lokalizacije potrebno je osmisliti koncept opisa mreže koji sadrži i podatke o poziciji vrhova grafa te o izmjerenim odnosima. Koncept koji se koristi u tu svrhu naziva se formacija točaka (engl. *point formation*). Formacija točaka predstavlja jednu od mogućih realizacija odgovarajućeg grafa mreže  $\mathbb{G}_{\mathbf{N}}$  u  $d$ -dimenzionalnom prostoru  $\mathbb{R}^d$ . Za problem koji se obrađuje u ovom radu, pretpostavka je da je prostor dvodimenzionalan odnosno  $d = 2$ . Formacija točaka  $\mathbb{F}_p$  je definirana kao skup od  $n$  točaka  $p \triangleq \{p_1, p_2, \dots, p_n\}$ , koje odgovaraju lokacijama vrhova grafa u  $\mathbb{R}^2$ , zajedno sa skupom veza  $\mathcal{L}$ , koje odgovaraju bridovima grafa i funkcijom  $f : \mathcal{L} \mapsto \mathbb{R}$ , koja odgovara mjerenju odnosa među čvorovima susjedima.

Rješenje problema lokalizacije za graf mreže  $\mathbb{G}_{\mathbf{N}}$ , je formacija točaka koja zadovoljava njegovu topologiju i izmjerene odnose među čvorovima susjedima. Prema tome, dvije različite formacije točaka  $\mathbb{F}_p$  i  $\mathbb{F}_q$  mogu predstavljati rješenje istog problema, ukoliko vrijedi  $T(q_i) = p_i, i \in 1, 2, \dots, n$  gdje  $T : \mathbb{R}^d \mapsto \mathbb{R}^d$  označava preslikavanje točaka koje čuva odnose među njima. Takve formacije i preslikavanja (transformacije) nazivaju se kongruentnima.

Rješenje problema lokalizacije može biti jedinstveno samo do kongruentne transformacije. Ukoliko su *sve* formacije točaka istoga grafa, mreže koje zadovoljavaju odnose među čvorovima, kongruentne, odnosno jedina transformacija koja je moguća je kongruentna, tada je takva formacija globalno kruta. Prema tome, može se reći da je i odgovarajući graf mreže globalno krut.

## 3.2. Udaljenost kao ograničenje

Kako je prethodno pokazano, izmjereni odnosi među čvorovima ograničavaju broj mogućih rješenja. Zato se izmjereni odnosi među čvorovima smatraju i nazivaju ograničenjima. Jedan od najčešće mjerenih odnosa među čvorovima je udaljenost. U tom slučaju, u definiciji formacije točaka, redefinira se funkcija  $f$  i označava se kao  $\delta : \mathcal{L} \mapsto \mathbb{R}$ , dok njene vrijednosti za točke  $(i, j) \in \mathcal{L}$  predstavljaju euklidsku udaljenost odnosno  $\|p_i - p_j\|$ .

Za tako definiranu mrežu, s udaljenostima kao ograničenjima, i dalje vrijedi zaključak da je problem lokalizacije rješiv, samo ukoliko su sve formacije točaka koje

imaju isti graf i zadovoljavaju funkciju udaljenosti  $\delta$ , kongruentne. Kako bi se odredilo da li je formacija točkaka  $\mathbb{F}_p$  globalno kruta, moraju se analizirati sve moguće formacije  $\mathbb{F}_q$  koje zadovoljavaju graf mreže i isti skup udaljenosti i odrediti da li su one kongruentne sa  $\mathbb{F}_p$ . Sve takve formacije  $\mathbb{F}_q$  moraju zadovoljavati jednadžbu

$$(q_i - q_j)^T \cdot (q_i - q_j) = \delta(i, j)^2. \quad (3.1)$$

Najprije se mogu sagledati formacije koje su najbližnije formaciji  $\mathbb{F}_p$ . To su formacije koje nastaju malim perturbacijama točkaka početne formacije. Takve formacije moraju zadovoljavati lineariziranu verziju jednadžbe (3.1):

$$(q_i - q_j)^T \cdot (\dot{q}_i - \dot{q}_j) = 0. \quad (3.2)$$

Ova jednadžba može se napisati i u matričnom obliku

$$R(\mathbb{F}_q)\dot{q} = 0, \quad (3.3)$$

u kojoj je  $\dot{q}$  vektor stupac, a  $R(\mathbb{F}_q)$  je matrica koja se naziva matrica krutosti (engl. *rigidity matrix*). Dimenzija matrice krutosti je  $|\mathcal{L}| \times dn$  gdje je  $d$  dimenzija prostora, a  $n$  broj vrhova.

Rješenja  $\dot{q}$  ovog sustava, nazivaju se savijanja prvog reda (engl. *first-order flexes*). U prostoru mogućih rješenja, postoji potprostor koji sadrži trivijalna rješenja. To su rješenja dobivena kongruentnim transformacijama. Dimenzija tog potprostora je 3 za formacije u ravnini, što odgovara zbroju dvije dimenzije za translaciju i jedne za rotaciju. Ako se sva moguća rješenja nalaze u tom potprostoru, formacija je prvog reda krutosti (engl. *first-order rigid*). Iz navedenog slijedi, da je formacija u ravnini kruta prvog reda, ukoliko je nulprostor ili jezgra matrice jednaka 3, odnosno rang matrice  $R(\mathbb{F}_p) = 2n - 3$ .

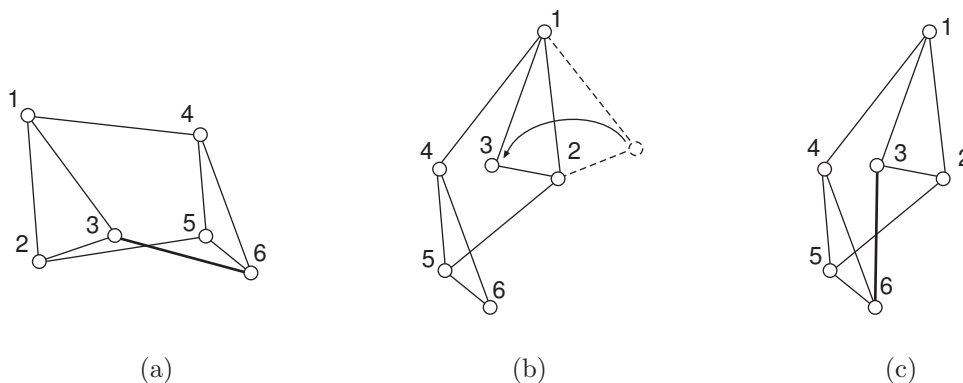
Krutost prvog reda osigurava da krutost formacije bude otporna na male perturbacije, odnosno biti će tolerantna na dovoljno male greške u mjerenjima udaljenosti, što je u praktičnoj primjeni vrlo bitno. Drugim riječima, poznavajući samo pripadajući graf može se sa sigurnošću reći da će formacija biti kruta prvog reda, bez potrebe za točnim udaljenostima. Takvi grafovi nazivaju se generički krutima.

### 3.2.1. Problem nejednakosti između krutosti i globalne krutosti

Problem se javlja ukoliko se traži dodatno svojstvo, a to je da graf  $\mathbb{G}$  ima samo jednu jedinstvenu realizaciju u obliku formacije točkaka  $\mathbb{F}_p$ , pri čemu se kongruentne formacije ne smatraju različitim realizacijama. U tom slučaju su udaljenosti između svih parova točkaka poznate, odnosno mogu se implicitno izračunati iz jedinstvenih položaja točkaka. Takve formacije se nazivaju globalno generički krute formacije.

U dvodimenzionalnim prostorima, različite realizacije krutih formacija prvog reda se sastoje ili od refleksija preko osi koja prolazi kroz skup reznih vrhova (engl. *cut set*) veličine 2 i/ili kao posljedica diskontinuiranih transformacija, pri kojima se formaciji oduzme jedan brid, a zatim nakon transformacije se taj brid vrati kao što je prikazano na slici 3.1. Prvi slučaj može se kontrolirati uvjetom da je minimalna povezanost grafa jednaka  $d + 1$ , odnosno za mreže u ravnini 3-povezan, a drugi uvjetom da je graf redundantno krut, odnosno oduzimanjem jednog brida on je i dalje generički krut. Ovaj zaključak sumiran je u sljedećem teoremu.

**Teorem 1.** *Graf  $\mathcal{G}$  sa  $n \geq 4$  vrha je generički globalno krut u  $\mathbb{R}^2$  ako i samo ako je 3-povezan i redundantno krut u  $\mathbb{R}^2$ .*



**Slika 3.1.** Primjer krute, ali ne i globalno krute formacije točaka u ravnini sa udaljenostima kao ograničenjima (a), oduzimanjem brida (3, 6) moguća je izmjena položaja trokuta (1, 2, 3) i (4, 5, 6) te refleksija točke 3 (b), konačna nekongruentna formacija koja zadovoljava isti skup ograničenja udaljenosti (c).

Nakon što su egzaktno definirani uvjeti za jedinstvenu lokalizaciju, može se zaključiti kako bi eventualna provjera tih uvjeta unutar raspodijeljenog algoritma lokalizacije bila vrlo zahtjevan zadatak. S obzirom na nesigurnost mjerenja udaljenosti prilikom lokalizacije, ne može se očekivati točno poklapanje izmjerenih udaljenosti i udaljenosti dobivenih iz estimiranih položaja točaka. Dapače, pogrešne formacije točaka često mogu bolje zadovoljavati skup izmjerenih udaljenosti nego što bi to zadovoljavala ispravna formacija. Prema tome, potrebno je u procesu lokalizacije dodati korak koji bi, na temelju dodatnih provjera vjerojatnosti pogrešnih realizacija, mogao odrediti da li se proces lokalizacije može nastaviti za čvorove koji se lokaliziraju. Jedan takav algoritam [16], koji se koristi principom robusnih četverokuta, opisan je u potpoglavlju 2.4.1. U toj metodi posebno je analiziran odnos između parametara četverokuta, varijance pogreške i vjerojatnosti strukturnih pogrešaka lokalizacije. Rezultati ispitivanja su pokazali da, ukoliko se parametar koji određuje robusnost četverokuta postavi na vrijednost pri kojoj je mogućnost strukturnih pogrešaka

manja od 1%, tada je potrebna vrlo visoka prosječna povezanost kako bi lokalizacija bila relativno uspješna. Primjerice, za realne vrijednosti varijance mjerenja ( $\sigma = 1\text{cm}$ ) i prosječnu povezanost čvorova 20, algoritam omogućava da tek 30% čvorova bude lokalizirano unutar istog koordinatnog sustava.

### 3.3. Azimut kao ograničenje

Na sličan način kao što se analizira slučaj kada je udaljenost ograničenje, sada se mogu analizirati uvjeti lokalizacije mreže kada je azimut ograničenje. Definicija grafa je identična, dok se formacija točaka sa ograničenim azimutima naziva paralelna formacija točaka. Cjelokupna analiza odnosi se na slučaj lokalizacije u ravnini.

#### 3.3.1. Paralelna formacija točaka

U slučaju mjerenja azimuta među čvorovima susjedima u definiciji paralelne formacije točaka, redefinira se funkcija  $f$  kao  $g : \mathcal{L} \mapsto [0, 2\pi)$ , čije vrijednosti u  $(i, j) \in \mathcal{L}$  predstavljaju kut pod kojim čvor  $i$ , u odnosu na svoju referentnu usmjerenost, vidi čvor  $j$ .

Dvije formacije točaka  $\mathbb{F}_p$  i  $\mathbb{F}_q$  su paralelne ako su im pripadajući grafovi mreže  $\mathbb{G}_\mathbb{N}$  i funkcije azimuta  $g$  iste. U zadanom slučaju funkcije azimuta  $g$  su uvijek iste, relativno na globalnu rotaciju, jer je pretpostavljena mreža u kojoj nijedan čvor nema kompas. Formacija točaka  $\mathbb{F}_q$  koja je paralelna nekoj proizvoljnoj formaciji  $\mathbb{F}_p$ , je rješenje sustava  $|\mathcal{B}|$  homogenih linearnih jednadžbi

$$(p_i - p_j)^\perp \cdot (q_i(t) - q_j(t)) = 0, \quad (i, j) \in \mathcal{L}, \quad t \geq 0 \quad (3.4)$$

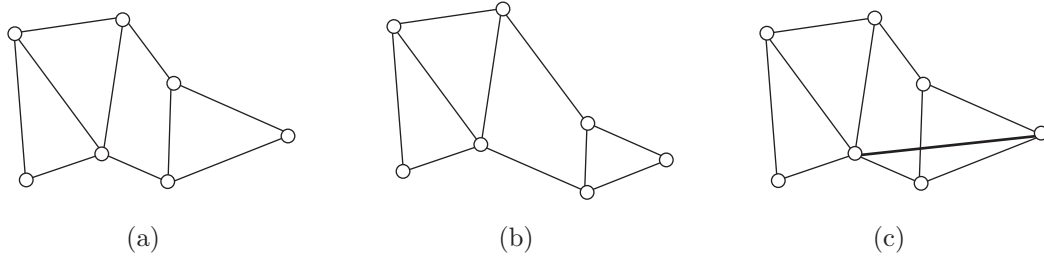
u kojima operator  $(\cdot)^\perp$  rotira vektor u ravnini za  $\frac{\pi}{2}$  suprotno od smjera kazaljke na satu i u kojima je  $q$  definiran relativno na globalnu rotaciju. Potrebno je naglasiti da se, ovakvom kontinuiranom transformacijom, može doći do svih mogućih (kongruentnih i nekongruentnih) paralelnih formacija točaka. Drugim riječima, sva moguća rješenja problema lokalizacije, povezana su trajektorijama  $q(t)$ .

Slično kao i za slučaj udaljenosti, ovaj sustav jednadžbi može se derivirati i napisati u matričnom obliku

$$T(\mathbb{F}_p)\dot{q} = 0. \quad (3.5)$$

U slučaju ograničenja azimuta potprostor trivijalnih rješenja je također trodimenzionalan, samo što za razliku od slučaja ograničenja udaljenosti, one proizlaze iz translacije i skaliranja. Rezultat je jednak i glasi – ukoliko je rang matrice krutosti  $T(\mathbb{F}_p)$  jednak  $2n - 3$  tada je formacija generički paralelno kruta. Primjer konstrukcije

takve formacije dodavanjem formaciji koja nije kruta još jednog ograničenja dan je na slici 3.2. Kako se kod mreža sa ograničenjem azimuta do svih paralelnih formacija



**Slika 3.2.** Formacija točaka koja nije paralelno kruta (a), njena nekongruentna transformacija (b), dodavanjem još jednog ograničenja čini je paralelno krutom (c).

točaka istog grafa može doći kontinuiranom transformacijom, tada se može zaključiti da je dovoljno ograničiti nulprostor matrice krutosti  $T(\mathbb{F}_p)$  na 3 odnosno njen rang na  $2n - 3$  kako bi ograničili ukupan broj mogućih realizacija na jedno jedinstveno rješenje. Drugim riječima, za mreže sa ograničenjem azimuta vrijedi da je generička krutost dovoljan uvjet za globalnu krutost.

Ovakav rezultat, koji govori da je jedinstvenost rješenja definirana u potpunosti topologijom mreže, omogućava razvoj algoritma lokalizacije koji će sa 100% sigurnošću moći predvidjeti mogućnost pojave višestrukih rješenja te poduzeti odgovarajuće akcije. O metodama kojima se kontrolira pojava višestrukih rješenja, detaljnije se govori u narednom potpoglavlju u kojem su opisane mrežne strukture i način na koji se koriste.

### 3.3.2. Problem globalne orijentacije

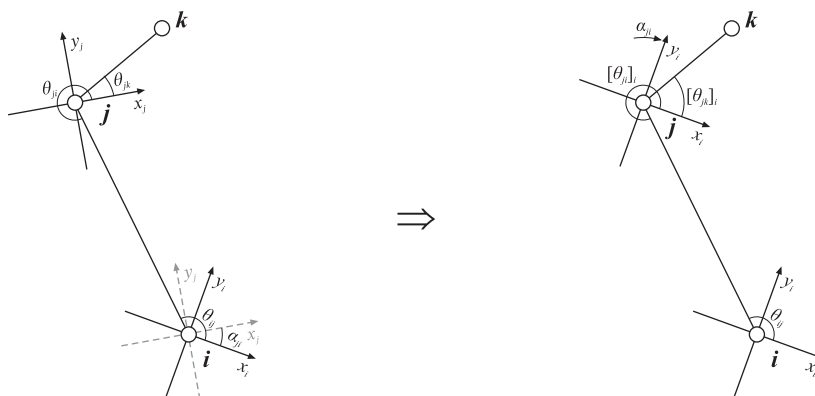
Ograničenja azimuta dobivaju se određivanjem smjera dolaska signala između svih parova čvorova susjeda. Kako je pri definiciji problema pretpostavljeno da čvorovi nisu opremljeni kompasima, estimacija njihove globalne orijentacije nije moguća. Prema tome, sva mjerenja smjera su relativna u odnosu na orijentaciju samog čvora koji provodi mjerenje.

Iako su mjerenja relativna, s obzirom da se između svakog para čvorova izvode po dva mjerenja, parovi čvorova mogu izračunati razliku između njihovih orijentacija i u skladu s tim prilagoditi izmjerene podatke jednom od tih sustava. Primjerice, na slici 3.3, mjerenje  $\theta_{jk}$  čvora  $j$  prema čvoru  $k$  može biti rotirano za kut  $\alpha_{ji}$ , kako bi se uskladilo sa orijentacijom čvora  $i$  gdje se kut  $\alpha_{ji}$  i prilagodeno mjerenje  $[\theta_{jk}]_i$  mogu izračunati na jednostavan način korištenjem sljedećih jednadžbi:

$$\alpha_{ji} = \theta_{ji} - \theta_{ij} - \pi \quad (3.6)$$

$$[\theta_{jk}]_i = \theta_{jk} + \alpha_{ji} \quad (3.7)$$

Ako se ova procedura iterira, primjerice, pri spajanju lokalnih koordinatnih sustava, naposljetku svi čvorovi mogu dobiti informaciju o svojoj orijentaciji unutar jedinstvenog koordinatnog sustava. Aplicira li se ova metoda na prethodno izvedene zaključke o jedinstvenosti rješenja, tada se mora uz translaciju i skaliranje uključiti i globalna rotacija kao jedna od trivijalnih paralelno krutih transformacija.



**Slika 3.3.** Primjer prilagodbe mjerenja između čvorova  $j$  i  $k$  lokalnom koordinatnom sustavu čvora  $i$ .

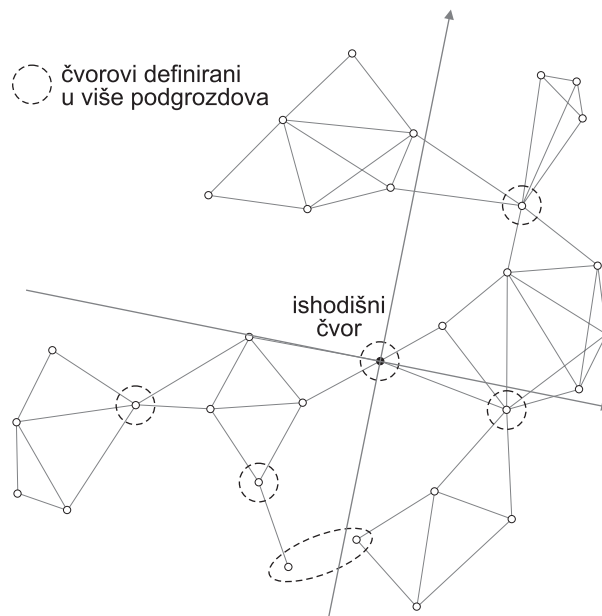
### 3.4. Mrežne strukture

Kako bi se na najbolji način iskoristili rezultati analize rješivosti problema lokalizacije za zadanu topologiju mreže, u ovom dijelu definiraju se nove mrežne strukture. Pod pojmom mrežne strukture, misli se na logičke dijelove mreže koji su bazirani na topologiji i svojstvima krutosti, a egzistiraju u memoriji čvorova. Potrebno je osigurati takav algoritam da svaki čvor ima ispravnu informaciju u kojoj se globalnoj strukturi nalazi. Ove strukture se koriste pri projektiranju i analizi raspodijeljenog algoritma lokalizacije temeljenog na određivanju smjera dolaska signala.

**Grozd** je skup čvorova čiji su izmjereni smjerovi dolaska signala definirani u koordinatnom sustavu ishodišnog čvora kao što je to prikazano na slici 3.4. *Ishodišni čvor grozda* je čvor u čijoj memoriji se grade i održavaju podaci o grozdu, a smješten je u ishodištu koordinatnog sustava grozda. Potrebno je naglasiti kako lokacije čvorova unutar grozda ne moraju biti jedinstvene.

**Podgrozdovi** su najveći mogući podskupovi grozdova čiji graf je paralelno krut kao što je prikazano na slici 3.5. Lokacije čvorova unutar grozdova podložne su samo trivijalnim transformacijama (translacija i skaliranje) i to samo na razini cijelog podgrozda. Neki čvorovi mogu biti članovi više podgrozdova, dapače, podgrozdovi mogu biti međusobno povezani samo kroz takve zajedničke čvorove.



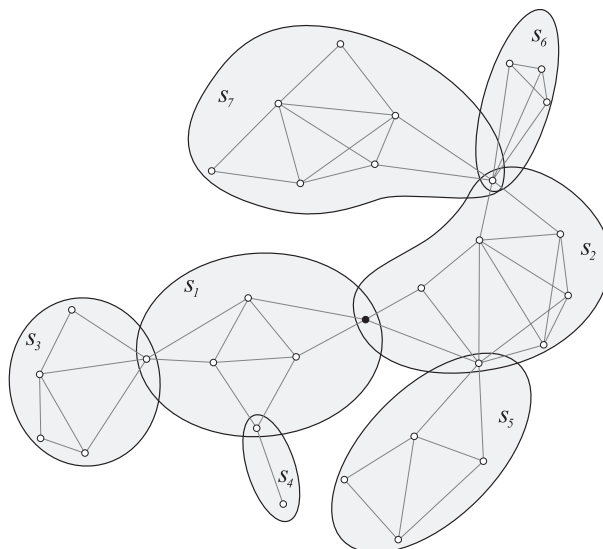


Slika 3.4. Primjer grozda i njegovog koordinatnog sustava.

