



FFI-rapport 2014/00657

Ruting i trådløse sensornettverk



Joakim Flathagen



Ruting i trådløse sensornettverk

Joakim Flathagen

Forsvarets forskningsinstitutt (FFI)

7. mai 2014

FFI-rapport 2014/00657

4496

P: ISBN 978-82-464-2374-6

E: ISBN 978-82-464-2375-3

Emneord

Perimetersikring

Rutingprotokoller

Sensornettverk

Styrkebeskyttelse

Godkjent av

Rune Lausund

Prosjektleder

Jon E. Skjervold

Avdelingssjef

Sammendrag

I militære operasjoner er det et stort behov for sikring av militære posisjoner. Forsvar av slike posisjoner gjøres ved en kombinasjon av fysisk sikring, overvåkning og reaksjon; som alle tre må balanseres. Elektronisk sensormateriell er et viktig virkemiddel for å oppnå tidlig varsling av inntrenging og dermed redusert reaksjonstid. Det finnes en rekke sensortyper på markedet beregnet for varsling av inntrenging. Av disse vil trådløse sensorer være å foretrekke ved beskyttelse av posisjoner av midlertidig karakter. Trådløse sensorer gir kortere deployeringstid og større frihet i valg av sensorplassering enn trådbaserte systemer. Det vil dessuten være mulig å benytte et større antall sensorer, noe som i kombinasjon med dataaggregering kan minske sannsynligheten for falske alarmer.

Trådløse sensorer introduserer på den annen side noen kommunikasjonsutfordringer. Da slike sensorer i tillegg er batteridrevne, og gjerne er tallrike, er det spesielt viktig at kommunikasjonsprotokollene er energieffektive og skalerbare. Rutingprotokollen sørger for at meldinger (alarmer og målinger) som produseres i et sensornettverk transporteres fra sensor til ønsket mottaker. I denne rapporten undersøkes følgende problemstillinger knyttet til ruting og energieffektivitet i trådløse sensornettverk: *i)* Samtidig posisjonering og ruting; *ii)* dataaggregering; *iii)* kommunikasjon mot eksterne systemer; *iv)* plassering av sinknoder og ruting i fler-sinksnettverk; og *v)* kommunikasjon i et miljø med radioforstyrrelser.

Det er utført en rekke simuleringer og laboratorieforsøk som adresserer disse problemstillingene. En hovedhensikt med arbeidet har vært å finne metoder som gir effektiv ruting i sensornettverk samtidig som en god energiutnyttelse sikres. En av de beste måtene å oppnå dette på er å bruke flere sinknoder i nettverket. Dette gir både høy effektivitet og god redundans og er en veldig enkel metode dersom scenarioet tillater bruk av multiple sinker. Dette kan også kombineres med dataaggregering. F.eks. ved beskyttelse av en base over et lengre tidsrom, vil det være fornuftig å benytte flere sinker for å øke redundansen og redusere antall hopp i nettverket. Samtidig kan man gjøre aggregering og filtrering av sensormeldinger i nettverket for å minske sannsynligheten for falske alarmer og for å øke energieffektiviteten ytterligere.

En annen hovedkonklusjon er at krysslagskoblinger mellom rutingprotokollen og andre protokoller kan bidra både til vesentlig ytelsesforbedring og økt funksjonalitet i sensornettverket. For eksempel kan krysslagskoblinger muliggjøre både aggregering, som gir lengre levetid til nettverket, og lokasjon, som er en nødvendig funksjon i mange sensornettverk. En siste anbefaling gjelder militære sensornettverk som er spesielt sårbare for radioforstyrrelser som f.eks. jamming. I rapporten anbefales det å benytte en hybrid rutingprotokoll. Denne protokollen gir god balanse mellom pakkeleveringsrate og energieffektivitet, både når nettverket er utsatt for høy og lav grad av radioforstyrrelse.

English summary

In military operations, there is a great need for protecting military positions. This is typically accomplished by a balanced combination of physical protection, surveillance and elimination of the threat. The use of electronic sensors is an important means to achieve early warning of intrusions and thus reduced reaction time. There is a variety of sensor types on the market intended for the detection of intrusion. Among these are wireless sensors preferable for protecting temporary camps and positions. Wireless sensors provide shorter deployment time and greater freedom regarding sensor placement compared to wired systems. Furthermore, a large number of sensors can be deployed, which can reduce the false-alarm probability when combined with other techniques such as data aggregation.

Wireless sensors do on the other hand introduce a range of communication challenges. Considering that wireless sensors are battery powered and often abundant, it is very important that the communication protocols are energy efficient and scalable. The routing protocol ensures that messages (alarms and measurements) produced in the sensor network is transported from the sensor to the desired recipient. This report investigates the following issues related to routing and energy efficiency in wireless sensor networks: *i*) Simultaneously positioning and routing; *ii*) data aggregation; *iii*) interoperability with external systems; *iv*) sink node placement and routing in a multi-sink network; and *v*) communication interfered environment.

A number of simulations and laboratory experiments are performed to address these issues. The main purpose of this work has been to find methods for efficient routing in sensor networks while ensuring good energy utilization. One the recommended ways to achieve this is to use multiple sink nodes in the network. The use of multiple sinks provides both high efficiency and redundancy. Multiple sinks can also be combined with data aggregation.

Another main conclusion is that cross-layer optimizations between the routing protocol and other protocols can contribute both to a significant performance improvement and to increase the functionality in the sensor network. For example, cross layer optimizations can enable data aggregation, which provides prolonged network lifetime, and a location service, which is a necessary feature in many sensor networks. One last recommendation in the report applies to military sensor networks subjected to interference or jamming. The report recommends a hybrid routing protocol for this purpose. This protocol provides a good balance between packet delivery rate and energy efficiency, both in circumstances when the network is subjected to high interference and when it is subjected to low interference.

Innhold

1	Innledning	7
1.1	Bakgrunn	7
1.2	Rapportens oppbygning	8
2	Samtidig ruting og posisjonering	9
2.1	Motivasjon	9
2.2	Bakgrunn	10
2.3	Samtidig ruting og posisjonering	10
3	Ruting og clustering	13
3.1	Motivasjon	13
3.2	Bakgrunn	14
3.3	Edge-Betweenness clustering i sensornettverk	15
4	Sinkplasseringen og sinkantallets innvirkning på ruting	18
4.1	Motivasjon	18
4.2	Bakgrunn	18
4.3	Sinkplassering	19
5	Samhandling mot eksterne systemer	22
5.1	Motivasjon	22
5.2	Bakgrunn	22
5.3	Bruk av Web Services mot NATO Network Enabled Capability	24
6	Hvilken innvirkning har jamming og interferens på ruting?	26
6.1	Motivasjon	26
6.2	Bakgrunn	26
6.3	Opportunistic Collection Tree Protocol	27
7	Oppsummering	29
Appendiks A Akronymer og forkortelser		31
Bibliografi		37

1 Innledning

1.1 Bakgrunn

Å benytte trådløse sensorer for taktiske formål er en over femti år gammel idé som først ble realisert med «Air Delivered Seismic Intrusion Detector (ADSID)» av US Air Force under Vietnamkrigen. Siden den tid har de fleste nasjoner eksperimentert med trådløse sensorsystemer av ulik art for det militære domenet. Parallelt med økt militær bruk har den sivile utviklingen innen mikroelektronikk og trådløs teknologi muliggjort stadig rimeligere og mer energieffektive sensorsystemer. Det er derfor, til tross for konseptuelle likheter, en markant forskjell mellom de første enkle systemene og dagens systemer (Figur 1.1).

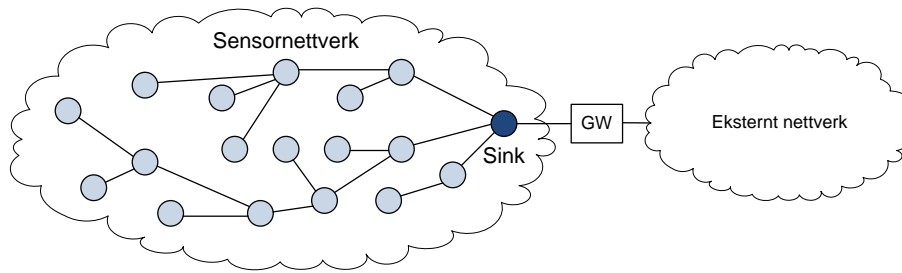


(a) Trådløs sensor (ADSID) droppet fra fly over Laos i 1967 (b) Seismisk trådløs marksensor i 2013

Figur 1.1 Trådløse sensorer har vært benyttet innen det militære domenet siden sekstitallet.

De tidlige systemene bestod av teknisk enkle sensornoder uten særlig prosesseringskapasitet. De var gjerne store og opererte uavhengig av hverandre og sendte ufiltrerte sensormålinger direkte inn til en basestasjon. Nye systemer er derimot basert på *multihopp nettverksteknologi* på lik linje med pakkerutingen i Internett. Dette muliggjør stor redundans og automatisk feilkorrigering. Videre kan sensornodene samarbeide under deteksjon slik at sannsynligheten for falske alarmer minimeres. Minimal størrelse på hver enkelt node og lave kostnader muliggjør systemer med et stort antall noder, men som likevel er enkle å deployere.

Figur 1.2 illustrerer et typisk sensornettverk. En rekke sensornoder er i figuren deployert i det området som man ønsker å overvåke eller som man vil utføre målinger i. Sensornodene knyttes sammen ved hjelp av trådløs kommunikasjon og en rutingprotokoll. Datapakker samles inn av en enkelt node («sink»). Denne sinknoden er igjen knyttet mot en gateway (GW) som utgjør grensesnittet mot et eksternt nettverk.



Figur 1.2 Prinsippet til et sensornettverk. Sensorene i nettverket produserer informasjon som rutes mot en «sink» som igjen sender informasjonen via en gateway mot det eksterne nettverket.

1.2 Rapportens oppbygning

Denne rapporten omhandler pakkeruting i trådløse sensornettverk. Oppgaven til en rutingprotokoll er i tradisjonelle kommunikasjonssystemer begrenset til å finne den optimale veien mellom en avsender og en destinasjon. Selve oppgaven med å rute pakken, det vil si å motta en pakke, lese destinasjonsadressen og sende den videre basert på informasjon i rutingtabeller, er overlatt til operativsystemet. I sensornettverk er disse to funksjonene ofte slått sammen, og rutingprotokollen har ansvar både for å finne rutene og for å videresende datapakker langs disse rutene.

Rutingprotokollen representerer dermed hele kjernen i nettverkssystemet, og som vi skal se i denne rapporten avhenger ytelsen til sensornettverket av hvordan denne protokollen er designet. I tillegg skal vi undersøke hvorvidt ulike protokoller og algoritmer i sensornettverket kan benytte krysslagskoblinger til å nyttiggjøre seg av funksjonalitet som finnes i rutingprotokollen, og motsatt; vi ser på hvorledes krysslagskoblinger kan bidra til at rutingprotokollen kan utnytte overliggende kommunikasjonslag.

Arbeidet har vært fokusert på fem problemstillinger som adresseres i fem ulike kapitler.

1. Kan samme protokoll samtidig utføre ruting og posisjonering (Kapittel 2)
2. Kan man oppnå en gevinst ved at rutingprotokollen oppretter clustre og gjør dataaggregering via disse? (Kapittel 3)
3. Hvordan kan eksterne systemer samvirke med sensornettverket på en robust og effektiv måte? (Kapittel 4)
4. Hvordan blir rutingen påvirket av antallet sinknoder og deres plassering? (Kapittel 5)
5. Hvordan påvirkes rutingen av jamming og interferens? (Kapittel 6)

For hvert av disse fem kapitlene presenterer vi problemstillingen i detalj, gir en oversikt over tidligere forskning, og presenter våre resultater innen temaet. Utdypende detaljer kan finnes i publiserte artikler [1–5] samt i Flathagens doktoravhandling [6]. Denne rapporten kan derfor leses som en forenklet og fornorsket oppsummering av tidligere publiserte arbeider. Dersom leseren skulle ønske utfyllende detaljer rundt implementasjon og forsøk henvises det med andre ord til artiklene.

Trådløse sensornoder

«Trådløse sensornoder» er ingen entydig definisjon og begrepet spenner vidt fra det ene ytterpunktet som omtaler store og avanserte radiokommuniserende sensorenheter som må transporteres på lastebil; til det andre ytterpunktet som beskriver sensorer på størrelse med støvkorn som spres i tusentalls og som knyttes i nettverk. Gjeldende praktiske implementasjoner av trådløse sensornettverk samt vår definisjon i denne rapporten er noe midt i mellom disse ytterpunktene: Vi definerer en trådløs sensornode som en node som inneholder en eller flere sensorer, en mikrokontroller, en radio og batteri. Alt dette er montert i en innkapsling på størrelse med en kortstokk.

