
Evaluering av ytelsen til den kommende ZigBee-standarden for trådløse sensornettverk

av Erik Undheim

Innlevert: 13.desember 2004



NORGES TEKNISK-NATURVITENSKAPELIGE UNIVERSITET
INSTITUTT FOR ELEKTRONIKK OG TELEKOMMUNIKASJON

Forord

Denne rapporten er resultatet av et fordypningsprosjekt skrevet av student Erik Undheim, høstsemesteret 2004, som går i 5.klasse ved linjen for *Digital kommunikasjon*, institutt for elektronikk og telekommunikasjon, NTNU. Prosjektet er skrevet på oppdrag for ABB Research Center på Billingstad.

Arbeidet har i hovedsak vært et litteraturstudie, hvor det å sette seg inn i den kommende ZigBee-standard har vært en stor del av oppgaven. I ABB er ZigBee-standarden først og fremst tenkt brukt i forbindelse med applikasjoner som trådløse sensornettverk for overvåkning og kontroll av industrielle prosesser. Dermed er det også studert litteratur som tar for seg trådløse sensornettverk og de kapasitetsbegrensninger slike nettverk må forholde seg til og som vil begrense ytelsen. På bakgrunn av dette har prosjektet gått under tittelen 'Evaluerings av ytelsen til den kommende ZigBee-standard for trådløse sensornettverk'. Prosjektet har også fungert som ett forstudie til masteroppgaven som skal igangsettes til våren 2004, under tittelen 'Latenstid i energisensitive trådløse sensornettverk'.

Det rettes en spesiell takk til professor Geir E. Øien, som har fungert som faglærer ved NTNU, samt PhD Niels A. Aakvaag ved ABB Research Center. Niels A. Aakvaag er initiativtageren til prosjektet og har også fungert som veileder i perioden det har pågått.

Erik Undheim

Innhold

1	Innledning	1
1.1	Bakgrunn	1
1.2	Problemstilling	2
1.3	Avgrensninger	2
1.4	Oppbygning	3
2	Generelle krav til trådløse sensornettverk	4
2.1	Lavt effektforbruk	4
2.2	Lave kostnader per node	4
2.3	Ulike nettverkstyper	5
2.4	Sikkerhet	5
2.5	<i>Data Throughput</i>	5
2.6	Latenstid	6
3	Begrensninger for trådløse sensornettverk	7
3.1	Mottatt signalstyrke	7
3.2	Interfererende kanaler	8
3.3	Effektforbruk	9
3.4	Energibesparelse ved å unngå for lange hopp	9
3.5	Akkumulert feilsannsynlighet ved bruk av multihopp	10
3.6	Økt latenstid i multihoppnettverk	11
3.7	Aksess til mediet	11
3.7.1	Generell definisjon av <i>throughput</i> B i MAC-laget	12
3.7.2	Rettferdig håndtering av trafikk som må videresendes	12
3.7.3	<i>Collision domain</i>	13
3.8	Ruting i nettverkslaget	14
3.9	Korteste vei er ikke godt nok	15
3.10	Feiltoleranse	15
3.11	Optimal energifordeling mellom noder i et multihopp nettverk	16
4	Oversikt over ZigBee-standard	19
4.1	IEEE 802.15.4 standarden	20
4.1.1	Komponenter i et IEEE 802.15.4 WPAN	20
4.1.2	Nettverktopologier	21
4.1.3	Arkitektur	22
4.1.4	Kanalaksess	23

4.2	ZigBee	24
4.2.1	Nettverkskomponenter definert av ZigBee	25
4.2.2	Nettverkstopologier definert av ZigBee	25
4.2.3	Valg av formasjon	26
4.2.4	Etablering av et PAN	27
4.2.5	Hvordan enheter kobler seg på nettverket(<i>Association</i>) . .	28
4.2.6	Nabotabell	28
4.2.7	Sending og mottagning	29
4.2.8	Stifinning(<i>routing</i>)	30
4.2.9	Stifinning i <i>mesh</i> -struktur (AODV-algoritmen)	30
4.2.10	Rask oppvåkning for ZigBee-enheter	32
5	Forventet ytelse til ZigBee	33
5.1	Ytelsen til ZigBee-nettverk med stjernetopologi	33
5.1.1	Nodeplassering	34
5.1.2	Effektforbruk	34
5.1.3	Sikkerhet for at pakker kommer frem	35
5.1.4	<i>Throughput</i>	35
5.1.5	Latenstid	36
5.2	Ytelsen til ZigBeenettverk med <i>mesh</i> -topologi	36
5.2.1	Begrensninger som følge av stifinningalgoritmen	37
5.2.2	<i>Throughput</i>	37
5.2.3	Batterilevetid	40
5.2.4	Batterilevetid som følge av nodeplassering - en kuriositet	40
5.2.5	Forventet latenstid	43
5.2.6	Robusthet mot linkfeil	44
5.2.7	<i>Beacon requests</i> i <i>mesh</i>	44
5.3	ZigBeenettverk med tretopologi	44
5.3.1	Throughput	45
5.3.2	Batterilevetid	45
5.3.3	Forventet latenstid	46
5.3.4	Lite robust mot linkfeil	46
6	Konklusjon og videre arbeid	47
6.1	Konklusjon	47
6.2	Videre arbeid	48

Figurer

3.1	Illustrering av energibesparelse ved bruk av to hopp i stedet for et hopp, for ulike plasseringer av mellomliggende node	10
3.2	Feilsannsynlighets permutasjonstree	11
3.3	a) Studie av rettferdig videresending av pakket i et tonode-nettverk til en <i>gateway</i> /sinknode GW; b) ideelt sett c) reell <i>throughput</i> til node 1 og 2 som en funksjon av ønsket datamengde transportert G. Illustrasjon hentet fra [1].	13
3.4	Det trådløse kollisjonsområdet som hører til linken GW-1, illustrasjon hentet fra [1]	13
3.5	Konnektivitet og redundansnivå. a) lineært nettverk R=1. b) <i>Mesh</i> R=2. c) Hexagonal <i>mesh</i> R=3. d) <i>Mesh</i> R=4	15
3.6	G_{\max} versus $P_{\text{ødelagt}}$	16
3.7	Optimal energi versus informasjonsmengde	18
4.1	ZigBee stakken, hentet fra [2]	19
4.2	Sjerne og <i>peer-to-peer</i> topologieksempler, fra [3]	21
4.3	Eksempel på superrammestruktur, fra [3]	24
4.4	Nettverkstopologier støttet av ZigBees nettverkslag	26
5.1	En rekke med $n=8$ noder som genererer og videresender data til en <i>gateway</i> , GW, fra [1]	38
5.2	Et trådløst <i>mesh</i> -nettverk med vilkårlig plassering av noder, fra [1]	39
5.3	Totalenergien brukt ved <i>optimal</i> -, <i>direct</i> - og <i>hop</i> -transmisjon i en lineær topologi som funksjon av antall sensornoder	41
5.4	Totalenergien brukt ved <i>optimal</i> -, <i>direct</i> - og <i>hop</i> -transmisjon i en kvadratisk topologi som funksjon av antall sensornoder	42
5.5	Optimal energifordeling mellom nodene når hver node potensielt kan sende all informasjon som skal inn til sinknoden	42
5.6	Optimal energifordeling mellom nodene når hver node potensielt 5 % av total informasjon som skal inn til sinknoden	43

Tabeller

4.1	Enhetstyper i et IEEE 802.15.4 nettverk, hentet fra [4]	21
4.2	Hovedegenskaper til nettverksstrukturene i ZigBee	27

Forkortelser og definisjoner

Forkortelser

BER	<i>bit error rate [bit feil/sekund]</i>
CAP	<i>contention access period</i>
CCA	<i>clear channel assessment</i>
DSSS	<i>direct sequence spread spectrum</i>
FFD	<i>full function device</i>
GTS	<i>guaranteed time slot</i>
LR-WPAN	<i>low rate wireless personal area network</i>
MAC	<i>medium access control-laget</i>
NPDU	<i>network service data unit</i>
PAN	<i>personal area network</i>
PHY	det fysiske laget
POS	<i>personal operating space</i>
QoS	<i>Quality of Service</i>
RFD	<i>reduced function device</i>
WPAN	wireless personal area network
WMN	<i>wireless mesh networks</i>
WSN	<i>wireless sensor network</i>

Definisjoner

<i>Association</i>	Et service for å etablere en enhets medlemskap i et trådløst PAN
<i>Beacon</i>	En ramme sendt ut av en koordinator eller PAN koordinator for å synkroniserte tilkoblede enheter
PAN <i>identifier</i> (PANId)	Unik identifikasjon av et nettverk
Enhet(<i>device</i>)	En enhet(enten FFD eller RFD) som inneholder en implementasjon av IEEE 802.15.4s MAC og PHY
Koordinator	En FFD-enhet som enheter kan koble seg til et nettverk via og synkronisere seg mot
PAN-koordinator	Hovedkoordinatoren i et IEEE 802.15.4 nettverk, en i hvert nettverk
ZigBee-enhet	Tilsvarende som en enhet, men med ZigBees nettverkslag implementert
ZigBee-ruter	Tilsvarende som en koordinator, men med ZigBees nettverkslag implementert
ZigBee-koordinator	Tilsvarende som en Pan-koordinator, men med ZigBees nettverkslag implementert

Sammendrag

ZigBee-standarden vil høyst sannsynlig stå klar rundt årsskiftet 2004/2005. Lovnader om ultralavt effektforbruk i nodene, og muligheten til å dekke store geografiske områder med nettverkstopologier som tre og *mesh*, gjør standarden svært attraktiv for en rekke anvendelser, også anvendelser som før ikke har vært mulig med tanke på nodeplassering og mobilitet. Egenskaper som lang batterilevetid og geografisk utbredelse går på bekostning av at det må stilles lave krav til faktorer som *throughput* og latenstid, men hvor lave krav? Standarden gir i det hele tatt ingen garantier for ytelsen til et ZigBee-nettverk. Det vil være opp til den enkelte implementasjon å ta de valg som gir nødvendig ytelse for hver spesifikke applikasjonstype.

I denne oppgaven går det nærmere inn på de faktorer som vil avgjøre et ZigBee-nettverks ytelse, særlig med tanke på større sensornettverk der en ZigBee-koordinator skal koordinere nettverk med opp til flere tusen sensornoder. Forventet ytelse til alle de tre nettverkstopologiene tilbudt av ZigBee blir diskutert ut i fra tilhørende egenskaper gitt av standarden. Stjernenettverk har sin klare fordel ved å unngå ressurskrevende ruting, samt være forutsigbar og enkel. Muligheten til å dekke større geografiske områder gir tre- og *meshtopologi* sine fordeler, men på bekostning av ytelse. *Mesh*-topologi har sin klare fordel ved å være robust mot linkfeil, noe som i sensornettverk vil kunne være avgjørende.

Det identifiseres at faktorer som vil være særlig avgjørende for *throughput*, latenstid og batterilevetid er nodeplassering, nodetetthet og antall noder i et nettverk. Blant annet kommer det frem at *throughput* i *mesh*-nettverk vil synke som $O(\frac{1}{n})$, der n er antall noder i et nettverk. Samtidig vil problemer med flaskehalser og ressurskrevende rutingalgoritmer begrense ytelsen ytterligere. Batterikapasiteten til nodene er også et gjennomgående tema, og det blir klart at noder som deltar i ruting vil ha en svært begrenset levetid slik standarden er spesifisert. Strømnettilkoblede rutingenheter vil i mange tilfeller være å foretrekke.

Kapittel 1

Innledning

1.1 Bakgrunn

ZigBee er en kommende standard for trådløse nettverk med lavt energiforbruk og moderate krav til datarate og rekkevidde. Dette gjør standarden særlig velegnet for trådløse sensornettverk, en teknologi det er knyttet seg store forventninger til i fremtiden, særlig innenfor kontroll og overvåkning av industriprosesser og større bygninger. Av andre tenkte applikasjoner kan det nevnes forbrukerelektronikk, eksternt tilbehør til pc, leker og personlig helseovervåkning. I dette prosjektet er hovedfokus på anvendelser av sensornettverk innen industrien, noe som setter spesielle krav til ytelse og topologityper.