**Bazni čvor** je čvor koji povezuje podgrozd sa susjednim podgrozdom koji je u strukturi stabla podgrozdova na putu prema ishodišnom čvoru odnosno korijenu stabla. Svaki podgrozd ima po jedan bazni čvor, a i sami ishodišni čvor je bazni čvor. Algoritam mora osigurati da bazni čvorovi imaju istu lokaciju unutar različitih podgrozdova čiji su članovi, što se postiže translacijom podgrozdova koji su udaljeniji od ishodišnog čvora. Na primjer, ako se transformira podgrozd  $s_2$ , lokacije baznih čvorova podgrozdova  $s_5$ ,  $s_6$  i  $s_7$  će se promijeniti, stoga ti podgrozdovi moraju biti translaticirani, tako da lokacija njihovog baznog čvora bude u skladu sa lokacijom u podgrozdu  $s_2$ . Potrebno je naglasiti da je, kako su rotacija i translacija ograničene unutar podgrozdova, jedina moguća transformacija, skaliranje i to samo onih krajnjih podgrozdova koji ne sadrže bazne čvorove drugih podgrozdova.

**Vezni podgrozd** je posebna vrsta podgrozda koji se sastoji od samo dva čvora, kao primjerice podgrozd  $s_4$  na slici 3.5. Jedan od tih čvorova je bazni čvor i lokacija mu je određena lokacijom u podgrozdu nižeg reda (onome koji je na putu prema ishodišnom čvoru), dok drugi čvor može biti (i) slobodan, (ii) bazni čvor podgrozda višeg reda ili (iii) običan čvor nekog drugog podgrozda, kao što je to slučaj u primjeru na slici. U zadnjem slučaju, čvor ne mora imati jedinstvenu lokaciju u oba podgrozda čiji je član.

**Inicijalni grozd** je grozd koji se sastoji samo od ishodišnog čvora i njegovih susjeda. Oni se stvaraju i lokaliziraju u svakom čvoru mreže, na samom početku lokalizacijskog procesa raspodijeljenog algoritma.

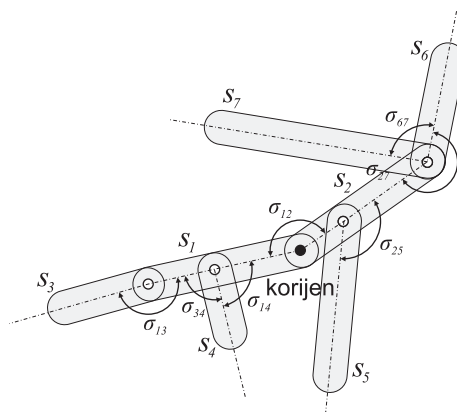


**Slika 3.5.** Grozd podijeljen na podgrozdove sa označenim baznim čvorovima na presjecima podgrozdova.

**Globalni grozd** je grozd koji sadrži sve čvorove mreže. Potrebno je naglasiti da bi stvaranje globalnog grozda bilo teoretski moguće, graf mreže mora biti povezan.

**Šume** su podgrozdovi globalnog grozda.

Daljnjom analizom grozd se može prikazati kao struktura slična stablu koja se sastoji od podgrozdova povezanih zajedničkim čvorovima sa korijenom u ishodišnom čvoru grozda, kao što je to prikazano na slici 3.6.



**Slika 3.6.** Podgrozdovi prikazani kao dijelovi stabla čije grane su ograničene izmjenjenim azimutima baznih čvorova  $\sigma_{ij}$  i njihovim lokacijama.

## 3.5. Operacije nad mrežnim strukturama

Kako bi se uspješno lokalizirali svi čvorovi mreže jasno je da se, nakon lokalizacije inicijalnih podgrozdova unutar algoritma, moraju spajati mrežne strukture, kako bi na kraju dobili jedan globalni grozd. Pokazalo se kako je osnovna operacija u tom procesu spajanje dva grozda. Pod operacijom spajanja podrazumijeva se registracija svih čvorova *izvorišnog* grozda u koordinatni sustav *odredišnog* grozda. Najprije je potrebno rotirati izvorišni grozd, kako bi se uskladile orijentacije koordinatnih sustava po principu koji je opisan u poglavlju 3.3.2, a zatim spojiti podgrozdove koji se mogu spojiti.

Kako bi se odredili minimalni uvjeti koji su potrebni da bi se dva podgrozda uspješno spojila u jedan, odnosno da ne postoji mogućnost višestrukih lokacija unutar podgrozda, potrebno je dodatno analizirati teoriju krutosti.

Rezultat spajanja dva podgrozda je novi podgrozd koji sadrži sve čvorove podgrozdova koji se spajaju. Novi podgrozd po definiciji mora biti paralelno kruta formacija. Metode za generiranje paralelno krutih formacija se sastoje od dvije operacije proširivanja [7].

### 3.5.1. Proširivanje

Prije formalnih definicija operacija proširivanja, definirana je minimalno kruta formacija (engl. *minimally rigid formation*) kao kruta formacija sa najmanjim mogućim brojem ograničenja. Stupanj čvora označava broj bridova koji su incidentni s njim. Sada se mogu definirati dvije operacije proširivanja.

**0-proširivanje (Dodavanje vrha):** Neka je  $\mathbb{G} = \{\mathcal{V}, \mathcal{L}\}$  graf sa vrhom  $i$  stupnja 2 u 2-dimenzionalnom prostoru; neka  $\mathbb{G}^* = \{\mathcal{V}^*, \mathcal{L}^*\}$  označava podgraf dobiven brisanjem vrha  $i$  i bridova koji su incidentni s njim. Slijedi da je  $\mathbb{G}$  minimalno paralelno krut ako i samo ako je i  $\mathbb{G}^*$  minimalno paralelno krut.

**1-proširivanje (Dijeljenje brida):** Neka je  $\mathbb{G} = \{\mathcal{V}, \mathcal{L}\}$  graf sa vrhom  $i$  stupnja 3, neka je  $\mathcal{V}_i$  skup vrhova povezanih sa vrhom  $i$ , i neka je  $\mathbb{G}^* = \{\mathcal{V}^*, \mathcal{L}^*\}$  podgraf dobiven brisanjem vrha  $i$  i njegova tri incidentna brida. Slijedi da je  $\mathbb{G}$  minimalno paralelno krut ako i samo ako postoji par vrhova  $j, k$  iz skupa  $\mathcal{V}_i$  takav da brid  $(j, k)$  nije član  $\mathcal{L}^*$  i da je graf  $\mathbb{G}' = \{\mathcal{V}^*, \mathcal{L}^* \cup (j, k)\}$  minimalno paralelno krut.

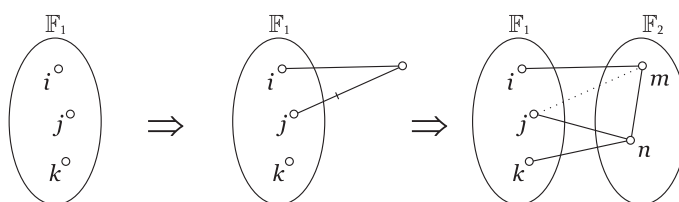
### 3.5.2. Spajanje podgrozdova

Spajanje dvije rigidne formacije sa udaljenostima kao ograničenjima, analizirano je i opisano u [25], dok je u ovom dijelu ta analiza proširena i primjenjena na mreže sa ograničenjima azimuta u dvodimenzionalnim prostorima formuliranjem i dokazivanjem sljedećih teorema.

**Teorem 2.** *Pretpostavka je da su dvije paralelno krute formacije  $\mathbb{F}_1$  i  $\mathbb{F}_2$  povezane skupom veza (bridova, ograničenja)  $\mathcal{L}$ . Slijedi da je  $\mathbb{F}_1 \cup \mathbb{F}_2 \cup \mathcal{L}$  globalno kruta ako su zadovoljena sljedeća dva uvjeta:*

1. *Barem tri rubna vrha veza iz skupa  $\mathcal{L}$  se nalaze u formaciji  $\mathbb{F}_1$  i barem dva se nalaze u formaciji  $\mathbb{F}_2$ .*
2.  *$\mathcal{L}$  sadrži barem tri veze.*

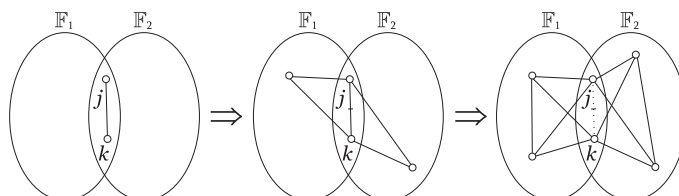
*Dokaz.* Izaberu se tri vrha  $i, j, k$  iz formacije  $\mathbb{F}_1$ . Koristeći slijed koji se sastoji od jednog dodavanja vrha i jednog dijeljenja brida može se povezati ta tri vrha koristeći tri veze sa vrhovima  $m, n$  u formaciji  $\mathbb{F}_2$ . Slika 3.7 prikazuje ove dvije operacije, a rezultirajuća formacija je paralelno kruta. Slijedi da se može, počevši od dva vrha  $m, n$ , konstruirati proizvoljnu paralelno krutu formaciju  $\mathbb{F}_2$  bez da se mijenja formacija  $\mathbb{F}_1$  i tri umetnute veze.  $\square$



**Slika 3.7.** Teorem 2 – dodavanje vrha i dijeljenje brida.

**Teorem 3.** *Ako dvije paralelno krute formacije  $\mathbb{F}_1$  i  $\mathbb{F}_2$  dijele najmanje dva vrha formacija  $\mathbb{F}_1 \cup \mathbb{F}_2$  je paralelno kruta.*

*Dokaz.* Ako su  $j$  i  $k$  vrhovi zajednički formacijama  $\mathbb{F}_1$  i  $\mathbb{F}_2$  tada se može dodati implicitno ograničenje azimuta između ta dva vrha u obliku brida  $(j, k)$  i to učiniti u obje paralelno krute formacije  $\mathbb{F}_1$  i  $\mathbb{F}_2$  koje sada dijele taj novi brid. Počevši od brida  $(j, k)$  slijedom operacija dodavanja vrhova i dijeljenja bridova može se, nezavisno, unutar svake od formacija, konstruirati proizvoljne paralelno krute formacije koje će dijeliti vrhove  $j$  i  $k$  i uvijek će se zadržati svojstvo da je formacija  $\mathbb{F}_1 \cup \mathbb{F}_2$  paralelno kruta.  $\square$



**Slika 3.8.** Teorem 3 – dodavanje vrha i dijeljenje zajedničkog brida  $(j, k)$ .

Važno je naglasiti da je spajanje podgrozdova u teoremu 2, izvedeno tako da su se uz korištenje zajedničkog čvora, umetale nove veze. Iako se u samoj mreži sa zadanom topologijom ne mogu stvarati nove veze, mogu se iskoristiti rezultati teorema tako da se spajaju podgrozdovi povezani već postojećim veznim podgrozdovima.

Definicije i rezultate iz ovog poglavlja koriste se kako bi projektirali i opisali metodu lokalizacije kojom se ostvaruje važan cilj, a to je stvaranje šuma sa što većim postotkom čvorova. Drugim riječima, algoritam treba lokalizirati čvorove unutar teoretski najbrojnijih mogućih, paralelno krutih struktura.

## 4. Prijedlog metode lokalizacije

U ovom poglavlju obrađeni su matematički i algoritamski principi po kojima je riješen problem lokalizacije definiran u poglavlju 2. Razlika između definiranog problema i sličnih problema rješavanih u radovima navedenim u tom poglavlju je ta što se ovdje rješava problem lokalizacije čvorova s pretpostavkom da u mreži ne postoje čvorovi koji znaju svoju apsolutnu lokaciju (neusidrena mreža). Jasno je da je takav problem općenitiji te da, ukoliko postoji mogućnost apsolutne lokalizacije barem dva čvora u ravnini, uvijek se može transformirati relativni koordinatni sustav u apsolutni.

Kao što je navedeno, prilikom lokalizacije čvorova uz mjerenje azimuta jedan od problema postojećih algoritama je propagacija mjerne pogreške. Autori u [19] predlažu metode za ograničenje grešaka propagacije kroz ograničenje propagacije samih mjerenja. Time se ujedno ograničava postotak čvorova koji će na kraju biti lokaliziran odnosno uvodi se potreba za velikim brojem čvorova koji poznaju apsolutnu lokaciju.

Rješenje ovog problema u ovom radu je zamišljeno kroz korištenje raspodijeljenog algoritma pri čemu se ne misli samo na raspodjelu dijelova zadatka pri računanju već i na međusobnu suradnju čvorova. Pristup se općenito može podijeliti u dvije faze, a to su lokalizacija inicijalnih grozdova i spajanje inicijalnih grozdova u globalni grozd.

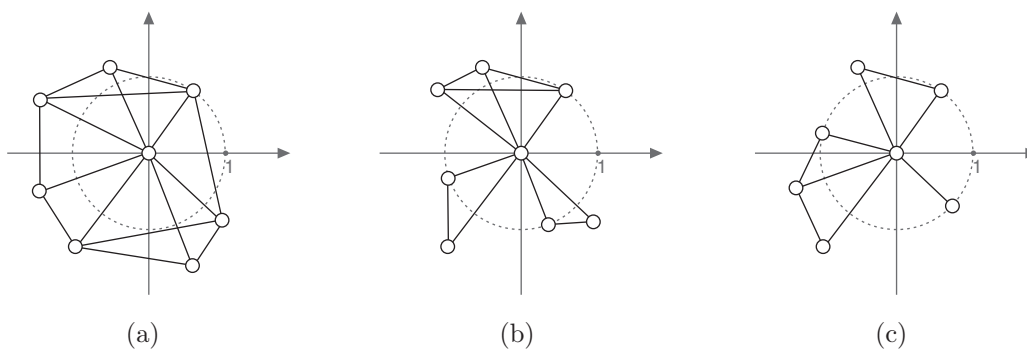
Takvim rješenjem omogućava se da performanse algoritma ne ovise o položaju i broju sidara. Problem položaja sidara može se analizirati odvojeno i kao takav odvojeno i rješavati. To je primjerice učinjeno u [21], u kojem se i formalno odvajaju apsolutna i relativna pogreška lokalizacije. Dodatno se stvara preduvjet za izbjegavanje akumulacije pogrešaka jer se iterativnim spajanjem inicijalnih grozdova usrednjuje pogreška koja je napravljena pri njihovoj lokalizaciji.

### 4.1. Metoda lokalizacije inicijalnih podgrozdova

U prethodnim poglavljima su opisana dva vrlo bitna preduvjeta uspješne lokalizacije. To je najprije izbor pristupa i izbor modaliteta mjerene veličine i mjerne metode, a zatim teoretska analiza uspješnosti lokalizacije s obzirom na topologiju i izabrani modalitet. Najvažniji rezultati te analize su definicije novih mrežnih struktura koje se

sada mogu iskoristiti.

Inicijalni grozdovi se sastoje od čvora koji izvodi njegovo formiranje, odnosno lokalizaciju, a to je u našem slučaju svaki čvor mreže, i od njegovih susjeda. Ako se sagledaju moguće kombinacije topologija podgrafova koji se sastoje od čvora i njegovih susjeda, dolazi se do zaključka da takve topologije ne moraju biti paralelno krute. Na slici 4.1 prikazani su primjeri inicijalnih grozdova s različitim topologijama i mogućnostima lokalizacije. Ukoliko je topologija paralelno kruta kao na slici 4.1(a), svi čvorovi grozda mogu biti jedinstveno lokalizirani do razine kongruentnih transformacija. Kako bi algoritam mogao raditi sa konkretnim lokacijama, kongruentne transformacije se mogu proizvoljno ograničiti. U ovom slučaju translacija će se ograničiti položajem ishodišnog čvora u ishodište, a skaliranje na način da se udaljenost do najbližeg čvora susjeda postavi na 1. Orijentacija je ograničena sa izmjerenim smjerovima prema susjedima. Slučaj na slici 4.1(b) je složeniji jer prikazuje inicijalni grozd koji se sastoji od dva podgrozda. U tom slučaju lokacije čvorova koji nisu u istom podgrozdu su nezavisne jedna o drugoj odnosno grozd je podložan nekongruentnim transformacijama. Jedino ispravno rješenje je lokalizirati svaki podgrozd nezavisno i to na isti način kao i u slučaju paralelno krutog grozda. Poveznica među podgrozdovima je ishodišni čvor, koji mora biti u ishodištu za sve podgrozdove. U trećem slučaju na slici 4.1(c), topologija je takva da jedan od podgrozdova, ima samo jednog člana ne računajući ishodišni čvor. Takav podgrozd se naziva veznim podgrozdom i jedina informacija koju je bitno poznavati je azimut tog čvora u koordinatnom sustavu ishodišnog čvora, a taj smjer zapisuje se u obliku jediničnog vektora što je u potpunosti u skladu sa prethodnim metodama zapisivanja lokacija čvorova u podgrozdu.



Slika 4.1. Moguće topologije inicijalnih grozdova.

### 4.1.1. Iterativna triangulacija

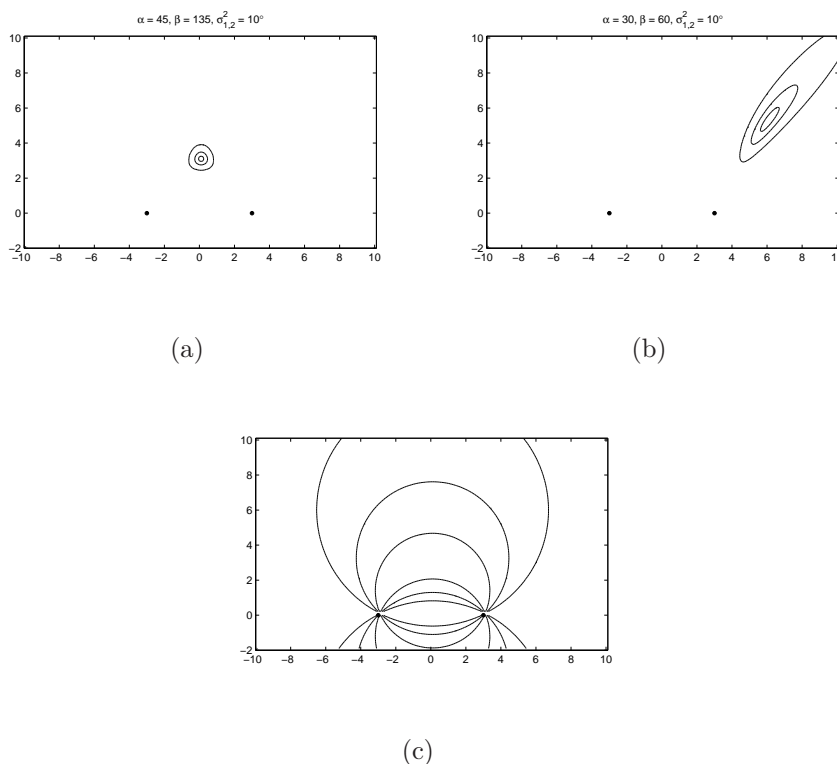
Postoji više metoda kojima se može lokalizirati čvorove podgrozda. Prvi i najizvorniji bi bila iterativna triangulacija. Postupak je sljedeći: najprije se postavi ishodišni čvor u ishodište, zatim se proizvoljno odabere jedan čvor i postavi na je-

diničnu kružnicu prema azimutu koji je izmjeren za taj čvor. Slijedi određivanje lokacije trećeg čvora koji je povezan sa ishodišnim čvorom te čvorom na jediničnoj kružnici. Dalje se postupak ponavlja sve dok se ne lokaliziraju svi čvorovi podgrozda.

Problemi kod ove metode su neiskorištavanje svih ograničenja azimuta i to što točnost lokalizacije ovisi o redosljedu po kojem se izvodi triangulacija. Uzrok drugom problemu je korelacija između mjerne nesigurnosti i položaja čvorova koji obavljaju triangulaciju. Ta korelacija prikazana je na slici 4.2. Na slikama 4.2(a)–(b) označene su linije koje označavaju jednaku vjerojatnost lokalizacije za dvije različite konfiguracije. Čvorovi koji obavljaju triangulaciju smješteni na koordinatama  $(-3, 0)$  i  $(3, 0)$ , a devijacija mjerenja kuta je  $\sigma = 10^\circ$ .

Prema tome, ukoliko se u prvom koraku učini velika pogreška, zbog nepovoljnog položaja čvorova ta greška se propagira u narednim triangulacijama.

Mjera kojom je moguće estimirati nesigurnost rezultata triangulacije naziva se geometrijsko slabljenje preciznosti (engl. *geometric dilution of precision* – *GDOP*). Na slici 4.2(c) prikazane su konture jednakog iznosa GDOP-a. Koristeći se metodom određivanja ove mjere, detaljno opisane u [26], može se odrediti najpovoljniji redosljed triangulacije.



Slika 4.2. Geometrijsko slabljenje preciznosti.



Na taj način se minimizira utjecaj redoslijeda triangulacije, ali problem neiskorištavanja svih ograničenja ostaje. Metoda koja rješava i taj problem je opisana u sljedećem potpoglavlju.