Mens sensortypen avhenger av bruksområdet til sensornoden, er mikrokontrolleren, radioen og batteriet som regel basert på tilnærmet lik teknologi på tvers av ulike bruksområder. Mikrokontrolleren som benyttes i trådløse sensornoder i dag har gjerne en 8- eller 16-bits arkitektur med et effektforbruk på 1-10mW. Typiske mikrokontrollere er Atmels AVR (8-bits) og Texas MSP430 (16-bit), men 32-bits arkitekturer som ARM Coretex M3 benyttes også. Blant operativsystemer for dette formålet er TinyOS, Contiki og Arduino de mest utbredte i dag.

Trådløse sensornoder har et bredt og stadig økende anvendelsesområde og vil snart være allsstedsnærværende. Begreper som smarthus, sensornettverk og «tingenes internett», omtaler alle trådløse sensornoder i familie med den typen som beskrives i denne rapporten.

2 Samtidig ruting og posisjonering

2.1 Motivasjon

I de fleste sensornettverk vil det være et behov for å vite sensornodenes eksakte posisjoner i terrenget. Innen militær overvåkning kan den nøyaktige posisjonen til en inntrenger kun bestemmes dersom sensorene har kjente posisjoner. Et annet eksempel er i et sensornettverk for miljøovervåkning. Her vil de målte data være av begrenset nytteverdi dersom man ikke vet ved hvilken lokasjon dataene er innhentet. Nøyaktig stedinformasjon kan også være viktig for enkelte tekniske delkomponenter i sensornettverket som for eksempel geografisk ruting, datafusjon, og datafiltrering [7–10]. Alle disse komponentene vil kunne oppnå forbedret ytelse dersom de har kjennskap til sensorenes posisjoner.

Bruk av satellittnavigasjonssystemer (f.eks GPS eller Galileo) er en opplagt og intuitiv metode for posisjonering. Det er spesielt i sensornettverk for utendørs bruk dette er aktuelt. Ulempen med satellittnavigasjon er at posisjoneringen er upresis dersom det ikke er fri sikt mellom mottaker og satellitter. I tillegg vil satellittnavigasjon øke både systemkostnadene og energiforbruket, spesielt i rimelige og enkle sensornettverk som denne rapporten omhandler. Her er det derfor undersøkt metoder for å utføre lokalisering i sensornettverk uten behov for ekstra maskinvare. Vår metode utnytter isteden informasjon som er tilgjengelig i selve rutingmekanismen og i de ulike kommunikasjonslagene.

2.2 Bakgrunn

De mest primitive metodene for å posisjonere sensornoder baserer seg på informasjon om nettverkets konnektivitet. Dette kan gi et grovt estimat av nodenes innbyrdes plassering. Et eksempel på en slik metode er presentert av Tran et al. [11]. Siden deres metode ikke krever noen spesiell maskinvare, kan den implementeres selv på meget rimelige sensorenheter. En slik løsning vil imidlertid gi veldig stor unøyaktighet i posisjonsestimater siden avstanden mellom nodene ikke måles. Unøyaktigheten blir spesielt høy i spredte nettverk med få sensornoder.

Ved å inkludere avstands- eller vinklestimater, kan man forbedre posisjoneringssystemets presisjon betydelig. Niculescu og Nath [12] baserer systemet på vinklestimater ved bruk av spesielle direktive antenner. Dette gir relativt nøyaktig posisjonering og kan også gi gode resultater i spredte nettverk. En annen tilnærming, som også er basert på spesialisert hardware, er å benytte ultralyd til å bestemme avstand mellom sensornodene [13]. Vinklestimater og ultralyd kan begge gi meget god posisjoneringspresisjon, men det gir også økt kompleksitet.

Arbeidene [14, 15] foreslår å bruke mobile noder til å hjelpe lokalisering av sensorene. Sensorposisjonene kan med dette prinsippet fastslås relativt nøyaktig ved at en mobil plattform som kan lokaliseres nøyaktig, beveges rundt i sensornettverket og kommuniserer med det. En slik mobil plattform kan være et kjøretøy, en soldat eller en autonom plattform. Slike metoder kan gi både en effektiv og rimelig løsning på posisjoneringen. Imidlertid er ikke løsningen like aktuell for alle typer sensornettverk. Presisjonen vil også avhenge av bevegelsesmønsteret til den mobile noden.

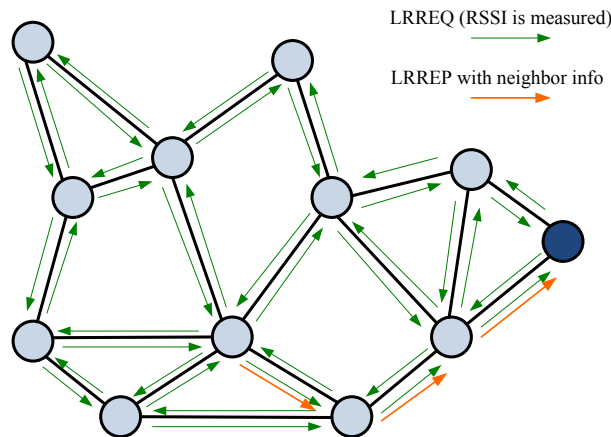
Tam et al. [16] estimerer nodeposisjoner i sensornettverk ved hjelp av en totrinns prosess. Først estimeres posisjonene ved hjelp av triangulering. Deretter benyttes en genetisk algoritme for å avgrense resultatet.

Genetisk algoritme

En genetisk algoritme er en søkemetode som etterligner prosessen ved naturlig evolusjon. Genetiske algoritmer betegnes også som *evolusjonære* algoritmer og genererer løsninger på optimaliseringsproblemer ved hjelp av teknikker inspirert av naturlig evolusjon, slik som arv, mutasjon, utvalg, og rekombinasjon. Genetiske algoritmer genererer et bredt sett av løsninger og gjør deretter mutasjoner og rekombinasjoner av de beste løsningene for å igjen generere nye generasjoner av løsninger. Denne prosessen gjøres gjentatte ganger inntil algoritmen terminerer med et resultat.

2.3 Samtidig ruting og posisjonering

Det fremgår av tidligere arbeider at det er mange mulige metoder som kan benyttes for å finne posisjonene til sensornodene. Av den relaterte forskningen som nevnt over, er vårt bidrag mest knyttet opp til arbeidet i [16]. Både deres og vårt forslag benytter en evolusjonær algoritme sentralt til å utlede nodes posisjoner. Det er imidlertid to sentrale bidrag i vårt arbeid som skiller det fra tidligere presentert forskning i denne kategorien:



Figur 2.1 Initialisering av et sensornettverk. En «sink» (til høyre i figuren) initialiserer sensornettverket ved å sende en LRREQ ut i nettverket som alle nodene svarer på i form av en LRREP. Lokalisering av nettverket foretas samtidig.

- Vår protokoll er holistisk og inkluderer hele posisjoneringssystemet, inkludert estimering av avstandsestimater, datainnsamling og selve lokaliseringsalgoritmen.
- Lokaliseringsalgoritmen benytter en evolusjonær prosess som kun behøver informasjon om avstandsestimater og posisjonen til tre ankeroder.

Den følgende beskrivelsen av posisjoneringssystemet er delt i to. Først beskrives ruting- og innsamlingsprotokollen. Deretter beskrives selve posisjoneringsalgoritmen.

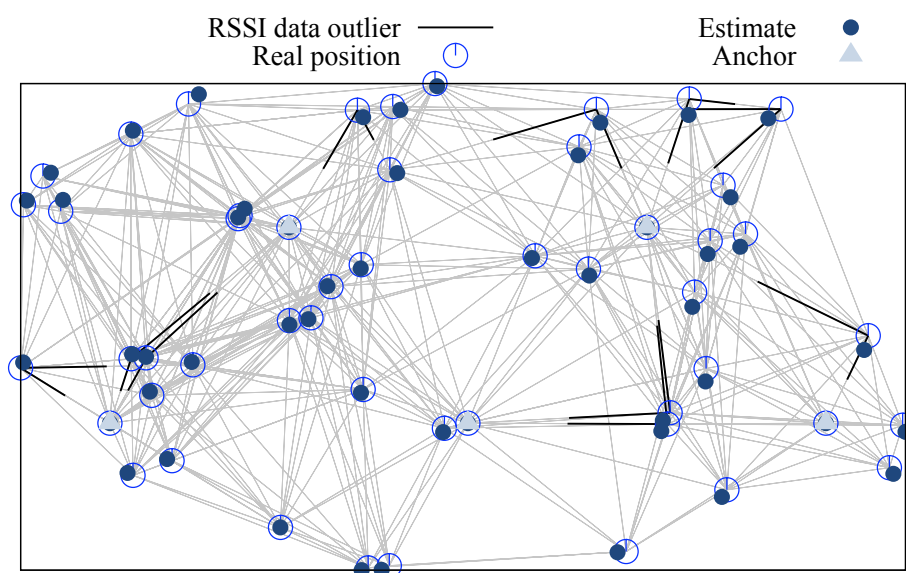
En posisjoneringssalgoritme for et trådløst sensornettverk krever at det hentes ut informasjon fra nettverket, noe som forbruker energi. Mekanismer for å trekke ut og samle inn slik informasjon er imidlertid lite prioritert i forskningslitteraturen, og svært få studier er gjort på hvordan man effektivt kan samle inn den informasjonen som er nødvendig. Som regel antas det at man kan oppnå denne funksjonaliteten ved å benytte en separat protokoll. Vi argumenterer med at man bør unngå å bygge lag på lag med protokoller i et nettverk som allerede har begrenset kapasitet. Fundamentet i vårt arbeid er at alle målinger nødvendig for posisjoneringen, samt innsamling av disse målingene, gjøres samtidig med rutingen. I vår implementasjon utføres dette ved å benytte krysslagskoblinger¹ til å utnytte rutingforespørsel (Route Request) og rutingtilbakemelding (Route Reply) i rutingprotokollen DYMO-low [17].

DYMO-low er beregnet for bruk på IEEE 802.15.4-enheter og er basert på prinsippet om å spre ruteforespørsler (RREQ) i nettverket og deretter sende rutetilbakemeldinger (RREP). Disse teknikkene ble introdusert i AODV [18] og er senere benyttet i DYMO [19]. Vår utvidelse av protokollen introduserer to nye meldinger, Localization Route Request (LRREQ) og Localization Route Reply (LRREP).

¹Krysslagskoblinger beskrives nærmere i faktaboks på side 17

Figur 2.1 viser protokollens hovedfunksjon bestående av en forespørselsfase og en tilbakemeldingsfase. Sinknoden starter forespørselsfasen ved å sende ut en LRREQ. Alle nodene måler signalstyrken (Received Signal Strength Indication (RSSI)) idet LRREQ pakken mottas og lagrer fortløpende disse målingene. Disse målingene benyttes til å gi et grovt distanseestimat. Deretter videresendes LRREQ som en tradisjonell RREQ. I og med at alle nodene videresender LRREQ én gang hver, vil hver enkelt node også motta én LRREQ (og dermed et distanseestimat via RSSI) fra hver av sine naboroder. Hver node vil så rapportere sine målinger som vedlegg i en LRREP melding som sendes tilbake til sink. LRREP meldingen går korteste vei tilbake, og oppretter samtidig rutingtreet med rot i sinknoden. Selve rutingprosessen har likheter med prinsippet bak Collection Tree Protocol (CTP) [20]. Imidlertid har vår protokoll to fordeler sammenliknet med denne. Den mest åpenbare er at vår protokoll utfører måling og innsamling av distanseestimer samtidig som rutingtreet opprettes. Derntest, mens CTP kun tillater ruter fra sensornoder til sink, muliggjør DYMO-low også ruting mellom to vilkårlige noder i nettverket (punkt-til-punkt ruting).

Selve posisjoneringsalgoritmen kjøres på en datamaskin som er koblet til sinknoden. Her benyttes en genetisk algoritme til å anslå nodeposisjoner basert på avstandsestimer fra signalstyrkemålinger. Algoritmen fungerer ved at den først foreslår en populasjon av tilfeldige sett av nodeposisjoner. Den evolusjonære utviklingen av populasjonen foregår ved å sammenlikne avstandene mellom de foreslåtte nodeposisjonene i hvert sett med målte (observerte) avstander. For hvert sett gis en poengsum (Fitness) basert på hvor godt forslaget passer med de målte data. Alle settene i populasjonen sorteres deretter etter denne poengsummen. Sannsynligheten for at et gitt sett bidrar til å generere neste generasjon samsvarer med den poengsummen som er gitt. På denne måten vil gode egenskaper bringes videre til neste generasjon, mens dårlige egenskaper undertrykkes. En ny generasjon skapes ved å kombinere innholdet i de beste settene på ulike måter.