I industrien har trådløs innhenting av informasjon fra sensorer i lengre tid vært forbundet med en viss usikkerhet. Terskelen for å erstatte eksisterende stabile og fungerende trådbundne løsninger med trådløse systemer har vært stor. Årsaken er både at eksisterende proprietære standarder har hatt for store begrensninger, samt det faktum at en ikke har lyktes i å enes om en felles standard, noe som i dag nærmest har blitt en forutsetning for at en teknologi skal lykkes og videreutvikles i riktig retning. Den senere års teknologiutvikling innen trådløse nettverk, med standarder som WLAN og Bluetooth, har vist at det er mulig å levere produkter med god nok QoS (*Quality of Service*). WLAN og Bluetooth er derimot dimensjonert for filoverføring, bilde og lyd, noe som gjør dem for energikrevende for sensornettverk, der nodeantall på flere tusen kan bli aktuelt. Behovet for en standard med lavere effektforbruk er påtrengende, men vil gå på bekostning av lavere datarate og latenstid, noe som i sensornettverk i større grad kan tolereres. *ZigBee Alliance* består av over 60 internasjonale industrikonsern som har gått i sammen for å enes om en felles standard, *ZigBee*, nettopp for anvendelser der lav datarate og latenstid kan tolereres og med en batterikapasitet på opp mot flere år. Blir standarden internasjonalt akseptert er potensialet enormt. Dette både med tanke på enklere utplassering, gjerne på steder som før ikke har vært tilgjengelige, samt ryddigere omgivelser som følge av mindre bruk av nettkabler. Svært billige, små og enkle noder vil også kunne åpne for helt nye anvendelser som før ikke har vært praktisk mulig. Om ZigBee blir en suksess vil avhenge sterkt av det endelige produktets ytelse.

ZigBee tar utgangspunkt i den allerede spesifiserte IEEE 802.15.4(2003)-

standarden for trådløse lavrate nettverk med kort rekkevidde, LR-WPAN. En sentral egenskap er muligheten et ZigBee-nettverk har til å ordne seg i tre ulike selvkonfigurerende topologityper; stjerne, tre og *mesh*. Mens et stjernenettverk har store begrensninger med tanke på geografisk utbredelse, kan tre- og *mesh*nettverk bre seg ut og dekke store geografiske områder ved hjelp av multihopp og ruting via relénoder. Dette på bekostning av økt kompleksitet i medium aksesslaget og nettverkslaget i protokollstakken. I et ZigBee-nettverk skjer all transmisjon i samme frekvensbånd og MAC-laget må forhindre pakkekollisjon. Rutingalgoritmene i nettverkslaget må sørge for at pakkene kommer frem til riktig adresse, noe som i en *meshtopologi*, der alternative stier fra sender til mottager vil kunne eksistere, er en svært ressurskrevende oppgave både med tanke på effektforbruk, latenstid og *throughput*. Nevnte egenskaper vil sette store begrensninger på ytelsen til et ZigBee-nettverk.

1.2 Problemstilling

ABB vurderer muligheten for å benytte ZigBee-standarden i noen av deres industrielle installasjoner. Standarden spesifiserer naturlig nok ingen krav til ytelse i form av latenstid, *throughput* og effektforbruk i nodene som følge av topologitype og antall noder i et ZigBee-nettverk. Denne oppgaven har til hensikt å belyse de egenskaper og avveininger som vil begrense og dermed bestemme den endelige ytelsen til et ZigBee-nettverk som følge av valgt topologitype. Det legges hovedvekt på ytelsen til et *mesh*-topologi, hvor informasjon fra alle nodene skal rutes til den samme sinknoden, ZigBee-koordinatoren. Det vil særlig sees på hvordan flaskehalser vil oppstå nær ZigBee-koordinatoren, både i form av noder og linkkollisjonsområder, og hvilke begrensninger dette vil gi i form av lavere *throughput* og levetid for ett nettverk. En annen viktig begrensning er hvordan akkumulert feilsannsynlighet i store multihoppnettverk vil gi seg utslag i form økt latenstid på grunn av retransmisjon og igangsetting av en ressurskrevende rutingsalgoritme. Evalueringen og analysen gjøres med utgangspunkt i IEEE 802.15.4 standarden for det fysiske lag(PHY) og medium aksesslaget(MAC), samt ZigBee's egen spesifikasjon av nettverkslaget, versjon 0.92.

1.3 Avgrensninger

Prosjektet er et litteraturstudie og baserer seg på teoretiske betraktninger av den ikke helt ferdige ZigBee-standarden. Oppgaven begrenser seg til å gjøre betraktninger på ZigBee-nettverk der all informasjon rutes til samme sinknode, noe som er en høyst aktuell situasjon for industrielle sensornettverk av den typen ABB ser for seg.

En komplett teoretisk analyse av forventet ytelse til et reellt ZigBee-nettverk ville vært en svært omfattende oppgave. Oppgaven begrenser seg dermed til å identifisere de viktigste faktorene som vil avgjøre den endelige ytelsen, samt å gå mer i detalj på noen av dem. Hovedvekt er lagt på topologitypen *mesh*, som

det er knyttet seg store forhåpninger til på grunn av dens unike egenskaper med tanke på robusthet mot linkfeil.

Hvordan AODV-algoritmen for ruting og stifinning i *mesh*-topologi vil påvirke latenstid, *throughput* og batterilevetid vil bli viet begrenset oppmerksomhet. En teoretisk betraktning av oppførselen til AODV-algoritmen i store nettverk vil være både for kompleks og idealisert til at et teoretisk resultat med reell nytteverdi vil la seg gjøre. Det er ikke uten grunn at simuleringsresultater er det mest benyttede verktøyet for evaluering av rutingalgoritmer, noe denne oppgaven ikke har benyttet seg av.

Oversikten som gis over ZigBee-standarden begrenser seg til kun å omhandle de detaljer som har betydning for rapportens formål.

1.4 Oppbygning

I kapittel 2 gis en kort gjennomgang av de viktigste kravene som er spesielle for trådløse sensornettverk. Videre i kapittel 3 belyses viktige teoretiske og fysiske begrensninger og rammebetingelsene et trådløst sensornettverk må forholde seg til, med særlig vekt på multihoppnettverk. Kapittel 4 gir en oversikt over de aktuelle standardene, ZigBee's spesifisering av nettverkslaget, samt de viktige underliggende lagene definert av IEEE 802.15.4. I kapittel 5 vil forventet ytelse og begrensninger til et ZigBee-nettverk for de ulike topologitypene diskuteres, for deretter ende opp i en konklusjon i kapittel 6.

Kapittel 2

Generelle krav til trådløse sensornettverk

Det kreves unike egenskaper av trådløse sensornettverk for de applikasjonene de er ment for. Forutsetningene for de avveininger som må tas vil ikke være de samme som for andre trådløse nettverk, der rask filoverføring mellom hvilke som helst noder vil være utgangspunktet for arkitekturen. For å forstå de avveininger som er blitt tatt i ZigBee, samt de underliggende lagene definert av IEEE 802.15.4, er det nødvendig med kjennskap til de egenskapene som kreves. Både Callaway og Hac [5, 6] diskuterer disse kravene i deres bøker, og hvordan protokollene kan designes for å tilfredstille dem. Callaways bok [5] står som referanse på ZigBees egne hjemmesider [7]. Under gis en kort gjennomgang av de identifiserte unike kravene til et sensornettverk.

2.1 Lavt effektforbruk

Applikasjoner hvor trådløse sensornettverk er tiltenkt krever typisk et mye lavere gjennomsnittlig effektforbruk enn standarder som finnes i dag. Batterier vil typisk ikke være oppladbare, i motsetning til hva som ofte er tilfellet eksempelvis for Bluetooth-enheter. Innenfor industrielle sensornettverk, med opptil tusenvis av noder, bør levetiden på hver node tilstrebes å være flere år for at ikke vedlikeholdsjobben med å skifte batterier skal bli for stor.

2.2 Lave kostnader per node

Kostnad per node er avgjørende i et nettverk med et stort antall noder, både med tanke på kostnader ved installasjon og vedlikehold.

For å oppnå lave kostnader per node er det avgjørende at kommunikasjonssprotokollene og designet av selve nettverket gjøres slik at dyre komponenter kan unngås. Protokollene bør tilstrebes å være så enkle som mulig slik at de krever et minimum av minne og prosessorkraft.

Den største kostnaden til mange nettverk er likevel selve administreringen og driften av nettverket. I følge Callaway [5] bør ekte lavkostnadsnettverk være

ad hoc, som vil si selvkonfigurerene og kunne vedlikeholde seg selv. Med selvkonfigurerende menes evnen et nettverk har til å detektere nye noder og organisere seg selv i en struktur uten menneskelig påvirkning. Med det å kunne vedlikeholde seg selv menes evnen nettverket har til selv å rette opp i feil som oppstår uten menneskelig påvirkning.

Motivasjonen for å utarbeide og bli enige om globale standarder er nettopp et ledd i det å kunne produsere komponenter til lavest mulig pris.

2.3 Ulike nettverkstyper

Et vanlig stjernenettverk, med en hovedenhet/basestasjon og slaveenheter, vil være tilstrekkelig for mange formål. Men på grunn av strenge krav til utsendt effekt, samt batteribegrensninger, er det store begrensninger i det geografiske området en hovedenhet/basestasjon kan betjene. Når større geografiske områder skal dekkes er en avhengig av nettverkstyper som muliggjør multihopp (for eksempel tre og *mesh* som støttes av ZigBee). Slike nettverkstopologier krever rutingtabeller, rutingalgoritmer, og innfører *overhead* fra vedlikehold av nettverket, som må støttes av nodene uten at det går ut over krav til effektforbruk og kompleksitet. Dette krever en nøye avveining av funksjonalitet.

2.4 Sikkerhet

Callaway [5] deler sikkerheten opp i to dimensjoner. Den ene er hvor sikkert nettverket i virkeligheten er, mens den andre er hvor sikkert nettverket oppfattes av brukeren, spesielt potensielle brukere. Opplevelsen av hvor sikkert nettverket er, er viktig siden brukerne har en naturlig skepsis til at deres data skal sendes over luften i stedet for i kabler som en fysisk kan se.

Sikkerhet er mer enn bare kryptering, autentisering og integritetssjekk av meldinger. ZigBee har en måte å sikre datapakker på ved kryptering, men dette vies ikke oppmerksomhet i denne rapporten, det henvises til [8] for detaljer. Men sikkerhet kan også innebære sikkerhet for at meldinger kommer frem, selv om linker skulle brytes. En *mesh*-topologi tilbyr denne form for sikkerhet, da pakker kan sendes på alternative stier ved hjelp av ruting, om en link skulle falle ut. Denne egenskapen vil måtte tilbys på bekostning av kompleksitet og effektforbruk.

2.5 Data Throughput

Trådløse sensornettverk har i utgangspunktet milde krav til *throughput* sammenlignet med for eksempel Bluetooth og andre typer WPANs og WLANs. Callaway [5] setter en maks gjennomsnittlig datarate/throughput for trådløse sensornettverk, gjennomsnittlig over en time, på 512 b/s, noe tilfeldig valgt. Flere anvendelser vil ikke kreve mer en 1 b/s eller lavere. Her er det snakk om informasjonsdataraten og ikke rådata sendt over kanalen, som vil være betydelig høyere. IEEE 802.15.4 i 2,4 GHz tilbyr en datarate, som da vil tilsvare rådata, på 250 kb/s.

KAPITTEL 2. GENERELLE KRAV TIL TRÅDLØSE SENSORNETTVERK6

De lave kravene til *throughput* impliserer at med en normal mengde protokoll-*overhead* (*headers*, adresser, stifinningsalgoritmer osv)), så vil kommunikasjonseffektiviteten til et nettverk være veldig lav. Sammenligner en med for eksempel TCP/IP, med pakkelengder i størrelsesorden 1500 bytes, vil et sensornettverk være uten sammenligning. Uansett hvilket design som velges så vil effektiviteten være veldig lav, noe som gir en form for frihet innenfor visse rammer. Det er selvsagt ønskelig å ha så høy *throughput* som mulig, men batterilevetid vil i mange tilfeller være overordnet.

2.6 Latenstid

Kravene til latenstid i trådløse sensornettverk er generelt lave sammenlignet med andre typer trådløse nettverk. Latenstid fra sekunder til minutter vil være akseptable i flere anvendelser. For noen applikasjoner vil lav latenstid være kritisk, og da er det opp til designeren av nettverket å gjøre de riktige avveininger mellom ulike parametre, topologityper og kompleksitet for å tilfredstille gitte behov. I ZigBee kan latenstid til en viss grad garanteres ved å tildele garanterte tidsrammer til utvalgte noder. En skal likevel se at prisen kan være dyr med hensyn på tapt kapasitet for andre noder.