## 4.2. Robusna angulacija

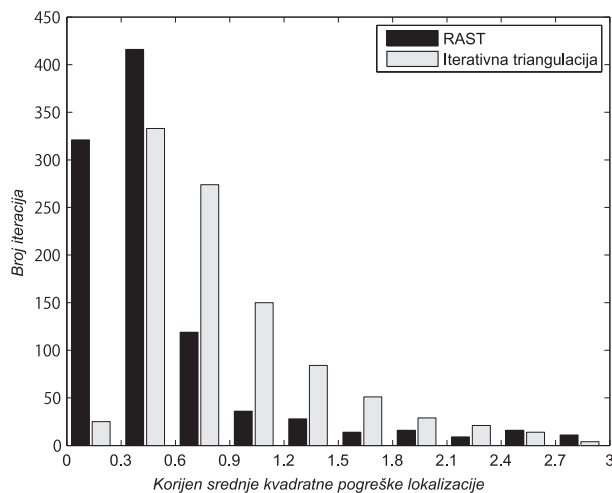
Angulacija je generalizirani koncept triangulacije kojom se određuje lokacija čvora uz pomoć skupa izmjerenih azimuta prema čvorovima čija je lokacija poznata. Za uspješnu lokalizaciju u ravnini, potreban broj čvorova koji poznaju svoju lokaciju je dva, pa se iz tog razloga najviše koristi metoda triangulacije. Ukoliko je mjerenje azimuta podložno grešci, više čvorova će biti potrebno kako bi dobili što točniju estimaciju.

U ovom slučaju cilj je lokalizirati čvorove inicijalnog grozda koristeći sve dostupne izmjerene azimute od ishodišnog čvora prema susjedima i obratno te među samim susjedima. Metoda koja istovremeno lokalizira u relativnom koordinatnom sustavu sve čvorove, bilo koje paralelno krute mreže koristeći mjerenja azimuta između čvorova susjeda, opisana je u [5] i naziva se robusna angulacija korištenjem potprostornih tehnika (engl. *Robust Angulation using Subspace Techniques – RAST*).

Kao što joj i ime govori, metoda se temelji na identifikaciji vektora koji razapinju potprostor rješenja. Kako je već navedeno, jedinstveno rješenje je u biti formacija točaka koja zadovoljava ograničenja smjera, a jedine moguće transformacije su skaliranje i translacija duž  $x$  i  $y$  osi. Njihov rezultat su kongruentne formacije točaka koje predstavljaju rješenja problema lokalizacije. Prema tome potprostor je razapet vektorima  $\mathbf{p}$ ,  $\mathbf{v}_x$  i  $\mathbf{v}_y$ , gdje vektori  $\mathbf{v}_x = [1 \ 0 \ 1 \ 0 \dots]^T$  i  $\mathbf{v}_y = [0 \ 1 \ 0 \ 1 \dots]^T$  reprezentiraju translaciju po  $x$  odnosno po  $y$  osi, a vektor  $\mathbf{p}$  odgovara skaliranju. Vrijednosti vektora  $\mathbf{p}$  su takve da reprezentiraju upravo lokacije u formaciji točaka koja ima centroidu u ishodištu i odgovara rješenju problema lokalizacije neusidrene mreže.

Potprostor se utvrđuje tehnikom rastava singularnih vrijednosti. Ukoliko nema dovoljno izmjerenih azimuta, odnosno grozd nije paralelno krut, potprostor će biti dimenzija većih od 3. Također za sam rastav je potrebno poznavati azimute u odnosu na jedinstveni smjer za cijelu mrežu. Drugim riječima, potrebno je odrediti međusobne orijentacije čvorova. Taj problem je rješiv na način koji je prikazan u odjeljku 3.3.2, a kako se mreža sastoji od čvorova susjeda koji i inače moraju slati svoja mjerenja ishodišnom čvoru, rješenje tog problema ne predstavlja dodatno komunikacijsko opterećenje. Ishodišni čvor tada može na osnovu tih mjerenja odrediti relativne orijentacije svih čvorova susjeda. Detaljan izvod računanja vektora  $\mathbf{p}$  dan je na kraju rada kao Prilog 1.

Veća točnost lokacija dobivenih metodom RAST u odnosu na iterativnu triangulaciju prikazana je histogramom na slici 4.3. Simulacije su izvedene na uzorku od 1000 iteracija slučajnih konfiguracija inicijalnih grozdova, koji su se sastojali od ishodišnog čvora i 7 čvorova susjeda.



Slika 4.3. Histogram.

### 4.3. Put spajanja grozdova u globalni grozd

Nakon što su određene lokacije čvorova u svim inicijalnim grozdovima, moglo bi se reći da je mreža lokalizirana. Jedini problem je što svaki čvor mreže ima svoje viđenje lokacije svojih susjeda u mreži te nema saznanje o mreži izvan domene svog inicijalnog grozda. Ono što je potrebno učiniti je da čvorovi postanu svjesni cjelokupne mreže i lokacije njihovog grozda u toj mreži, odnosno lokacije njih samih unutar globalnog grozda. Činjenica koja nam pomaže u tom procesu je ta da se svi inicijalni grozdovi međusobno preklapaju te ih se može odgovarajućim kongruentnim transformacijama njihovih koordinatnih sustava spajati.

Spajanje, kao raspodijeljeni proces, mora se odvijati organizirano, po određenoj mrežnoj strukturi, kako bi se izbjegli potpuni zastoji (engl. *deadlock*) i petlje (engl. *loop*) te kako bi se sa sigurnošću detektirao kraj tog procesa (engl. *global termination detection*).

Jedan od najefikasnijih načina organizacije povezane mreže kojom se određuju komunikacijski putovi i postiže kontrola nad trenutnim funkcijama čvorova je konstruiranje razapinjajućeg stabla (engl. *spanning tree*) kao mrežne strukture. Uz pomoć stabla može se provoditi kontrola, a paralelno uz konstruiranje stabla može se obaviti i izbor vođe (engl. *leader election*) koji će tu kontrolu izvoditi. Obično u takvim strukturama vođa se nalazi u korijenu stabla.

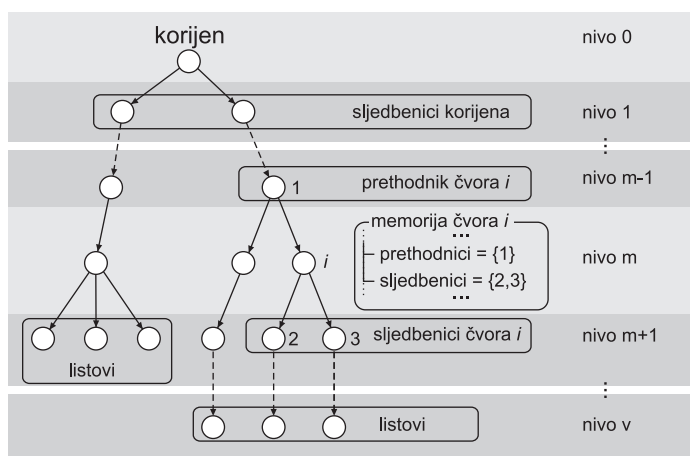
#### 4.3.1. Optimalno stablo

Više o detaljima raspodijeljenog algoritma za konstrukciju stabla, komunikacijskoj cijeni konstrukcije i prednostima u 5. poglavlju, dok će se u nastavku opisati sama

struktura stabla i na koji način će se ona koristiti u lokalizaciji.

Stablo se u teoriji grafova definira kao graf u kojem su bilo koja dva vrha povezana samo sa jednim putem. Drugim riječima, stablo ne sadrži cikluse. Ako se pretpostavi da je mreža povezana i sadrži  $n$  čvorova, onda stablo predstavlja razapinjajući podgraf grafa te mreže koji se sastoji od  $n - 1$  bridova te se naziva *razapinjajuće stablo*.

Ukoliko se odredi jedan vrh stabla koji će biti korijen, a to može biti bilo koji vrh, tada se stablo naziva korijensko (engl. *rooted tree*). Korijensko stablo može se opisati i kao usmjereni graf (digraf) kojem su bridovi, koji se u digrafu nazivaju lukovi (engl. *arcs*), usmjereni od korijena prema ostalim vrhovima. Takva definicija je moguća jer u stablu svaki brid povezuje dva različita nivoa stabla te je usmjeren od nižeg prema višem kao što je prikazano na slici 4.4. Čvorovi susjedi nižeg nivoa nazivaju se *prethodnici*, a višeg nivoa *sljedbenici*. Vrhovi koji imaju prethodnike i sljedbenike nazivaju se unutarnjim vrhovima, a vrhovi koji imaju samo prethodnike se nazivaju listovi stabla.



Slika 4.4. Korijensko stablo visine  $v$ .

Graf mreže može imati više različitih korijenskih razapinjajućih stabala. Ako se težinski graf definira kao graf u kojem svaki od bridova posjeduje težinu  $w(e)$  odnosno definira se funkcija  $w : \mathcal{E} \mapsto \mathbb{R}$ , tada postoji jedno ili više stabala sa ukupnom težinom bridova koja je minimalna u odnosu na sva druga stabla. Takva stabla nazivaju se optimalna stabla (engl. *minimum-cost spanning tree – MST*). Ukoliko je težina bridova jedinstvena za svaki brid, tada postoji samo jedan mogući MST.

### 4.3.2. Idealna težina bridova

Pod pretpostavkom da kvaliteta lokalizacije, odnosno samog spajanja inicijalnih grozdova, ovisi o putu po kojem se oni spajaju, potrebno je predložiti funkciju  $w$  po kojoj će se određivati težine bridova. O toj funkciji naposljetku ovisi i izgled samog

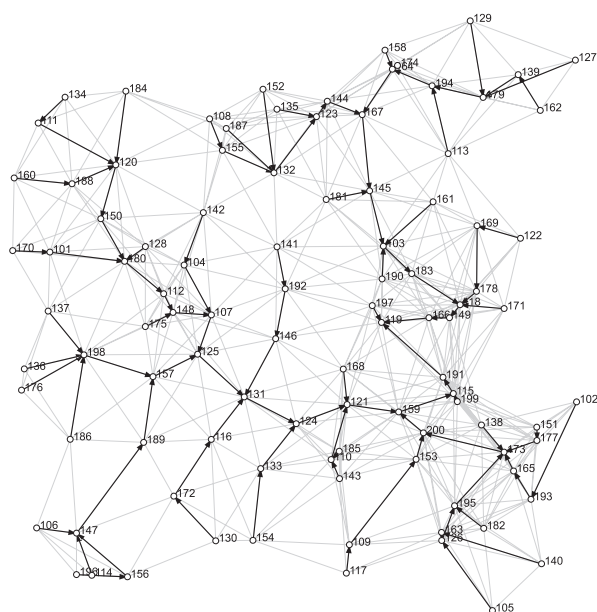
stabla, pa se ona mora odrediti na način da spajanje grozdova po tom stablu unosi što manju grešku u proces lokalizacije. U nastavku je navedeno i opisano nekoliko prijedloga funkcije  $w$ .

Za početak se traži funkcija koja će svakom bridu grafa dati jedinstvenu težinu, tako da i MST bude jedinstven. Prema iznesenoj teoriji koja će u narednom dijelu ovog poglavlja biti upotpunjena i formalnim metodama spajanja, može se zaključiti da će se najkvalitetnije spajanje dva grozda obaviti ako ti grozdovi imaju veliki broj zajedničkih čvorova. Prema tome, težina brida  $e$  bi bila jednaka broju zajedničkih čvorova između grozdova koje taj brid povezuje. Kako se stvara stablo s minimalnom ukupnom težinom, a žele se koristiti bridovi s najvećim brojem zajedničkih čvorova, težinu je potrebno pomnožiti sa  $-1$ . Nadalje potrebno je zadovoljiti uvjet da bridovi imaju jedinstvenu težinu, što u ovom slučaju, gdje je težina cijeli broj, nije slučaj, pa se mora svakoj težini pridodati slučajni broj manji od 1. Ova verziju stabla naziva se MAXNODE.

Sljedeća funkcija koju je potrebno ispitati ne ovisi o parametrima lokalizacije – ovisi samo o ID broju čvorova koje povezuje brid. Kako bi svi bridovi imali jedinstvenu težinu, prvo se gleda manji ID od dva čvora koje brid povezuje, a ako je on isti onda se gleda vrijednost ID broja drugog čvora. Iako funkcija djeluje nasumična, s obzirom da ne postoji korelacija između lokacije čvora i njegovog ID broja, stablo generirano na taj način ima vrlo specifičan izgled. Označava se sa ID.

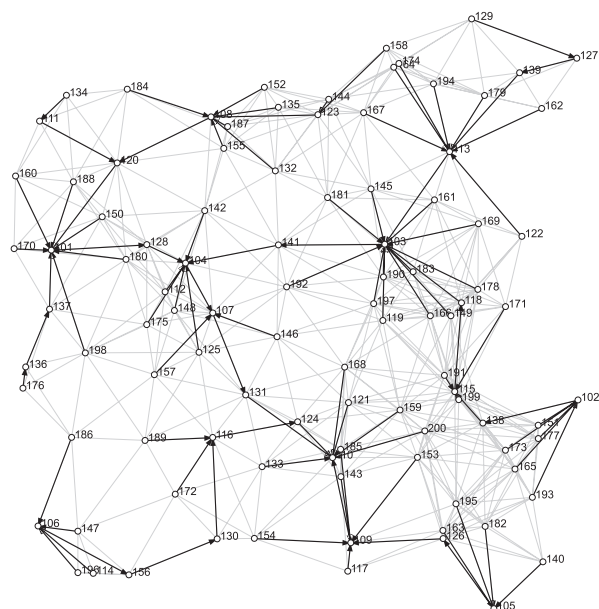
Treći oblik stabla koji se ispituje u ovom radu ne ovisi o težini već o visini stabla. Visina stabla je najveći nivo stabla odnosno dužina najdužeg puta od korijena prema listu. Ovo je komunikacijski najzahtjevnija struktura za kreiranje jer je za raspodijeljeno generiranje stabla s minimalnom visinom potrebno kreirati tablice usmjeravanja s najkraćim putem (engl. *shortest path routing tables*) za svaki čvor mreže. Stablo s minimalnom visinom označava se sa MHST.

Primjeri stabala za sve tri funkcije prikazani su na slici 4.5 na kojoj se vidi kako se za istu topologiju dobivaju sasvim različite strukture.

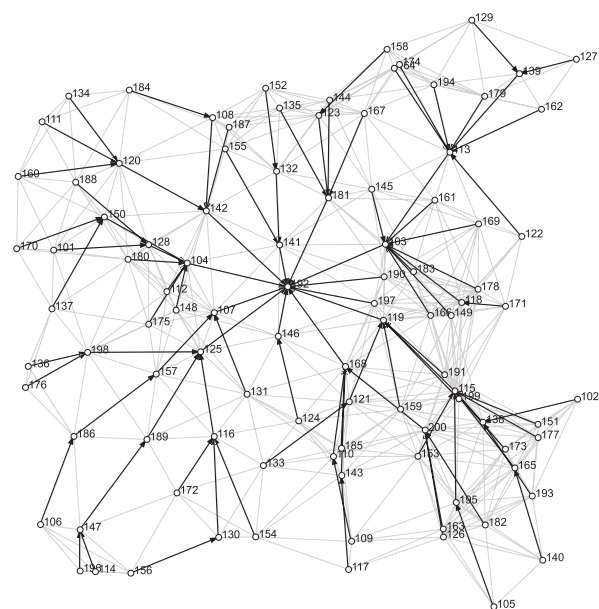


(a) MAXNODE

Slika 4.5. Izgledi stabala za istu topologiju mreže i za različite predložene metode kreiranja.



(b) ID



(c) MHST

**Slika 4.5.** Izgledi stabala za istu topologiju mreže i za različite predložene metode kreiranja (nastavak).

## 4.4. Spajanje grozdova

Nakon što je utvrđeno kojim će se redosljedom grozdovi spajati potrebno je analizirati koji su sve scenariji mogući prilikom spajanja dva grozda i koje će se akcije u svakom od tih scenarija primijeniti. Spajanje grozdova je proces lokaliziranja svih čvorova *izvorišnog* grozda  $C_i$  unutar koordinatnog sustava *odredišnog* grozda  $C_j$  i on se odvija lokalno u čvoru  $j$ .

Najprije je potrebno primijetiti kako se spajanje grozdova  $C_i$  i  $C_j$  u biti svodi na spajanje njihovih podgrozdova  $s_s \in C_i$  i  $s_d \in C_j$ . Treba odrediti koji podgrozdovi će biti prvi međusobno spojeni i zatim ih spojiti. Te dvije akcije se odvijaju naizmjenično, sve dok svi čvorovi izvorišnog grozda ne postanu članovi odredišnog grozda. Nakon toga pokazati će se kako je potrebno provjeriti da li je moguće spojiti međusobno neke od podgrozdova novokreiranog grozda. Ova se akcija naziva *međuspajanje* i obavlja se na samom kraju procesa spajanja.

### 4.4.1. Opis metoda spajanja

U ovom odjeljku analizirane su metode na raspolaganju prilikom spajanja dva podgrozda. Sve metode se temelje na izračunu parametara transformacije kojima se koordinate čvorova izvorišnog grozda prebacuju u koordinatni sustav odredišnog grozda. Parametri transformacije su faktor skaliranja  $\xi$ , vektor translacije  $\vec{t}$  i matrica rotacije  $\mathbf{R}$ , a metode na raspolaganju su:

**STITCHNODES** ovom metodom računaju se parametri koristeći se lokacijama čvorova koji su zajednički za oba podgrozda. Kao što je i izneseno u teoremu 3, ako postoje barem dva zajednička čvora između dviju paralelno krutih formacija, mogu se izračunati jedinstveni parametri transformacije.

**STITCHLINKS** ako podgrozdovi imaju samo jedan zajednički čvor, ali unutar odredišnog čvora postoji vezni podgrozd sa po jednim čvorom u svakom od dva podgrozda (spaja te podgrozdove), prema teoremu 2 mogu se jedinstveno odrediti svi parametri transformacije.

**SIMPLEADDITION** ako između podgrozdova postoji samo jedan zajednički čvor tada se mogu odrediti samo dva parametra – faktor skaliranja  $\xi$  i vektor translacije  $\vec{t}$ . U tom slučaju, svi čvorovi izvorišnog podgrozda čine novi podgrozd u odredišnom grozdu, pri čemu zajednički čvor postaje bazni čvor.

Slučaj u kojem nema zajedničkih čvorova između nijednog od podgrozdova susjednih grozdova nije moguć, jer barem ishodišni čvorovi tih grozdova imaju međusobno jedan drugoga u svojim inicijalnim grozdovima. Drugim riječima, svaki podgrozd iz  $C_i$  može postati dijelom grozda  $C_j$  koristeći se trima nabrojanim metodama.

Zajednička transformacija za sve podgrozdove iz  $C_i$  je rotacija, jer su sva ograničenja azimuta definirana u odnosu na orijentacije svojih ishodišnih čvorova  $i$  i  $j$ . Kako bi se uskladili rotacijom, vektor lokacija čvorova iz  $C_i$  množi se sa matricom rotacije  $\mathbf{R}$

$$\mathbf{R} = \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) \\ \sin(\varphi) & \cos(\varphi) \end{bmatrix} \quad (4.1)$$

gdje je  $\varphi$  kut između orijentacija ishodišnih čvorova  $\varphi = \alpha_i - \alpha_j$ .

S obzirom na pogreške mjerenja smjerova najbolja estimacija **matrice rotacije**  $\mathbf{R}$  postiže se metodom STITCHNODES. Stoga se ta metoda koristi kako bi se rotiralo sve podgrozdove izvorišnog grozda. Izračun estimacije matrice  $\mathbf{R}$  koristeći lokacije  $N$  zajedničkih čvorova gdje je  $N \geq 2$  je sljedeći [3]:

- Pronađi centroide zajedničkih čvorova za oba podgrozda

$$\bar{p}_s = \frac{1}{N} \sum_{i=1}^N p_{s,i} \quad \bar{p}_d = \frac{1}{N} \sum_{i=1}^N p_{d,i} \quad (4.2)$$

gdje su  $p_s$  i  $p_d$  lokacije zajedničkih čvorova u izvorišnom odnosno odredišnom koordinatnom sustavu.

- Translatiraj sve lokacije tako da su definirane u odnosu na centroide

$$\bar{p}'_{s,i} = p_{s,i} - \bar{p}_s \quad \bar{p}'_{d,i} = p_{d,i} - \bar{p}_d \quad (4.3)$$

- Izračunaj matricu  $\mathbf{M}$  veličine  $2 \times 2$

$$\mathbf{M} = \sum_{i=1}^N \bar{p}'_{d,i} (\bar{p}'_{s,i})^T. \quad (4.4)$$

- Za matricu  $\mathbf{M}^T \mathbf{M}$  izračunaj svojstvene vrijednosti  $\lambda_1, \lambda_2$  i svojstvene vektore  $\hat{u}_1, \hat{u}_2$ . Matrica rotacije se tada može izračunati kao

$$\mathbf{R} = \mathbf{M} \left( \frac{1}{\sqrt{\lambda_1}} u_1 \hat{u}_1 + \frac{1}{\sqrt{\lambda_2}} u_2 \hat{u}_2 \right). \quad (4.5)$$

Estimacija **faktora skaliranja**  $\xi$  u STITCHNODES metodi se računa koristeći

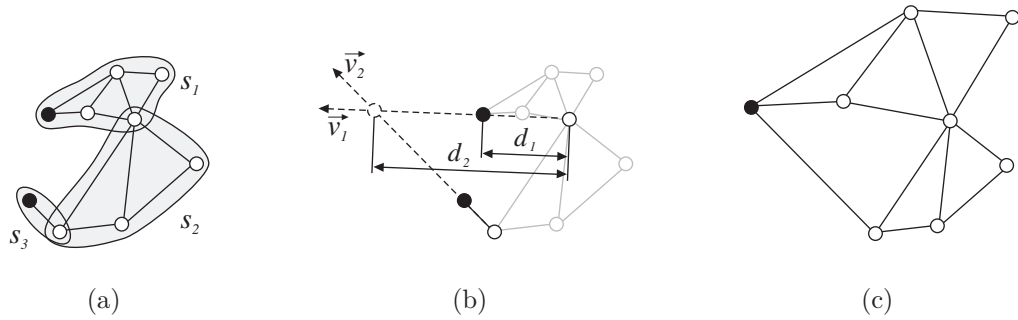
$$\xi = \sqrt{\frac{\sum_{i=1}^N \|\bar{p}'_{d,i}\|^2}{\sum_{i=1}^N \|\bar{p}'_{s,i}\|^2}}. \quad (4.6)$$



U STITCHLINKS metodi faktor skaliranja se može izračunati na način kao što je prikazano na slici 4.6(a). Dva podgrozda  $s_1$  i  $s_2$  se spajaju koristeći jedan zajednički čvor i jedan zajednički vezni podgrozd –  $s_3$ . Orijeatcije koordinatnih sustava grozdova, a samim time i pripadajućih podgrozdova su prethodno usklađene korištenjem metode STITCHNODES. Zatim se podgrozdovi transliraju tako da lokacija zajedničkog čvorova bude ista u oba podgrozda. Ispunjene kružnice označavaju lokacije istih čvorova u  $s_1$  i  $s_3$ . Faktor skaliranja može se izračunati preko sjecišta između vektora skaliranja  $\vec{v}_1$  i vektora veznog podgrozda  $\vec{v}_2$  na slici 4.6(b). Nakon što se izračunaju odgovarajuće udaljenosti  $d_1$  i  $d_2$  koristi se sljedeći izraz

$$\xi = \frac{d_2}{d_1}. \quad (4.7)$$

Konačni rezultat na slici 4.6(c) je jedan podgrozd sa čvorovima iz sva tri podgrozda. Kao što je već navedeno, u SIMPLEADDITION metodi čvorovi izvorišnog podgrozda



**Slika 4.6.** Primjer određivanja faktora skaliranja  $\xi$  za STITCHLINKS metodu.

završavaju u novom podgrozdu za koji nije definiran faktor skaliranja. U simulaciji se koristi faktor  $\xi = 1$ .