Figur 2.2 Resultat fra eksperiment med evolusjonær lokalisering. Eksperimentet består av femti sensornoder med ukjent posisjon og fem ankeroder med kjent posisjon.

En forutsetning for å kunne angi posisjonene i et kjent koordinatsystem er at det eksisterer minst tre noder i nettverket som har kjente posisjoner. Disse nodene kalles gjerne ankerknoder. Ankerknodenes posisjoner kan gis av GPS eller være angitt manuelt. Ved hjelp av kjente ankerposisjoner, vil den evolusjonære algoritmen kunne konvergere mot et sett med nodeposisjoner som kan georefereres.

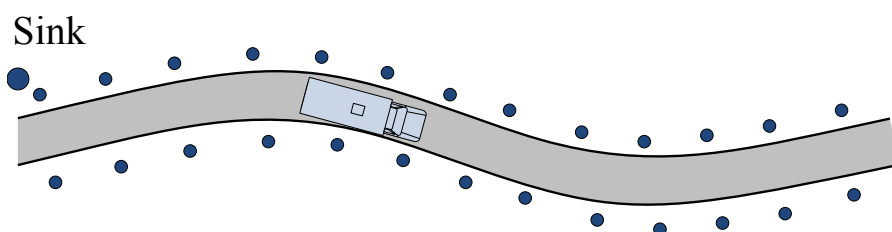
Resultatene fra simuleringer med metoden viser at nøyaktig posisjonering er mulig selv med 10-50 % målefeil i distanseestimatene (se Figur 2.2). Ytelsen reduseres imidlertid betraktelig dersom radiorekkevidden er kort og den avhenger også av den geografiske fordelingen til ankerknodene.

Våre forslag til endringer i DYMO-low kan redusere antall rutingmeldinger med en faktor på n sammenlignet med standard DYMO-low (der n representerer antall noder i nettverket). Det er verdt å merke seg at en tilsvarende optimalisering nylig er blitt foreslått for DYMO-low sin arvtager LOAD-NG av Yi et al. [21]. Vårt foreslåtte system for ruting og posisjonering er fleksibelt; det betyr at lokaliseringsalgoritmen kan byttes ut med en annen sentralisert lokaliseringsalgoritme. Motsatt kan også den evolusjonære algoritmen dra nytte av informasjon som er gitt av en annen type rutingprotokoll, som for eksempel Objective Link State Routing Protocol (OLSR) [22]. Avslutningsvis er det verdt å nevne at den evolusjonære algoritmen med fordel kan dra nytte av mer presise avstandsestimater en hva som er mulig med RSSI.

3 Ruting og clustering

3.1 Motivasjon

Ved etablering av et sensornettverk for militær overvåkning vil det være hensiktsmessig dersom flere sensorer er i stand til å detektere samme hendelse til samme tid. Dette gir for det første økt redundans, siden en feil eller svikt i en sensornode oppveies ved at andre sensorer er operative. For det andre vil det også minske sannsynligheten for falske alarmer, siden man da kan filtrere bort alarmer som genereres av én enkeltstående sensornode. Konvensjonell ruting behandler imidlertid alle alarmmeldinger som sendes i nettverket individuelt og ignorerer mulig korrelasjon i meldingene. Dette er en dårlig utnyttelse av den knappe energien og den begrensede transmisjonskapasiteten som er tilgjengelig.



Figur 3.1 Prinsippskisse for sensornettverk til overvåkning av en veistrekning. Sensorer er plassert på begge sider av veien og rapporterer om aktivitet via et multihopp-nettverk tilbake til sink.

Ved å benytte dataaggregering kan spesielle aggregeringsnoder vente på flere rapporter, enten fra samme node (temporal redundans), eller fra nærliggende noder (romlig redundans), før hendelsen samles i en rapport og sendes til sinknoden. En slik strategi kan føre til en vesentlig reduksjon av datatrafikken. I tillegg er det også mulig å la aggregeringsnoden filtrere sensordata slik at man kan redusere sannsynligheten for falske alarmer. Merk at alle sensortyper har en viss grad av unøyaktighet, og derfor har en sannsynlighet for å produsere feilaktige målinger. Effekten av aggregering og filtrering vil derfor avhenge av hvilke typer sensorer som benyttes.

I dette kapitlet fokuserer vi primært på dataaggregering i sensornettverk for overvåkning. Vi ser spesielt på aggregering i clustre (virtuelle grupper eller klynger) og adresserer følgende problemstillinger:

1. Kan nettverks-clustering kombinert med dataaggregering øke levetiden til nettverket?
2. Bør clustrene utledes distribuert eller sentralt?
3. Er det mulig å benytte informasjon fra rutingprotokollen til å konstruere clustre?

3.2 Bakgrunn

Ulike dataaggregeringsalternativer kan kategoriseres basert på hvilken nettverksstruktur som benyttes i aggregeringen. Nodene kan struktureres virtuelt enten som en kjede, et tre eller i klynger. Ved kjedeaggregering, som i PEGASIS [23], opprettes en lineær kjede for aggregering. Hver node i kjeden overfører data bare til sin nærmeste nabo, som kombinerer disse dataene med egne målinger, og videresender dette langs kjeden. I Directed Diffusion [24] organiseres nodene i en trestruktur med rot i sinknoden. En organisering i en trestruktur gir mer effektiv ruting og gir som regel best ytelse og energieffektivitet av de to strategiene. Ulempen med begge alternativene er at aggregeringen vil gi en forsinkelse siden hver node må vente på andre noders rapporter før den sender sin egen informasjon. Aggregeringsforsinkelsen i hver node avhenger av dens posisjon i aggregeringstreet (eller kjeden).

Den generelle aggregeringsforsinkelsen øker gjerne med antall noder i nettverket [25]. Utfordringen i et aggregeringssystem er derfor å balansere god energieffektivitet med lav forsinkelse. Både trebaserte og kjedebaserte strukturer er best egnet for applikasjoner der alle nodene i nettverket produserer aggregerbar informasjon *regelmessig*. I et tre-, eller en kjedestruktur er det ingen garanti for at noder som er fysisk nær hverandre også er nær hverandre i aggregeringstopologien. Den lange aggregeringsforsinkelsen kan gjøre det vanskelig å identifisere unike hendelser i nettverket. Å spore et bestemt kjøretøy blant flere kjøretøy i et veiscenarie kan for eksempel bli umulig. Trestrukturaggregering er forøvrig benyttet i kapittel 5 og er ytterligere forklart i den tilhørende artikkelen [4].

Klynger (eller clustre) baserer seg på å opprette virtuelle grupper av noder som har felles egenskaper, f.eks. geografisk plassering. LEACH [26] var et tidlig forslag på et slikt cluster-basert sensornettverk. Her ble sensornodene organisert i virtuelle grupper og aggregering ble gjort på dedikerte clusternoder (clusterheads). Denne tilnærmingen fører til en vesentlig reduksjon av aggregeringsforsinkelsen sammenliknet med kjede- og trestrukturer siden det er færre noder i nettverket som er involvert i å aggregere (og dermed forsinke) data. Å benytte clustre er derfor ofte et bedre alternativ i et

overvåkningsscenarie hvor forsinkelse er viktig. Begrensningen i LEACH er at den kun støtter ett rutinghopp fra hver clusterhead til sink, men den har senere blitt utvidet til multihopp av Lai et al. [27]. Gong et al. [28] tar en annen tilnærming og foreslår å bruke modifisert K-means clustering, og utlede clustre sentralt ved hjelp av geografisk informasjon om sensornodene.

3.3 Edge-Betweenness clustering i sensornettverk

K-means clustering, som presentert av Gong et al. [28], har to begrensninger som reduserer den praktiske anvendbarheten til systemet. For det første, siden algoritmen baseres på kjennskap til alle noders geografiske posisjon så krever algoritmen at et posisjoneringssystem eksisterer i nettverket. Dette kan ikke alltid garanteres, og som beskrevet i kapittel 2, er det ikke trivielt å oppnå tilfredsstillende presisjon i posisjoneringssystemet. For det andre antar K-means implisitt at geografisk tilstøtende noder også kan kommunisere med hverandre over ett hopp. Denne forutsetningen kan sjelden oppfylles siden bygninger og vegetasjon kan forhindre kommunikasjon selv mellom noder som er geografisk svært nær hverandre. Dette gjør at algoritmen underestimerer avstanden (i kommunikasjonshopp) mellom enkelte noder, noe som igjen kan føre til suboptimale clustre.

Vårt forslag adresserer disse to begrensningene ved å benytte en clusteralgoritme (EB-clustering) som ikke tar hensyn til nodenes geografiske posisjoner, men som i stedet benytter informasjon om nettverkets topologi. Denne informasjonen hentes via krysslagsmekanismer. «Edge-betweenness community detection» (EB-clustering) er opprinnelig en metode foreslått av Newman og Girvan [29]. Algoritmen er kjent fra fysikk-literaturen og har blant annet blitt brukt til å finne interaksjoner i sosiale nettverk og i mobile ad-hoc-nettverk [30].

EB-clustering søker å finne regioner i nettverket med tette forbindelser. Dette gjøres ved å beregne «betweenness», som er et mål på hvilke linker i nettverket som er de mest sentrale. Den mest sentrale linken i ett nettverk, med andre ord den linken med høyest betweenness-verdi, er den linken som vil bli benyttet flest ganger dersom alle nodene skal snakke med alle over korteste sti. Ved å finne disse linkene, og iterativt fjerne dem, finner man implisitt også de fremtredende clustrene i nettverket. I motsetning til de fleste eksisterende clusteralgoritmer legger ikke EB-clustering noen a priori begrensninger for clusterstrukturene (f.eks. clusterdiameter, antall noder i en cluster eller antall clustre). Forutsetningen for at EB-clusteringalgoritmen skal fungere godt er oppdatert informasjon om nettverkstopologien. Vi foreslår derfor tre forskjellige metoder for å fremskaffe topologiinformasjonen. Alle metoder utnytter topologikunnskap som innhentes ved hjelp av krysslagskoblinger fra rutingprotokollen:

1. Distribuert clustering ved hjelp av OLSR.
2. Sentralisert clustering ved hjelp av OLSR.
3. Sentralisert clustering ved hjelp av DYMO-low.

Distribuert clustering gir den unike fordelingen at alle noder selv kan fastslå sitt eget clustermedlemskap

uten å kommunisere eksplisitte meldinger til andre noder. Siden det ikke kommuniseres noen eksplisitte meldinger mellom nodene kan fremgangsmåten også beskrives som «passiv clustering». OLSR er en link-state protokoll og hver node i nettverket har derfor en noenlunde oppdatert informasjon om nettverkstopologien². Denne informasjonen kan brukes direkte i EB-Clustering for å utlede clustrene. Men for å sikre full konsistens i clusterbestemmelsen, det vil si at alle nodene utleder nøyaktig de samme clustrene, må også topologiinformasjonen være nøyaktig, og identisk for alle nodene. Dersom man benytter standard innstillinger for OLSR, kan man bare oppnå delvis topologioversikt, og det er da ingen garanti for at alle nodene utleder identiske clustre. I våre eksperimenter utvidet vi topologikunnskapen til OLSR ved å benytte mekanismer fra [31] og [32]. Det ble da mulig å oppnå konsistent clustergenerering på en fullstendig distribuert måte.



Figur 3.2 Akseovervåking av en 190m lang veiakse fra [33]. 40 sensornoder med radar, IR og akustisk sensor utplassert.

Selv om et distribuert design er ønskelig i de fleste trådløse nettverk av pålitelighetshensyn, har de fleste sensornettverk allerede en sentral enhet, nemlig sinken. Denne sinknoden utgjør allerede et «single-point-of failure». Av denne grunn reduseres ikke påliteligheten til systemet nevneverdig ved å gjøre enkelte protokollfunksjoner sentralt. Vi har derfor studert clustergenerering sentralt ved bruk av både OLSR og DYMO-low. Vi observerte at sentral clustergenerering med OLSR er mulig selv med den delvise topologioversikten som standard OLSR gir. Det tredje alternativet vi har studert er å generere clustre ved hjelp av den reaktive protokollen DYMO-low. Vi har konstruert en enkel protokollutvidelse som gjør det mulig å samle inn informasjon fra alle linker i nettverket samtidig som rutingen pågår. Dette gjør det mulig for sinknoden å ha fullstendig topologioversikt og dermed kunne bruke EB-clustering til å generere de (topologimessig) mest relevante clustrene. Ved å bruke DYMO-low, senkes overheaden betydelig sammenliknet med OLSR.