Kapittel 3

Begrensninger for trådløse sensornettverk

Ytelsen til et trådløst sensornettverk er begrenset av en rekke faktorer. Begrensningene vil typisk merkes av brukeren ved at data fra sensorene ikke når sinknoden¹ og dermed heller ikke applikasjonen, at data bruker for lang tid fra sensor til applikasjonen, at hele nettverket henger seg opp som følge av at antall pakker i nettverket overstiger kapasiteten og at batteriet til noder blir utladet med uakseptabelt korte intervaller.

Årsakene til det som for brukeren vil oppfattes som en begrensning i ytelsen vil gjøre seg utslag i alle lagene av protokollstakken. Det er foreløpig gjort svært få studier av den teoretiske kapasiteten og ytelsen til trådløse sensornettverk. Kapasiteten til ad hoc nettverk er det derimot flere resultater å vise til, hvor blant annet Gupta og Kumar har bidratt med et velkjent resultat [9] i form av en øvre grense for *throughput* som følge av antall noder. Resultatene kjennetegnes gjerne ved at forutsetninger som er tatt gjør dem lite anvendbare for sensornettverk. Gupta og Kumar [9] forutsetter trafikk mellom alle noder, og vil ikke representere trådløse sensornettverk, hvor gjerne all trafikk går mot samme node, sinknoden, noe som vil begrense *throughput* ytterligere. I dette kapitlet vil noen av de viktigste teoretiske begrensninger trådløse sensornettverk må forholde seg til presenteres. Mye baseres på de to artiklene [10, 1].

Fokuset er lagt på begrensninger de tre laveste protokollagene må forholde seg til; PHY-, MAC- og nettverkslaget. Det er ikke entydig i hvilket lag de teoretiske begrensningene vil gi seg utslag i, kapitlet er dermed gitt en forholdsvis løs struktur.

3.1 Mottatt signalstyrke

En avgjørende faktor for sannsynligheten for at en link detter ut er mottatt signalstyrke. Mottatt signalstyrke svekkes som følge av avstanden til senderen,

¹Med sinknode menes oppsamlingsnoden hvor sensordata fra nettverket blir sendt mot, samlet opp, og sendt videre til ønsket applikasjon. I ZigBee vil sinknoden typisk være ZigBee-koordinatoren

KAPITTEL 3. BEGRENSNINGER FOR TRÅDLØSE SENSORNETTVERK8

samt stokastiske forstyrrelser som følge av refleksjoner, spredning, fysiske hindringer eller dopplereffekt ved mobilitet. Avanserte modeller kan brukes for å modellere disse effektene, men for systemperspektivet i dette prosjektet er det nødvendig med en forenklet tilnærming. Teoretiske modeller og praktiske målinger har kommet frem til en meget enkel modell som kan brukes for å beskrive signaleffekttap i ulike miljø, [16, 17, 18] i [10]. Gjennomsnittlig tap $L_{p, \text{dB}}(r, n)$ er uttrykt kun ved hjelp av avstanden r og veitaprateeksponenten n som vist i uttrykk 3.1. Parameteren r_0 er en referanseavstand.

$$L_{p, \text{dB}}(r, n) = L_{f, \text{dB}}(r_0) + 10n \lg\left(\frac{r_0}{r}\right) \quad (3.1)$$

Veitap-rate-eksponenten n representerer raten for hvor mye signalttetapet øker med distanse r . Modellen i 3.1 har vist seg å være anvendbar både i miljøer innendørs og utendørs. Eksperimentelle resultat, [16, 18] i [10], har viser at n har typiske verdier på $3,5 < n < 4$ i mobilkanaler utendørs og $2 < n < 6$ på kanaler innendørs. Distansetap opp til referanseavstanden r_0 kan enten bestemmes ut fra målinger nær senderantennen eller ved hjelp av frittromsdempningen ved hjelp av formelen:

$$L_{f, \text{dB}}(r) = 10 \lg\left(\frac{G_t G_r \lambda^2}{4\pi^2 r^2}\right) \quad (3.2)$$

G_t og G_r i 3.2 er henholdsvis antenneforsterkningen i sender og mottaker. Parameteren r_0 må velges større en nærfeltet til antennen (Fraunhoferdistansen).

Ut i fra 3.1 kan en få et uttrykk for mottatt effekt P_t i avstand r fra senderen med utsendt effekt P_t :

$$P_{r, \text{dB}} = 10 \lg(P_t) + L_{p, \text{dB}}(r, n) \text{ dB} \quad (3.3)$$

Utsendt effekt, P_t , må holde seg innenfor regionale bestemmelser for tillatt utstråling både innenfor tildelt båndbredde, og *out-of-band* nivå. Maks rekkevidde bestemmes av mottatt SNR-nivå og vil noe forenklet bestemmes av maks utsendt effekt, faktoren n for omgivelsene, samt mottatt støy fra omgivelsene.

3.2 Interfererende kanaler

I et reellt trådløst sensornettverk vil et stort antall noder konkurrere om å sende på samme frekvenskanal. MAC-laget i ZigBee må lytte på kanalen og vurderer om den er ledig før eventuell transmisjon kan foregå. En terskel for interferensstøy som kan tolereres fra andre noder må dermed være til stede. Geografisk spredte noder vil da kunne sende og motta på samme tid, og transmisjon andre steder i nettverket vil oppfattes som en additiv interferensstøy. Grenseavstanden for akseptabel linkkvalitet, r_g , vil da være avhengig av 3.4, signal-til-interferens forholdet (SIR - *signal-to-interferens ratio*), ved mottakeren. P_r vil være gitt av 3.3.

$$SIR = \frac{P_r}{N_0 + \sum_{i \in \text{interfererendenoder}} I_i} \quad (3.4)$$

3.3 Effektforbruk

En vanlig måte å beskrive gjennomsnittlig effektforbruk til en node på i trådløse sensornettverk på er som vist i uttrykk 3.5 [5].

$$P = \alpha \cdot P_0 + (1 - \alpha) \cdot P_s \quad (3.5)$$

hvor

P = tidsgjennomsnittlig effektforbruk, W

α = *duty-cycle*; den brøkdel av tiden transceiveren er aktiv

P_0 = effektforbruk når transceiveren er aktiv, W

P_s = effektforbruk når transceiveren står i stand-by, W

For å redusere gjennomsnittlig effektforbruk bør både P_0 og P_s reduseres maksimalt. Men for en gitt teknologi og applikasjon vil det være en nedre grense for hvor mye disse parametrene kan reduseres. I de aller fleste transceivere vil effektforbruket når den er aktiv, under sending eller lytting på kanalen, være betydelig høyere enn når den står i stand-by.

Under denne forutsetningen vil en kunne oppnå lavt gjennomsnittlig effektforbruk og lenger batterilevetid med en lav *duty-cycle*. Lav *duty-cycle* ser en ut fra ligning 3.5 vil redusere den dominerende P_0 betraktelig. Lav *duty-cycle* vil være anvendbart i sensornettverk der det er lave krav til samplingshastighet og latenstid for sensordata. I ZigBee vil en i neste kapittel se at lav *duty-cycle* oppnås ved å innføre ikkeaktive perioder hvor ingen transmisjon vil foregå, samt å gi nodene mulighet til å skru av *transceiveren* når de ikke har data å sende.

I en sensornode vil også selve sensoren bruke effekt i tillegg til *transceiveren*. Dette effektforbruket kan i mange tilfeller sees vekk i fra, i og med at sensoren ofte har minimalt effektforbruk sett opp mot radiotransmisjonen av informasjon sensoren generer.

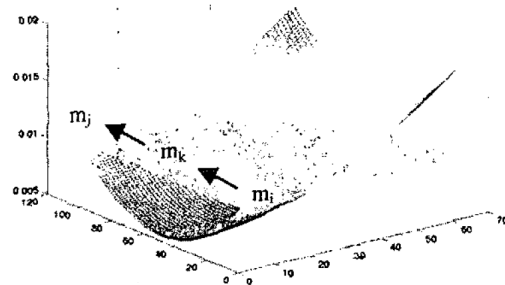
3.4 Energibesparelse ved å unngå for lange hopp

Et av formålene med multihoppnettverkene er å kunne dekke større geografiske områder. Dekningsområdet til et stjernenettverk er begrenset av sinknodens eget POS (*Personal Operating Space*), samtidig som det må kreves at tilkoblede enheters POS er like stort. Den totale effektforbrukøkningen ved å doble dekningsområdet til et stjernenettverk vil være meget stor. I et multihoppnettverk vil dekningsområdet i en retning kunne doubles ved å plassere en relénod på grensen til sinknodens dekningsområde. Dette vil kreve en mye mindre økning av effektforbruket enn ved å øke utsendt effekt til nodene for å nå dobbel så langt.

For å vise energibesparelsen tas det utgangspunkt i at signalet kun dempes av frittromsdempningen gitt i uttrykk 3.2 for avstanden r . P_{t-r} er utsendt effekt for å dekke avstanden r i en retning. Deles veien opp i to ved å sette inn en videresendende node i mellom vil distansen til mellomliggende node være $r/2$ i stedet for r for et-hoppstilfellet. Ut i fra uttrykk 3.2 ser en at en

halvering av avstanden r fører til en firedobling av mottatt signalstyrke på grunn av kvadreringen av r i nemneren. Dermed vil nødvendig utsendt effekt til mellomliggende node være $P_{t-r}/4$ og totalt utsendt effekt for begge hopp vil bli $P_{t-r}/4 + P_{t-r}/4 = P_{t-r}/2$. Dermed vil to hopp over avstanden r kreve halvparten av effekten et hopp ville brukt.

Det er her antatt at den mellomliggende noden er plassert midt i mellom sender og mottager. I [10] er den totale energikostnaden ved ulike plassering av mellomliggende node illustrert som i figur 3.1. Figuren er utydelig men poenget kommer frem. I eksempelet er det antatt en avstand på 100 meter mellom sender og mottager definert som m_i og m_j . Nødvendig sendeeffekt ved bare ett hopp mellom dem er på 0,01 W. 0,01 W er illustrert med et plan, noe som kommer dårlig frem av figuren. Mottatt signalstyrke er antatt holdt på -70 dBm. Totalt effektforbruk med to-hopp er den mest tydelige mørke kurven. En ser at det totale effektforbruket er avhengig av posisjonen til den mellomliggende noden m_k , med minimalt effektforbruk ved plassering midt mellom m_i og m_j . I dette punktet ser en at effektforbruket er godt under 0,01 W som et-hoppsalternativet ville brukt.



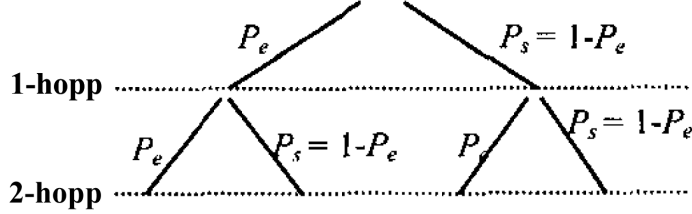
Figur 3.1: Illustrering av energibesparelse ved bruk av to hopp i stedet for et hopp, for ulike plassering av mellomliggende node

En isolert energibetraktning taler altså for multihopp over store avstander, med mellomliggende relénoder så midt i mellom som mulig ved ideelle omgivelser. På samme vis vil det kunne argumenteres for uendelig antall mellomliggende noder som vil gi en totaleffekt som går mot null. Det som da ikke er tatt med i betraktningen er økt kompleksitet i forhold til kanalaksess og ruting for høyereliggende lag. I tillegg innfører også multihopp økt sannsynlighet for retransmisjon som følge av aggregering av pakkefeil i nettverkslaget.

3.5 Akkumulert feilsannsynlighet ved bruk av multihopp

Om det er to hopp fra sensoren og inn til sinknoden må samme pakke sendes to ganger over to ulike linker. En gang av kildenoden og en gang av relénoden. Antar en ideelle forhold som samme støyforhold og samme avstand mellom nodene vil feilsannsynligheten P_e være lik for begge transmisjonene. Transmisjonene

over linkene og utfallet av dem kan illustreres som et tre vist i figur 3.2, der P_s er sannsynlighet for feilfri transmisjon.



Figur 3.2: Feilsannsynlighets permutasjonstree

Ved å traversere treet kan en enkelt regne ut sannsynligheten for feilfri transmisjon $P_{s,sti}$ over en sti, som i to-hoppstilfellet, om hvert hopp regnes som ukorrelerte, vil bli:

$$P_{s,sti} = (1 - P_e)(1 - P_e) \quad (3.6)$$

Ut i fra en generalisering av uttrykk 3.6 kan en enkelt komme til et generelt uttrykk for feilsannsynligheten for transmisjon over en sti $P_{s,sti}$ med h antall hopp, der hvert hopp i har feilsannsynlighet $P_{e,i}$ [10] og er ukorrelerte:

$$P_{e,sti} = 1 - \sum_{i=1}^h (1 - P_{e,i}) \quad (3.7)$$

Dette illustrerer at multihopptopologier vil være mye mer sårbare for linkfeil, siden feilen blir aggregert gjennom stien. En dårlig link ødelegger hele stien. Uttrykk 3.7 vil avgjøre sannsynligheten for minst en retransmisjon av pakken på veien gjennom stien. Dermed er konklusjonen at innføring av multihoppstier vil føre til mer *overhead* som følge av retransmisjoner.