Procedura estimacije **vektora translacije**  $\vec{t}$  za sve tri metode je jednostavna. Potrebno je translirati prethodno rotiranu i skaliranu centroidu zajedničkih čvorova izvorišnog podgrozda tako da se uskladi sa centroidom zajedničkih čvorova odredišnog podgrozda

$$\vec{t} = \bar{p}_d - \xi \mathbf{R} \bar{p}_s. \quad (4.8)$$

Naposljetku, lokacije čvorova u odredišnom podgrozdu se računaju koristeći sljedeći izraz:

$$p_d = \vec{t} + \xi \mathbf{R} p_s. \quad (4.9)$$

#### 4.4.2. Izbor podgrozdova i metode spajanja

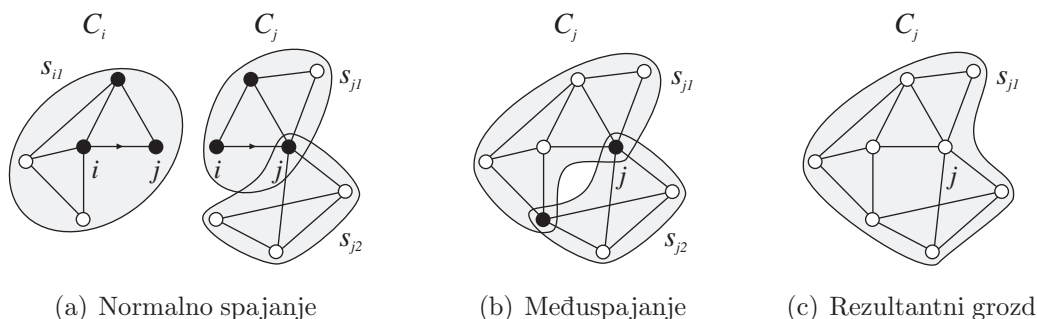
Izbor podgrozdova obavlja se na temelju usporedbe broja zajedničkih čvorova i broja zajedničkih veza između podgrozdova. Algoritam 1 prikazuje pseudokod algo-

ritma spajanja. Algoritam se sastoji od dvije *while* petlje – prva služi za spajanje izvorišnih podgrozdova  $s_s \in C_i$  čvora  $i$ , a druga za međuspajanje podgrozdova u novostvorenom grozdu.

Izbor redoslijeda parova podgrozdova  $s_s$  i  $s_d$ , koji se trebaju spojiti i izbor metode, je zadatak funkcije **selectStitch**. Odabrani podgrozdovi i metode se vraćaju u glavni program u strukturi **stitch**. Funkcija **selectStitch** je podijeljena u dva dijela. U prvom dijelu matrice **commonnodes** i **commonlinks** se ispunjavaju brojem zajedničkih čvorova i veza između odgovarajućih podgrozdova  $s_s$  i  $s_d$ , pri čemu se koristi funkcija **Common**. Ovisno o tipu spajanja (normalno ili međuspajanje) i o odnosu skupova čvorova uspoređivanih podgrozdova, vrijednosti se mogu naknadno podesiti na 0 ili  $-1$ . Na primjer, ako je podgrozd  $s_s$  podskup  $s_d$ , tada je jasno da spajanje nije potrebno i u tom slučaju vrijednost odgovarajućeg elementa matrice se postavlja na 0, a ukoliko je dodatno tip spajanja međuspajanje tada se podgrozd  $s_s$  može obrisati, pa se vrijednost postavlja na  $-1$ . U drugom dijelu vrijednosti tih matrica se koriste za odabir para podgrozdova i metode spajanja.

### 4.4.3. Međuspajanje

Posljednja faza spajanja grozdova je međuspajanje. Potreba za ovom fazom na prvi pogled nije posve jasna. Na slici 4.7 je prikazan primjer spajanja dva grozda u kojem rezultatni grozd ne bi bio ispravno konstruiran, odnosno njegovi podgrozdovi ne bi bili najveći mogući paralelno kruti podgrafovi. U prvom koraku podgrozd  $s_{i1}$  se spaja sa  $s_{j1}$  koristeći tri (ispunjena) zajednička čvora na slici 4.7(a). Kao rezultat ovog spajanja dobili smo još jedan zajednički čvor između podgrozdova  $s_{j1}$  i  $s_{j2}$  na slici 4.7(b) te sada postoji mogućnost spajanja tih čvorova međusobno unutar grozda  $C_j$ . Kako bi prikaz bio jasan, podgrozdovi su odgovarajuće rotirani i skalirani i prije samog spajanja.



Slika 4.7. Primjer u kojem je potrebno međuspajanje.

---

**Algoritam 1:** Algoritam za spajanje izvorišnih podgrozdova  $s_s \in C_i$  čvora  $i$  sa odredišnim podgrozdovima  $s_d \in C_j$  čvora  $j$ .

---

```

R=[];
stitch.method=[];
while stitch.method ~='null' do
    stitch=selectStitch(stitch,'extra');
    switch stitch.method do
        case 'stitchnodes'
            [Ci,Cj,R]=Stitchnodes(Ci,Cj,stitch,R);
        case 'stitchlinks'
            [Ci,Cj]=Stitchlinks(Ci,Cj,stitch);
        case 'simpleaddition'
            [Ci,Cj]=Simpleaddition(Ci,Cj,stitch,R);
    end
end
Ci=Cj;
stitch.method=[];
while stitch.method ~='null' do
    stitch=selectStitch(stitch,'intra');
    switch stitch.method do
        case 'stitchnodes'
            [Ci,Cj]=Stitchnodes(Ci,Cj,stitch);
        case 'stitchlinks'
            [Ci,Cj]=Stitchlinks(Ci,Cj,stitch);
        case 'deletesc'
            end
            ←Ci(stitch.sc2);
            ←Cj(stitch.sc2);
    end
end
function stitch=selectStitch(stitch,type);
begin
    forall ss ∈ Ci do
        forall sd ∈ Cj do
            if type == 'intra' and sd ⊇ ss then
                commonnodes(ss,sd)=0;
                commonlinks(ss,sd)=0;
            else if type == 'intra' and sd ⊆ ss then
                commonnodes(ss,sd)=-1;
                commonlinks(ss,sd)=0;
            else if sd ⊆ ss then
                commonnodes(ss,:)=0;
                commonlinks(ss,:)=0;
                break;
            else
                [commonnodes(ss,sd),commonlinks(ss,sd)]=Common(Ci,Cj,Cs,Cd);
            end
        end
    end
end
switch max(commonnodes) do
    case ≥ 2
        stitch=struct(method → 'stitchnodes',sc1 → sd,sc2 → ss);
    case == 1
        if commonlinks > 0 then
            stitch=struct(method → 'stitchlinks',sc1 → sd,sc2 → ss);
        else if type == 'extra' then
            stitch=struct(method → 'simpleaddition',sc1 → sd,sc2 → ss);
        end
    case == -1
        stitch=struct(method → 'deletesc',sc1 → sd,sc2 → ss);
    otherwise
        stitch=struct(method → 'null',sc1 → sd,sc2 → ss);
    end
end
end
end

```

---

## 5. Raspodijeljeni algoritam

Problem lokalizacije je rješiv uz pomoć centraliziranog algoritma. Bez obzira koju metodu koristili, princip je sličan, a svodi se na tri faze – mjerenje, prikupljanje informacija u centralnom čvoru te izvođenjem samog algoritma koji uz pomoć informacija iz same mreže pokušava što bolje estimirati lokacije čvorova. U teoriji ovakvo rješenje je dovoljno dobro, ali u praksi se javljaju problemi, pogotovo u slučajevima u kojima se pokušava osigurati učinkovita lokalizacija bez obzira na veličinu mreže. Limiti centraliziranog algoritma su računalna snaga procesora centralnog čvora, njegova potrošnja energije te potrošnja energije za prosljeđivanje poruka čvorova u njegovoj bližjoj okolini, koji predstavljaju i komunikacijska i energetska uska grla. Zatim, kod centraliziranih algoritama postoji problem robusnosti, jer činjenica da cijeli rad mreže ovisi o radu jednog čvora nije poželjna. Svi su ti problemi rješivi uglavnom uz pomoć dodatnih hardverskih mogućnosti, no takva rješenja su u kontrastu s nekim trendovima poput smanjivanja veličine i cijene čvorova odnosno mreže.

Nadalje, bežične mreže osjetila bi trebale biti u mogućnosti svoje zadatke obavljati energetski autonomno u dužim vremenskim periodima, pa je kontrolirana potrošnja te ograničene energije ključni element za postizanje tog cilja. Osim hardverskih unaprijeđenja moguće je korištenje raspodijeljenog sustava na način koji će najbolje iskoristiti sve prednosti mreže, poput raspodijeljenog prikupljanja podataka i računanja. Drugim riječima, cilj je napisati kvalitetan raspodijeljeni algoritam.

Kako bi ostvarili taj cilj, potrebno je formalno definirati značajke i mogućnosti takvog raspodijeljenog okruženja. Raspodijeljeno računalno okruženje koje se pretpostavlja i koristi u ovom radu temelji se na karakteristikama današnjih komercijalnih bežičnih mreža osjetila, a formalna analiza se temelji na principima opisanim u [27]. U prvom dijelu ovog poglavlja revidiraju se osnovni pojmovi i svojstva, dok se u drugom dijelu izneseno koristi kako bi opisali izvedbu algoritma prema prethodno opisanoj metodi.

## 5.1. Raspodijeljeno računalno okruženje

Raspodijeljeno računalno okruženje fizički u potpunosti odgovara opisu mreže koja je korištena u prethodnim poglavljima. U ovom dijelu fizički izgled, odnosno topologija, upotpunjava se sa računalnim principima po kojima ona funkcionira. Ti principi se svode na dvije osnovne akcije koje su potrebne kako bi mreža funkcionirala, a to su računanje i komunikacija. Cilj koji je potrebno ostvariti tim akcijama može biti rješavanje problema, odnosno izvršavanje zadataka koji su zadani od strane korisnika izvan mreže ili od strane samih čvorova unutar mreže.

### 5.1.1. Entiteti

U skladu sa navedenim akcijama raspodijeljeno računalno okruženje sastoji se od skupa računalnih entiteta (engl. *computational entities*) –  $\mathcal{E}$ , u našem slučaju bežičnih čvorova i poruka (engl. *messages*). Entitet  $i \in \mathcal{E}$  ima mogućnost pohranjivanja informacija u ograničenu memoriju  $M_i$  koja se sastoji od određenog broja registara. Jedan od registara se naziva statusni i može poprimiti vrijednost jednog od konačnog broja definiranih stanja  $\mathcal{S}$  koja određuju režim rada unutar raspodijeljenog algoritma po kojem se entitet trenutno vlada. Ostali dijelovi entiteta su procesor (CPU), alarm, i komunikacija, pri čemu je jedan od glavnih ciljeva energetske djelotvornosti algoritama što manje korištenje procesorskog, a posebice komunikacijskog dijela. U daljnjem tekstu, alarmi i sve što je povezano s njima, neće biti opisano jer se ne koriste unutar algoritma.

Entitet može djelovati izvršavajući sljedeće akcije:

- lokalno pohranjivanje i obrada podataka
- odašiljanje poruka
- promjena vrijednosti statusnog registra.

Jednu od najvažnijih akcija u algoritmu lokalizacije, a to je mjerenje, može se definirati kao akciju odašiljanja poruka i lokalne obrade signala, jer se većina mjernih metoda temelji upravo na tim akcijama.

### 5.1.2. Komunikacija

Komunikacija među entitetima se svodi na odašiljanje i primitak poruka koje se definiraju kao konačan niz bitova. Komunikacija se provodi isključivo između dva entiteta. Skup entiteta s kojima određeni čvor  $i$  može komunicirati je u skladu sa dosadašnjim opisima mreže podskup skupa svih entiteta. Postoje dva definirana skupa,

a to su onaj koji sačinjava čvorove prema kojima čvor  $i$  može odašiljati poruke  $N_{out} \subseteq \mathcal{E}$  i skup čvorova od kojih može primiti poruke  $N_{in} \subseteq \mathcal{E}$ . U ovom slučaju ta dva skupa su jednaka i čvorovi unutar njih se nazivaju susjedi, a odnos susjedstva prikazuje se neusmjerenim grafom  $G$  na način koji je već opisan.

Pri obradi pristiglih poruka prioritet imaju poruke koje su pristigle prve, a ukoliko su dvije ili više poruka pristigle u isto vrijeme obrađuju se po proizvoljnom poretku. Ako se promatra ponašanje mreže u vremenskoj domeni, tada poruke koje su prve poslone ne moraju biti prve i primljene. Od samog zahtjeva za slanje u čvoru pošiljaocu, do primitka poruke u čvoru primatelju u realnim sustavima postoji mnogo akcija koje ti čvorovi moraju obaviti, a koje se protežu kroz sve komunikacijske slojeve (priprema za slanje, rezervacija komunikacijskog kanala, slanje, primanje, ispravak grešaka, obrada poruke itd.). Količina vremena potrebna za obavljanje svih tih akcija je nepredvidiva, pa se može definirati sljedeći aksiom:

*u odsustvu pogrešaka, komunikacijsko kašnjenje je konačno.*

Drugim riječima, u odsustvu pogrešaka, svaka poslana poruka biti će i dostavljena u svom izvornom obliku. Ova pretpostavka, iako pojednostavljuje realno stanje stvari, ostavlja veliki problem pri projektiranju algoritama, a to je činjenica da je jedina vremenska sinkronizacija među čvorovima moguća korištenjem poruka. Drugim riječima, vrijeme u klasičnom smislu (ono mjereno satom unutar samog čvora), zbog neodređenosti komunikacijskog kašnjenja, gubi značaj.

Sljedeća definirana pretpostavka je da je čvor sposoban raspoznavati pojedine čvorove među svojim susjedima, odnosno ima mogućnost poslati poruku točno određenom susjedu i kada primi poruku zna od kojeg susjeda je poruka pristigla.

Struktura i opis polja u porukama koje se koristi u raspodijeljenom okruženju u ovom radu, prema definiranim pretpostavkama, prikazana je tablicom 5.1.

Ime polja	Tip podatka	Opis
source	integer	ID čvora pošiljaoca
nexthop	integer	ID čvora susjeda koji je sljedeći čvor na putu prema odredištu
destination	integer	ID odredišnog čvora
header	string	zaglavlje poruke, određuje funkciju poruke i strukturu poslanih podataka odnosno polja data
data	struct	bilo koji podaci

Tablica 5.1. Message structure

### 5.1.3. Ponašanje entiteta

S obzirom na akcije koje entiteti mogu obavljati razlikuju se dva tipa događaja (engl. *event* – *e*) to su:

- primitak poruke
- spontani impuls.

Entitet se ponaša reaktivno, što znači da ne izvršava nikakve operacije u odsustvu događaja. Na događaje reagira izvršavajući određeni nedjeljivi niz operacija kojeg se naziva akcijom. Koju akciju će entitet izvršiti, ovisi o tipu događaja *e* i stanju *s* u kojem se entitet nalazi:

$$s \times e \longrightarrow \text{akcija} \quad (5.1)$$

Niz potpunih i jednoznačnih pravila koja povezuju statuse i događaje sa određenim akcijama za sve čvorove, naziva se raspodijeljeni algoritam  $B(\mathcal{E})$ . Ako se svi entiteti ponašaju prema istom algoritmu  $B(i) = B(j)$ ,  $\forall i, j \in \mathcal{E}$ , tada sustav ima homogeno ponašanje, a to je ono što se želi postići u zadanoj mreži.

Rezultat pojedine akcije, a sukladno tome, i cjelokupnog algoritma u pravilu ovisi o puno više parametara, primjerice vrijednostima ostalih registara i/ili sadržaju poruke koji je pokrenuo određenu akciju. Skup vrijednosti svih registara naziva se unutarnje stanje entiteta, a skup svih unutarnjih stanja u mreži zajedno sa događajima koji su pokrenuti i izvršiti će se u budućnosti naziva se konfiguracija sustava  $\mathcal{C}$ .

Algoritam mora osigurati prelazak sustava iz početne konfiguracije u konfiguraciju u kojoj će se riješiti određeni problem koji je zadan. Pri tome, ista početna konfiguracija nad kojom se izvrši isti algoritam, ne mora rezultirati jednakom konačnom konfiguracijom. Jedan od faktora koji može biti uzrok tome je nepredvidivost komunikacijskog kašnjenja. Kako bi algoritam bio ispravan potrebno je osigurati da svako njegovo izvršenje koje započinje iz skupa definiranih početnih konfiguracija, završi u konačnom vremenu, u jednoj od definiranih završnih konfiguracija, te da mu se ona više ne mijenja.

Postavlja se pitanje koje je zadatke sam sustav sposoban obaviti i kako formalno definirati ciljeve tih zadataka. U informacijskim sustavima svi zadaci se mogu objediniti pod imenom prikupljanje informacija. Prikupljanje informacija rezultira povećanjem razine znanja (engl. *knowledge*) sustava. Lokalno znanje definira se kao sadržaj lokalne memorije entiteta i informacije koje proizlaze iz tog sadržaja. Ako  $p$  označava informaciju koju posjeduje entitet  $i$  u vremenu  $t$  tada se može pisati:

$$p \in \text{LK}_t[i]. \quad (5.2)$$

Razine znanja koje slijede definiraju se za grupu entiteta  $W \subseteq \mathcal{E}$ , a nazivaju se implicitno znanje (engl. *implicit knowledge*) IK – skup informacija od kojih je svaka poznata barem jednom od čvorova grupe, eksplicitno znanje (engl. *explicit knowledge*) EK – skup informacija koje su poznate svim članovima grupe i zajedničko znanje (engl. *common knowledge*) CK – skup informacija koje su poznate svim članovima grupe i za koje svi članovi grupe znaju da su poznate svim članovima grupe.

Za zadani problem lokalizacije rješenje je u biti generiranje informacije o lokaciji čvorova pri čemu će lokalno znanje biti oznaka šume (ili više šuma) u kojoj se čvor nalazi i njegova lokacija u toj šumi. Implicitno znanje za sve čvorove mreže je topologija cijele mreže s podjelom na šume, lokacije svih čvorova unutar pojedine šume i odnos šuma u globalnom grozdu. Jednostavnom distribucijom podataka duž stabla to znanje može postati zajedničko znanje.

#### 5.1.4. Pretpostavke – ograničenja primjene algoritma

Svaki problem u raspodijeljenom okruženju mora, osim dobro definiranog zadatka, sadržavati i dobro definirane pretpostavke čijim se zadovoljenjem omogućava normalno djelovanje algoritma. Svaka od definiranih pretpostavki ograničava djelovanje algoritma samo na slučaj kada je ona zadovoljena, pa ih se stoga naziva i ograničenjima. Neke pretpostavke koje se koriste u zadanom problemu lokalizacije su već spomenute, a sada ih se i formalno definira.

Prva je već spomenuta pretpostavka dvosmjernih komunikacijskih veza (engl. *bidirectional links*). Ona pretpostavlja da entitet zna tko su mu susjedi, da svim susjedima može poslati poruku te da od svih susjeda može primiti poruku.

$$N_{out}(x) = N_{in}(x), \forall x \in \mathcal{E} \quad (5.3)$$

Važno svojstvo mreža je da svaki čvor u mreži može komunicirati sa svim ostalim čvorovima, izravno ili preko čvorova koji se nalaze između njih. Drugim riječima, pretpostavlja se takva topologija da je odgovarajući graf mreže povezan, a pretpostavka se zove povezanost (engl. *connectivity*).

Pri projektiranju algoritma potrebno je što više se koncentrirati na rješavanje samog zadatka, a manje na probleme koji se mogu javiti u komunikacijskim slojevima tako složenog sustava. Zato je potrebno pretpostaviti da će sve poruke koje su poslone u konačnom vremenu i stići, odnosno da se u tom procesu neće desiti neispravljlive greške. Ova pretpostavka naziva se potpuna pouzdanost (engl. *total reliability*).

Posljednja u nizu pretpostavki je ta da svaki čvor mreže inicijalno ima jedinstvenu oznaku (engl. *initial distinct value*). Ova pretpostavka je već spomenuta pri opisu težina bridova za konstrukciju razapinjajućih stabala. Ona je dodatno potrebna kako bi se moglo izabrati jedan čvor koji će koordinirati izvršavanjem zadatka. Ako ona



nije zadovoljena, problem konstruiranja razapinjajućeg stabla i problem izbora vođe su nerješivi.