²Merk at OLSR sjelden er benyttet som rutingprotokoll i sensornettverk. Uten modifikasjoner har den for høye ressurskrav til å være effektiv på små sensornoder. Det finnes imidlertid klasser av sensornettverk der OLSR vil kunne være riktig valg, for eksempel dersom det kreves interoperabilitet med mobile noder

Systemet vårt kan benyttes for mange scenarier, men vi har valgt å fokusere på veiovervåkning/akseovervåkning (se Figur 3.2 og rapporten [33] for et eksempel på mulig bruk av systemet). I slike sensornettverk, vil det være flere sensornoder som observerer en og samme hendelse (gjerne et kjøretøy) samtidig. Dataaggregering i clustre vil kunne forbedre falsk-alarmer og øke kapasiteten i nettverket. Vi har simulert et nettverk på 200 sensornoder langs en vei og sammenliknet EB-clustering med K-means [28]. Resultatene viser at vår protokoll (som ikke behøver geografisk informasjon om alle sensornoder) alltid gir like god eller bedre ytelse enn K-means (som behøver geografisk informasjon). Sammenliknet med et vanlig nettverk (uten clustering og dataaggregering) oppnår vårt system en energibesparelse på opp mot 90%³.

Oppsummert så skiller vår implementasjon seg fra de relaterte tidligere arbeider på de følgende punkter:

1. Mens arbeidene [26–28] bruker eksplisitte kontrollmeldinger til å opprette clustre, har vårt system evnen til å passivt utnytte den underliggende rutingprotokollen for å få topologikunnskap.
2. Vårt system gjør at forskjellige trafikk-klasser kan ta ulike veier mot sinken. Trafikk som ikke skal aggregeres (for eksempel bilder) kan ta korteste vei mot sink, mens trafikk som skal aggregeres (alarmer og målinger) vil bli rutet via en aggregeringsnode (som kan resultere i noe lengre vei).

Krysslagskoblinger (Cross-layer)

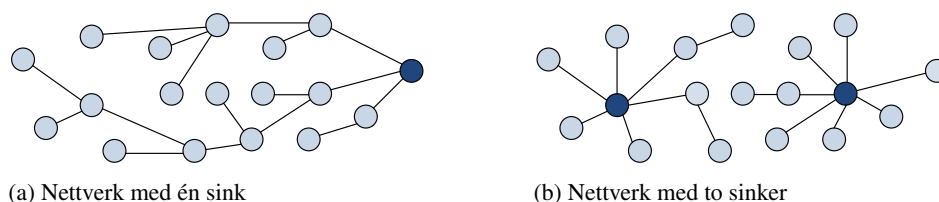
I kommunikasjonssystemer deles implementasjonen av kommunikasjonskomponentene opp i flere lag. Resultatet av en slik oppdeling er at hver funksjon (f.eks. adressering, transport) blir håndterbar. Ved at det benyttes definerte grensesnitt mellom lagene kan man gjøre forandringer i ett lag uten at de andre lagene påvirkes. Selv i mikrokontrollerbaserte sensorsystemer er det vanlig å dele kommunikasjonssystemet opp i fysisk lag, nettverkslag, transportlag og applikasjonslag. Dette gjør systemet mer generisk, fleksibelt og lettere å vedlikeholde. «Krysslagskoblinger» innebærer at det lages nye grensesnitt mellom lagene, at man slår sammen flere lag, eller at man skaper andre avhengigheter mellom lagene. Det er ofte fristende å implementere slike koblinger siden det kan gi store ytelsesforbedringer i form av mer effektiv utnyttelse av radiokapasitet eller minne samt øke batterilevetiden til sensorsystemet. Imidlertid kan disse krysslagskoblingene gjøre systemet vanskelig å utvide i fremtiden. Det kan også resultere i en uoversiktlig programkode og dermed øke sannsynligheten for feil. Krysslagskoblinger bør derfor gjøres med en viss varsomhet, og er derfor mest aktuelle i systemer hvor ytelseskravene er svært høye.

³Det er verdt å merke seg at disse tallene kun gjelder simuleringer og det er en rekke faktorer i et implementert nettverk som vil kunne påvirke resultatet, som sensortyper, datatrafikkmengde, størrelse på nettverket, parameterinnstillinger på radio osv.

4 Sinkplasseringen og sinkantallets innvirkning på ruting

4.1 Motivasjon

Ruting i trådløse sensornettverk utføres gjerne av en datainnsamlingsprotokoll (collection protocol) som bygger et rutingtre med rot i én sink (som vist i Figur 1.2). I store nettverk kan både nettverkets levetid og skalerbarhet økes ved å utplassere flere sinker. I tillegg til å redusere gjennomsnittlig antall hopp mellom en node og en sink, forbedres også fordelingen av både energiforbruket og transmisjonsforbruket i nettverket, se Figur 4.1. Dette gjør at kapasiteten økes i nettverket. Bruk av multiple sinker gir også redundans dersom en av sinkene feiler, blir vandalisert eller stjålet, eller går tom for energi. Enten man benytter en eller flere sinker, vil den plassering man velger for sinkene avgjøre effektiviteten til rutingsystemet og hvilken levetid som kan forventes. I dette kapitlet fokuserer vi på løsninger for å finne optimale plasseringer for et gitt antall sinker.



Figur 4.1 Bruk av flere sinker i nettverket gir lavere gjennomsnittlig antall hopp enn om det benyttes kun én sink. Energieffektiviteten blir forbedret og kapasiteten kan økes.

4.2 Bakgrunn

Oyman et al. [34] foreslår å finne optimal plassering av flere sinker ved hjelp av en K-means clusteralgoritme. Den optimale plasseringen for k sinker er gitt av centroidene til de k clustrene som produseres av algoritmen. Begrensningen i denne algoritmen ligger i at alle sensornodenes posisjon må være kjent. Vincze et al. [35] letter på dette kravet ved at algoritmen kan benytte approksimerte posisjoner til noder som har ukjent plassering. Begge disse metodene studerer imidlertid *ubegrenset* sinkplassering. Det vil si at de tillater at de k sinkene kan plasseres hvor som helst. I praksis betyr dette at de forslagene som algoritmene produserer kun fungerer i praktisk bruk dersom det ikke er noen fysiske begrensninger som forhindrer sinkplassering. Dette kan sjelden garanteres, og de plasseringene som algoritmene foreslår (selv om de er optimale i teorien) kan vise seg å være på toppen av bygninger eller trær, eller gjøre at sinknoden får dårlig eller ingen konektivitet med det omkringliggende sensornettet. Strengt tatt er metodene [34, 35] derfor ikke praktisk gjennomførbare.

Dai et al. [36] har som mål å løse dette problemet ved å foreslå sinkposisjoner kun på steder som er man vet er innenfor rekkevidde med resten av sensornettverket. For å oppnå dette, begrenses sinkplassering bare til lokasjoner som er okkupert av sensornoder som allerede har god kommunikasjon med nettet. Utplassering foregår dermed ut ved å bytte ut sensornoder som ligger på gode lokasjoner med sensornoder som også kan opptre som sinker. Forslaget er derimot begrenset til enkle nettverk med ensartede link-lengder og link-vekter og er av denne grunn ikke spesielt nyttig for nettverk med andre

(og mer realistiske) topologier. Forslagene som er presentert i [37] og [38] kan betraktes som mer aktuelle i praksis. Ved å bruke noen begrensninger som input til algoritmen kan man her til en viss grad kontrollere utfallet av algoritmen; nærmere bestemt hvor sinkene skal plasseres. Ulempen er at man må ha stor kunnskap både om nettverket og topologien (for eksempel ved hjelp av detaljerte kart) for å kunne mate algoritmen med nødvendig informasjon.

I neste avsnitt beskriver vi vårt forslag til algoritmer for sinkplassering. Vårt forslag er ment å adressere de utfordringene som er nevnt over, ved at den ikke er begrenset til noen spesielle nettverkstopologier. I tillegg er ikke eventuelle topologibegrensninger inndata til algoritmen; den lærer disse selv ved å hente informasjon fra rutingprotokollen.

4.3 Sinkplassering

For å effektivt kunne bestemme optimal plassering for en eller flere sinker, må man enten innsamle eller estimere nettverksinformasjon. De ulike metodene for sinkposisjonering kan kategoriseres basert på hvilken type informasjon de behøver, som kan være enten:

1. kunnskap om den geografiske posisjonen til alle sensornoder (geo-metoder), eller
2. kunnskap om nettverkstopologien (topologi-metoder).

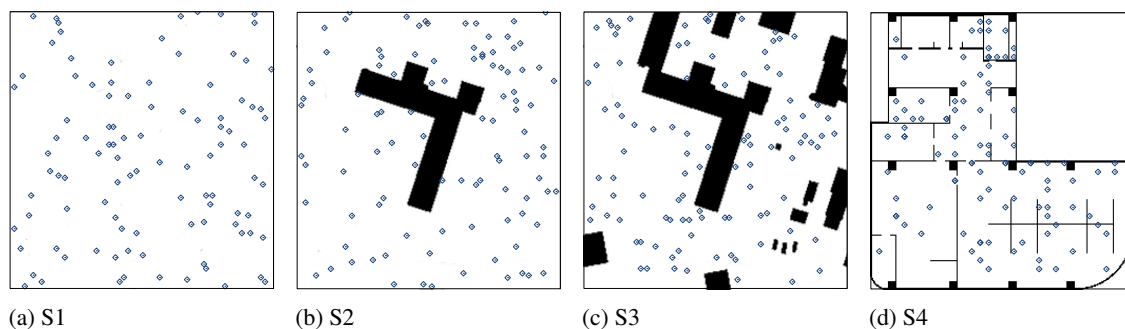
Vi har undersøkt fire ulike sinkposisjoneringsalgoritmer, to i hver av disse kategoriene. Den første metoden er basert på K-means (KSP) og er identisk med metoden presentert i [34]. Denne tjener dermed som en basis i sammenlikningen. De tre andre metodene har vi utviklet selv. Utover K-means har vi studert en geo-metode som er basert på K-medoid clustering [39] (KDP). K-medoids bygger som navnet tilsier på *medoider* i stedet for cluster-tyngdepunkt (*means*). En medoid defineres her som det mest sentrale objektet i et cluster. Ved å benytte denne metoden for sink-plassering vil utfallet av algoritmen (k noder), være de k nodene som er de mest sentrale i nettverket. Å plassere sinknodene på disse nodeposisjonene vil gi lavest gjennomsnittlig Euklidsk avstand til sink(ene). Ved å begrense algoritmen til å kun foreslå kjente sink-lokasjoner (det vil si lokasjoner som allerede er benyttet av sensornoder og som har god konnektivitet) kan en nettverksoperatør være sikker på at sinker plassert i disse lokasjonene også vil ha tilfredsstillende konnektivitet.

Alle geo-metoder i litteraturen (inkludert de to nevnt over) lider imidlertid av følgende svakheter:

1. De geografiske posisjonene til sensornodene må være kjent. For å beregne og innhente nodeposisjoner må en lokalisering- og innsamlingsprotokoll være tilstede i nettverket.
2. Metodene er basert på Euklidsk avstand. Det vil si at algoritmene antar at alle sensornoder har identisk radiorekkevidde og at geografisk nærliggende noder kan nå hverandre med ett nettverkshopp. Dette er imidlertid ikke alltid mulig i et deployert miljø.

For å overvinne disse to begrensningene har vi foreslått to topologi-baserte sinkplasseringsmetoder. Disse benytter informasjon om nettverkstopologien i stedet for å basere seg på noderes geografiske

posisjoner. Den første metoden lager en konnektivitetsmatrise basert på linkinformasjonen og benytter deretter Dijkstra's algoritme [40] for å beregne antall hopp mellom alle nodene i nettverket i motsetning til kortest avstand i meter. Ved å erstatte Euklidisk avstand med hopp-avstand i K-medoid algoritmen, vil utfallet fra algoritmen være de k nodene i nettverket som i gjennomsnitt gir færrest antall hopp i nettverket. Metoden kalles Shortest Path Positioning (SPP). Vi har også foreslått en utvidelse av denne algoritmen som i tillegg til link-informasjon benytter informasjon om linkkvaliteter i beregningen. Denne metoden kalles Routing Metric Placement (RMP).