3.6 Økt latenstid i multihoppnettverk

Latenstiden i et multihoppnettverk er i utgangspunktet triviell. En forsinkelse på d for hvert hopp gir en forsinkelse på $d \cdot h$ for h hopp. Dette vil i de fleste relle situasjoner være langt fra sannheten. Faktorer som retransmisjon, pakkeoverhead, begrenset aksess til mediet, pakkekollisjon og forsinkelser som følge av stifinningsalgoritmen vil i praksis gjøre latenstidbetraktninger langt mer komplekse og vil være utenfor denne rapportens omfang. Nettverkssimuleringsprogram som NS-2 [11] egner seg godt for slike betraktninger, noe som sannsynligvis vil være en del av et videre arbeid med å finne forventet latenstid i større ZigBee-nettverk.

3.7 Aksess til mediet

I trådløse sensornettverk bruker stort sett alle nodene et samme felles medium for kommunikasjon over samme frekvens. Alle noder innenfor hverandres POS

vil motta hverandres utsendte pakker, og adressefeltet til hver av dem vil bli sjekket. Dermed vil to pakker utsendt på samme tidspunkt føre til en kollisjon og begge pakkene vil gå tapt. MAC-protokollens hovedoppgave er å unngå slike kollisjoner, samtidig som god effektivitet, latenstid og rettferdig videresending av pakker opprettholdes.

Her vil det gjennomgås de teoretiske begrensningene og definisjoner som brukes i kapittel 5.2 for å vurdere kapasiteten til MAC-laget av et ZigBee nettverk med *meshtopologi*.

3.7.1 Generell definisjon av *throughput* B i MAC-laget

B defineres som den *throughput* det er mulig å oppnå av MAC-laget i et et-hopps nettverk. Eksakt verdi av B avhenger av flere parametre både i MAC-laget selv og det fysiske laget under [1]:

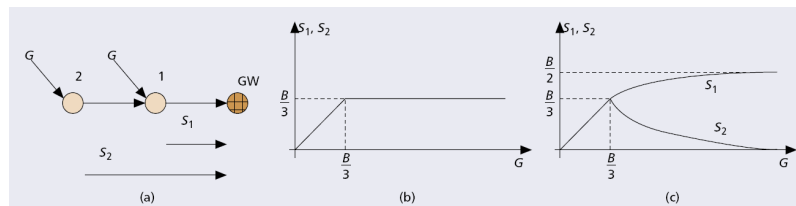
- Radioteknologien, rådataratene til det fysiske laget.
- Effektiviteten til det aktuelle MAC-laget
- Størrelsen og distribusjonen av pakker sent gjennom sinknoden. *Overhead*-størrelsen i MAC-laget kan være svært stor i trådløse lavratenettverk.
- Feilraten, BER, bør tas med om den er signifikant.
- Antall noder i systemet vil ha en innvirkning. Jo flere noder, jo større er sjansen for kollisjon.

Sichitiu, Jun og Peddabachagari [12] viser hvordan en kan finne en eksakt B for IEEE 802.11 om disse parametrene er kjent. Eksakte beregninger av B i et IEEE 802.15. 4 nettverk er utelatt fra denne rapporten. Kapasiteten B vil det derimot antas at finnes, og vil bli brukt i senere beregninger av totalkapasiteten til nettverk avhengig av topologitype valgt.

3.7.2 Rettferdig håndtering av trafikk som må videresendes

I multihopps topologityper vil det finnes noder, som i tillegg til data fra seg selv, også må fungere som en relénod for andre. Situasjonen er som vist i figur 3.3. Node 1 og node 2 ønsker å sende datamengden G. Node 1 må, i tillegg til egne data, motta og videresende data fra node 2 til sinknoden (GW). Ideelt sett så vil node 1, ettersom datamengden G øker, gjøre en rettferdig køing av data fra node to og seg selv, slik at dataene som når sinknoden består av like mye data fra begge nodene. I virkeligheten viser det seg imidlertid at, når datamengdene begge nodene ønsker å sende økes, så vil dataene fra noden nærmest sinknoden gradvis, men etterhvert fullstendig dominere dataene som når sinknoden (GW). Dette skjer når dataene node 2 må sende blir større enn det den har kapasitet til.

Denne oppførselen kan både forklares teoretisk, og er også blitt verifisert i nettverkssimulatorer som OPNET og ns-2 [1]. I denne rapporten antas det derimot, som også antatt i [1], det ideelle tilfellet med lik fordeling av *throughput* mellom noder.

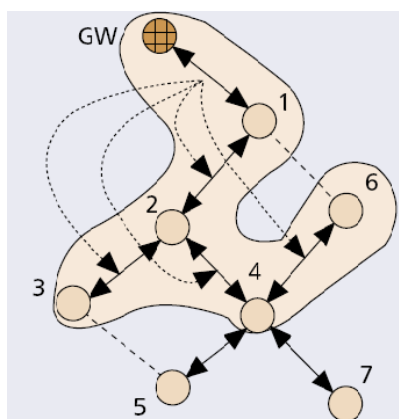


Figur 3.3: a) Studie av rettferdig videresending av pakket i et tonode-nettverk til en gateway/sinknode GW; b) ideelt sett c) reell *throughput* til node 1 og 2 som en funksjon av ønsket datamengden transportert G . Illustrasjon hentet fra [1].

3.7.3 Collision domain

Siden all kommunikasjonen i et ZigBee-nettverk må dele samme båndbredde blir MAC-protokollens hovedoppgave å tilrettelegge transmisjonene slik at kollisjon unngås. Den eneste måten å oppnå dette på er å sørge for at bare en node i et gitt geografisk område, hvor alle noder kan høre hverandre, sender av gangen. Som det kommes tilbake til i kapittel 4.1 bruker MAC-laget til ZigBee-standarden CCA (*clear channel assessment*) for å vurdere om kanalen er i bruk av andre ZigBee-enheter. Styrken på dette signalet avgjør om noden vil forsøke å sende eller vente en tilfeldig periode, eller et tilfeldig antall tidsluker til, før et nytt forsøk gjøres (*unslotted* eller *slotted* CSMA/CA).

I figur 3.4 definerer de hele pilene aktive linker brukt for å videresende trafikk til og fra sinknoden (GW). Linjene som ikke er heltrukne viser koblinger mellom noder som kan motta hverandres transmisjoner. De prikkede pilene peker på de linkene som ikke kan være aktive samtidig med den linkene de peker ut i fra, GW-1.



Figur 3.4: Det trådløse kollisjonsområde som hører til linken GW-1, illustrasjon hentet fra [1]

Kollisjonsområdet, *Collision domain*, til link i defineres som settet av linker inneholdt i link i , samt alle andre linkene som ikke kan være aktive for at link

i skal kunne sende uten kollisjon. Det beige feltet i figur 3.4 viser kollisjonsområdet til link GW-1.

Hidden terminal problem

I figur 3.4 er en større forenkling gjort. Det er antatt at både GW og node 1 vil kunne detektere interfererende ZigBee-enheter i samme område. I virkeligheten vil kanskje ikke GW kunne høre utsendt signal fra node 4, men det vil node 1, og kollisjon vil kunne oppstå om GW prøver å sende til node 1 samtidig som node 4 prøver å sende til node 2, et problem ofte omtalt som *hidden terminal problem*. Ved bruk av CCA for CSMA/CA-algoritmen senses bare kanalen rett før sending av noden som skal sende, altså et asymmetrisk MAC-lag. Det blir aldri sjekket om mottageren er klar for å motta, slik som for eksempel RTS/CTS-algoritmen (Request to send/Clear to send) gjør i IEEE 802.11² [1], og som betegner et symmetriske MAC-lag. Dermed vil kollisjoner kunne oppstå ved at GW sender kanalen som ledig, mens node 4 i realiteten sender samtidig noe som fører til kollisjon. *Collision domain*-modellen kunne tatt høyde for assymetriske linker noe som ville gitt en noe lavere øvre grense for *throughput*. For enkelthets skyld antas det i denne oppgaven symmetriske linker ved hjelp av RTS/CTS. I denne oppgaven er poenget først og fremst å vise effekten kollisjonsområder har på *throughput*.

3.8 Ruting i nettverkslaget

Både i tre- og *meshtopologi* er ruting nødvendig for at pakker skal kunne videreføres riktig vei mot sinknoden. I en treetopologi foregår dette meget enkelt ved hierarkisk ruting. Om en tenker seg sinknoden øverst i hierarkiet vil all sensorinformasjon rutes oppover i treet. I *mesh* blir ruting mer komplisert. Strukturen til *mesh* kan på mange måter sammenlignes med Internet, der rutene til en hver tid må holde styr på nabo- og rutingstabeller. Enkelt sagt så vil stifinningsalgoritmen oversvømme hele nettverket for å bestemme den mest hensiktsmessige ruten mellom to noder. Det sier seg selv at bruken av denne algoritmen må begrenses til det minimale.

I ZigBee er AODV-algoritmen valgt. Dette fordi den er en *reactive* algoritme som etablerer en rute kun i det den trengs. Den vedlikeholder ikke optimale ruter mellom alle nodene til en hver tid slik som flere andre rutingalgoritmer tilbyr, noe som også er dens viktigste begrensning. Kostnadene ville vært for store i form av noderes batterilevetid, latenstid og *throughput*. AODV-algoritmen tilbyr typisk lav *overhead*, men vil ha høy latenstid første gangen en pakke skal sendes til en node hvor ruten ennå ikke er etablert.

²RTS/CTS blir brukt i en IEEE 802.11 for å overkomme dette problemet, og i [13] blir denne standarden sammenlignet med IEEE 802.15.4 ved simuleringer, nettopp for å se konsekvensene av å ikke bruke RTS/CTS. Konklusjonen er at effekten slår tydelig inn ved stor trafikk. Ved lavere trafikk sees derimot liten effekt, og den *overhead* bruken av RTS/CTS innfører ville vært for kostbar for de lavrateapplikasjoner IEEE 802.15.4 er ment for.

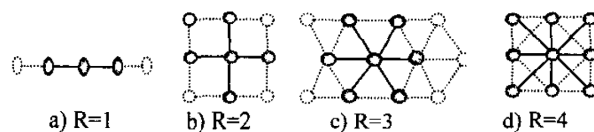
3.9 Korteste vei er ikke godt nok

I en *mesh*-topologi velger rutingalgoritmen den mest hensiktsmessige ruten mellom to noder. Tradisjonelt sett har disse algoritmene basert seg på å finne den ruten med minimalt antall hopp i et ad hoc nettverk. I [14] viser eksperimentelle resultat at det vanligvis vil finnes flere minimum hopp ruter, der flere av dem vil ha dårlig *throughput*. Dermed blir den totale kapasiteten til nettverket dårligere enn det den kunne vært ved mer intelligent valg av stifinningsalgoritmen. Årsaken til dette er at stifinningsalgoritmen ikke tar hensyn til gode og dårlige linker (les mottatt SNRnivå). Det argumenteres derfor i [14] for at linkkvaliteten må tas med i betraktningen når den mest hensiktsmessige ruten skal velges mellom to noder. Dette gjøres i ZigBee ved at fysisk lag måler linkkvaliteten, LQI (Link Quality Index), som deretter kan brukes av nettverkslaget når beste rute skal finnes.

3.10 Feiltoleranse

Stjerne og distribuert topologi betegner på mange måter to hovedklasser av nettverkstopologier. Stjernetopologi er mer sårbar for feil i den sentrale noden eller i noen av linkene inn til den sentrale noden, enn i en distribuert topologi. Hovedfordelen med distribuerte nettverkstopologier er at det tilbys multiple stier over hvor kommunikasjon kan foregå. Tretopologi er en utvidet form for stjernetopologi.