### 5.1.5. Metode projektiranja algoritma

Složeno ponašanje raspodijeljenog algoritma i mnogi skriveni scenariji razlog su što se pri projektiranju raspodijeljenog algoritma potrebno služiti intuicijom i konstantnom analizom mogućih ishoda. Tehnički, algoritam se prvo mora kvalitetno idejno osmisliti, a zatim mu se definira pseudokod kojeg je potrebno temeljito analizirati s obzirom na mnoge neželjene ishode koje je vrlo teško prepoznati u toj fazi. Za mnoge uobičajene zadatke preporuča se korištenje niza definiranih i dokumentiranih algoritama čije djelovanje je temeljito ispitano.

#### Pseudokod

Pseudokod mora sadržavati sve potrebne informacije za pisanje samog koda. To uključuje definiranje svih statusa te među njima dodatno definiranje onih s kojima algoritam započinje i onih s kojima može završiti  $\mathcal{S}$ ,  $\mathcal{S}_{INIT}$  i  $\mathcal{S}_{TERM}$ . Zatim je potrebno navesti sva ograničenja te na posljetku napisati pseudokodove akcija za svaku kombinaciju statusa i događaja, i to grupiranu prema statusu entiteta. Ako se određena kombinacija ne definira, pretpostavlja se izvršavanje **null** akcije, odnosno ne reagiranja na događaj. U nizu operacija unutar akcije operacija promjene statusa mora biti posljednja.

#### Detekcija stabilnih konfiguracija

Ukoliko se rješenje problema sastoji od više zadataka koji se ne mogu izvršavati simultano, potrebno je sa sigurnošću odrediti kada je jedan zadatak izvršen te pokrenuti sljedeći. Konfiguracije koje uz izostanak vanjskih događaja ostaju nepromijenjene nazivaju se *stabilne*. Uzroci zbog kojeg je nastala stabilna konfiguracija mogu biti različiti, primjerice zbog greške u algoritmu može sustav doći stanje u kojem čvorovi čekaju na određeni događaj koji treba biti pokrenuti od strane drugih čvorova koji također čekaju. Ovakav slučaj se naziva zastoje (engl. *deadlock*) ili kružno čekanje (engl. *circular wait*) i predstavlja veliki problem pri projektiranju algoritama. Drugi primjer stabilnih konfiguracija može biti regularni završetak izvršavanja algoritma ili određenog dijela algoritma (zadatka). I u jednom i u drugom slučaju potrebno je osigurati mehanizam kojim se može otkriti takvo stanje sustava.

## 5.2. Analiza projektiranog algoritma

Raspodijeljeni algoritam lokalizacije podijeljen je primarno na niz zadataka sukladno sa fazama opisanim u 4. poglavlju. U konačnici algoritam se sastoji od sljedećeg niza zadataka koji se detaljno obrađuju u narednim odjeljcima:

1. inicijalizacija algoritma
2. odabir vođe
3. mjerenje i izmjena mjernih podataka među susjedima
4. lokalizacija inicijalnih podgrozdova
5. konstrukcija optimalnog stabla
6. spajanje podgrozdova.

Kao što se vidi iz popisa, osim glavnih faza, pod brojevima 3 – 6 bilo je potrebno dodati još dva zadatka koji nisu mogli biti predviđeni prije formalne definicije raspodijeljenog računalnog okruženja i pretpostavki koje su pridružene tom okruženju. Problem odabira vođe je rješiv uz pomoć nekoliko različitih metoda. Odabrana je metoda kojom se uz inicijalizaciju i odabir vođe konstruira i optimalno stablo. Ovakav odabir omogućava rješavanje zadataka 1, 2 i 5 jednim algoritmom.

Kako zadaci 2 i 5 moraju završiti prije izvršenja zadataka 3 i 6 u samom algoritmu tih zadataka mora se osigurati i detekcija njihovog globalnog okončanja. Tek nakon što je sa sigurnošću ustanovljena stabilna konfiguracija u kojoj je svaki čvor obavio svoj dio zadatka, inicijalizira se algoritam koji izvršava sljedeći zadatak. U ostalim slučajevima entitet može nastaviti sa sljedećim zadatkom ukoliko je on sam obavio sve radnje u trenutnom zadatku. Takav slučaj okončanja se zove lokalno okončanje (engl. *local termination*).

### 5.2.1. Inicijalizacija algoritma

Za početak izvršavanja algoritma obično je odgovoran događaj koji se naziva spontani impuls. Spontani impuls se karakterizira kao događaj koji dolazi izvan sustava i entitet nema predodžbu tko ga šalje ili hoće li se i kada će se on dogoditi. Spontani impuls može se primjerice aplicirati na početku algoritma na samo jedan čvor mreže te se tada takva pretpostavka naziva jedinstveni inicijator (engl. *unique initiator*). Ova pretpostavka je vrlo snažna i njeno osiguravanje u mreži se mora posebno planirati. Kako su ograničenja primjene algoritma s tom pretpostavkom velika, ona nije primijenjena na zadani problem.

Ukoliko ne postoji jedinstveni inicijator, pretpostavka je da se spontani impuls pojavljuje kod najmanje jednog, a najviše u svim čvorovima mreže i to u neodređenom vremenskom trenutku. Algoritam koji slijedi mora djelovati ispravno, neovisno od vremena kada i kojem se skupu čvorova impuls dogodio. Odabir algoritma koji se koristi u sljedećem zadatku zadovoljava sve iznesene uvjete, pa će pri opisu tog algoritma, u narednom odjeljku, biti govora i o inicijalizaciji.

### 5.2.2. Odabir vođe i konstrukcija korijenskog optimalnog stabla

Raspodijeljena konstrukcija stabla grafa  $G$  je proces u kojem je razina lokalnog znanja čvora podignuta iz početne, u kojoj je svaki čvor svjestan samo svojih susjeda, u sustav sa razinom lokalnog znanja u kojem je svakom čvoru  $i$  poznata informacija koji od susjeda su mu susjedi u stablu te, ako čvor nije korijen, koji od tih susjeda je na nižem nivou stabla. Lokalno znanje mora biti takvo da se izborom podskupa susjeda, koji su u stablu, i njihovom dodatnom klasifikacijom po nivoima, gledajući sa aspekta cjelokupne mreže, stvorilo razapinjajuće stablo. To stablo je optimalno u smislu da mu je zbroj težina bridova minimalan i uz to je i jedinstveno, što je direktna posljedica uvjeta da su mu svi bridovi različitih težina. Ukoliko nije definirana posebna težinska funkcija, koja osigurava da ne postoje dva brida s istim težinama, tada taj uvjet se može zadovoljiti na način kojim je definirana težina bridova u ID stablu u odjeljku 4.3.2.

U ovom odjeljku opisuje se algoritam kojim se rješava ukupno tri zadatka (1, 2 i 5). Algoritam koji u potpunosti zadovoljava naše potrebe zove se *Mega-Merger*, a princip rada detaljno je opisan u [27]. Radi se o algoritmu koji može započeti svoj rad sa jednim ili više inicijatora i koji stvara optimalno korijensko stablo te samim time provodi i odabir vođe odnosno korijena stabla. Za razliku od niza algoritama specijaliziranih za različite tipove topologija (stablo, prsten, kocka i sl.) ili one koji zahtijevaju određenu razinu znanja o samim karakteristikama mreže, ovaj algoritam je univerzalan i može djelovati u bilo kakvoj mreži, a jedine pretpostavke koje uvjetuje su one koje su već navedene. Iako je relativno komplicirane izvedbe, vrlo je efikasan i sa aspekta broja poruka i vremena potrebnog za izvršenje.

U ovom odjeljku, u kratkim crtama, opisan je princip rada algoritma, neke skrivene probleme i na koji način su oni riješeni. Popis korištenih registara s opisom svih funkcija i pseudokod algoritma dan je kao Prilog 2 na kraju rada.

Za što lakši opis algoritma koristi se posebna terminologija u kojoj su vrhovi predstavljeni kao gradska naselja, dok su bridovi komunikacijskog grafa, s različitim težinama, prikazani kao ceste različitih duljina koja povezuju ta naselja. Cilj je povezivanjem naselja najkraćim cestama u manje gradove naposljetku stvoriti metropolu koja će se sastojati od svih  $n$  naselja i od točno  $n - 1$  najkraćih cesta koje povezuju sva

naselja. Drugim riječima, stvara se optimalno stablo. Kako bi se minimizirao broj poruka između naselja i izbjegli mogući problemi poput zastoja, spajanje naselja i gradova u metropolu mora se odvijati po strogo definiranim pravilima.

Svaki grad je korijensko stablo i sastoji se od centra (korijen), cesta i naselja. Gradovi imaju svoje jedinstveno ime i razinu, a točne informacije o imenu i razini moraju znati sva naselja. U našem slučaju ID centra je ime grada. S obzirom na kašnjenje u komunikaciji, sva naselja nemaju u svakom trenutku pravu informaciju o imenu i razini grada u kojem se nalaze. Gradske ceste su samo one ceste koje su dio korijenskog stabla grada. Kako bi grad bio pravo korijensko stablo, naselja moraju znati koja od gradskih cesta vodi prema centru. Na početku izvršavanja algoritma, svaki čvor predstavlja grad koji se sastoji samo od centra bez naselja i cesta te ima razinu 1.

U nastavku se opisuju osnovna pravila algoritma i neki od mogućih problematičnih scenarija. U tablici 5.2 dana je struktura i opis svih tipova poruka koje se koriste u samoj izvedbi algoritma.

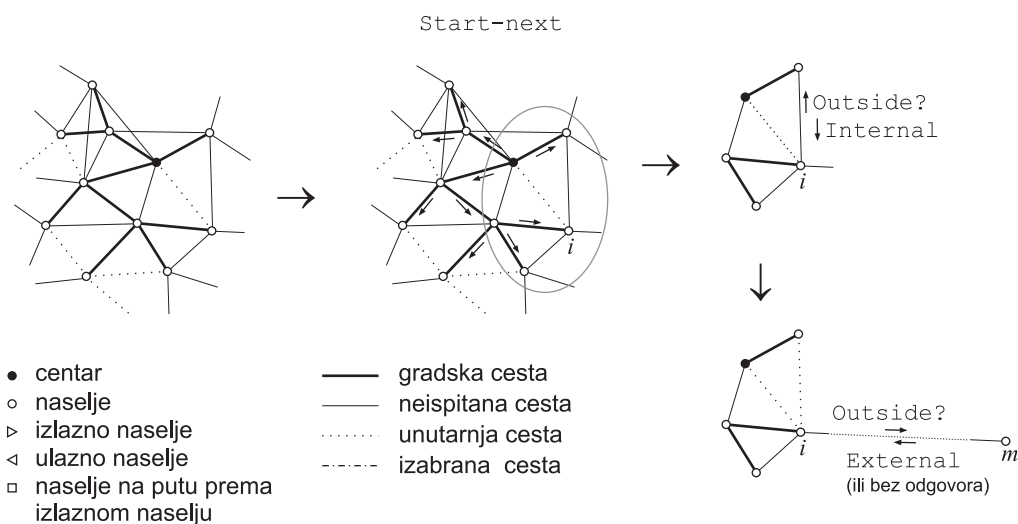
Zaglavlje poruke	Podatkovno polje	Opis
Outside?	name level	Poruka kojom se želi saznati da li cesta vodi izvan grada ili je unutranja
Internal		Odgovor na Outside? – unutrašnja cesta
External		Odgovor na Outside? – cesta prema drugom gradu
Let-us-merge	name level	Upit za spajanje, šalje se ime i razina grada
Notification	name level	Obavijest o absorpciji
Start-next	name level	Pokretanje potrage za najkraćom cestom koja vodi van grada
Start-next-reply	di	Odgovor na Start-next prema roaddown čvoru koji uključuje informaciju o najkraćoj vanjskoj cesti iz tog dijela stabla

**Tablica 5.2.** Opis poruka koje se koriste u algoritmu *Mega-Merger*

Kako bi formirali optimalno stablo, gradovi se moraju međusobno spajati i to isključivo korištenjem najkraćih cesta koje ih povezuju. Nakon što se odredi najkraća cesta koja povezuje grad s nekim drugim gradom, centar odašilje zahtjev za spajanjem (*Let-us-merge*) koristeći put od gradskog naselja s kojim je ta cesta incidentna do naselja u drugom gradu s kojim se želi spojiti. Dok se ne obavi spajanje, grad ne obavlja nikakve druge aktivnosti. Nakon što se dva grada spoje, nastaje novi grad koji se sastoji od naselja i cesta iz obaju gradova i spojne cesta kojom su se dva grada povezala.

Izbor najkraće ceste, koja vodi iz grada, obavlja centar koji duž stabla šalje zaht-

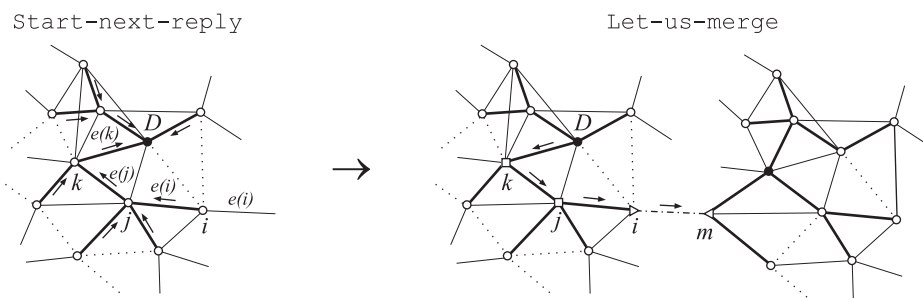
jev **Start-next** kako je prikazano na slici 5.1. Čvorovi po primitku prosljeđuju poruku svojim sljedbenicima i sami kreću u potragu za najkraćom cestom koja vodi van grada. Naselje neće uzimati u obzir ceste koje se već koriste, jer one vode prema naseljima istog grada (gradske ceste), ali i neke od nekorisćenih cesta mogu voditi prema naseljima u istom gradu (unutarnje ceste). Kako je prikazano na slici za čvor  $i$ , čvorovi potragu počinju najkraćim neispitanim cestama i to slanjem upita **Outside?**, kojim se želi saznati da li cesta zaista vodi van grada ili je unutarnja. Upit se sastoji od imena i razine grada iz kojeg upit dolazi. Naselje koje je primilo upit provjere može sa sigurnošću odgovoriti da je u istom gradu (**Internal**), ako su im imena gradova ista. Tada se cesta označava kao unutarnja i više se ne koristi. Naselje koje primi upit ne može biti sigurno da je u drugom gradu ako su im imena različita, zato jer ta informacija ne mora biti trenutno važeća odnosno grad može biti već apsorbiran od strane grada iz kojega je poruka stigla, a zbog kašnjenja u komunikaciji upitano naselje još nije primilo obavijest o tome. U tom slučaju pomaže razina grada jer ako su one iste ili je naselje koje odgovara na upit više razine  $\text{razina}(m) \geq \text{razina}(i)$  tada do apsorpcije nije moglo doći i naselje odgovara sa **External**. Ako je slučaj da se imena razlikuju i naselje koje je primilo upit je u gradu niže razine, tada je apsorpcija moguća, naselje  $m$  ne može biti sigurno da je u istom gradu te obustavlja svoj odgovor dok mu razina ne naraste na istu ili veću od one iz upita. Rezultat obustave odgovora na upit je taj da je grad iz kojeg je upit odaslan u zastoju.



**Slika 5.1.** Pokretanje novog ciklusa potrage za najkraćom cestom koja vodi van grada i lokalno ispitivanje cesta naselja  $i$ .

Čim je dio mreže u zastoju dolazi do opasnosti da je i uzrok njegovom zastoju neki drugi dio mreže koji je u zastoju te da na taj način dođe do kružnog zastoja. Ako je  $C_0$  grad u zastoju, tada on može biti u zastoju jedino ako čeka odgovor od drugog grada  $C_1$  niže razine  $C_1 < C_0$ . Ako vrijedi  $C_0 > C_1 > C_2 > \dots > C_k$  tada ne može vrijediti i  $C_k > C_0$  čime bi se zatvorio krug, pa ne može doći ni do kružnog zastoja.

Nakon što je naselje pronašlo najkraću cestu koja vodi van grada, šalje obavijest (**Start-next-reply**) o njoj i njenoj duljini prema centru. Ukoliko su sve incidentne ceste unutarnje, duljina ceste je  $\infty$ . Kako bi se smanjio broj podataka povratnih poruka, u izbor najkraće ceste naselja ne uključuju se samo ceste koje su sami čvorovi našli nego i ceste koje su dobili od svojih sljedbenika. Tek kad skupe sve poruke od svojih sljedbenika o njihovim najkraćim cestama, provode usporedbu i šalju obavijest sa oznakom i duljinom najkraće ceste njihovog dijela stabla. Nakon što centar dobije podatke o najkraćim cestama od svojih sljedbenika, može izvršiti odabir i poslati zahtjev za spajanjem (**Let-us-merge**) prema najkraćoj vanjskoj cesti cijeloga grada. Ovaj proces prikazan je na slici 5.2 na kojoj je najkraća cesta naselja  $i$  označena sa  $e(i)$ , usporedbom postaje ujedno i najkraća cesta naselja  $j$  i  $k$  na putu prema centru  $D$  ( $e(i) = e(j) = e(k)$ ). Centar provodi sličnu usporedbu kao i čvorovi  $j$  i  $k$  – odabire čvor  $i$  kao izlazno naselje te šalje zahtjev do njega preko naselja  $j$  i  $k$ . Naselje  $i$  prosljeđuje naposljetku zahtjev svoga grada za spajanjem vanjskom naselju  $m$ .

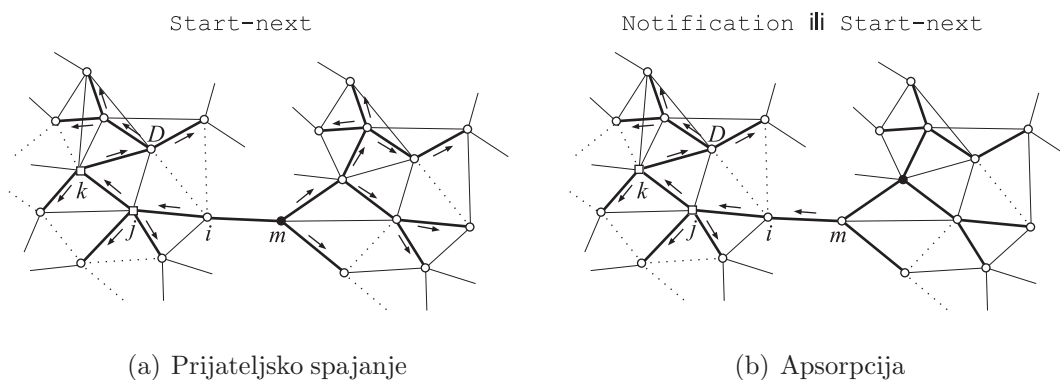


**Slika 5.2.** Odabir najkraće ceste i slanje zahtjeva za spajanje.

Gradovi se mogu povezati na više načina u ovisnosti na kojoj su razini. Ako im je razina ista i oba su odabrala istu cestu kao cestu s najmanjom dužinom koja vodi do susjednog grada, tada se obavlja **prijateljsko spajanje** (engl. *friendly merger*) kako je prikazano na slici 5.3(a). Pri prijateljskom spajanju, novi grad ima razinu povećanu za 1 u odnosu na gradove od kojih je nastao. Novi centar i samim time i ime grada, bira se između dva naselja incidentna sa spojnom cestom ( $i$  i  $m$ ) i to po kriteriju manjeg ID-a. Tada naselja  $i$  i  $m$  šalju novi **Start-next** zahtjev za potragom najkraće ceste koja vodi van novog grada (zahtjev je ujedno i obavijest o novom gradu i razini) čvorovima u svojim bivšim gradovima. Čvorovi na putu do starih centara moraju obaviti dvije akcije (1) okrenuti smjer incidentnih cesta prema novom centru i (2) nakon okretanja obavijest proslijediti svojim sljedbenicima.

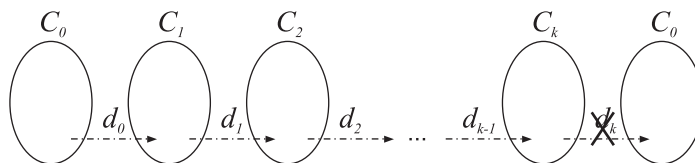
Ako je grad koji se želi povezati niže razine od grada kojem je taj zahtjev upućen, tada se obavlja **apsorpcija** grada niže razine prema slici 5.3(b). Slijede dvije moguće akcije koje ovise o tome da li je ulazno naselje grada koji apsorbira već poslalo **Start-next-reply**. Ako jest, u novi se grad šalje samo obavijest (**Notification**) o apsorpciji, a ako nije onda se potraga proširuje sa **Start-next** zahtjevom u novi dio grada.

U svim drugim slučajevima, zahtjev je obustavljen i grad koji je poslao zahtjev miruje dok se ne promjene razine ili uvjeti tako da nastupi jedan od opisanih slučajeva. Grad koji miruje ne zna, niti treba znati da je zahtjev obustavljen. Iz njegove perspektive zastoje izgleda samo kao kašnjenje u komunikaciji.



**Slika 5.3.** Dva moguća scenarija nakon što je upućen *Let-us-merge* zahtjev duž izabrane ceste. Treća mogućnost obustave odgovora je samo privremena.

I ovom slučaju mora se isključiti mogućnost kružnog zastoja. Grad u zastoju  $C_0$  čeka grad iste ili niže razine  $C_1$  da postigne višu razinu. Izabrana cesta grada  $C_0$ , označena sa  $d_0$  u tom slučaju mora biti veće duljine od ceste koju je odabrao  $C_1$ , inače bi  $C_1$  odabrao  $d_0$  za spajanje. Nadalje, kako bi nastao kružni zastoje,  $C_1$  mora također biti u zastoju zbog odabira grada  $C_2$  itd. sve do  $C_k$  koji čeka na  $C_0$ . Ako su odgovarajuće izabrane ceste za spajanje svakog pojedinog grada označene sa  $d_0 \dots d_k$ , onda mora vrijediti odnos  $d_0 > d_1 \dots > d_k$ . Kako bi se zatvorio krug mora vrijediti i  $d_k > d_0$ , što je kontradiktorno prethodnom zaključku, pa se može zaključiti kako zbog ovog tipa zastoja ne može doći do kružnog globalnog zastoja (slika 5.4).