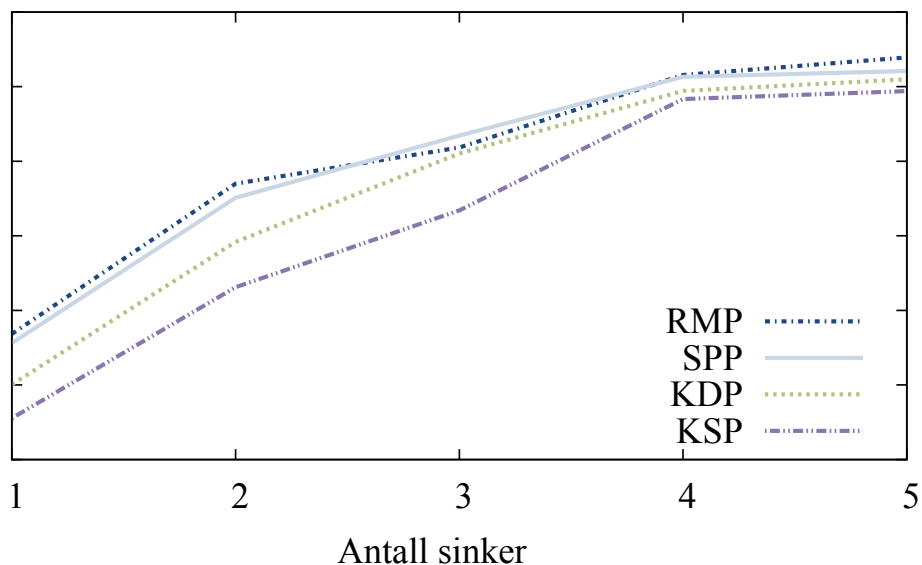


Figur 4.2 De fire scenarioene som er benyttet i simuleringene: (S1) Åpent terreng, (S2) en bygning, (S3) flere bygninger, (S4) innendørs.

I tillegg til de fire sinkplasseringsstrategiene som nevnt over, implementerte vi en enkel og naiv sinkplasseringsmekanisme som legger sinken i midten av nettverket for å simulere en nettverksoperatørs intuisjon. Uttestingen er gjort med simuleringer. For å sikre at resultatene ikke er påvirket av et bestemt nettverksoppsett har vi testet fire ulike scenarioer, se Figur 4.2. Det første scenarioet (S1) representerer et åpent område uten hindringer. Det andre scenarioet (S2) er det samme området, men med en stor bygning i midten. På det tredje scenarioet (S3) har vi lagt inn flere bygninger. Det fjerde scenarioet (S4) er et innendørs scenario som representerer et kontorlokale. Simuleringsverktøyet (ns2) inkluderer bygninger og andre hindringer når linkkvalitet beregnes.

Resultatene viser at miljøet som sinkplasseringsstrategiene testes i har en stor innvirkning på ytelsen. Selv den aller enkleste metoden vi benyttet, som legger ut en sink midt i nettverket, fungerer godt under ideelle forhold slik som i S1. Imidlertid gir de enkle strategiene dårlige resultater i et område som har hindringer for radiosignalene. I komplekse miljøer (S3-S4) er det kun de mest avanserte strategiene som gir gode resultater. Grunnen til dette er at disse strategiene tar hensyn til den faktiske nettverkstopologien mens de enkle strategiene antar (feilaktig) topologiens karakteristikk basert på geografisk informasjon. Figur 4.3 viser relativ levetidsforskjell i nettverk med 1–5 sinker i scenario S3. Våre resultater indikerer at tidlige sinkplasseringsstrategier, uttestet under forenklede simuleringer, ikke nødvendigvis er nyttige i en reell implementasjon.

Merk at selv om våre resultater peker i retning av en sinkplasseringsstrategi som er avansert og benytter både komplett linkinformasjon og linkkvaliteter, har vi identifisert to tilfeller der valget av sinkplasseringsstrategi er irrelevant. Det første tilfellet er i veldig enkle scenarioer som i S1. Slike åpne og homogene områder er imidlertid ikke spesielt sannsynlige annet enn i en simulator.



Figur 4.3 Relativ levetidsforskjell mellom ulike sinkplasseringsstrategier med sinkantall fra en til fem. RMP og SPP baseres på kunnskap om nettverkstopologien (topologi-metoder) mens KDP og KSP baseres på kunnskap om den geografiske posisjonen til alle sensornoder (geo-metoder).

Det andre tilfellet opptrer dersom man har et veldig høyt antall sinker tilgjengelig: Dersom en stor prosentandel av sensornodene også opererer som sinknoder blir deployeringsstrategien etter hvert av liten relevans. Denne tendensen ser vi i Figur 4.3. Siden begge disse ytterpunktene er lite sannsynlige i praktisk bruk, er det vår klare anbefaling at man benytter en sinkplasseringsstrategi som baserer seg på linkinformasjon fra rutingprotokollen. Det å øke sinkantallet kan gi en stor gevinst i levetid og kapasitet, men det er verdt å merke seg at en god sinkplasseringsstrategi kan gi stor effekt også i tilfeller hvor man kun har én sink.

Dataaggregering eller multiple sinker som effektiviseringstiltak?

Dataaggregering kan være en meget effektiv metode for å effektivisere energiforbruket i et trådløst sensornettverk. Det forutsetter imidlertid at nettverket transporterer data som lar seg aggregere. En annen begrensning med dataaggregering er at det er vanskelig å forene kravene om god aggregeringseffektivitet og lav forsinkelse. Bruk av *multiple sinker* kan være en like effektiv og ofte enklere metode. Den er også mer generell, siden gevinsten (lengre levetid) er uavhengig av type data som transporteres i nettverket. Metoden gir også redundans dersom en sinknode skulle feile eller bli utsatt for vandalisme. Multiple sinker krever imidlertid at det etableres et «backbone»-nettverk som knytter sinkene sammen. I en militær sammenheng er redundans viktig, og multiple sinker er derfor foretrukket fremfor dataaggregering. Men metodene kan også kombineres.

5 Samhandling mot eksterne systemer

5.1 Motivasjon

I internasjonale operasjoner vil militære sensornettverk behøve integrasjon med NATOs kommunikasjonsinfrastruktur. Denne infrastrukturen vil kunne bestå av et bredt spekter med systemer som har ulike krav til informasjonen som skal produseres av sensornettverket. Det er derfor viktig at sensornettverket har et lettfattelig og fleksibelt grensesnitt som kan benyttes til å sende spørringer inn i nettverket og til å returnere rapporter ut av nettverket.

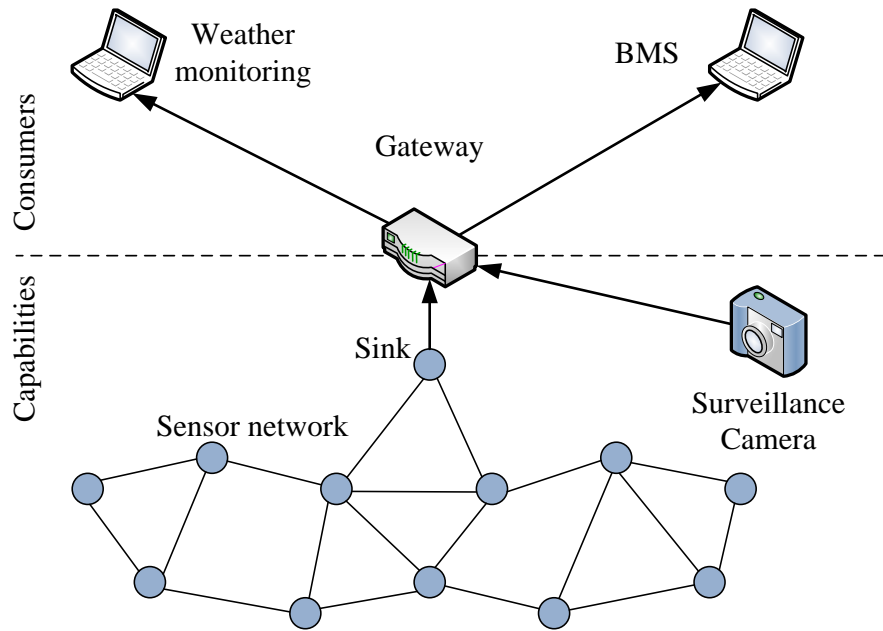
Det å sikre en god energieffektivitet er den sentrale utfordringen i sensornettverk. Algoritmer og protokoller som benyttes i sensornettverket bør derfor tilpasses spørringene som den eksterne infrastrukturen produserer slik at energiforbruket i nettverket til en hver tid minimeres. På denne måten vil sensornettverket kunne utføre de ønskede oppgaver med lavest mulig energiforbruk, og dermed kan lengst mulig levetid oppnås. I dette kapitlet undersøker vi hvordan man kan oppnå interoperabilitet med eksterne systemer uten å øke belastningen på sensornettverket.

5.2 Bakgrunn

Directed Diffusion [24] var en av de første protokollene som kombinerte pakkeruting og spørring i trådløse sensornettverk. I Directed Diffusion er spørringene formatert som interessemeldinger som spres fra sinken og ut til alle sensornoder. Mens disse interessemeldingene spres blir det satt opp ruter fra hver sensornode tilbake til sinken. Directed Diffusion utfører i tillegg aggregering av sensorinformasjon idet informasjonen rutes på vei til sinknoden. Datamodellen er imidlertid basert på at spørringer må gjentas med visse intervall, og er ikke spesielt godt egnet for hendelsesinitierte alarmscenarioer, slik som taktisk overvåkning. Videre er formatet som benyttes i Directed Diffusion proprietært og derfor mindre egnet i en interoperabilitetskontekst.

TinyDB [41] er et system som baseres på prinsipper bak databasebehandling. Men her gjøres spørringer mot sensorsystemet i stedet for mot en database. Systemet, som baseres på SQL-liknende spørringer, muliggjør flere samtidige spørringer fra ulike konsumenter. Men også her, kan selve formatet betraktes som proprietært og komplisert. I alle fall sett opp mot en større infrastruktur.

Som nevnt innledningsvis, er interoperabilitet mot NATOs systemer viktig for militære sensorsystemer. «Web Services» ses på som en nøkkelteknologi for oppnå denne interoperabiliteten i NATO Network Enabled Capability (NNEC). For et sensornettverksgrensesnitt gir Web Services lang større fleksibilitet enn hva som oppnås med Directed Diffusion og TinyDB. Et «Web Services» basert sensornettverk kan oppnås enten ved å la hver enkelt sensornode i nettverket få et «Web Services-grensesnitt», eller ved å designe en gateway mellom sensornettverket og det eksterne nettverket. En slik gateway vil benytte spesielle sensornettverksprotokoller på den interne siden og Web Services standarder på den eksterne siden, se Figur 5.1. Delicato et al. [42] foreslår en løsning som tilhører den første kategorien. I dette systemet integreres full SOAP-støtte på alle sensornodene. Siden SOAP er en relativt ordrik protokoll basert på XML, vil det resultere i en betydelig overhead.

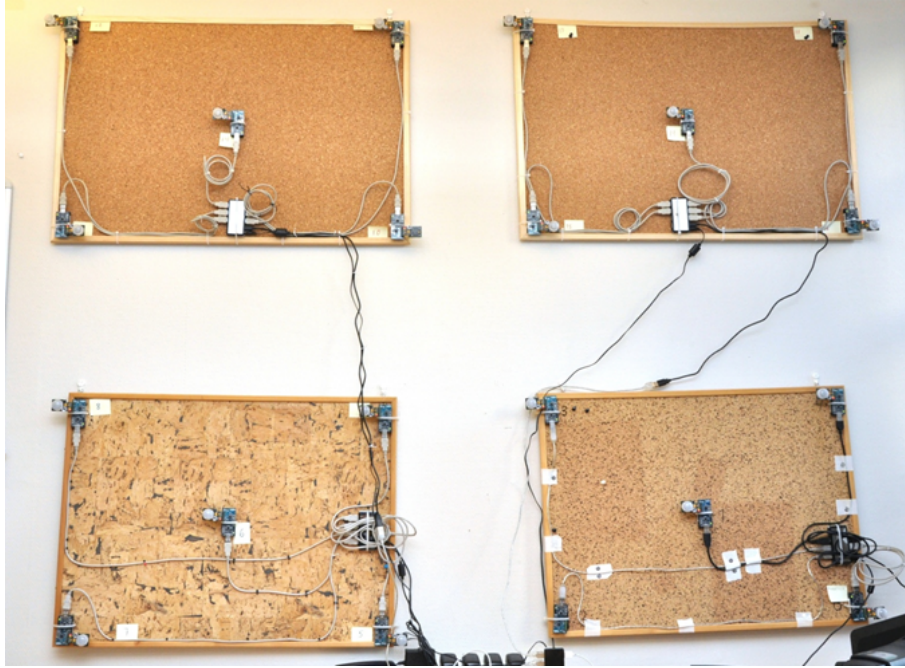


Figur 5.1 Sensornettverket er en tjeneste som tilbyr kapasiteter til ulike eksterne nettverkkomponenter. En gateway kan kombinere tjenester fra sensornettverket med andre tjenester (for eksempel et kamera).