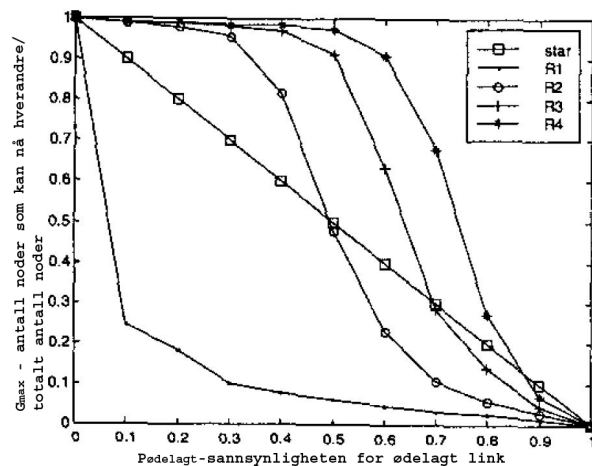
Redundansstørrelsen R i et distribuert nettverk er definert som link-til-noderaten i et array av noder, eller konnektiviteten. Minimum konnektivitet blir brukt som referanse og kan sies å ha et redundansnivå på $R=1$, som vil være tilfelle i en linær topologi der alle nodene ligger på ei rekke som vist i figur 3.5a). *Mesh*-topologier kan ha en redundansstørrelse på $R \geq 2$, som vist i figur 3.5b)-d).



Figur 3.5: Konnektivitet og redundansnivå. a) linært nettverk $R=1$. b) *Mesh* $R=2$. c) Hexagonal *mesh* $R=3$. d) *Mesh* $R=4$

Feiltoleransen i et nettverk med $R \geq 2$ forebygger en total kollaps om en node eller link skulle falle ut.

I [10] er en estimering av hvordan redundansstørrelsen påvirker konnektiviteten mellom to noder gjort ved simulering. Simuleringen består av en symmetrisk nettverkstopologi bestående av 100 noder med en redundansstørrelse på R . Hver link blir så ødelagt med en sannsynlighet $P_{\text{ødelagt}}$. G_{max} betegner antall noder som fremdeles vil kunne nå hverandre på antall noder i nettverket. Grafen i figur 3.6 viser G_{max} versus $P_{\text{ødelagt}}$.

Figur 3.6: G_{\max} versus $P_{\text{ødelagt}}$

Dette er en banalt enkel simulering, men den gir et godt bilde av poenget med å lage mer komplekse topologier som *mesh*. R1 i grafen tilsvarer en linær topologi som vist i a) i figur 3.5. Forsvinner en link i en slik topologi vil flere en to noder miste kontakten med hverandre, noe som kommer klart frem av kurven til R1 i figuren. Dette illustrerer problemer som vil kunne oppstå i en trestruktur hvor en enkelt link vil kunne føre til at et stort antall noder mister kontakten med hverandre. Som vist for en *star*-topologi, så vil konsekvensene av at en link brytes ikke bety annet enn at den ene noden mister kontakten med hovednoden. Med $R=4$ vil svært mange noder fremdeles ha kontakt med hverandre selv med en sannsynlighet for linkfeil på 0,6.

3.11 Optimal energifordeling mellom noder i et multihopp nettverk

I trådløse multihopp sensornettverk vil noder som står nærmere sinknoden måtte videreføre sensordata fra noder lenger vekk fra sinknoden, i tillegg til egne sensordata, se figur 3.3. Dermed vil det i nærheten av sinknoden oppstå flaskehalser med en mye større trafikkbelastning enn noder lenger ute i nettverket. Denne belastningen vil resultere i at noder i et nettverk vil ha et svært ulikt energiforbruk, og dermed batterilevetid, etter hvor de er plassert i nettverket.

Krishnamachari og Ordóñez har i deres artikkel [15] fokusert på teoretiske betraktninger av hvor mye informasjon det er mulig å trekke ut fra et WSN ut fra tilgjengelig energi i nodene. Formålet er å trekke ut så mye informasjon som mulig av et energibegrenset sensornettverk til en sinknode som har ubegrenset tilgang på energi over strømmettet. Modellen antar n sensorer plassert i bestemte posisjoner. Parameteren ϵ_i bestemmer totalt energiforbruk av node i , som vil være summen av energien brukt til sensing, sending og mottagning. Hver node

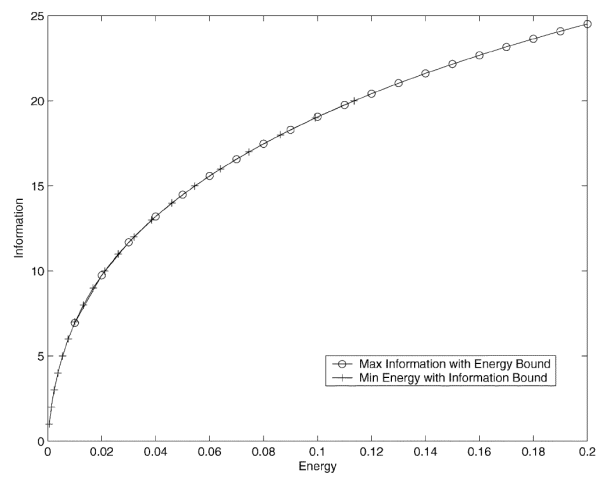
bruker C enheter energi per bit mottatt, og β enheter energi per bit generert av sensoren. I modellen antas det at informasjonsraten og transmisjonseffekten, som har notasjonen f_{ij} og P_{ij} mellom node i og j , kan varieres. Det kommes frem til følgende uttrykk for totalt energiforbruk av sensornodene i et WSN (For mer detaljer rundt utledningen henvises det til [15]):

$$\sum_{i=1}^n \epsilon_i = \sum_{i=1}^n (\beta f_{in+1} + P_{in+1}) + \sum_{i=1}^n \sum_{j=1}^n (C f_{ij} + P_{ij}) \quad (3.8)$$

Forholdet mellom informasjonsraten og transmisjonseffekten, f_{ij} og P_{ij} , for en link er gitt av Shannons's kapasitetsuttrykk for en additiv gausisk støykanal (AWGN), der det antas at signalet svekkes med en faktor d_{ij}^{-2} med en avstand d_{ij} mellom node i og j , noe som tilsvarer kun frittromsdempning.

Uttrykk 3.8 for totalt energiforbruk har en logisk oppbygning. Det første leddet summerer energikostnaden nodene bruker på å sense informasjonen, samt energikostnaden med å sende informasjonen til sinknoden. Det andre leddet representerer kostnaden ved å sende informasjon mellom alle nodeparene $i - j$.

Denne modellen kan brukes til en rekke interessante informasjons- og energibetraktninger av et WSN ved hjelp av ulineær optimalisering av uttrykket, noe som gjøres i [15]. Det kan finnes maksimal informasjonsmengde som kan utvinnes og mottas av sinknoden som funksjon av tilgjengelig energi for gitte nettverkskonstellasjoner og rutingmønstre. Motsatt kan også minimal energi som funksjon av informasjonsmengden som skal utvinnes finnes. Figur 3.7 viser hvordan energimengden brukt for ønsket utdradd informasjonsmengde øker eksponentielt med hensyn på informasjonsmengden. Dermed vil det måtte gjøres en avveining på antall sensorer en ønsker skal sende informasjon til samme sinknode, og dermed utgjøre et nettverk. Dette fordi energiforbruket øker eksponentielt med hensyn på antall sensorer. For mange tilfeller vil det være hensiktsmessig og sette opp to mindre nettverk i stedet for ett stort. I kapittel 5 vil noen flere interessante resultat basert på denne modellen presenteres. Nå skal det derimot gis en oversikt over de viktigste løsningene som er valgt for ZigBee-standarden for å opna de kravene som stilles til lavrate sensornettverk.

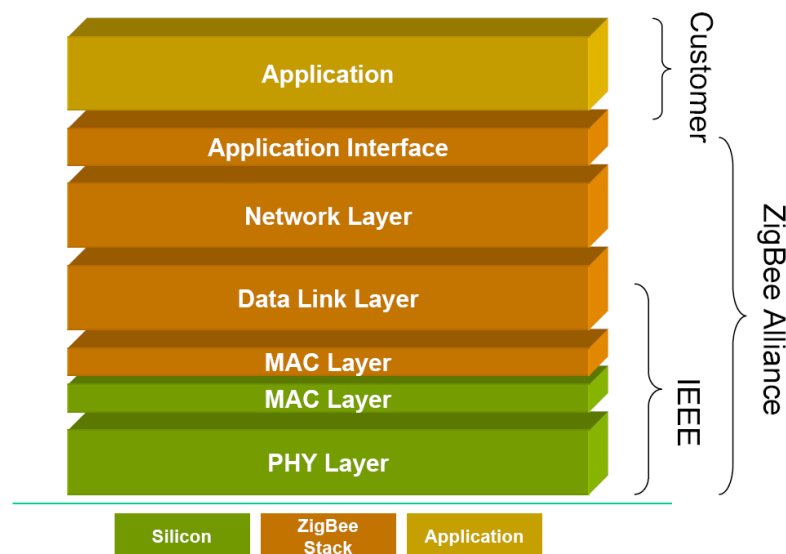


Figur 3.7: Optimal energi versus informasjonsmengde

Kapittel 4

Oversikt over ZigBee-standarden

Arkitekturen til protokollstakken for ZigBee-standarden er vist Figur 4.1. Den baserer seg på den standardiserte syvlagsmodellen OSI [16], men definerer bare de lagene som er relevante for ønskede funksjonaliteter for ønsket markedsområde. IEEE 802.15.4 standarden definerer de to laveste lagnene: det fysiske laget(PHY) og medium aksess laget(MAC). Med IEEE 802.15.4 som fundament sørger ZigBee alliansen for nettverkslaget(NWK), sikkerhetløsninger, samt et rammeverk for applikasjonslaget[8].



Figur 4.1: ZigBee stakken, hentet fra [2]

I dette kapittelet vil det gis en oversikt over de viktigste funksjonalitetene til IEEE 802.15.4, samt nettverkslaget definert av ZigBee alliansen [8]. Det vil bli lagt hovedvekt på de funksjonaliteter som vil være avgjørende for det trådløse nettverkets ytelse.

4.1 IEEE 802.15.4 standarden

Standarden definerer protokollen og den trådløse forbindelsen mellom to enheter for et *low rate - wireless personal area network*(LR-WPAN). LR-WPAN er betegnelsen på et trådløst nettverk utviklet av IEEE 802.15 'working group', beregnet på å koble sammen elektroniske enheter med korte avstander på ti meter eller mindre(*personal operating space(POS)*) med milde krav til datarate. IEEE 802.15.4s hovedformål er å levere en standard for ultralav kompleksitet, ultra-lave kostnader, ultralavt energiforbruk, og lav data rate trådløs kommunikasjon mellom billige enheter.

Karakteristiske egenskaper på et LR-WPAN av typen 802.15.4 er:

- Datarater på 250 kb/s, 40kb/s eller 20 kb/s
- Stjerne eller *peer-to-peer* type topologier
- Mulighet for allokerede tidsluker, *guaranteed time slots*(GTSer)
- *Carrier sense multiple access with collision avoidance*(CSMA-CA) kanalaksess
- *Fully acknowledged* protokoll for sikker dataoverføring
- Lavt energiforbruk
- Energideteksjon (ED)
- *Link Quality indication*(LQI)
- 16 kanaler i 2450 MHz-båndet, 10 kanaler i 915 MHz-båndet og 1 kanal i 868 MHz-båndet.

I de påfølgende underkapittelene vil det gis en nøyere beskrivelse av de viktigste egenskapene til IEEE 802.15.4. Det vil først gis en generell beskrivelse av komponenter som inngår i et IEEE 802.15.4 WPAN og topologiene de kan ordne seg i, for deretter kort se på viktige funksjonaliteter som tilbys av de to lagene, PHY og MAC, som standarden er bygget opp av. Til slutt vil det sees nærmere på hvordan MAC laget skal kontrollere aksess til kanalen, en sentral del av standarden. For mer detaljert informasjon henvises det til [3].

4.1.1 Komponenter i et IEEE 802.15.4 WPAN

For å muliggjøre billige enheter tilbyr et IEEE 802.15.4 WPAN to typer enheter, *reduced-function device*(RFD) og *full-function device*(FFD). En RFD vil kun inneholde en minimal implementasjon og vil bare kunne sende og motta til og fra en FFD. RFD-enheter er ment for ekstremt enkle applikasjoner, som for eksempel en lyssvitsj eller en passiv infrarød sensor. En FFD har utvidet implementasjon for å kunne kommunisere med begge typer enheter i alle typer nettverk, og vil kunne operere som PAN koordinator, koordinator eller som en vanlig enhet. Deres egenskaper er oppsummert i tabell 4.1.