**Slika 5.4.** Isključivanje kružnog zastoja.

Unatoč jasno definiranim pravilima zbog komunikacijskih kašnjenja, istodobnih akcija, odgađanja odgovora i zastoja, postoji jako puno mogućih problematičnih scenarija. Posljedica je da trenutno ne postoji zadovoljavajući potpuni dokaz da će u svakom od tih scenarija algoritam uspjeti konstruirati stablo i izabrati vođu. Unatoč nepostojanju dokaza, algoritam to čini vrlo uspješno. Proces projektiranja algoritma prema zadanom opisu također je pun zamki te je bila potrebna iscrpna analiza pseudokoda i testiranje samog kôda u raznim scenarijima kako bi se one otkrile i izbjegle.

Ovaj se algoritam može iskoristiti više puta, što znači da je samo potrebno odabrati drugu funkciju ili registar kao nosioca težine bridova te se istim algoritmom može konstruirati potpuno različito optimalno stablo.

### 5.2.3. Mjerenje, izmjena mjernih podataka među susjedima i lokalizacija inicijalnih podgrozdova

U ovom dijelu opisan je algoritam kojim se organizira mjerenje, izmjena mjernih podataka među susjedima i lokalizacija inicijalnih podgrozdova. Algoritam treba zadovoljavati sljedeće zahtjeve:

1. algoritam započinje korijen stabla
2. lokalizacija inicijalnih podgrozdova može započeti tek nakon što je čvor prikupio sva mjerenja svojih susjeda
3. korijen treba sa sigurnošću otkriti globalno okončanje algoritma.

Na početku svi čvorovi imaju status FOLLOWER, osim korijena koji je u statusu ROOT. On spontano pokreće prvu akciju algoritma – mjeri azimut svih susjeda, zatim im svima šalje poruku *Bearing* sa mjernim podacima te nakon slanja promjeni svoj status u CLUSTERLOC. Susjedi, nakon što prime poruku, rade identičnu stvar te se na taj način algoritmom preplavljuju (engl. *flooding*) poruka širi duž svih bridova. Na taj način, naposljetku, svi čvorovi ulaze u status CLUSTERLOC. Za razliku od širenja *Start-next* poruke u algoritmu *Mega-Merger*, u ovom slučaju poželjno je koristiti sve bridove, a ne samo one koji su dio korijenskog stabla. Razlog je taj što je širenje brže, a broj poruka nije veći od minimalno potrebnog jer svi čvorovi ionako moraju izmijeniti mjerne podatke sa svim svojim susjedima (uvjet 2) prije nego lokaliziraju svoje inicijalne podgrozdove.

Svaki čvor u statusu CLUSTERLOC prati koliko mu se susjeda javilo, a u trenutku kada mu i zadnji pošalje poruku sa mjernim podacima, može lokalizirati svoj inicijalni podgrozd na način koji je opisan u potpoglavlju 4.2.

Algoritam se nakon lokalizacije nastavlja u suprotnom smjeru – prema korijenu, ali ne ovaj put, kako bi se minimizirao broj poruka, isključivo duž stabla. Kreće se od listova koji odmah nakon lokalizacije šalju poruku *Ack* svojim prethodnicima i mijenjaju status u ASLEEP. Ostali čvorovi prate pristigle *Ack* poruke te samo ako su ih primili od svih svojih sljedbenika, i uz to su još i sami izvršili lokalizaciju, mogu prosljediti poruku *Ack* dalje prema korijenu i prijeći u status ASLEEP. Kada korijen naposljetku primi potvrdu da su svi njegovi sljedbenici poslali *Ack* poruke, odnosno prešli u ASLEEP status, tada može zaključiti da je algoritam globalno okončan. Pseudokod ovog algoritma dan je kao Prilog 3 na kraju rada.



U konačni algoritam dodatno je ugrađena mogućnost slanja težine bridova (*cost*) susjedima. Ta mogućnost se može iskoristiti, primjerice, kada se želi stvoriti novo optimalno stablo duž kojega će se spajati grozdovi, a težina takvog stabla je vezana uz parametar koji čvor izračuna tek lokalizacijom inicijalnih podgrozdova.

#### 5.2.4. Spajanje podgrozdova

Sljedeći i završni zadatak je spajanje svih grozdova u jedan globalni grozd sa što manje globalnih podgrozdova. Zadatak je podijeljen na dva dijela.

Prvi dio je opcionalan, a sastoji se od kreiranja novog optimalnog stabla ponovnim pokretanjem algoritma *Mega-Merger* sa težinama bridova izračunatim nakon lokalizacije inicijalnih podgrozdova. Kao što je već spomenuto, jedan od ciljeva ovog rada je pronaći težinu koja bi dala najbolje rezultate lokalizacije, pa je stoga ostavljena i ova mogućnost određivanja težine bridova.

U drugom dijelu se duž odabranog stabla provodi spajanje. Spajanje se obavlja na sličan način kao i prosljeđivanje *Ack* poruka pri lokaliziranju inicijalnih podgrozdova. Dakle, kreće se od listova koji se spajaju sa svojim prethodnicima. Zatim se unutarnji čvorovi spajaju sa svojim prethodnicima, ali tek kada su se svi grozdovi sljedbenika spojili s njihovim grozdom itd., sve do korijena koji provodi posljednja spajanja i zaključuje algoritam lokalizacije.

## 6. Rezultati

Kako bi se što bolje procijenila kvaliteta projektiranog raspodijeljenog algoritma lokalizacije, simuliran je njegov rad na računalu. Simulacije su obavljene u simulatoru za bežične mreže osjetila koji je posebno projektiran za ovu namjenu kao dio rada na algoritmu. Ostvaren je koristeći se programskim jezikom visoke apstrakcije – MATLAB® i to kao proširenje postojećeg simulatora za mobilne robote AMORsim [28, 29].

Sveobuhvatna, realna simulacija rada jedne ad hoc bežične mreže osjetila, obuhvaća simulaciju rada svih komunikacijskih slojeva, simuliranje okoline te propagacije elektromagnetskih signala. Uz to može obuhvatiti i simulaciju realnog rada osjetila i razne dodatne opreme ukoliko postoji. Nadalje, za svaki dio komunikacijskog modela postoji niz standardiziranih protokola koje bi također trebalo implementirati i uključiti u simulaciju. Naravno, izrada takvog simulatora, ali i samo ispravno korištenje vrlo je zahtjevan zadatak u kojem korisnik mora točno znati koje pretpostavke i ograničenja postavlja i zašto.

U ovom slučaju, zadatak koji mreža mora obaviti i rješenje vrlo su specifični. Iako se temelje na korištenju najnaprednije komunikacijske infrastrukture, kvaliteta njihovog rada i dobivenih rezultata je potpuno autonomna. Vodeći se tom činjenicom već su se u 5. poglavlju uvele pretpostavke na rad ostalih dijelova sustava (apsolutna pouzdanost, dvosmjerne veze i povezanost) i time se argumentirano ograničila upotrebljivost algoritma, ali i dobila potrebna sloboda u pristupu rješenju. U samim simulacijama vrijede sve navedene pretpostavke te je time omogućena visoka razina fleksibilnosti samog simulatora. Ostatak parametara koji su izdvojeni kao ključni, s obzirom na njihov utjecaj na rad algoritma, pažljivo su modelirani, a dobiveni rezultati dani su u nastavku.

### 6.1. Kriteriji evaluacije algoritma

Prije samih rezultata dan je opis kriterija kojima se procjenjuje kvaliteta dobivenih rezultata lokalizacije. Kriteriji su odabrani tako da se može što lakše izvršiti usporedba

sa postojećim algoritmima te kako bi se izdvojila i analizirala greška nastala uslijed rada algoritma od greške nastale zbog realnih ograničenja mreže.

### 6.1.1. Korijen srednje kvadratne pogreške lokacije

Standardni kriterij za procjenu kvalitete lokalizacije je korijen srednje kvadratne pogreške

$$E_{rms} = \sqrt{\frac{1}{n} \sum_{i=1}^n e_i^2} \quad (6.1)$$

gdje je pogreška udaljenost između estimirane i stvarne lokacije čvora mreže  $i$

$$e_i = \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2}. \quad (6.2)$$

Jasno je da tako definiran kriterij ovisi o više parametara i direktna usporedba na temelju samo tog kriterija nije moguća. Primjerice, fizička veličina mreže, broj čvorova, njihov raspored i udaljenost izravno utječu na veličinu pogreške.

### 6.1.2. Cramér-Rao limit

Kako bi se utjecaj velike većine parametara mreže na koje se ne može utjecati algoritmom učinio mjerljivim, te na taj način omogućila direktna usporedba samih algoritama lokalizacije, koristi se Cramér-Rao limit (engl. *Cramér-Rao bound – CRB*).

CRB je donja granična vrijednost matrice kovarijanci bilo kojeg nepristranog estimatora (engl. *unbiased estimator*). Vrlo bitno svojstvo CRB-a je da njegova vrijednost ne ovisi o tome koji se algoritam koristi, već on pruža uvid u kvalitetu najbolje moguće estimacije za zadane parametre mreže. Na taj način moguće je izolirati grešku koju unosi algoritam (sve iznad CRB-a) i grešku na koju algoritam nema utjecaj, već je ona posljedica parametara koji su zadani, primjerice topologijom i/ili mjernim pogreškama. Zbog toga se CRB smatra idealnom referentnom točkom pri usporedbi algoritama lokalizacije.

Matematički izvod CRB-a za problem lokalizacije uz pomoć mjerenja udaljenosti dan je u [30]. U [31] dana je procedura za računanje CRB-a za isti problem, ali korištenjem mjerenja vremena dolaska signala TOA, jačine primljenog signala RSS i smjera dolaska signala AOA.

Zadani problem lokalizacije, izražen kao problem estimacije glasio bi: *koristeći se sa  $M \times 1$  mjernim vektorom  $X$  u kojem su popisane izmjerene vrijednosti azimuta  $\theta_{ij}$  između svih  $M/2$  parova susjednih čvorova i  $i$   $j$  potrebno je estimirati  $3n \times 1$  vektor pozicije (lokacija i orijentacija)  $P = \{x_1, y_1, \alpha_1, x_2, y_2, \alpha_2, \dots, x_n, y_n, \alpha_n\}$ . Kako je pretpostavljeno u definiciji problema da orijentacije nisu poznate, vektor pozicije se sastoji od tri nepoznata parametra po čvoru.*

Realno mjerenje kuta dolaska signala podložno je mjernim pogreškama. Te pogreške mogu se modelirati koristeći normalnu razdiobu, tako da je mjerni vektor  $X$  definiran sa multivarijantnom normalnom funkcijom gustoće vjerojatnosti

$$f_X(x, P) = \frac{1}{(2\pi)^{3n} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2} [X - \mu(P)]^T \Sigma^{-1} [X - \mu(P)]} \quad (6.3)$$

gdje je  $\Sigma = \sigma^2 I$  matrica kovarijanci. Matrica je poprimila tu formu zato jer su varijance svih mjerenja, po pretpostavci korištenja iste opreme i metodologije mjerenja, – jednake ( $\sigma$ ).  $\mu(P)$  je vektor točnih azimuta i to tako da  $m$ -ti element odgovara azimutu čvora  $j$  s obzirom na čvor susjed  $i$  koji je izvršio mjerenje

$$\mu_m(P) = \arctan \left( \frac{y_j - y_i}{x_j - x_i} \right) - \alpha_i. \quad (6.4)$$

CRB je definiran kao inverz Fisherove informacijske matrice (engl. *Fisher Information Matrix – FIM*)  $I(P)$ , koja je dana sa sljedećim izrazom:

$$I(P) = E \left[ \left[ \frac{\partial}{\partial P} \ln f(X, P) \right]^2 \right] \quad (6.5)$$

gdje  $E[\cdot]$  predstavlja očekivanu vrijednost.

Koristeći (6.3) i (6.5) FIM se može prikazati kao

$$I(P) = \frac{1}{\sigma^2} [G'(P)]^T [G'(P)]. \quad (6.6)$$

gdje je  $G'(P)$  matrica veličine  $M \times 3n$  čiji  $mn$ -ti elementom je jednak:

$$G'(P)_{mn} = \frac{\partial \mu_m}{\partial P_n} = \begin{cases} 0, & \text{ako je } P_n \notin \{x_i, y_i, x_j, y_j, \alpha_i\} \\ \frac{y_j - y_i}{d_{ij}^2}, & \text{ako je } P_n = x_i \\ \frac{y_i - y_j}{d_{ij}^2}, & \text{ako je } P_n = x_j \\ \frac{x_i - x_j}{d_{ij}^2}, & \text{ako je } P_n = y_i \\ \frac{x_j - x_i}{d_{ij}^2}, & \text{ako je } P_n = y_j \\ 1, & \text{ako je } P_n = \alpha_i. \end{cases} \quad (6.7)$$

$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$  je udaljenost između stvarnih lokacija čvorova  $i$  i  $j$ .

FIM se lako može izračunati koristeći (6.6), ali pri izračunavanju njenog inverza – matrice CRB, može se naići na određene probleme. Kao što se to tvrdi u [32], FIM u slučaju problema lokalizacije neusidrene mreže je nedovoljnog ranga. U [33] dan je dokaz sljedećeg teorema.

**Teorem 4.** *Za problem lokalizacije neusidrene mreže (korištenjem mjerenja udaljenosti), sa ukupno  $n$  čvorova, rang Fisherove informacijske matrice  $I(P)$  je  $2n - 3$ .*

Uzrok smanjenom rangu je rezultat ukupno 3 stupnja slobode koje posjeduje rješenje, s obzirom da su rotacije i translacije valjanog rješenja ujedno i sama rješenja tog problema. U zadanom problemu rang FIM-a je nešto drukčiji i iznosi  $3n - 4$  s obzirom da postoji još jedan parametar po čvoru za estimaciju (orijentacija) i još jedan stupanj slobode (skaliranje). U svakom slučaju, FIM je singularna matrica i njen inverz  $I^{-1}$  ne postoji. Unatoč tome, ista ta matrica je punog ranga, ako se promatra u potprostoru razapetom s  $3n - 4$  ortonormirana svojstvena vektora  $\vec{v}_1, \dots, \vec{v}_{3n-4}$ , koji odgovaraju svojstvenim vrijednostima matrice  $I$  različitim od nule, kao što je prezentirano u [33]. Tada je njen inverz dan sa  $Q^{-1} = V^T I^\dagger V$ , gdje je  $V = (v_1, \dots, v_{3n-4})$ , te je  $I^\dagger$  Moore-Penroseov pseudoinverz od matrice  $I$ . Taj pseudoinverz se naziva i *relativni CRB*, iz razloga što predstavlja odgovarajući CRB za problem neusidrene, odnosno relativne lokalizacije.

Jasno je da se za neusidrenu lokalizaciju ne može limitirati točnost estimacije pojedinog čvora, jer bi se estimirano rješenje uvijek moglo translahirati tako da se poklopi sa pravom vrijednošću. Prema tome, jedini relevantan limit estimacije mora biti invarijantan na transformacije, a taj limit je ukupni limit estimacije označen kao  $V_{total} = Tr(I^\dagger)$ .

Kako bi u svrhu usporedbe bilo moguće prikazati CRB u obliku korijena srednje vrijednosti kvadratne pogreške (6.1), mora se izostaviti njegov orijentacijski dio i izdvojiti ukupni limit estimacije samo za lokaciju. Stoga se za Cramér-Rao limit korijena srednje vrijednosti kvadratne pogreške koristi sljedeći izraz:

$$\begin{aligned} E_{rms}(\text{CRB}) &= \sqrt{\frac{1}{n} \sum_{i=1}^n (\text{E}((\hat{x}_i - x_i)^2) + \text{E}((\hat{y}_i - y_i)^2))} \\ &= \sqrt{\frac{1}{n} \sum_{i=1}^n (I_{3i-2}^\dagger + I_{3i-1}^\dagger)}. \end{aligned}$$

gdje je  $(\hat{x}_i, \hat{y}_i)$  estimirana lokacija čvora  $i$ .

## 6.2. Utjecaj postavnih parametara

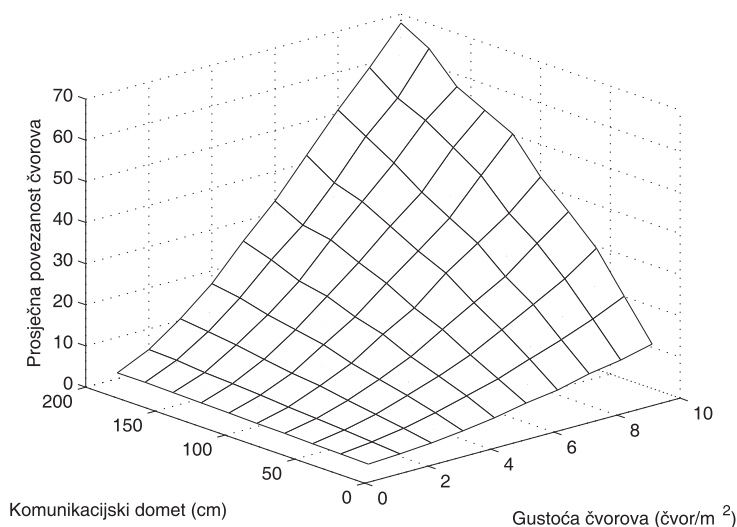
Pogreške pri estimiranju lokacije čvorova mogu imati različite uzroke. S obzirom na izvor pogreške, ukupnu pogrešku estimacije može se podijeliti na dio na koji se može utjecati algoritmom lokalizacije – algoritamska pogreška (engl. *algorithmic error*) i na onaj dio koji je inherentan zbog same postavke mreže – postavna pogreška (engl. *setup*)

*error*) [30]. Izvor algoritamske pogreške je sam po sebi jasan, dok izvori postavne pogreške mogu biti broj i lokacije čvorova u mreži, komunikacijski domet (topologija mreže) te mjerne pogreške. Jedan način izdvajanja algoritamske pogreške opisan je u dijelu u kojem se govori o Cramér-Rao limitu. Drugi način evaluacije algoritma je simulacija njegovog rada pri različitim vrijednostima postavki mreže.

### 6.2.1. Postavni parametri mreže

U ovom odjeljku obavlja se analiza međusobne ovisnosti nekih od postavnih parametara. Ti parametri su komunikacijski domet (engl. *communication range*) koji je predstavljen kružnicom zadanog polunjera, gustoća čvorova (engl. *node density*), odnosno prosječni broj čvorova po jedinici površine i srednja povezanost čvorova (engl. *mean node connectivity*) ili srednja vrijednost broja susjeda. Identifikacija odnosa među njima je jedan od preduvjeta za daljnju analizu rada algoritma, jer ti parametri imaju veliki utjecaj jedan na drugog i dodatno na kvalitetu estimacije.

U [30] je dana vjerojatnost određene prosječne povezanosti, s obzirom na gustoću čvorova i komunikacijski domet te je izvedena funkcija kumulativne distribucije. U ovom dijelu revidiraju se ti rezultati sa par modifikacija. U obavljenim simulacijama komunikacijski domet je uz gustoću čvorova dodan kao novi parametar, a prati se srednju vrijednost povezanosti. Simulacije su izvršene u mreži sa 100 nasumično postavljenih čvorova, unutar kvadratne površine čija veličina ovisi o traženoj gustoći, variranjem komunikacijskog dometa.

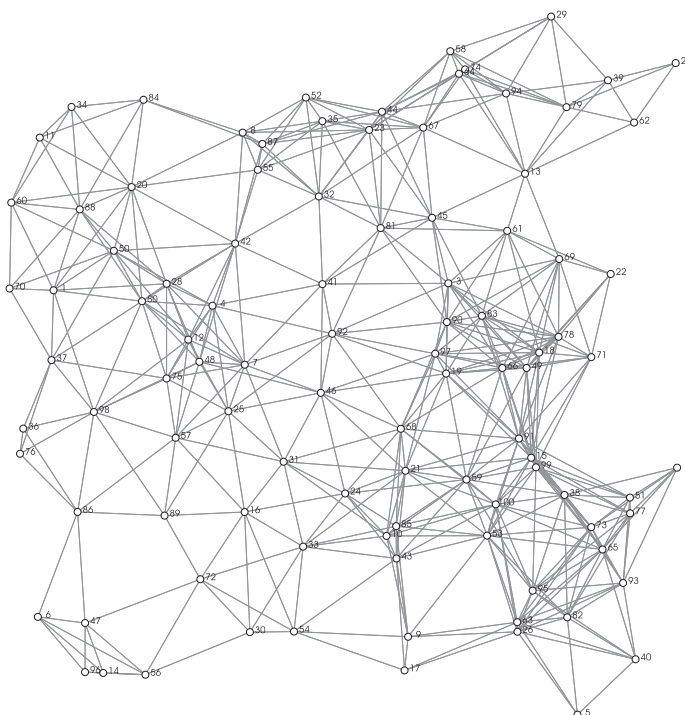


Slika 6.1. Odnos između dijela postavnih parametara mreže.

Kao što se vidi i iz rezultata na slici 6.1, prosječna povezanost je u potpunosti određena sa dometom i gustoćom čvorova, te se može koristiti kao jedan od parametara simulacije bez potrebe da se u ispitivanje uključuju druga dva. U nastavku poglavlja dani su rezultati simulacija u kojima se variralo prosječnu povezanost i veličinu mreže.

Kako u mreži nije pretpostavljeno postojanje čvorova koji poznaju svoju lokaciju, jedino moguće rješenje je u relativnom koordinatnom sustavu, pa prije evaluacije pogreške potrebno je taj relativni koordinatni sustav transformirati, kako bi se što bolje slagao sa stvarnim koordinatnim sustavom u kojemu je mreža i zadana. Ta transformacija se izvodi na sličan način kako je to prikazano u 2. poglavlju, s tim da su zajednički čvorovi svi čvorovi mreže.