Listing 1 XML spørring som ber om at alarmrapporter sendes til en gitt IP-adresse dersom minst fire IR-sensorer samtidig detekterer et objekt. Det er kun ønskelig at det gis alarmer når det er mørkt (mindre enn 1 lux).

```
<GetIntruder>
  <MinPIRDetections>4</MinPIRDetections>
  <LightMaxThreshold>1lux</LightMaxThreshold>
  <Duration>30d</Duration>
  <IncidentReport>http://10.0.0.2/</IncidentReport>
</GetIntruder>
```

Komprimering vil kunne redusere overheaden med XML, og det er vist at nye standarder som Efficient XML kan muliggjøre XML på taktisk nivå [43]. Tidligere forskning på FFI viser dog at prosessorbelastningen og minneforbruket øker betydelig ved bruk av komprimeringsbiblioteker [44]. Dette gjør komprimering som løsning mindre egnet for små sensornoder. Av denne grunn, har vi, i motsetning til andre alternative implementasjoner, for eksempel [45], valgt å ikke benytte XML-kompresjon i vår implementasjon. I stedet foreslår vi å benytte små, korte og proprietære meldinger internt i sensornettverket, og benytte XML ut av sensornettverket. Et liknende oppsett foreslås i WSN-SOA [46]. Her reduseres XML-formatet ned til en størrelse som er håndterbar for 802.15.4-nettverk. Bressan et al. [47] presenterer liknende ideer, men benytter en ny standard «Constrained RESTful Environments (CoRE)» i sensornettverket.



Figur 5.2 Lab med 20 sensornoder

5.3 Bruk av Web Services mot NATO Network Enabled Capability

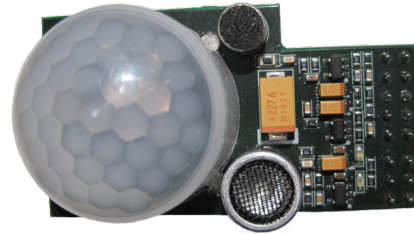
Det er undersøkt hvordan sensornettverket kan være en kapasitet i NNEC ved bruk av «Web Services». Det finnes ulike måter for å realisere en kapasitet som en tjeneste. For eksempel, en tjeneste kan lages fra bunnen av, en tjeneste kan presenteres som en front-end til et eksisterende system, eller tjenesten kan være en kombinasjon av eksisterende tjenester. På grunn av de knappe ressursene i et sensornettverk er det ikke formålstjenlig å tjenesteorientere hver enkelt sensornode. I stedet bruker vi en tilnærming hvor vi «pakker inn» mekanismene i sensornettverket ved hjelp av en gateway. Det eksterne systemet kan dermed bruke Web Services for å kommunisere med sensornettverket, mens internt i sensornettverket brukes eksisterende og enklere mekanismer. På denne måten vil hele nettverket fremstå som en tjeneste.

Vår foreslåtte arkitektur inkluderer dermed en «Web Services wrapper» som gjør at eksterne forbrukere kan samhandle med sensornettverket ved hjelp av XML og webtjenester. Kommunikasjonen foregår i begge retninger. Eksterne systemer formulerer spørringer i XML-format som sendes inn til en gateway (se Listing 1) og videredistribueres i sensornettverket på egne formater. Rapporter som produseres på bakgrunn av disse spørringene vil igjen sendes ut av nettverket og formateres som XML-rapporter i gateway. En slik todeling av strukturen gir en langt mer effektiv bruk av ressursene i sensornettverket enn om hele nettverket skal baseres på de samme protokollene.

Et annet viktig bidrag for å begrense ressursbruken, som også tilbys i vår arkitektur, er at rutingen er adaptiv basert på XML-spørringene i som sendes sensornettverket. Dersom et eksternt system ber om informasjon som lar seg aggregere, vil nettverket kunne aggregere data samtidig som den blir rutet gjennom sensornettverket. Denne type aggregering kan redusere energiforbruket vesentlig.

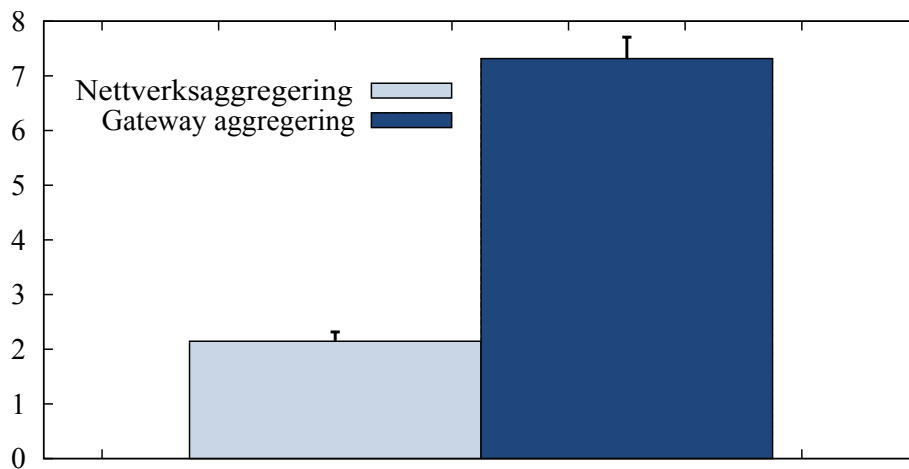


(a) Foto: OpenWSN



(b) Foto: FFI

Figur 5.3 Sensornoden består av (a) TelosB mote, (b) spesiellagd sensor-kort med PIR og mikrofon



Figur 5.4 Meldinger som transporteres i nettverket per sekund for to aggregeringsalternativer: i) Nettverksaggregering, som foregår langs rutingtreet; og ii) aggregering på gateway.

Dersom informasjonen ikke lar seg aggregere blir den sendt direkte til sink/gateway og ut av nettverket. Dersom meldingen ikke aggregeres, reduseres tiden det tar fra meldingen produseres til den når mottaker. For å oppnå denne fleksibiliteten har vi utvidet rutingprotokollen Collection Tree Protocol (CTP) [20] med en enkel aggregeringstjeneste. Merk at i dette konseptet gjøres aggregeringen langs rutingtreet og ikke på definerte noder slik som i Kapittel 3.

Arkitekturen er uttestet i et sensornettverk (Figur 5.2) bestående av 20 TelosB-baserte sensornoder med Passiv Infrarød Sensor (PIR) og mikrofonensorer (Figur 5.3). Vår arkitektur består av en gateway som kommuniserer ved hjelp av SOAP mot et eksternt nettverk mens det benyttes små binære meldinger internt i sensornettverket. Alternativet til dette systemet er å sende SOAP-meldinger direkte til hver sensornode. Disse to alternativene er sammenliknet. Resultatene viser at distribusjonstiden ble redusert med 80% og energibruken med 90% ved å bruke vårt reduserte format internt i nettverket i stedet for å benytte SOAP-meldinger hele veien til sensornoden. Videre ble to ulike tilnærminger for dataaggregering undersøkt. Den ene var å aggregere data langs rutingtreet (nettverksaggregering), mens den andre metoden gikk ut på å aggregere data på et felles punkt idet datameldingene går ut av sensornettverket (gatewayaggregering). Figur 5.4 viser at nettverksaggregering kan spare opp til 80%

av transmisjonene i nettverket⁴.

Konklusjonen fra uttestingen er at en Web Services-basert arkitektur har stor nytteverdi selv på små og begrensede sensornheter så lenge formatene tilpasses det underliggende systemet som skal benyttes. Med dagens enkle sensornoder er det ikke formålstjenlig (dog ikke umulig) å implementere full støtte for Web Services. Et forenklet format gjør det enklere for å utføre både meldingslesing, meldingsopprettelse og aggregering. Vår tilnærming gjør det mulig for sensornettverksdesignere å fokusere på energieffektive løsninger i selve sensornettverket, mens mer ordrike og standardiserte protokoller kan benyttes på utsiden for å sikre et enkelt grensesnitt mot omverden.

6 Hvilken innvirkning har jamming og interferens på ruting?

6.1 Motivasjon

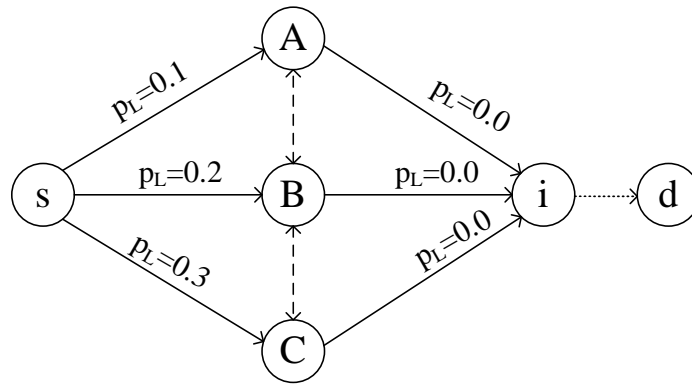
Både radioforstyrrelser fra andre radiosystemer, bevegelser i området eller fiendtlige jamme-angrep kan føre til svært uforutsigbar kommunikasjon i trådløse sensornettverk. Resultater fra deployerte sensornettverk viser at typisk pakkeleveringsrate er på mellom 70 og 99% [20, 48, 49], og den kan være så lav som 20-40% [50–52]. En del av disse dårlige resultatene kan selvsagt skyldes implementeringsfeil, dårlig planlegging eller «uflaks». En sannsynlig årsak er også eksterne påvirkninger (vind, personell i bevegelse) eller interferens. Det er mange tiltak som kan iverksettes for å minske sensornettverkets følsomhet for slike påvirkninger; man kan for eksempel forbedre radiokretsene, bytte til bedre antenner eller bytte til en mer robust mac-protokoll. Dette er imidlertid endringer som er vanskelige å implementere etter at sensorsystemet er konstruert, og det kan i slike tilfeller være hensiktsmessig å forbedre protokoller lengre opp i protokollhierarkiet. I dette kapitlet studerer vi derfor hvordan ulike *rutingprotokoller* oppfører seg når sensornettverket er påvirket av forstyrrelser og vi foreslår mekanismer for å maksimere leveringsraten i slike tilfeller.

6.2 Bakgrunn

Tradisjonelle rutingprotokoller for trådløse sensornettverk forsøker å håndtere dynamikken i den underliggende nettverksstrukturen ved hjelp av ulike metrikker, for eksempel ved å måle antall hopp som i TYMO [53], kvaliteten på radiolinken som i MultihopLQI [54] eller Expected Transmission Count (ETX) som i CTP [20]. De metrikkberegningene som brukes i disse protokollene har imidlertid problemer med å mestre de raske endringene i linkkvalitet som oppstår i et upålitelig trådløst medium. Av denne grunnen kan rutingbeslutningene som tas være basert på historiske og utdaterte beregninger. Denne observasjonen har ført til utviklingen av et noe annerledes rutingparadigme, opportunistisk ruting [55–57]. Opportunistisk ruting skiller seg fra de tradisjonelle rutingteknikkene siden den utnytter, snarere enn å forsøke å skjule, det trådløse mediums broadcastnatur.

Opportunistisk ruting er basert på det faktum at det kan være mange potensielle sensornoder i en nodes

⁴Merk at forsøket er gjort med et relativt lite sensornettverk (20 noder). Den relative forskjellen mellom nettverksaggregering og gatewayaggregering vil øke (i favør av nettverksaggregering) med økende antall noder.