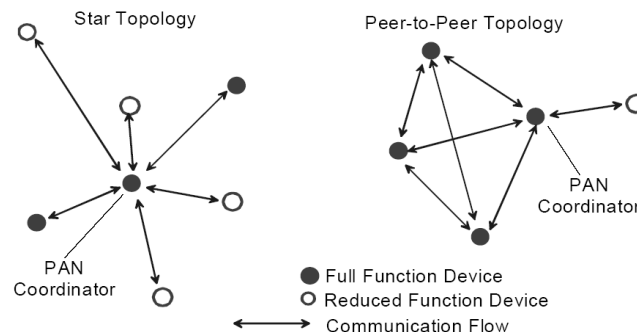
<i>Reduced Function Device</i> (RFD)	<i>Full Function Device</i> (FFD)
Kun bare fungere som endeenhet	Kan fungere i alle topologityper, se 4.1.2
Kan ikke bli koordinator	Kan bli PAN koordinator
Kan bare 'snakke' med FFDer	Kan bli koordinator
Enkel implementering - min. RAM og ROM	Kan 'snakke' med alle andre enheter(RFD/FFD)
Generelt batteridrevet	Generelt tilkoblet strømmettet

Tabell 4.1: Enhetstyper i et IEEE 802.15.4 nettverk, hentet fra [4]

To eller flere enheter i samme POS, hvorav minst en FFD opererer som PAN koordinator, og som kommuniserer over samme fysiske kanal utgjør et IEEE 802.15.4 LR-WPAN.

4.1.2 Nettverkstopologier

Forming av nettverksstrukturer er det nettverkslaget definert av ZigBee som utfører og vil bli nærmere presentert i kapittel 4.2.2. Med utgangspunkt i de to typer komponenter definert i 4.1.1 er likevel rammebetingelsene lagt for to typer topologier, stjerne og *peer-to-peer*, se figur 4.2.



Figur 4.2: Sjerne og *peer-to-peer* topologieksempler, fra [3]

Ved bruk av stjernetopologi kan linker bare etableres mellom en PAN-koordinator i midten og enheter rundt. Enhetene kan både være RFDer og FFDer. Alle stjernenetter vil operere uavhengig av hverandre og det er PAN-koordinatorens oppgave å velge en kanal som ikke interfererer med andre. 802.15.4-standarden [3] nevner at applikasjoner for hjemmautomasjon, eksterne pc-enheter, leker og spill og medisinske anvendelser typisk vil kunne dra nytte av en slik topologitype.

Ved bruk av *peer-to-peer* topologi vil hver enhet(les FFD) være i stand til å kommunisere med andre enheter i eget dekningsområde. En FFD må være PAN-koordinator. Videre struktur kan bli bygget videre ut ifra PAN-koordinatoren og danne grunnlaget for mer komplekse nettverksstrukturer som *mesh* og tre. Dette er opp til nettverkslaget definert av ZigBee å velge. Det er slike type

nettverkstrukturer som vil være interessante innenfor industriell kontroll, automasjon og overvåkning.

4.1.3 Arkitektur

LR-WPAN-standarden er som nevnt bygget opp av to lag, PHY- og MAC-laget. I neste kapittel gis en kort oversikt over ansvarsfordelingen dem i mellom.

PHY-laget

PHY-laget fungerer som et grensesnitt mellom MAC-laget og den fysiske radiokanalen. I tråd med OSI-modellen skal den levere tjenester til laget over, MAC-laget. De viktigste egenskapene til PHY-laget er:

- Aktivisering og deaktivering av transceiveren
- Estimering av mottatt signalstyrke på en gitt kanal(*energy detection*(ED)), uavhengig om mottatt signal er et IEEE 802.15.4 type signal
- Regne ut kvaliteten på en etablert link fra avsender(*link quality indication*(LQI) for hver mottatt pakke
- Lytte på kanalen og vurdere om den er ledig: 'Clear Channel Assessment'(CCA) for CSMA-CA
- Kunne tune transceiveren inn på kanalene IEEE 802.15.4 spesifisere at skal støttes
- Sende og motta data ut på luften. 2.4 GHz PHY bruker en 16-symbols quasi-orthogonal modulasjonsteknikk modulert på bærebølgen ved hjelp av O-QPSK og DSSS.
- Holde seg innenfor lokale bestemmelser for *out-of band* effektnivå. Europeiske bestemmelser (ETSI EN 300 328) begrenser utsendt effekt til ikke høyere enn 10 dBmW, i tillegg til krav til *out-of-band* nivå.

Standarden definerer ikke dekningsområde for et LR-WPAN på grunn av at overføringskarakteristikken er for dynamisk og uforutsigbar. Den kan i mange tilfeller overstige det definerte område POS på ti meter.

MAC-laget

MAC-laget fungerer som et grensesnitt mellom det fysiske laget og nettverkslaget, som i ZigBee standarden vil være definert av ZigBee. De viktigste egenskapene for MAC-laget er:

- Generere *beacons* hvis enheten er en koordinator, se kapittel 4.1.4. Koordinatoren kan selv bestemme om den vil operere i *beacon-enabled mode* hvor superrammestruktur er brukt.
- Kunne synkronisere seg mot mottatte *beacons*, se kapittel 4.1.4.

- Opprette nye PAN-forbindelser og bryte dem
- Utføre CSMA-CA for kanalaksess
- Håndtere og vedlikeholde *guaranteed time slot*(GTS) mekanismen
- Lage en pålitelig link mellom to enheters MAC-lag. For å oppnå dette brukes blant annet *frame acknowledgement* og retransmisjon, feildeteksjon ved hjelp av 16-bits CRC, samt CSMA-CA
- En FFD må kunne utføre aktiv og passiv skanning av kanalene for å finne hvilke kanaler det finnes etablerte IEEE 802.15.4 nettverk på. Et aktivt skan gjøres ved at en ber om å få tilsendt en beacon på hver kanal, for deretter lytte etter hvilke kanaler en får respons på. I et passivt skan lyttes det bare på kanalen etter utsendte beacons fra koordinatorene på de ulike kanalene.

4.1.4 Kanalaksess

Sentralt i IEEE 802.15.4-standarden er hvordan kanalaksess koordineres av MAC-laget. Det tilbys to typer kanalaksess; *non beacon mode* og *beacon mode*.

Non Beacon Mode

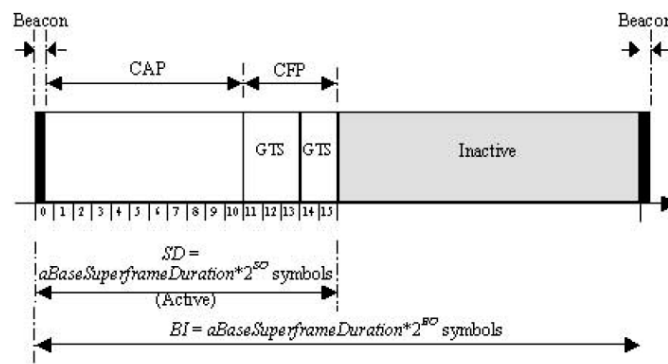
Non beacon-enabled nettverk bruker en versjon av CSMA-CA uten tidsluker(*slotter*). Hver gang en enhet ønsker å sende en type dataramme(mer detaljer rundt datarammetyper i [3]) må den vente en tilfeldig tidsperiode, *random back-off*. Ved hjelp av CCA må enheten så lytte på kanalen for se om den er ledig, og om den er ledig, sende dataene. Om den ikke er ledig skal den vente enda en tilfeldig tidsperiode før den lytter på kanalen og prøver på nytt. *Acknowledgement*-rammer skal bli sent umiddelbart uten bruk av CSMA-CA mekanismen.

Beacon Mode

Standarden tilbyr også valgfri bruk av superrammesstruktur. Det er PAN-koordinatoren som bestemmer formatet på superrammen. Som figur 4.3 viser, består rammen av en aktiv del og en valgfri ikke-aktiv del, innbundet av *beacons*. Den aktive delen består av 16 tidsluker og deles inn i to perioder hvorav en *contention access period*(CAP) og en valgfri *contention free period*(CFP).

Beacon-rammen blir alltid sendt av koordinatoren i første tidsluke. Den sendes ut av en koordinator for å synkronisere alle tilkoblede enheter, og for å beskrive strukturen på superrammen. Tidsintervallet mellom *beacon*-rammer kan være mellom 15.36 ms og 251,66s avhengig av kravene til latenstid og energiforbruk.

Den valgfrie CFPen kan romme opp til åtte såkallede *guaranteed time slots*(GTSer), og en GTS kan okkupere mer en en tidslukeperiode. Garantert tidsramme muliggjør garantert båndbredde for viktige enheter. En tilstrekkelig stor CAP må likevel tilegnes for *contention based access* for at andre typer nye enheter skal



Figur 4.3: Eksempel på superrammestruktur, fra [3]

kunne koble seg på nettverket. Alle *contention based* overføringer skal være unnagjort før CFP-luken begynner, og alle GTS-overføringer må være unnagjort før enden av CFP.

En slottet versjon av CSMA-CA er brukt for kanalaksess i *Beacon Mode* under CAP. Hver gang en enhet ønsker å sende data i løpet av CAP må den lokalisere slutten på neste *back-off*-luke, for deretter vente et tilfeldig antall *back-off*-luker. Ved hjelp av CCA må enheten deretter lytte på kanalen for å vurdere om den er ledig før den aksesserer kanalen. Om den ikke er ledig må den vente i et tilfeldig antall *back-off*-slots før samme prosedyre gjentas.

Superrammen kan også ha en ikke-aktiv periode etter CAP og CFP, før neste *beacon*, se figur 4.3. I denne perioden skal koordinatoren ikke utføre noen form for interaksjon med tilknyttede enheter og kan gå inn i *low power mode*, som innebærer og skru av *transceiveren*. I følge [17] vil det med maks tidsintervall mellom *beacons* på 251,66s være teoretisk mulig for PAN koordinatoren å skru av *transceiveren* i 251,64s, noe som tilsvarer en svært lav *duty-cycle*. Så lav *duty-cycle* vil bare være mulig for helt spesielle applikasjoner, som for eksempel styring av et fåtall lyssvitsjer to ganger i døgnet.

Bruk av *beacons* gir også enheter redusert energiforbruk ved at deres mottakere bare trenger å være påskrudd i det tidsrommet *beacons* blir sendt ut. Da kan dem våkne opp, synkronisere seg mot en *beacon*, sende pakken, for deretter legge seg i søvn igjen. Eventuelt kan den synkronisere seg mot en *beacon* og sjekke om det er pakker som venter på å bli sendt til seg selv. Det er *beacon*-rammen som varsler tilkoblede enheter om de har pakker som venter på å bli sendt til dem. Det henvises til spesifikasjonen [3] for mer detaljer rundt hvordan sending og mottagning blir koordinert for de to tilbudte aksessmetodene.

4.2 ZigBee

Nettverkslagsprotokollen til ZigBee er spesifisert i [8]. Den skal inneholde rett funksjonalitet til å fungere sammen med de underliggende lagene definert av IEEE 802.15.4, samt den funksjonaliteten som er nødvendig for å tilfredstille de applikasjonene ZigBee tar sikte mot. ZigBees nettverkslag skal inneholde

mekanismer for å koble seg på og koble seg av et nettverk, samt sørge for sikkerhet til datarammer og rute dem til rett mottaker. Stifinning(*route discovery*) og vedlikehold(*route maintenance*) av stier tilfaller også nettverkslaget, samt oppdagning av ett-hops naboer og lagring av relevant naboinformasjon. Nettverkslaget til en PAN-koordinator er ansvarlig for å starte et nytt nettverk, og tilegne adresser til nytilkoblede enheter.

Dette kapitlet gir en kort oversikt over de viktigste delene av nettverksspesifikasjonen. Etter en kort gjennomgang av komponenttyper og topologier, vil de viktigste prosedyrene for å forme og opprettholde et nettverk beskrives, samt hvordan data rutes i de ulike topologiene.

4.2.1 Nettverkskomponenter definert av ZigBee

Et ZigBeenettverk kan inneholde tre typer enheter; ZigBee-enheter, ZigBee-rutere og en ZigBee-koordinator. Dette tilsvarer de komponentene IEEE 802.15.4 benevner enhet, koordinator og PAN koordinator, men ZigBee spesifikasjonen beskriver deres funksjonaliteter for nettverkslaget.