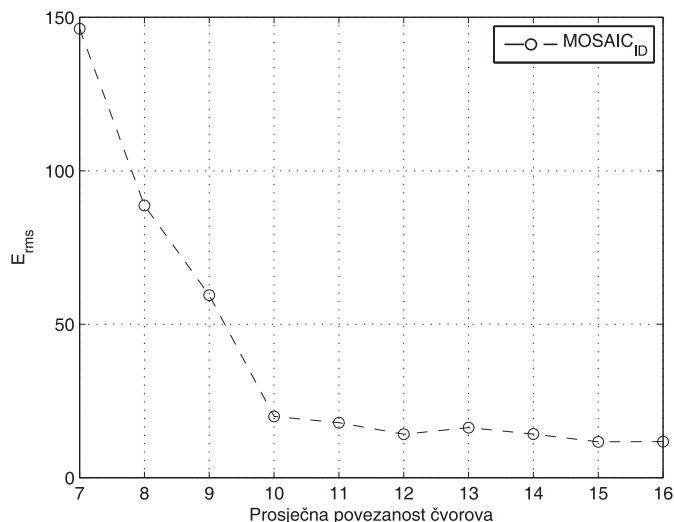
### 6.2.2. Povezanost



Slika 6.2. Lokacije čvorova u ispitnoj mreži.

Kako bi varirali prosječnu povezanost u mreži, generirana je mreža sa nasumičnim rasporedom čvorova koji je ostao nepromijenjen u svim simulacijama, (slika 6.2) te se za željeni iznos povezanosti računao potreban komunikacijski domet. Minimalna prosječna povezanost za koju su se provodile simulacije je 7, jer ispod te vrijednosti mreža nije bila povezana, što je bila jedna od pretpostavki.

Na slici 6.3 prikazani su rezultati simulacija. Za svaku srednju vrijednost povezanosti obavljeno je 20 simulacija.



**Slika 6.3.** Kvaliteta lokalizacije s obzirom na prosječnu povezanost.

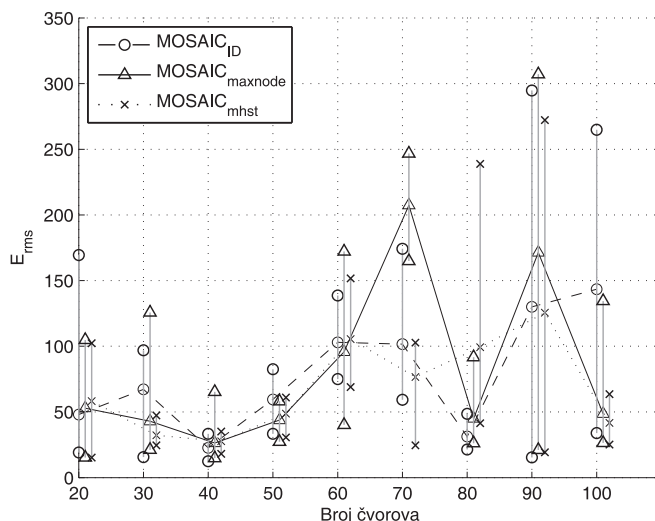
Izravna posljedica povećanja srednje povezanosti čvorova je povećani broj mjerenja azimuta, a gledajući sa aspekta krutosti, povećanje broja ograničenja. Kako su dodavanjem novih ograničenja čvršće definirani položaji čvorova, greška lokalizacije bi se trebala smanjivati što je i pokazano na slici 6.3. Djelovanje tog parametra je efektivno do određene razine povezanosti, koja u ovom slučaju iznosi 10. Nakon te točke dolazi do zasićenja te povećanje povezanosti ima sve manji utjecaj na smanjenje greške lokalizacije. Dobiveni rezultati su u skladu sa rezultatima u [30], gdje je obavljena slična analiza korištenjem CRB-a u mreži sa ograničenjem udaljenosti.

### 6.2.3. Veličina mreže

Među navedenim postavnim parametrima identificiran je još jedan koji ima veliki utjecaj na kvalitetu estimacije, a nije u vezi sa prosječnom povezanošću. Kako je prosječna povezanost nedimenzionalni parametar, potrebno je dodatno analizirati utjecaj dimenzija same mreže. Ako se prosječna povezanost drži konstantnom, veličina mreže je funkcija broja čvorova. U provedenim simulacijama prosječna povezanost je držana na vrijednosti 10, varijanca mjernog šuma je  $6^\circ$ , a broj čvorova je variran između 20 i 100. Za svaku definiranu vrijednost broja čvorova generirana je jedna nasumična mreža i na njoj je obavljeno 10 simulacija. Rezultati su prikazani na slici 6.4.

Povećanje broja čvorova mreže  $n$  ima izravan utjecaj na broj spajanja grozdova koji iznosi  $n - 1$ . Prema tome, očekivano je lagano povećanje pogreške sa povećanjem broja čvorova što je i potvrđeno rezultatima simulacija.





Slika 6.4. Kvaliteta lokalizacije s obzirom na veličinu mreže.

## 6.3. Lokalizacija uz prisutnost mjernog šuma

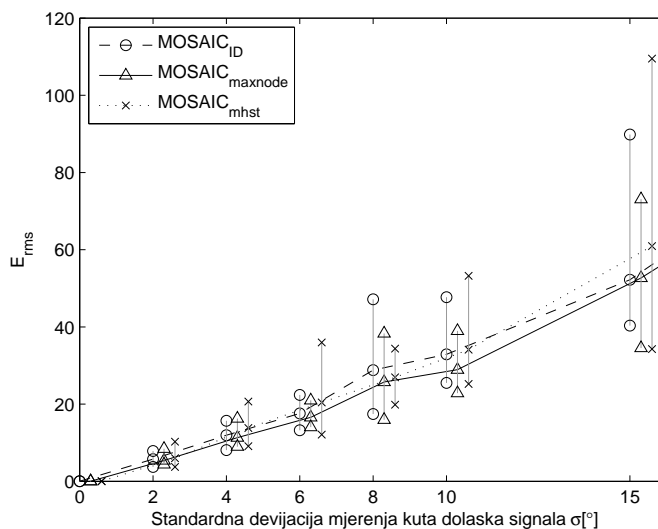
Jedan od najvažnijih postavnih parametara obrađuje se u ovom poglavlju – mjerni šum. Analiza ponašanja algoritma s obzirom na mjerni šum je standardna procedura za ispitivanje kvalitete lokalizacije. U provedenim simulacijama mjerni šum je modeliran kao slučajna varijabla sa normalnom distribucijom  $\mathcal{N}(\theta, \sigma_{AOA}^2)$ , gdje je  $\theta$  pravi azimut, a  $\sigma_{AOA}^2$  je varijanca zajednička za sva mjerenja u mreži. Za sve simulacije u ovom poglavlju koristi se mreža prikazana na slici 6.2. Mreža se sastoji od 100 čvorova sa prosječnom povezanošću 10, što odgovara topologiji koja je prikazana na istoj slici. Standardna devijacija mjerenja kuta dolaska signala varirana je od  $\sigma = 0^\circ$  savršena mjerenja, do  $\sigma = 15^\circ$  slaba kvaliteta mjerenja. Za svaku točku grafa obavljeno je 10 simulacija te su prikazane minimalne, maksimalne i prosječne dobivene vrijednosti.

### 6.3.1. Međusobna usporedba algoritama sa različitim funkcijama težine optimalnih stabala

U ovom odjeljku prikazani su rezultati simulacije različitih izvedbi algoritma, s obzirom na funkciju koja definira težinu bridova pri formiranju optimalnih stabala za povezivanje grozdova. Opis funkcija dani su u odjeljku 4.3.2, a prikazi topologija za odabranu mrežu na slici 4.5 na 47. stranici.

Usporedba grešaka za sve tri verzije prikazana je na slici 6.5. Iz dobivenih rezultata se može vidjeti da sve tri verzije imaju jednaku kvalitetu. S obzirom da se može izvući zaključak kako kvaliteta estimacije ne ovisi o putu kojim se spajaju grozdovi, ovaj

rezultat može se označiti kao neočekivan.



**Slika 6.5.** Usporedba kvalitete lokalizacije uz prisustvo mjernog šuma za tri različite verzije algoritma s obzirom na težinsku funkciju bridova pri konstrukciji stabla za spajanje grozdova.

Za daljnje analize i usporedbe koristi se ID verzija koja iskorištava inicijalno konstruirano stablo dobiveno u fazi izbora vođe. Ta verzija je najpogodnija što se tiče brzine, broja potrebnih poruka, a posljedično, i količine energije.

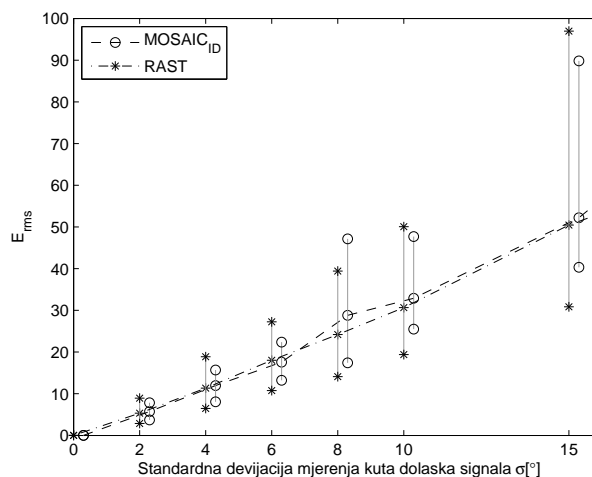
### 6.3.2. Usporedba sa centraliziranom verzijom algoritma

Kako bi što bolje odredili kvalitetu prebacivanja algoritma iz centralizirane verzije u raspodijeljenu, u ovom odjeljku obavlja se izravna usporedba ova dva pristupa. Kao centralizirana verzija, odabran je upravo algoritam RAST [21], koji se koristi za lokalizaciju inicijalnih grozdova, kao što je opisano u odjeljku 4.1. Rezultati su dani na slici 6.6.

Kao što se vidi, rezultati su slični za oba algoritma, čak uz malo dosljednije rezultate raspodijeljene verzije. Može se zaključiti kako raspodijeljena verzija uz sve prednosti koje ima nad centraliziranom nije prouzročila dodatne gubitke u kvaliteti same lokalizacije, što je dokaz da algoritam uspješno odrađuje svoj zadatak.

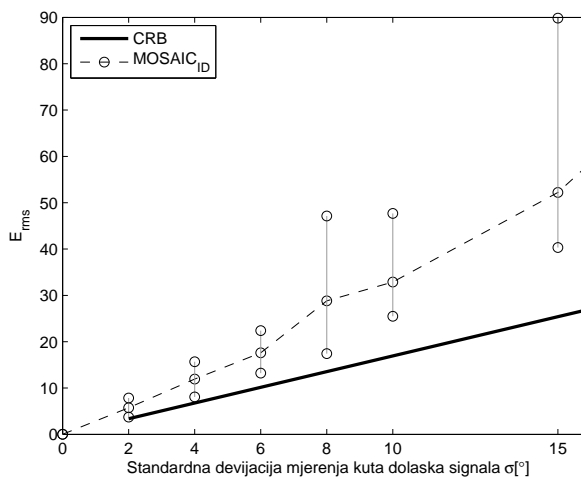
### 6.3.3. Usporedba sa Cramér-Rao limitom

Nezavisna analiza algoritma, kao što je već zaključeno može biti obavljena jedino ako se uspije razdvojiti postavnu i algoritamsku pogrešku. Za taj zadatak koristi se



**Slika 6.6.** Usporedba kvaliteta lokalizacije ID verzije algoritma sa centraliziranom verzijom algoritma (RAST).

Cramér Rao limit opisan u odjeljku 6.1.2. Ova usporedba je vrlo bitna, jer CRB predstavlja rezultate koji bi bili ostvareni koristeći teoretski idealan algoritam za danu topologiju i mjerne pogreške. Na slici 6.7 prikazani su rezultati simulacija.



**Slika 6.7.** Usporedba kvaliteta lokalizacije ID verzije algoritma sa Cramér-Rao limitom.

Rezultati pokazuju kako je dio pogreške uzrokovan isključivo algoritmom predstavlja između 30% – 50% ukupne pogreške. Ovakvi rezultati kazuju kako još uvijek postoji margina u kojoj je moguće poboljšanje, ali i kako algoritam ima sasvim zadovoljavajuće performanse pogotovo za kvalitetnija mjerenja u kojima je varijanca šuma do  $6^\circ$ .

## 7. Zaključak

Područje istraživanja ovog magistarskog rada je razvoj metode te projektiranje raspodijeljenog algoritma za lokalizaciju bežičnih ad hoc mreža osjetila, koji se temelji na određivanju kuta dolaska signala među susjednim čvorovima.

Veliki tehnološki napredak u razvoju mikro elektromehaničkih sustava i sustava za bežičnu komunikaciju omogućio je razvoj složenijih sustava, koji se sastoje od velikog broja malih, računalno ograničenih i međusobno povezanih entiteta. Potencijal koji takvi sustavi nude, u prvom redu se odnosi na raspodijeljena mjerenja, nadgledanja te praćenja koja mogu obuhvatiti velika područja, što je sa dosadašnjim metodama bilo praktički neostvarivo, a zatim i na mogućnosti koje bi takvi sustavi nudili kao aktuatori. Ove mogućnosti otvorile su niz raznovrsnih primjena koje danas tek ulaze u komercijalnu upotrebu, a veliki broj ideja još čeka da tehnologija dovoljno sazrije.

Jedan od bitnih čimbenika, koji određuju iskoristivost podataka dobivenih bežičnom mrežom osjetila, je poznavanje lokacije samih osjetila. U mnogim primjenama sami podaci koji nisu stavljeni u kontekst apsolutne ili relativne lokacije nisu korisni. Prednost poznavanja lokacije pojedinih entiteta se očituje i u mogućnostima naprednijeg upravljanja komunikacijom, poglavito usmjeravanjem i boljim iskorištavanjem medija. S obzirom na energetske cijene komunikacije i u tom smislu se povećava učinkovitost mreže.

Tom motivacijom definirani su ciljevi koji se žele postići, a to je jednostavan, robusan, učinkovit i precizan raspodijeljeni algoritam koji omogućava lokalizaciju najvećeg mogućeg broja čvorova s obzirom na topologiju i mjerne mogućnosti čvorova. Većina postojećih metoda i algoritama ispituje mogućnosti lokalizacije mjerenjem udaljenosti između čvorova.

U ovom radu željelo se istražiti koje su metode na raspolaganju ako se umjesto udaljenosti mjeri kut dolaska signala, s obzirom na manju povezanost koja je potrebna u takvim mrežama, da bi rješenje problema lokalizacije bilo jedinstveno. Projektiran je raspodijeljeni algoritam za lokalizaciju bežičnih mreža. Algoritam pretpostavlja korištenje zašumljenih mjerenja kuta dolaska signala, između čvorova koji su u komunikacijskom dometu. Pretpostavlja se da u mreži ne postoje čvorovi koji znaju svoju točnu lokaciju niti orijentaciju. Za lokalizaciju inicijalnih grozdova (ishodišni čvor i

oni čvorovi s kojim može ostvariti izravnu komunikaciju) koristi se klasična metoda višedimenzionalnog skaliranja (MDS), primijenjena na nepotpun skup mjerenja kao što je to opisano u [21]. Kako bi formalno definirali uvjete za spajanje inicijalnih i svih ostalih grozdova korišteni su rezultati iz teorije krutosti. Pri projektiranju algoritma dana je definicija i opis svih dijelova raspodijeljenog računalnog okruženja, zajedno sa ograničenjima koja su pretpostavljena pri projektiranju algoritma. Naposljetku, algoritam je analiziran kroz niz simulacija u kojima su izvršene usporedbe sa nekim postojećim rješenjima i sa Cramér Rao limitom, kao standardnim pokazateljem kvalitete estimiranih rezultata.

U simulacijama algoritam se pokazao vrlo dobrim, pogotovo pri usporedbi sa centraliziranom verzijom uz variranje šuma mjerenja. Usporedba sa CRB pokazala je zadovoljavajući omjer algoritamske i postavne pogreške, ali i mogućnost napretka u tom smislu.

Glavni doprinosi ovog rada su:

- proširenje teorije krutosti kako bi se formalno definirali uvjeti za spajanje grozdova
- formalno definiranje novih mrežnih struktura koji se temelje na teoriji krutosti, u cilju projektiranja učinkovitog algoritma lokalizacije
- projektiranje i analiza raspodijeljenog algoritma, uz pomoć kojeg se čvorovi lokaliziraju unutar najvećih mogućih (paralelno krutih) struktura, bez propagacije mjernih pogrešaka i sa preciznošću koja je u potpunosti usporediva sa centraliziranom verzijom algoritma
- smanjenje komunikacijskog i računalnog opterećenja eksperimentalnom metodom, kojom je isključen utjecaj izbora stabla duž kojeg će se obavljati spajanje grozdova, te je time omogućeno korištenje najjednostavnije verzije stabla (ID).

## 7.1. Planovi budućeg istraživanja

Projektirani algoritam se u simulacijama pokazao kao vrlo dobro rješenje, ali konačna potvrda i iskoristivost može biti ostvarena tek implementacijom. Problemi koji bi se mogli pojaviti pri implementaciji, poput mnogih drugih primjera implementacija u bežičnim mrežama, predstavljaju veliki izazov i motivaciju za budući rad i istraživanje.

Prvi problem koji je uočen je ostvarivanje kvalitetnih mjerenja kuta dolaska signala. Prema saznanjima do kojih se uspjelo doći, ne postoji komercijalno rješenje ovog problema primjenjivo na same bežične mreže osjetila te bi to mogao biti i prvi korak u budućem istraživanju.

Osim implementacije obavljena ispitivanja su ukazala na postojanje margine unutar koje su moguća metodološka i algoritamska unapređenja. Također, postoji mogućnost dodatne analize principa po kojima bi se mogao smanjiti utjecaj postavne pogreške. Ove i mnoge druge teme ukazuju se kao logičan nastavak istraživanja obavljenog u ovom radu.

# Literatura

- [1] Samba Sesay, Zongkai Yang, and Jianhua He. Survey on mobile ad hoc wireless networks. *Information Technology Journal*, pages 168–175, 2004.
- [2] [www.xbow.com](http://www.xbow.com).
- [3] Ivan Stojmenovic, editor. *Handbook of Sensor Networks: Algorithms and Architectures*. Series on Parallel and Distributed Computing. John Wiley & Sons, Inc., 2005.
- [4] F. Zhao and L.J. Guibas. *Wireless Sensor Networks: An Information Approach*. Morgan Kaufmann Publishers, Elsevier Inc., 2004.
- [5] Joshua N. Ash and Lee C. Potter. Robust system multiangulation using subspace methods. In *IPSN '07: Proceedings of the 6th international conference on Information processing in sensor networks*, pages 61–68, New York, NY, USA, 2007. ACM Press.
- [6] B.K.P. Horn, H. Hilden, and S. Negahdaripour. Closed-form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society of America A*, 5:1127, 1988.
- [7] T. Eren, W. Whiteley, A. S. Morse, P. N. Belhumeur, and B.D.O. Anderson. Sensor and network topologies of formations with direction, bearing and angle information between agents. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 3064–3069, December 2003.
- [8] A. Savvides, H. Park, and M. B. Srivastava. The bits and flops of the n-hop multilateration primitive for node localization problems. In *WSNA '02 September 28, 2002, Atlanta, Georgia, USA*, September 2002.
- [9] A. Savvides, C. Han, and M. B. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of MOBICOM'01, Rome, Italy*, July 2001.

- 
- [10] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *6th ACM International Conference on Mobile Computing and Networking (ACM MOBICOM), Boston, MA, August 2000*, August 2000.
- [11] *nanoNET Chirp Based Wireless Networks, White paper*, 2007.
- [12] Guoqiang Mao, Barış Fidan, and Brian D. O. Anderson. Wireless sensor network localization techniques. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 51(10):2529–2553, 2007.
- [13] K. Romer. The lighthouse location system for smart dust. In *Proceedings of MobiSys 2003 (ACM/USENIX Conference on Mobile Systems, Applications and Services)*, 2003.
- [14] N. B. Priyantha, A. K. L. Miu, H. Balakrishnan, and S. Teller. The cricket compass for context aware mobile applications. In *7th ACMConf. Mobile Computing and Networking (MOBICOM)*. ACM, July 2001.
- [15] A. Nasipuri and K. Li. A directionality based location discovery scheme for wireless sensor networks. In *WSNA '02 September 28, 2002, Atlanta, Georgia, USA*. ACM, September 2002.
- [16] D. Moore, J. Leonard, D. Rus, and S. Teller. Robust distributed network localization with noisy range measurements. *ACM SenSys*, November 2004.
- [17] Nissanka B. Priyantha, Hari Balakrishnan, Erik Demaine, and Seth Teller. Anchor-free distributed localization in sensor networks. Technical Report TR-892, MIT LCS, April 2003.
- [18] D. Niculescu and B. Nath. Ad hoc positioning system (aps). In *Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE*, volume 5, pages 2926–2931 vol.5, 25-29 Nov. 2001.
- [19] D. Niculescu and Badri Nath. Ad hoc positioning system (aps) using aoa. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. IEEE*, volume 3, pages 1734–1743, 30 March – 3 April 2003.
- [20] P. Biswas, H. Aghajan, and Y. Ye. Integration of angle of arrival information for multimodal sensor network localization using semidefinite programming. Technical report, Wireless Sensor Networks Lab, Stanford University, May 2005.
- [21] J.N. Ash and R.L. Moses. Relative and absolute errors in sensor network localization. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 2, pages II–1033–II–1036, 15-20 April 2007.