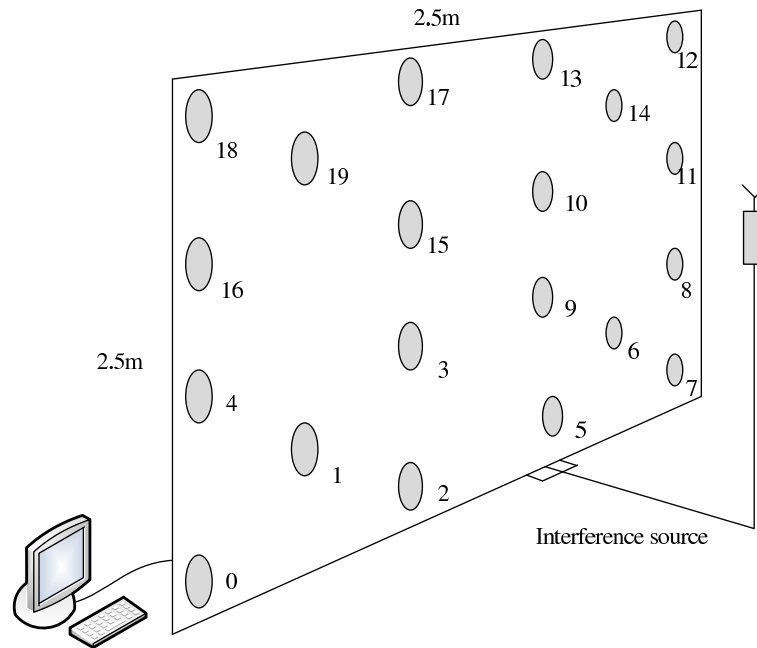
Figur 6.1 *Opportunistisk ruting utnytter det trådløse mediets broadcastnatur. Node s forhåndsvelger ikke en dedikert node for neste-hopp-ruting, men utnytter at det er flere noder i området som kan bidra til videresending av pakken.*

nærhet som er i stand til å motta en datapakke og som kan videresende den. Extremely Opportunistic Routing (ExOR) [55] var en av de første forslagene til denne type ruting. Her velger senderen av en pakke en delgruppe av alle sine nærliggende noder som kan bidra til videreruting. Disse kandidatene listes i pakkeheaderen sortert på avstand fra den endelige destinasjonen (i vårt tilfelle sinknoden). Hver node som mottar denne pakken forsinkes videresending av pakken en tid som er proporsjonal med avstanden til sinknoden. Med andre ord; den noden som er nærmest sinknoden vil viderefremidle pakken først. Noder som har en tilsvarende pakke klar for videresending, men som «hører» at en annen node videresender vil dermed la være å sende sin kopi videre. Opportunistisk ruting har vist seg å være veldig effektivt i upålitelige trådløse nettverk. Ulempen med opportunistisk ruting er imidlertid at dupliserte pakker oppstår i nettverket. Overheaden som følger med opportunistisk ruting er derfor et problem som er vanskelig å håndtere og umulig å eliminere.

6.3 Opportunistic Collection Tree Protocol

Siden lav overhead og høy pakkeleveringsrate er vanskelig å forene er det ikke trivielt å bestemme hva som vil være «best» av tradisjonell ruting og opportunistisk ruting. Hvilken protokollfamilie som gir best resultater avhenger av kvaliteten på radiokanalene i nettverket. La oss se på et konkret, dog forenklet eksempel (Figur 6.1). En pakke som sendes fra node *s* kan ta tre ulike veier på vei til *d*. Pakketapet er målt til å være mellom 10–30%. En tradisjonell rutingprotokoll vil typisk velge ruten via A, og så lenge pakketapet er stabilt, vil det også være det aller beste alternativet, og 10% pakketap kan forventes.

Dersom pakketapet varierer raskere enn hva rutingprotokollen klarer å håndtere, økes sannsynligheten for at suboptimale ruter benyttes. La oss nå se på opportunistisk ruting. Siden både A, B og C vil kunne nås fra *s*, vil opportunistisk ruting «se» et pakketap på 0,6%, som er produktet av de tre linkenes pakketap. Dette er en åpenbar forbedring sammenliknet med en tradisjonell rutingprotokoll. Men bakkdelen her er at siden A og C ikke «hører» hverandre vil de begge viderefremidle pakken til



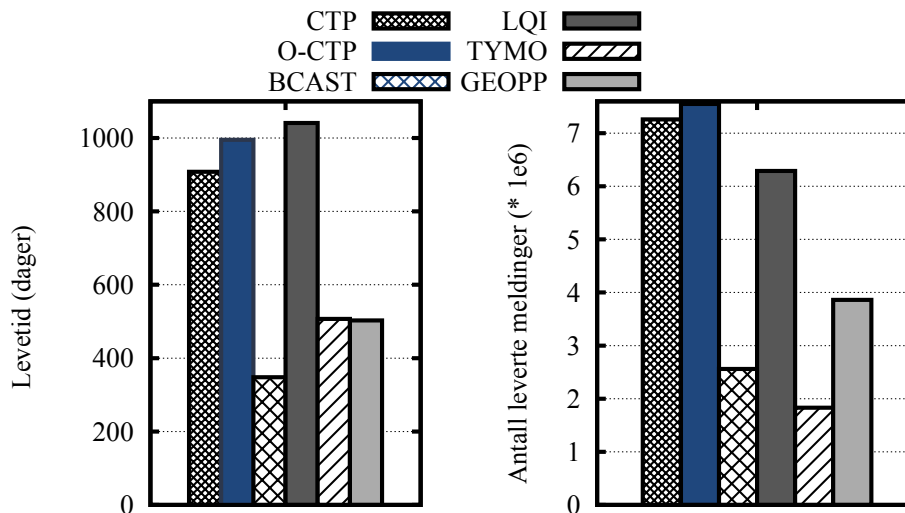
Figur 6.2 Testoppsettet består av 20 TelosB sensornoder og en 2.4GHz softwarekontrollert interferenskilde.

i , noe som dermed resulterer i duplikate pakker. Det er dermed en balanse mellom hvilke av de to protokollene som er mest gunstig. Mengden interferens man forventer vil kunne være avgjørende for hvilken protokoll man bør velge. Av tidligere forskning er det ingen arbeider som har belyst muligheten til å utføre automatisk veksling mellom tradisjonell ruting og opportunistisk ruting basert på de underliggende nettverksegenskaper. I vårt arbeid har vi forsøkt å undersøke fordelene ved å gjøre et automatisk bytte mellom disse to ulike rutingparadigmene.

Vi har konstruert og implementert en ny hybrid opportunistisk protokoll (O-CTP), som bruker tradisjonell ruting basert på CTP [20] når nettverket er stabilt og har lavt pakketap. Dersom nettverket blir utsatt for forstyrrelser eller jamming, bytter protokollen til opportunistisk videresending. Den opportunistiske delen av protokollen bygger på en forenklet versjon av ExOR [55]. Et sett triggere gjør ulike vurderinger av nettverkskvaliteten og lar protokollen bytte til opportunistisk ruting dersom det antas at dette gir høyere pakkeleveringsrate.

Uttestingen av O-CTP ble foretatt ved å konstruere en testbed bestående av 20 TelosB sensornoder (Figur 5.2). En interferenskilde kontrollert fra programvare gjorde det mulig å kontrollere interferensen i fire ulike mønstre (se testoppsett på Figur 6.2). Vi sammenliknet O-CTP med fem andre rutingprotokoller: CTP, Broadcast (BCAST), MultihopLQI, TYMO og en ren opportunistisk rutingprotokoll (GEOPP). Resultatene viste at O-CTP gir den beste balansen mellom pakkeleveringsrate og overhead av de seks protokollene. O-CTP ga for alle interferensnivåene inntil 15% høyere pakkeleveringsrate og kun en tyvendedel av overheaden.

Figur 6.3 viser antatt levetid til et sensornettverk der nodene er utstyrt med to AA-batterier. Nettverket



Figur 6.3 Ytelsen til ulike rutingsprotokoller i et scenario med varierende interferens. O-CTP er den protokollen som leverer flest datapakker i løpet av nettverkets levetid.

eksponeres for varierende interferens. Figuren viser i tillegg antallet datapakker som mottas i løpet av nettverkets levetid. O-CTP gir ca 10% lengre levetid og leverer 10% flere pakker enn CTP. MultihopLQI gir lengst levetid, men leverer omlag 20% færre pakker enn O-CTP.

CTP er bygd for å være helt uavhengig av hvilken radiokrets som benyttes. Vi ønsket ikke å bryte denne uavhengigheten og de triggerne som bytter til opportunistisk ruting er derfor basert på lag-3 funksjoner. Vi tror imidlertid at protokollen kan prestere bedre dersom den utnytter målinger direkte fra det fysiske laget ved hjelp av krysslagskoblinger. Et annet minus med O-CTP er at den i realiteten er bygd på to protokoller. Dette øker kompleksiteten og kodestørrelsen. Lagringsminnet på de TelosB-enhetene som ble benyttet er kun 48KB, og optimalisering av kodestørrelsen er derfor overordnet viktig for alle komponenter som skal benyttes på plattformen.

7 Oppsummering

Denne rapporten er en oppsummering av arbeidet angående ruting i trådløse sensornettverk. Arbeidene er utført i prosjektene 1141 «SASS, Situational Awareness Sensor Systems» og 1196 «Styrkebeskyttelse for landbaserte operasjoner». Vi har studert tre ulike metoder for å øke energieffektiviteten til nettverket. Den første metoden, som er beskrevet i kapittel 3, oppretter clustre (eller virtuelle grupper) i nettverket. Disse clustrene konstrueres basert på informasjon om nettverkstopologien, slik at sensornoder som har god forbindelse med hverandre også blir medlem i den samme clustergruppen. Alle nodene i et slikt cluster aggregerer sensordata i en dedikert node før de samlede data sendes til sinknoden. Denne teknikken gir en flerdobling i nettverkets levetid sammenlignet med å benytte tradisjonell ruting. I kapittel 5 beskriver vi en annen dataaggregeringsmetode. Her blir aggregeringen gjort samtidig som sensormeldingene videresendes langs rutingtreet. Dette er en enklere metode, som ikke behøver administrasjon av clustre. Denne aggregeringsmetoden gir derimot mindre effekt siden rutingtreet stadig endrer seg i takt med variasjoner i det underliggende nettverket.

En tredje metode, som er beskrevet i kapittel 4, er å bruke flere sinknoder i nettverket. Dette gir både høy effektivitet og god redundans. Det er en veldig enkel og høyst aktuell metode dersom scenarioet tillater bruk av multiple sinker. Kombinasjon av disse tre metodene er selvsagt mulig. I overvåkningsøyemed, f.eks. ved beskyttelse av en base over et lengre tidsrom, vil det være fornuftig å benytte flere sinker for å øke redundansen og redusere antall hopp i nettverket. Samtidig kan man gjøre aggregering og filtrering av sensormeldinger i nettverket for å redusere sannsynligheten for falske alarmer og for å øke energieffektiviteten ytterligere.

Tre bidrag i denne rapporten omhandler hvordan krysslagskoblinger mellom rutingprotokollen og andre protokoller kan bidra til å øke ytelsen eller funksjonaliteten til sensornettverket. I kapittel 2 studerte vi en metode for å la rutingprotokollen bidra til lokalisering av sensornodene. Ved å benytte rutingprotokollens topologioversikt sammen med målinger av signalstyrken mellom nodene, viste vi at sensornodenes posisjon kunne bestemmes med tilfredsstillende nøyaktighet. En variant av denne rutingmetodikken ble også benyttet i kapittel 3. I kapittel 5 undersøkte vi hvordan rutingprotokollen kan samhandle med en ekstern protokoll. Spesifikt så vi på hvordan et BMS eller et annet system kunne forespørre sensornettverket om spesiell informasjon og der rutingprotokollen gjør ulike aggregeringsvalg basert på dette. Avslutningsvis studerte vi i kapittel 6 hvordan to rutingprotokoller kunne jobbe sammen for å gi økt ytelse i et sensornettverk utsatt for høyt pakketap og ustabiliteter.