ZigBee-enhet

- En ZigBee-enhet skal kunne koble seg på og av et nettverk
- Kan være enten en RFD eller FFD

ZigBee-ruter

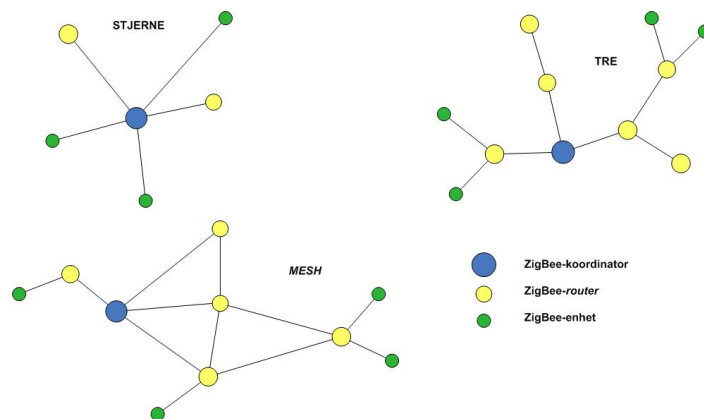
- En ZigBee-ruter skal i tillegg kunne la enheter koble seg på og av nettverket. Dette gjør den enten ved at den får beskjed fra MAC-laget om at en enhet ønsker å koble seg på eller av, eller ved en eksplisitt forespørsel fra applikasjonslaget.
- En ZigBee-ruter må også kunne tildele en passende adresse til enheter som kobler seg på nettverket, samt opprettholde en nabotabell der alle naboene og deres adresser er listet opp.
- En ZigBee-ruter må ha den funksjonalitet som trengs for å rute pakker til rett destinasjon i de multihoppstopologiene definert av ZigBee.
- Dette er funksjonalitet som bare kan kreves av en FFD.

ZigBee-koordinator

- En ZigBee-koordinator har samme funksjonaliteter som en ZigBee-ruter, men har ansvaret for å starte nettverket.

4.2.2 Nettverkstopologier definert av ZigBee

Ut i fra rammebetingelsene definert av IEEE 802.15.4, se kapittel 4.1.2, har ZigBee spesifisert tre typer nettverkstopologier; stjerne, tre og *mesh*, se figur 4.4. Stjernetopologi fungerer som forklart i kapittel 4.1.2, enhetene har bare



Figur 4.4: Nettverkstopologier støttet av ZigBees nettverkslag

skiftet navn som i figuren, mens *tre* og *mesh* er spesialtilfeller av *peer-to-peer*-nettverk.

I både stjerne, tre og mesh er ZigBee-koordinatoren et naturlig valg som sinknode (den noden hvor informasjonen fra sensorene tappes), men dette er det opp til applikasjonslaget å velge.

Tre

Trestruktur er en modifikasjon av en stjerneformasjon. En eller flere av enhetene koblet til ZigBee-koordinatoren må være en ZigBee-ruter, hvor nye ZigBee-rutere og/eller ZigBee-enheter videre kan koble seg på. Dermed oppstår en treformasjon som utspringer seg fra ZigBee-koordinatoren. Nye ZigBee-enheter kobler seg til allerede tilkoblede enheter med rutingkapasitet i et forelder-avkom forhold, hvor den nytilkoblede enheten selvsagt er avkommet i forholdet. Er ZigBee-enheten en RFD, som beskrevet i kapittel 4.1.1, kan den bare opptre som en endeenhet, også kalt løvnode.

Mesh

I et *mesh*-nettverk vil det være full kobling mellom alle ZigBee-rutere og ZigBee-koordinatoren. RFDer kan koble seg på hvilken som helst FFD, men kan bare opptre som en endeenhet og ikke fungere som ruter.

FFDene har funksjonalitet for å kunne rute pakkene i optimale stier ved hjelp av stifinningsalgoritmen AODV, se kapittel 4.2.9

4.2.3 Valg av formasjon

Valg av struktur vil være av stor praktisk betydning og må nøye vurderes etter applikasjonsstype. De har hver sine fordeler og ulemper og i tabell 4.2.3 er de mest tilsynelatende egenskapene listet opp. En nøyere vurdering av egenskapene vil bli presentert i kapittel 5.

	Stjerne	Tre	Mesh
Fordeler	<ul style="list-style-type: none"> • enkel å implementere • maks ett hopp gir minimal latenstid • lavt energiforbruk (ingen stifinring muliggjør lange perioder i sove-modus) 	<ul style="list-style-type: none"> • kan dekke et større område • enkel stifinring (opp eller ned) • bruk av <i>beacons</i> gir enkel synkronisering av noder 	<ul style="list-style-type: none"> • kan dekke et større område • stiredundans gir større feiltoleranse • valg av optimale stier kan gi større kapasitet
Ulemper	<ul style="list-style-type: none"> • geografisk begrenset av PAN koordinatorens POS • sårbar for linkfeil 	<ul style="list-style-type: none"> • ingen stiredundans, sårbar for linkfeil • større latenstid • økt energiforbruk 	<ul style="list-style-type: none"> • størst kompleksitet; høyt energiforbruk • større latenstid • større pakkeoverhead pga. stifinring

Tabell 4.2: Hovedegenskaper til nettverksstrukturene i ZigBee

4.2.4 Etablering av et PAN

Bare en ZigBee-koordinator som ikke allerede er med i et nettverk kan starte et nytt PAN. Her gir en forenklet beskrivelse av hvordan det blir gjort.

Det første ZigBee-koordinatoren gjør er å be MAC-laget skanne tilgjengelige kanaler ved hjelp av *energy detection*(ED), se kapittel 4.1.3. Det blir laget en liste med kanalene i ordnet rekkefølge etter hvor mye energi som er detektert i hver kanal, for deretter å fjerne kanalene hvor energinivået er over det som kan aksepteres. I hver av de kanalene med akseptert energinivå skal ZigBee-koordinatoren deretter utføre et aktivt skan, se kapittel 4.1.3, for å finne antall allerede etablerte ZigBee-nettverk innenfor rekkevidde. På bakgrunn av disse to opplysningene velger ZigBee-koordinatoren ut den kanalen med lavest detektert energinivå hvor det samtidig er etablert færrest ZigBee-nettverk fra før. Finner ZigBee-koordinatoren en passende kanal velger den en unik *PAN-identifer* og en adresse for nettverket, for deretter gi beskjed til applikasjonslaget om at alt er klart for å starte opp nettverket.

ED måler energien på en kanal uavhengig om det er et IEEE 802.15.4 nettverk som sender ut interererende signal, og man unngår dermed å forstyrre

eller bli forstyrret av andre trådløse systemer som befinner seg i nærheten. Dette er nødvendig da 2,4 GHz båndet er et ulisensiert bånd hvor systemer som blant annet WLAN og Bluetooth befinner seg.

Under oppstart har også ZigBee-koordinatoren ansvaret for å bestemme viktige nøkkelparametre for de underliggende lag definert av IEEE 802.15.4. ZigBee støtter både bruk av *beacons* og superrammestruktur i MAC-laget i stjerne- og tretopologier. Tiden mellom beacons, samt lengden på aktiv periode av superrammen, se kapittel 4.1.4, er viktige parametre som ZigBee-koordinatoren må bestemme. I en *mesh*-topologi skal ikke ZigBee-rutere sende ut *beacons* regelmessig. Mesh-topologiens 'ad-hoc'e natur, hvor alle FFDer kan kommunisere med hverandre ville medført et uhåndterlig antall utsendte beacons. Fordelene ved bruk av *beacons* og superrammestruktur kan utnyttes i en tretopologi, men det krever en nøye planlegging av *beacon*intervall for å unngå kollisjoner. ZigBee har laget en egen algoritme for å unngå slike kollisjoner, det henvises til kapittel 10.4 i [8] for ytterligere detaljer.

4.2.5 Hvordan enheter kobler seg på nettverket (*Association*)

Det er bare en ZigBee-koordinator eller en ZigBee-ruter som kan la enheter koble seg til nettverket. Avsnittet under gir en enkel beskrivelse av prosedyren en ny enhet må bruke for å koble seg på et ZigBee-nettverk.

Enheter som vil koble seg på utfører først en aktiv skanning av kanalene gjennom MAC-laget. Resultatet viser alle ZigBee-enheter det er mulig å koble seg til. Disse enhetene lagres i en nabotabell. Om det er første gang enheten kobler seg til ZigBee-nettverket velger den en passende enhet å koble seg på fra den nyopprettede nabotabellen. Den passende enheten må ha en linkkostnad(LQI) på maksimum 3, se i 4.2.9. Hvis nabotabellen har flere like passende enheter å velge i skal den med kortest avstand til PAN koordinatoren velges.

ZigBee-ruteren tildeler deretter en unik nettverksadresse til den nye enheten og legger den inn i sin egen nabotabell. Hvordan adresser tildeles enheter er en sentral del av spesifikasjonen men blir ikke kommentert videre i denne oppgaven. Hvis enheten som ønsker å koble seg på finnes i nabotabellen fra før tildeles ikke en ny nettverksadresse, men ellers følges samme prosedyre.

Spesifikasjonen inneholder også en prosedyre for å koble på nye enheter direkte, ved å prekonfigurerer adressen til enheten i den ZigBee-ruteren den skal kobles til.

4.2.6 Nabotabell

Nabotabellen til ZigBee-rutere skal inneholde informasjon om alle enheter innenfor POS. Denne informasjonen har ulike anvendelser i flere viktige prosedyrer. Parametre som må være med om hver nabo er:

- PAN *identifier*
- Utvidet adresse om enheten er en forelder eller et avkom
- Nettverksadresse

- Enhetstype
- Koblingstype(*Relationship*)

Tabellen kan utvides om ønskelig med informasjon som for eksempel; RxOn-WhenIdle, utvidet adresse uansett nabotype, beaconvstand, dypde, gjennomsnittlig LQI. Dette er opp til de som implementerer systemet å velge.

4.2.7 Sending og mottagning

Sending

Nettverkslaget skal tilby applikasjonslaget å transportere applikasjonsprotokoll-data mellom to eller flere enheter. Dette gjør den ved å tilføye en header på dataene fra applikasjonslaget. Datarammen generert av nettverkslaget kalles *Network level protocol data unit*(NPDU). De viktigste feltene i headeren til en NPDU er destinasjonsadresse- og avsenderadressefeltet, samt rammekontrollfeltet. Rammekontrollfeltet beskriver hva slags type data rammen inneholder samt andre typer kontrolldata, som for eksempel protokollversjon.

En NPDU klar for sending blir sendt videre ned i protokollstakken til MAC-laget, og deretter sendt til mottakeren via PHY-laget.

Mottagning og synkronisering

Hvordan enheter mottar data og synkroniserer seg med en ZigBee-koordinator eller ZigBee-ruter avhenger av om den er i et *beacon enabled* eller i et *non beacon enabled* nettverk.

I et *beacon enabled* nettverk kan enheter sjekke om det er data som venter på å bli sendt til seg selv ved å undersøke mottatte *beacons*. Et av feltene i en *beacon*-ramme inneholder adresser til de enheter som har data som venter på å bli sendt til dem. For å motta disse dataene må enheten sende en beskjed til ZigBee-ruteren i løpet av superrammens CAP-tidsluke om at den er klar for å motta data, for deretter vente på å få dataene tilsendt.

Lytting etter utsendte *beacons* er måten enheter kan synkronisere seg mot rutere og koordinatorene på i et *beacon enabled* ZigBee nettverk.

I et *non beacon enabled* nettverk kan enheter sjekke om det er data som venter på å bli sendt til seg selv ved å 'polle' en ZigBee-koordinator eller en ZigBee-ruter. 'Polling' innebærer å sende en forespørsel direkte til gjeldene enhet å spørre om den har data som venter. Om den har ventende data blir disse så sendt til enheten ved hjelp av CSMA-CA uten tidsluker. Om den ikke har ventende data så blir en tom pakke sendt på samme måte.

'Polling' er måten enheter kan synkronisere seg mot rutere og koordinatorene på i et *non beacon enabled* ZigBee-nettverk.

En ser at hovedforskjellen er at all transmisjon foregår mer strukturert i et *beacon enabled* ZigBee-nettverk. Når data skal sendes eller mottas i et *non beacon enabled* nettverk så vil det kunne skje umiddelbart ved hjelp av CSMA-CA uten tidsluker. I et *beacon enabled* nettverk må det derimot først synkroniseres mot utsendte *beacons*, for deretter innrette seg etter superrammestrukturen beskrevet i den, for så å kunne sende innenfor CAP.