- 
- [22] D. Arbula. Distributed algorithm for anchor-free network localization using angle of arrival. In *IEEE International Symposium on Industrial Electronics (ISIE'08)*, Cambridge, UK, 2008.
- [23] T. Eren, O.K. Goldenberg, W. Whiteley, Y.R. Yang, A.S. Morse, B.D.O. Anderson, and P.N. Belhumeur. Rigidity, computation, and randomization in network localization. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 4, pages 2673–2684, 7–11 March 2004.
- [24] Tolga Eren, Walter Whiteley, and Peter N. Belhumeur. A theoretical analysis of the conditions for unambiguous node localization in sensor networks. Technical report, Department of Computer Science, Columbia University, 2004.
- [25] T. Eren, B.D.O. Anderson, W. Whiteley, A.S. Morse, and P.N. Belhumeur. Merging globally rigid formations of mobile autonomous agents. In *Autonomous Agents and Multiagent Systems, 2004. AAMAS 2004. Proceedings of the Third International Joint Conference on*, pages 1258–1259, July 2004.
- [26] D.J. Torrieri. Statistical theory of passive location systems. *Aerospace and Electronic Systems, IEEE Transactions on*, 20(2):183–198, March 1984.
- [27] Nicola Santoro. *Design and analysis of distributed algorithms*. Series on Parallel and Distributed Computing. John Wiley & Sons, Inc., 2007.
- [28] Jakub Hrabec. Autonomous mobile robotics toolbox simrobot. <http://www.uamt.feec.vutbr.cz/robotics/simulations/amrt/simrobot.zip>, June 2001.
- [29] Toni Petrinic, Edouard Ivanjko, and Ivan Petrovic. Amorsim - a mobile robot simulator for matlab. In *Proceedings of 15th International Workshop on Robotics in Alpe-Adria-Danube Region, Balatonfüred, Hungary*, June 15–17 2006.
- [30] A. Savvides, W. Garber, S. Adlakha, R. Moses, and M.B. Srivastava. On the error characteristics of multihop node localization in ad-hoc sensor networks. In *Proceedings of the Second International Workshop on Information Processing in Sensor Networks (IPSN'03)*, pages 317 – 332, Paolo Alto, California, April 2003.
- [31] N. Patwari, J.N. Ash, S. Kyperountas, III Hero, A.O., R.L. Moses, and N.S. Correal. Locating the nodes: cooperative localization in wireless sensor networks. *Signal Processing Magazine, IEEE*, 22(4):54–69, July 2005.
- [32] R.L. Moses and R. Patterson. Self calibration of sensor networks. In *Proceedings of SPIE*, volume 474 of *Unattended Ground Sensor Technologies and Applications IV*, April 2002.

- 
- [33] Cheng Chang and A. Sahai. Estimation bounds for localization. In *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, pages 415–424, 4–7 October 2004.

# Prilozi

## Prilog 1. Robusna angulacija

Svaka izmjerena vrijednost azimuta između prijemnika  $\mathbf{r}$  i odašiljača  $\mathbf{t}$ , može se predstaviti kao jedinični vektor rotiran za  $90^\circ$  od pravog azimuta

$$\mathbf{u}_{rt} = \begin{bmatrix} \sin \theta_{rt} \\ \cos \theta_{rt} \end{bmatrix} \quad (\text{P.1})$$

gdje je  $\theta_{rt}$  azimut odašiljača prema prijamniku u globalnom koordinatnom sustavu. Kako su orijentacije čvorova u globalnom koordinatnom sustavu nepoznate, poznat je samo izmjereni azimut u relativnom koordinatnom sustavu prijamnog čvora  $\phi_{rt}$ . Prema tome, potrebno je estimirati i globalnu orijentaciju prijamnog čvora  $\alpha_r$  kako bi se iz

$$\theta_{rt} = \phi_{rt} + \alpha_r \quad (\text{P.2})$$

izračunao globalni azimut  $\theta_{rt}$ .

Ovaj dio algoritma se razlikuje od originalnog algoritma opisanog u [5]. U slučaju lokalizacije inicijalnih podgrozdova, estimiranje globalnih orijentacija čvorova je jednostavno, jer su svi čvorovi pojedinih podgrozdova povezani sa ishodišnim čvorom. Jednostavnom metodom opisanom u odjeljku 3.3.2, svim se čvorovima može uspješno estimirati globalna orijentacija u odnosu na orijentaciju ishodišnog čvora.

Kako je  $\mathbf{u}_{rt}$  orijentiran pod kutom od  $90^\circ$  prema pravom azimutu, odnosno razlici lokacija odašiljačkog i prijamnog čvora, tada vrijedi sljedeća jednačba:

$$\mathbf{u}_{rt}^T(\mathbf{p}_t - \mathbf{p}_r) = 0 \quad (\text{P.3})$$

gdje su  $\mathbf{p}_t$  i  $\mathbf{p}_r$  lokacije odašiljačkog i prijamnog čvora.

Sustav sastavljen od jednačbi (P.3), koje će reprezentirati mjerenja azimuta između svih parova susjednih čvorova, može se napisati u matičnom obliku:

$$U^T K \mathbf{p} = 0 \quad (\text{P.4})$$

gdje je  $U$  blok dijagonalna matrica, sastavljena od vektora  $u_{rt}$  svih parova susjeda, a  $\mathbf{p}$  vektor stupac koordinata svih  $n$  čvorova. Matrica  $K$  je posebno konstruirana matrica sastavljena od elemenata 0,  $-1$  i  $1$ , na taj način da se umnoškom sa vektorom pozicija kreiraju razlike lokacija na odgovarajućim pozicijama rezultantne matrice.

Rješenje problema relativne lokalizacije je partikularno rješenje linearnog homogenog sustava (P.4), pri čemu je uz dovoljan broj mjerenja dimenzija nulprostora matrice  $U^T K$  jednaka 3. Dimenzije odgovaraju kongruentnim transformacijama paralelno krutih formacija. Jedna dimenzija se može predstaviti vektorom translacije po  $x$ -osi  $\mathbf{v}_x = [1 \ 0 \ 1 \ 0 \ \dots]^T$ , druga vektorom translacije po  $y$ -osi  $\mathbf{v}_y = [0 \ 1 \ 0 \ 1 \ \dots]^T$ , a treća dimenzija predstavlja vektor skaliranja, koji je jednak upravo vektoru relativnih lokacija  $\mathbf{p}$ .

Dodavanjem vektora  $\mathbf{v}_x$  i  $\mathbf{v}_y$  matrici  $U^T K$

$$A = \begin{bmatrix} U^T K \\ \mathbf{v}_x^T \\ \mathbf{v}_y^T \end{bmatrix} \quad (\text{P.5})$$

reducira se nulprostor te vrijedi  $\mathcal{N}(A) = \text{span}(\mathbf{p})$ . Prema tome sustav:

$$A\mathbf{p} = 0 \quad (\text{P.6})$$

je rješiv koristeći se rastavom singularnih vrijednosti  $A = U_A \Sigma_A V_A^T$ , pri čemu je najbolja estimacija vektora relativnih lokacija  $\mathbf{p}$  singularni vektor, koji odgovara najmanjoj singularnoj vrijednosti matrice  $A$

$$\hat{\mathbf{p}} = V_A^{(2n)}. \quad (\text{P.7})$$

## Prilog 2. Pseudokod algoritma Mega-merger

```

: S = {ON, DOWNTOWN, DISTRICT}
: SINIT = {ON}
: TERM = {DOWNTOWN, DISTRICT}

ON
  Spontaneously
  begin
    initialize;
    become DOWNTOWN;
  end
  Receiving ("Outside?",name *,level *)
  begin
    initialize;
    if level *==level then
      send("External") to sender;
    else
      [sender name * level *] → outsidequeue;
    end
    become DOWNTOWN;
  end
DOWNTOWN
  Receiving ("Outside?",name *,level *)
  begin
    if name *==name then
      send("Internal" to sender);
      remove sender from roads;
    else if level *<=level then
      send("External") to sender;
    else
      [sender name * level *] → outsidequeue;
    end
  end
  Receiving ("Internal")
  begin
    remove sender from roads;
    if size(roads)>0 then
      next =mindist(roads);
      send("Outside?",name,level) to next;
    else
      di =∞;
      [id di] → shortest;
      updateShortestDowntown;
    end
  end
  Receiving ("External")
  begin
    di =[min(id sender) max(id sender)];
    [sender di] → shortest;
    updateShortestDowntown;
  end
  Receiving ("Let-us-merge",name *,level *)
  begin
    mergeDowntown;
  end
  Receiving ("Notification",name *,level *)
  begin
    name =name *;
    remove sender from roads, roadsup, active, shortest;
    roaddown =sender;
    send("Notification",name *,level *) to active;
    become DISTRICT;
    minroadup =[0 1];
    if level *>level then
      level =level *;
      resolveQueues;
    end
  end
  end

  Receiving ("Start-next",name *,level *)
  begin
    name =name *;
    remove sender from roads, roadsup, active;
    roaddown =sender;
    send("Notification",name *,level *) to active;
    become DISTRICT;
    shortest =[];
    minroadup =[0 0];
    if level *>level then
      level =level *;
      resolveQueues;
    end
    newSearch;
  end
  Receiving ("Start-next-reply",di *)
  begin
    if di *==∞ then
      remove sender from active;
    else
      [sender di *] → shortest;
    end
    updateShortestDowntown;
  end
DISTRICT
  Receiving ("Outside?",name *,level *)
  begin
    if name *==name then
      send("Internal" to sender);
      remove sender from roads;
    else if level *<=level then
      send("External") to sender;
    else
      [sender name * level *] → outsidequeue;
    end
  end
  Receiving ("Internal")
  begin
    remove sender from roads;
    newSearch;
  end
  Receiving ("External")
  begin
    di =[min(id,sender) max(id,sender)];
    [sender di] → shortest;
    if size(shortest)==size(active)+1 then
      di =min(shortest (:,2));
      minroadup =[shortest (min(shortest (:,2),1)) 1]
    end
    ;
    send("Start-next-reply",di) to roaddown;
  end
  Receiving ("Let-us-merge",name *,level *)
  begin
    if sender ==roaddown then
      send("Let-us-merge",name *,level *) to minroadup
      (1);
      minroadup (2)=2;
      if minroadup (1)∉ roadsup then
        if minroadup (1)∈ mergequeue then
          remove minroadup (1) from mergequeue;
          friendlyMerger;
        end
      end
    else
      mergeDowntown;
    end
  end
  end
end

```

```

Receiving ("Notification",name *,level *)
begin
  name =name *;
  if sender ==minroadup (1) then
    remove sender from roads, roadsup, active,
    shortest;
    roaddown → roadsup;
    roaddown →active;
    roaddown =sender;
    minroadup =[0 1];
  end
  send("Notification",name *,level *) to active;
  if level *>level then
    level =level *;
    resolveQueues;
  end
end

Receiving ("Start-next",name *,level *)
begin
  name =name *;
  if sender ==roaddown then
    send("Start-next",name *,level *) to active;
  else
    remove sender from roads, roadsup, active,
    shortest;
    roaddown → roadsup;
    roaddown →active;
    roaddown =sender;
    send("Start-next",name *,level *) to active,
    roaddown;
  end
  minroadup =[0 0];
  if level *>level then
    level =level *;
    resolveQueues;
  end
  shortest =[]; newSearch;
end

Receiving ("Start-next-reply",di *)
begin
  if di *==∞ then
    remove sender from active;
  else
    [sender di *] → shortest;
  end
  updateShortestDistrict;
end

PROCEDURE initialize
begin
  name =id;
  level =1;
  roads =sort(neighbors);
  roaddown =[];
  roadsup =[];
  active =[];
  shortest =[];
  minroadup =[0 0];
  mergequeue =[];
  outsidequeue =[];
  roads →next;
  send("Outside?",name,level) to next;
end

PROCEDURE updateShortestDowntown
begin
  if size(shortest)==size(active)+1 then
    di =min(shortest (:,2));
    if di ≠ ∞ then
      minroadup =[shortest (min(shortest (:,2)),1) 2];
      send("Let-us-merge",name,level) to minroadup
      (1);
      if minroadup (1)∉ roadsup then
        if minroadup (1)∈ mergequeue then
          remove minroadup (1) from
          mergequeue;
          friendlyMerger;
        end
      end
    else
      % global termination
    end
  end
end

PROCEDURE updateShortestDistrict
begin
  if size(shortest)==size(active)+1 then
    di =min(shortest (:,2));
    if di ≠ ∞ then
      minroadup =[shortest (min(shortest (:,2)),1)
      1];
    else
      % local termination
    end
    send("Start-next-reply",di) to roaddown;
  end
end

PROCEDURE mergeDowntown
begin
  if level *==level then
    if sender ==minroadup and minroadup (2)==2 then
      friendlyMerger;
    else
      sender →mergequeue;
    end
  else if level *<level then
    sender →roadsup, active;
    remove sender from roads;
    if minroadup (2)==0 then
      send("Start-next",name,level) to sender;
    else
      send("Notification",name,level) to sender;
    end
  end
end

PROCEDURE friendlyMerger
begin
  level =level +1;
  if DISTRICT then
    roaddown →roadsup, active;
  end
  if minroadup (1)>id then
    name =id;
    send("Start-next",name,level) to active;
    minroadup (1)→roadsup, active;
    if DISTRICT then
      become DOWNTOWN;
    end
  else
    name =minroadup (1);
    roaddown =minroadup (1);
    send("Start-next",name,level) to active;
    if DOWNTOWN then
      become DISTRICT;
    end
  end
  remove minroadup (1) from roads;
  minroadup =[0 0];
  resolveQueues;
  shortest =[];
  newSearch;
end

PROCEDURE newSearch
begin
  if size(roads)>0 then
    next =mindist(roads);
    send("Outside?",name,level) to next;
  else
    di =∞;
    [id di ]→shortest;
    if DISTRICT then
      updateShortestDistrict;
    else if DOWNTOWN then
      updateShortestDowntown;
    end
  end
end
end

```

```

PROCEDURE resolveQueues
begin
  if size(mergequeue) > 0 then
    for i=1 to size(mergequeue) do
      mergequeue(i) → roadsup, active;
      remove mergequeue(i) from roads;
      if minroadup(2) = 0 then
        send("Start-next", name, level) to mergequeue(i);
      else
        send("Notification", name, level) to mergequeue(i);
      end
    end
    mergequeue = [];
  end
  if size(outsidequeue) > 0 then
    for i=size(outsidequeue) to 1 do
      if level * >= outsidequeue(i,3) and name * ≠ outsidequeue(i,2) then
        send("External") to outsidequeue(i,1);
        remove outsidequeue(i,:) from outsidequeue;
      else if level * = outsidequeue(i,3) and name * = outsidequeue(i,2) then
        send("Internal") to outsidequeue(i,1);
        remove outsidequeue(i,:) from outsidequeue, roads;
      end
    end
  end
end
end
end

```

Ime registra	Tip podatka	Opis
name	id	Ime grada
level	int	Razina grada
roads	[id distance]	Neistražene ceste sortirane prema udaljenostima
roaddown	id	Ime ceste (id čvora s druge strane) koja vodi prema centru, prethodnik u stablu
roadsup	[id]	Imena cesta prema ostalim susjednim naseljima u istom gradu, sljedbenici u stablu
active	[id]	Roadsup susjedi koji nisu lokalno terminirani, njima se šalju start next poruke i čeka ih se kod merge-link potrage
shortest	[roadup distance]	Najkraće ceste koje vode van grada od čvora i njegovih prvih roadsup susjeda
minroadup	[minroadup act]	Ime roadsup susjeda koji je na putu do najkraće ceste tog dijela stabla; act ∈ {0, 1, 2} 0 – potraga nije završila u čvoru, 1 – potraga je završila u čvoru, 2 – potraga je završila i čvor je na putu od centra prema izabranoj cesti
outsidequeue	[sender name level]	Spremnik za za odgovor na Outside? upit gradova (trenutno u zastoju), služi dok se čeka na povećanje razine
mergequeue	[sender]	Spremnik za absorpciju sa popisom gradova (trenutno u zastoju) koji su tražili spajanje, služi dok se čeka na povećanje razine

**Tablica P.1.** Registri koji se koriste u algoritmu *Mega-Merger*

## Prilog 3. Pseudokod algoritma za lokalizaciju inicijalnih grozdova

```

: S = {ROOT, FOLLOWER, CLUSTERLOC, ASLEEP}
: SINIT = {ROOT, FOLLOWER}
: STERM = {ROOT, ASLEEP}
ROOT
  Spontaneously
  begin
    bearings = AOA(neighbors);
    send("Bearings",bearings) to neighbors;
    ack = 0;
    become CLUSTERLOC;
  end
FOLLOWER
  Receiving ("Bearings",bearings *)
  begin
    bearings = AOA(neighbors);
    send("Bearings",bearings) to neighbors;
    bearings * → bearings;
    ack = 0;
    become CLUSTERLOC;
  end
CLUSTERLOC
  Receiving ("Bearings",bearings *)
  begin
    bearings * → bearings;
    if size(unique(bearings)) == size(neighbors)+1 then
      costOut = calculate();
      send("Cost",cost) to neighbors;
      localizeSubcluster;
      ack = ack + 1;
      checkAck;
    end
  end
  Receiving ("Cost",costOut)
  begin
    costOut → costIn;
    checkAck;
  end
  Receiving ("Ack")
  begin
    ack = ack + 1;
    checkAck;
  end
PROCEDURE checkAck
  begin
    if ack == size(upneighbors + 1) then
      if ~isempty(downneighbor) then
        send("Ack") to downneighbor
      end
      ack = ack + 1;
    end
    if size(costIn) == size(neighbors) and ack == size(upneighbors)+2 then
      cost = combine(costOut,costIn);
      if ~isempty(downneighbor) then
        become ASLEEP;
        % local termination
      else
        % global termination
      end
    end
  end
end
end

```



# Curriculum Vitae

## Osnovne informacije

Ime: **Damir Arbula**

Radno mjesto: Znanstveni novak na  
Tehničkom fakultetu,  
Zavod za automatiku, elektroniku i računarstvo,  
Sveučilište u Rijeci, Hrvatska.

e-mail: damir.arbula@riteh.hr

## Education

1997–2002 Diploma iz područja elektrotehnike,  
smjer radiokomunikacije i profesionalna elektronika,  
Fakultet elektrotehnike i računarstva,  
Sveučilište u Zagrebu, Hrvatska.

1993–1997 Gimnazija Andrija Mohorovičić,  
Rijeka, Hrvatska.

## Strani jezici

Jezici:

1. Hrvatski
2. Engleski
3. Talijanski

## Radno iskustvo

- 2004–2008 Znanstveni novak na  
Tehničkom fakultetu,  
Zavod za automatiku, elektroniku i računarstvo,  
Sveučilište u Rijeci, Hrvatska.
- 2004 Projektant u tvrtci Control engineering iz Rijeke.
- 2002–2004 Grafički dizajner i fotograf u firmi Visuality studio iz Rijeke.
- 2000–2001 Glazbeni urednik i tehničar u Studentskom centru.

## Nastavne aktivnosti

Elementi automatizacije postrojenja	2004–2008
Sustavi digitalnog upravljanja	2005–2008
Primjena računala	2006
Osnove automatske regulacije	2006–2008

## Popis radova

D. Arbula. Distributed algorithm for anchor-free network localization using angle of arrival. In IEEE International Symposium on Industrial Electronics (ISIE'08), Cambridge, UK, 2008.

# Sažetak

Problem lokalizacije u bežičnim mrežama osjetila i ad hoc bežičnim mrežama rješiv je korištenjem izmjerenih udaljenosti ili azimuta između susjednih čvorova. Uz pomoć tih ograničenja rješiv je jedino problem relativne lokalizacije, dok za specifičiji problem apsolutne lokalizacije potreban je skup čvorova - sidara kojima je poznata apsolutna lokacija.

U ovom radu obrađen je pristup projektiranja i evaluacije raspodijeljenog algoritma za lokalizaciju u neusidrenoj mreži. Algoritam se koristi samo sa azimutima između čvorova susjeda, izmjerenim unutar relativnog koordinatnog sustava samog čvora nepoznate orijentacije. Algoritam se temelji na podjeli mreže u grozdove. Nakon što se lokaliziraju inicijalni čvorovi, njihovim međusobnim spajanjem lokalizira se cjelokupna mreža. Korištenjem teorije krutosti grafova definirane su nove mrežne strukture i njihovi odnosi, na taj način da algoritam može odrediti relativne lokacije maksimalnog broja čvorova za zadani broj izmjerenih azimuta.

Performanse algoritma ispitivane su kroz niz simulacija, a dobiveni rezultati su usporedivi sa centraliziranom verzijom algoritma. Dodatnom analizom pokazalo se kako performanse ne ovise bitno o putu i poretku kojim se spajaju grozdovi. Usporedbom performansi sa Cramér Rao limitom pokazuje se kako pogreška uzrokovana algoritmom čini 30% – 50% ukupne pogreške lokalizacije.

**Ključne riječi:** bežične mreže osjetila, ad hoc mreže, teorija krutosti grafova, lokalizacija, kut dolaska signala, raspodijeljeni algoritmi.

# Abstract

A localization problem in wireless sensor and ad hoc networks can be solved using measured distances and/or directions between neighboring nodes as constraints. With internode constraints only general the relative localization problem can be solved as for more specific absolute positions, a set of prelocalized nodes called anchors is needed.

In this thesis approach in design and evaluation of a distributed anchor-free localization algorithm is presented. It is using only direction constraints relative to node's coordinate system as their orientations are assumed to be unknown. The algorithm is based on division of a network in clusters. After initial clusters are localized, by stitching them together it can localize whole network. Using graph rigidity theory we defined new structures and their relationships so the algorithm can provide relative locations to maximum number of nodes for given number of constraints.

The performance was evaluated through simulations and obtained results are similar to centralized version of algorithm. Additional analysis showed that the performance is not dependant on a path and order used for cluster stitching. By comparing the performance to the Cramér Rao bound, it is revealed that the error induced by the algorithm represents 30% – 50% of the total localization error.

**Keywords:** wireless sensor networks, ad hoc networks, graph rigidity theory, localization, angle of arrival, distributed algorithms.