Appendiks A Akronymer og forkortelser

AODV	Ad hoc On-Demand Distance Vector Routing
BCAST	Broadcast Protocol
BMS	Battlefield Management System
CoRE	Constrained RESTful Environments
CTP	Collection Tree Protocol
DYMO	Dynamic MANET On-demand Routing Protocol
DYMO-low	Dynamic MANET On-demand for 6LoWPAN Routing
EB	Edge Betweenness
ETX	Expected Transmission Count
ExOR	Extremely Opportunistic Routing
GEOPP	Geographical Opportunistic Routing
GW	Gateway
GPS	Global Positioning System
LRREP	Localization Route Reply
LRREQ	Localization Route Request
LQI	Link Quality Indication
MANET	Mobile Ad-hoc Network
NNEC	NATO Network Enabled Capability
OLSR	Objective Link State Routing Protocol
PIR	Passiv Infrarød Sensor
REST	Representational State Transfer
RREP	Route Reply
RREQ	Route Request
RSSI	Received Signal Strength Indication
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol

TYMO	DYMO on TinyOS
UNIK	Universitetscenteret på Kjeller
XML	Extensible Markup Language

Referanser

- [1] J. Flathagen and R. Korsnes, "Localization in wireless sensor networks based on Ad hoc routing and evolutionary computation," in *MILITARY COMMUNICATIONS CONFERENCE, 2010-MILCOM 2010*. IEEE, 2010, pp. 1062–1067.
- [2] J. Flathagen, O. V. Drugan, P. E. Engelstad, and Ø. Kure, "Increasing the Lifetime of Roadside Sensor Networks Using Edge-Betweenness Clustering," in *Communications (ICC), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1–6.
- [3] J. Flathagen, Ø. Kure, and P. E. Engelstad, "Constrained-based Multiple Sink Placement for Wireless Sensor Networks," in *Mobile Adhoc and Sensor Systems (MASS), 2011 IEEE 8th International Conference on*. IEEE, 2011, pp. 783–788.
- [4] J. Flathagen and F. T. Johnsen, "Integrating Wireless Sensor Networks in the NATO Network Enabled Capability using Web services," in *MILITARY COMMUNICATIONS CONFERENCE, 2011-MILCOM 2011*. IEEE, 2011, pp. 828–833.
- [5] J. Flathagen, P. E. Engelstad, , and Ø. Kure, "O-CTP: Hybrid Opportunistic Collection Tree Protocol for Wireless Sensor Networks," in *IEEE Conference on Local Computer Networks, LCN*. IEEE, 2012, pp. 947–955.
- [6] J. Flathagen, "A routing and cross-layering approach for energy and bandwidth efficiency in wireless sensor networks," Ph.D. dissertation, NTNU, 2013, ISBN 978-82-471-4659-0.
- [7] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," *Ad Hoc Networks*, vol. 3, no. 3, pp. 325 – 349, 2005. [Online]. Available: <http://www.sciencedirect.com/science/article/B7576-4B3JKH2-3/2/0baf704778ae8cf6e3651eb9f329e4c0>
- [8] S. Chen, G. Fan, and J.-H. Cui, "Avoid 'void' in geographic routing for data aggregation in sensor networks," *Int. J. Ad Hoc Ubiquitous Comput.*, vol. 1, no. 4, pp. 169–178, 2006.
- [9] D. Espes and Z. Mammeri, "Backbone-based location-aided routing algorithm to reduce control packets of AODV protocol," in *Mobility '06: Proceedings of the 3rd international conference on Mobile technology, applications systems*. New York, NY, USA: ACM, 2006, p. 15.
- [10] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, "GHT: a geographic hash table for data-centric storage," in *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. New York, NY, USA: ACM, 2002, pp. 78–87.
- [11] D. A. Tran and T. Nguyen, "Localization In Wireless Sensor Networks Based on Support Vector Machines," *IEEE Trans. Parallel Distrib. Syst.*, vol. 19, no. 7, pp. 981–994, 2008.

- [12] D. Niculescu and B. Nath, “Ad hoc positioning system (APS) using AOA,” *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 3, pp. 1734–1743 vol.3, April 2003.
- [13] L. Girod and D. Estrin, “Robust range estimation using acoustic and multimodal sensing,” *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, vol. 3, pp. 1312–1320 vol.3, August 2002. [Online]. Available: <http://dx.doi.org/10.1109/IROS.2001.977164>
- [14] M. L. Sichitiu and V. Ramadurai, “Localization of wireless sensor networks with a mobile beacon,” *Mobile Ad-hoc and Sensor Systems, 2004 IEEE International Conference on*, pp. 174–183, October 2004.
- [15] S. Lee, E. Kim, C. Kim, and K. Kim, “Localization with a mobile beacon based on geometric constraints in wireless sensor networks,” *Wireless Communications, IEEE Transactions on*, vol. 8, no. 12, pp. 5801–5805, December 2009. [Online]. Available: <http://dx.doi.org/10.1109/TWC.2009.12.090319>
- [16] V. Tam, K. Cheng, and K. Lui, “Using micro-genetic algorithms to improve localization in wireless sensor networks,” *J. of Comm., Academy Publisher*, vol. 1, 2006.
- [17] K. Kim, G. Montenegro, S. Park, I. Chakeres, and C. Perkins, “Dynamic MANET On-demand for 6LoWPAN (DYMO-low) Routing,” Internet Engineering Task Force, Internet-Draft, Jun. 2007, Expired.
- [18] C. Perkins, E. Belding-Royer, and S. Das, “Ad hoc On-Demand Distance Vector (AODV) Routing,” RFC 3561 (Experimental), Jul. 2003. [Online]. Available: <http://www.ietf.org/rfc/rfc3561.txt>
- [19] I. Chakeres and C. Perkins, “Dynamic MANET On-demand (DYMO) Routing,” Internet Engineering Task Force, Internet-Draft draft-ietf-manet-dymo-19, Sep. 2009, work in progress. [Online]. Available: <http://www.ietf.org/internet-drafts/draft-ietf-manet-dymo-19.txt>
- [20] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, “Collection tree protocol,” in *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '09. New York, NY, USA: ACM, 2009, pp. 1–14.
- [21] J. Yi, T. Clausen, and A. de Verdiere, “Efficient Data Acquisition in Sensor Networks: Introducing (the) LOADng Collection Tree Protocol,” in *Proceedings of the 8th IEEE International Conference on Wireless Communications, Networking and Mobile Computing - WiCom-2012*. IEEE, 2012.
- [22] T. Clausen and P. Jacquet, “Optimized Link State Routing Protocol (OLSR),” RFC 3626 (Experimental), Oct. 2003. [Online]. Available: <http://www.ietf.org/rfc/rfc3626.txt>

- [23] S. Lindsey, C. Raghavendra, and K. Sivalingam, "Data gathering algorithms in sensor networks using energy metrics," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 13, no. 9, pp. 924 – 935, sep. 2002.
- [24] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, ser. MobiCom '00. New York, NY, USA: ACM, 2000, pp. 56–67.
- [25] I. Solis and K. Obraczka, "The impact of timing in data aggregation for sensor networks," in *2004 IEEE International Conference on Communications (IEEE Cat. No.04CH37577)*. IEEE, 2004, pp. 3640–3645 Vol.6. [Online]. Available: <http://dx.doi.org/10.1109/ICC.2004.1313222>
- [26] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," in *HICSS '00*, 2000, p. 8020.
- [27] W. K. Lai, C.-S. Shieh, and Y.-T. Lee, "A cluster-based routing protocol for wireless sensor networks with adjustable cluster size," in *ChinaCOM 2009*, Aug 2009, pp. 1 –5.
- [28] Y. Gong, G. Chen, and L. Tan, "A Balanced Serial K-Means Based Clustering Protocol for Wireless Sensor Networks," *WiCOM '08. Proceeding*, pp. 1 –6, oct. 2008.
- [29] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, pp. 026 113+, Feb 2004.
- [30] O. V. Drugan, T. Plagemann, and E. Munthe-Kaas, "Detecting Communities in Sparse MANETs," *IEEE/ACM Transactions on Networking*, Accepted for publication in 2011.
- [31] T. Clausen, P. Jacquet, and L. Viennot, "Investigating the Impact of Partial Topology in Proactive MANET Routing Protocols," in *WPMC*. IEEE, 2002.
- [32] P. E. Villanueva-Pena, T. Kunz, and P. Dhakal, "Extending network knowledge: making OLSR a quality of service conducive protocol," in *IWCMC '06*. New York, NY, USA: ACM, 2006, pp. 103–108.
- [33] J. Flathagen, T. Mjelde, R. Korsnes, V. Pham, J. Sander, "Eksperimenter med trådløst sensor-nettverk for perimetersikring og akseovervåkning," FFI-Rapport-2012/02009, 2012.
- [34] E. I. Oyman and C. Ersoy, "Multiple sink network design problem in large scale wireless sensor networks," in *In: Proc. of 1 st Int. Conf. on Communications (ICC 2004)*, 2004.
- [35] Z. Vincze, R. Vida, and A. Vidacs, "Deploying multiple sinks in multi-hop wireless sensor networks," in *In: Pervasive Services, IEEE International Conference on*, July 2007, pp. 55–63.
- [36] S. Dai, C. Tang, S. Qiao, K. Xu, H. Li, and J. Zhu, "Optimal Multiple Sink Nodes Deployment in Wireless Sensor Networks Based on Gene Expression Programming," in *ICCSN 2010*, February 2010, pp. 355–359.

- [37] S. Misra, S. D. Hong, G. Xue, and J. Tang, “Constrained relay node placement in wireless sensor networks to meet connectivity and survivability requirements,” in *In: IEEE INFOCOM 2008 Proceedings*, April 2008, pp. 281–285.
- [38] D. Yang, S. Misra, X. Fang, G. Xue, and J. Zhang, “Two-Tiered Constrained Relay Node Placement in Wireless Sensor Networks: Efficient Approximations,” in *In: IEEE Secon 2010 Proceedings*, June 2010, pp. 1–9.
- [39] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley-Interscience, March 1990.
- [40] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Num Math*, vol. 1, no. 1, pp. 269–271–271, Dec. 1959.
- [41] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, “TinyDB: an acquisitional query processing system for sensor networks,” *ACM Trans. Database Syst.*, vol. 30, pp. 122–173, March 2005.
- [42] F. C. Delicato, P. F. Pires, L. Pirmez, and L. F. R. d. C. Carmo, “A flexible web service based architecture for wireless sensor networks,” in *Proceedings of the 23rd International Conference on Distributed Computing Systems*, ser. ICDCSW ’03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 730–.
- [43] J. Schneider, “Efficient XML: Taking Net-Centric Operations to the Edge,” in *13th International Command and Control Research and Technology Symposium (ICCRTS)*, June 2008.
- [44] D. Hadzic *et al.*, “Web services i nettverk med begrenset datarate (in Norwegian),” in *FFI report 2006/03886*, 2006.
- [45] N. Glombitza, D. Pfisterer, and S. Fischer, “Integrating wireless sensor networks into web service-based business processes,” in *Proceedings of the 4th International Workshop on Middleware Tools, Services and Run-Time Support for Sensor Networks*, ser. MidSens ’09. New York, NY, USA: ACM, 2009, pp. 25–30.
- [46] J. Leguay, M. Lopez-Ramos, K. Jean-Marie, and V. Conan, “An efficient service oriented architecture for heterogeneous and dynamic wireless sensor networks,” in *Local Computer Networks, 2008. LCN 2008. 33rd IEEE Conference on*, October 2008, pp. 740–747.
- [47] N. Bressan, L. Bazzaco, N. Bui, P. Casari, L. Vangelista, and M. Zorzi, “The Deployment of a Smart Monitoring System Using Wireless Sensor and Actuator Networks,” October 2010, pp. 49–54.
- [48] S. Lin, G. Zhou, K. Whitehouse, Y. Wu, J. Stankovic, and T. He, “Towards stable network performance in wireless sensor networks,” in *RTSS’09*. IEEE, 2009, pp. 227–237.

- [49] T. Liu, A. Kamthe, L. Jiang, and A. Cerpa, “Performance Evaluation of Link Quality Estimation Metrics for Static Multihop Wireless Sensor Networks,” in *IEEE SECON '09*, june 2009, pp. 1–9.
- [50] V. Shnayder, B. Chen, K. Lorincz, T. Fulford-Jones, and M. Welsh, “Sensor networks for medical care,” in *SenSys '05*, 2005, pp. 314–314.
- [51] A. Wood, J. Stankovic, and G. Zhou, “DEEJAM: Defeating energy-efficient jamming in IEEE 802.15. 4-based wireless networks,” in *SECON'07*. IEEE, 2007, pp. 60–69.
- [52] J. Vanhie-Van Gerwen, E. De Poorter, B. Latré, I. Moerman, and P. Demeester, “Real-life performance of protocol combinations for wireless sensor networks,” in *2010 IEEE SUTC*, 2010, pp. 189–196.
- [53] R. Thouvenin. TYMO source code. [Online]. Available: <http://www.tinyos.net/tinyos-2.x/tos/lib/net/tymo/>
- [54] G. Tolle. MultihopLQI source code. [Online]. Available: <http://www.tinyos.net/tinyos-2.x/tos/lib/net/lqi/>
- [55] S. Biswas and R. Morris, “Opportunistic routing in multi-hop wireless networks,” *ACM SIGCOMM'04*, vol. 34, no. 1, pp. 69–74, 2004.
- [56] S. Chachulski, M. Jennings, S. Katti, and D. Katabi, “Trading structure for randomness in wireless opportunistic routing,” *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 4, pp. 169–180, 2007.
- [57] J. Carnley, B. Sun, and S. Makki, “TORP: TinyOS Opportunistic Routing Protocol for Wireless Sensor Networks,” in *Consumer Communications and Networking Conference (CCNC), 2011 IEEE*. IEEE, pp. 111–115.