4.2.8 Stifinning(*routing*)

En sentral del av nettverkslaget oppgave er å definere hvordan data skal rutes for de ulike topologiene. I en stjernetopologi er alle noder ett hopp unna ZigBee-koordinatoren og noen form for ruting er ikke nødvendig. I tre- og meshtopologier vil derimot veien fra avsender til mottager kunne inneholde mer enn ett hopp. Da er det nødvendig å finne den korteste ruten. En trestruktur er et strengt hierarki ut fra ZigBee-koordinatoren, og bare en korteste vei mellom en node og ZigBee-koordinatoren finnes. Stifinningen for en trestruktur i ZigBee kalles gjerne hierarkisk stifinning. Det eneste som må avgjøres i hver ZigBee-ruter på veien mellom avsender og destinasjon er å vurdere om destinasjonen er av egne avkom. Hvis ikke sendes pakken opp til ruterens forelder. Med ZigBee-koordinatoren som sinknode vil det meste av trafikken kun gå oppover i treet. I en *mesh*-topologi vil det kunne finnes flere alternative ruter mellom samme avsender og destinasjon. Da er ikke korteste vei like enkelt å avgjøre.

4.2.9 Stifinning i *mesh*-struktur (AODV-algoritmen)

Valget mellom to ruter til samme destinasjon i en *meshtopologi* kan for eksempel stå mellom en rute med bare ett-hopps avstand og en med to-hopps avstand. I mange tilfeller vil ett-hopps ruten være mest fordelaktig, men om avstanden mellom nodene i ett-hopps ruten er lang, eller om fysiske hindringer forstyrrer linken, vil to-hopps ruten kunne være et bedre valg. I et ZigBee-nettverk med *mesh*-topologi er AODV-algoritmen valgt som stifinningsalgoritme. Det er en kjent algoritme opprinnelig utviklet for Internett [18, 19]. Valg av beste rute er særlig avgjørende for latenstiden og energiforbruket, og er dermed avgjørende for ytelsen til ett ZigBee-nettverk. Som sagt i kapittel 3.8 så er AODV-algoritmen *reactive*, med de fordeler og ulemper det medfører.

Stifinningsprosedyren(*Route discovery*)

Denne prosedyren settes i gang av AODV-algoritmen når destinasjonsadressen ikke finnes i stitabellen. AODV bruker en tradisjonell stitabell(*Route Table*). I motsetning til stitabellen i den ofte sammenlignede rutingalgoritmen DSR [20], som både kan inneholde flere stier til samme destinasjon, og som også inneholder alle hopp på veien, inneholder tabellen bare en rad per destinasjon. Innholdet i stitabellen består av destinasjonsadressen og adressen til neste hopp på veien til destinasjonen, samt ett statusfelt.

Route discovery er en omfattende prosedyre hvor hele nettverket involveres for å finne beste rute fra avsender til destinasjon og må dermed unngås i det lengste. Den opptar store ressurser som vil begrense både *throughput*, latenstid og effektforbruk. Her gis en kort gjennomgang av de viktigste egenskapene til prosedyren. Det henvises til [18, 19, 8] for ytterligere detaljer.

Selve algoritmen starter med at enheten som må etablere en ny sti, *route-discovery*, sender ut en RREQ(*Route Request*) til alle naboene. Den inneholder blant annet destinasjonsadressen, hvem som er kilden til RREQen og akkumulert stikostnad. Alle naboenheter som mottar en RREQ, og som ikke

er destinasjonen, lager en midlertidig plass til stifinningsprosedyren som er underveis i en stifinningstabell (*route discovery table*) og setter opp en reversert sti tilbake til kilden. Hver enhet som mottar en RREQ må først avgjøre om den har mottatt en RREQ med kortere akkumulert stikostnad tidligere. Hvis en RREQ med kortere akkumulert stikostnad finnes fra før i stifinningstabellen slettes den nylig mottatte, og ingenting sendes videre. Hvis mottatte RREQ derimot har lavere akkumulert stikostnad enn tidligere mottatte RREQer så kringkastes den videre til sine naboer igjen og stifinningstabellen oppdateres. Slik oversvømmes nettverket med RREQ pakker helt til destinasjonen nås. Da sjekkes destinasjonens stitabell for å sjekke om den allerede har en sti med lavere akkumulert kostnad til kilden. Hvis ikke oppdaterer enheten stitabellen og svarer med å sende ut en *route reply* (RREP). RREP følger den reverserte stien tilbake til kilden mens stikostnadspostene i tabellene blir oppdaterte. Kilden som først sendte ut RREQ vil få inn flere RREP, men bare de med lavere akkumulert stikostnad enn tidligere mottatte fører til en oppdatering av stitabellen. En *timer* følger RREQ pakkene og bestemmer maks lengde for en prosedyres varighet.

AODV opererer med symmetriske linker. Det vil si at en link fra A til B vil ha samme stikostnad som fra B til A.

Med en definert stifinningsalgoritme og nabotabell har en den funksjonaliteten som trengs for å kunne behandle mottatte pakker og eventuelt sende dem videre til rett mottager.

Når en pakke mottas i nettverkslaget

Kommer pakken fra enhetens applikasjonslag og adressen ikke er enhetens egen, avgjøres først om destinasjonsadressen er en nabo. Hvis den er det sendes pakken direkte til destinasjonen. Hvis ikke, og enheten har rutingkapasitet, sjekkes stitabellen for å se om destinasjonen står der. Står destinasjonen der rutes pakken til tilhørende neste-hopp-adresse i tabellen. Står ikke destinasjonsadressen i stitabellen må stifinningsalgoritmen aktiveres for destinasjonsadressen før rett neste hopp kan avgjøres.

Kommer pakken fra underliggende lag følges samme prosedyre. Hvis pakken er til enheten selv sendes den opp til applikasjonslaget, hvis ikke slås det opp i stitabellen og sendes til neste hopp mot destinasjonen, eller eventuelt sette i gang *route-discovery*.

Beregning av linkkostnad (*path cost*)

For å kunne sammenligne og velge en optimale rute fra flere alternativer er en nødt til å kunne representere hver sti med en kostnad. Den enkleste måten å gjøre dette på er vurdere hver sti etter antall hopp. Denne metoden er enkel, men vil i flere tilfeller ikke være tilstrekkelig god som diskutert i kapittel 3.9. Hver link i ett nettverk vil være forskjellig, både med tanke på avstand og ulike typer fading som følge av omgivelsene. ZigBee overlater mye frihet til implementasjonen å velge hvordan dette skal gjøres, men definerer likevel en link kostnad $C\{l\}$ på link l som må være ett tall mellom 1 og 7, der 1 er

minimum og 7 er maksimum kostnad. Kostnaden til en sti med flere linker blir lagt i sammen til en akkumulert linkkostnad.

Måten å regne ut $C\{l\}$ på er opp til implementasjonen. Ett forslag som er foreslått er å basere estimatet av $C\{l\}$ på gjennomsnittlig LQI, som er ett mål på linkkvalitet MAC-laget i IEEE 802.15.4 regner ut for hver mottatt pakke. En tabell kan da avgjøre hvilken linkkostnad, mellom 1 og 7, målt LQI tilsvarer. LQI-estimatet baserer seg enten på SNR estimering, ED eller en kombinasjon av disse.

4.2.10 Rask oppvåkning for ZigBee-enheter

Muligheten ZigBee-enheter har til å ligge i 'søvn' for deretter våkne opp, sende ut data, for så å legge seg i søvn igjen er sentral for lang batterilevetid. Tiden enheter bruker på å våkne opp å legge seg i søvn igjen er meget rask, og er i [4] sammenlignet med Bluetooth. I listen under vises noen estimerte tall som på mange måter beskriver forskjellen mellom hvilke applikasjoner de to standardene er laget for.

- Typisk tid det vil ta for en ZigBee-enhet og koble seg til ett ZigBee-nettverk er 30 ms. Tilsvarende tall i Bluetooth er typisk 3 s.
- Typisk tid det vil ta for en ZigBee-enhet å gå fra sovende til aktiv tilstand er 15 ms. Tilsvarende tall i Bluetooth er typisk 3 s.
- Typisk aksestid for en aktiv slavekanal er 15 ms i ZigBee. Tilsvarende tall i Bluetooth er typisk 2 ms.

Det er likevel viktig å ha klart for seg at de estimerte tallene over først og fremst stammer fra kommersielle presentasjoner for markedsføring av ZigBee. En oppkoblingstid på 30 ms, som beskrevet i listen over, baserer seg for eksempel på et *beacon enabled* nettverk med en *beaconorder* på 0, altså den kortest mulige intervallengden mellom utsendte *beacons*. I teorien kan da oppkoblingen skje i løpet av neste superramme og 30 ms i oppkoblingstid er teoretisk mulig. I et *non beacon mode*, som er situasjonen i et *mesh*-nettverk, vil oppkoblingstiden bli betydelig høyere. Etter å ha brukt aktivt skan (sende ut *beacon request* på alle kanaler), velges den mest hensiktsmessige koordinatoren å koble seg til, for deretter å sende ut et *association request* pakke. Etter å ha mottatt *ack* tilbake fra koordinatoren må enheten vente en *aResponsWaitTime* før data kan sendes eller mottas. *aResponsWaitTime* er en MAC-konstant på 30720 symbol som vil bety 491,52 ms i 2,4 GHz båndet. Konklusjonen er at en oppkoblingstid på minst 0,5 sekund må tilregnes i *mesh*-nettverk. I *beacon enabled* nettverk vil estimert tid for oppkobling være avhengig av intervallet mellom *beacons*, *beaconorder*, noe som for mange applikasjoner vil være større enn 0 som antatt i punktlisten over.

Kapittel 5

Forventet ytelse til ZigBee

Med ytelsen til et ZigBee nettverk menes i denne oppgaven hvordan et reellt ZigBee nettverk vil oppføre seg og yte for ulike konstellasjoner og oppsett med hensyn på latenstid, *throughput* og batterilevetiden til nodene. De ulike valgene som er tatt i ZigBee-standarder er alle med den hensikt å gi et ZigBee-nettverk best mulig ytelse i forhold til de kravene beskrevet i kapittel 2. Mange valg er likevel, som beskrevet i forrige kapittel, opp til implementeringen å velge ut i fra applikasjonstypen nettverket er tiltenkt. De avgjørende valgene må tas med hensyn på topologitype, antall sensorer det må kunne håndtere, hvor mye *throughput* hver sensor krever, hvor geografisk spredt sensorene må plasseres, hvor lang levetid en ønsker for hver batteridrevne sensor og hvor lang latenstid en kan tålerere på en utsendt datapakke fra en sensor. Hensikten med dette kapittelet er å belyse de konsekvensene de ulike valgene vil ha på ytelsen til nettverket. Dette på bakgrunn av de teoretiske begrensningene et nettverk må forholde seg til beskrevet i kapittel 3.

Et særlig avgjørende valg er topologitype, som vil ha innvirkning på alle parametrene. Innenfor begge multihop-topologiene vil plasseringen av noder i forhold til hverandre ha avgjørende betydning og vies mye oppmerksomhet, men hovedvekt vil bli lagt på *mesh*-topologi. Kapittelet er dermed strukturert ut i fra de tre topologitypene; stjerne, *mesh* og tre.

5.1 Ytelsen til ZigBee-nettverk med stjernetopologi

Stjernetopologi er den enkleste topologien å analysere teoretisk. Den er også forholdsvis enkel å ha kontroll over i praksis. ZigBee-standarder legger opp til bruk av *beacons* og superrammestruktur for stjernetopologier. Med bare en koordinator, ZigBee-koordinatoren, vil sensorenhetene bare ha en *beacon* de trenger å forholde seg til og synkronisere seg mot, noe som gjør nettverket både forutsigbart og enkelt å implementere.

Begrensningene i et stjernenettverk har først og fremst sammenheng med løsningene i PHY og MAC laget, definert av IEEE 802.15.4. Nettverkslaget sin oppgave blir i hovedsak å starte nettverket, og deretter kunne koble enheter på og av ved hjelp av tjenester tilbudt av de underliggende lag. Noen form for stifinng er ikke nødvendig. Det finnes kun to alternativer, enten sende

til ZigBee-koordinatoren eller motta fra ZigBee-koordinatoren. Pakke-*overhead* produsert av nettverkslaget kan en se bort i fra i de fleste tilfeller.

5.1.1 Nodeplassering

Nodeplasseringen vil være avgjørende for ytelsen til et stjernenettnettverk. Den stokastiske naturen til radiokanalen, samt frittromsdempningen, vil begrense avstanden fra ZigBee-koordinatoren en enhet kan plasseres. I kapittel 3.1 brukes parameteren n som et mål på hvor mye signalet svekkes over avstanden r mellom to noder. For vanskelige omgivelser både innendørs og utendørs, gjerne med hindringer i veien, vil signalet kunne svekkes med en n på 4-6. I større prosessanlegg, der mange typer metaller og andre materialer i forskjellige størrelser og former preger omgivelsene, vil forholdene være vanskelige for radiosignaler. Da må en høy n påregnes. Jo lenger unna en node plasseres fra ZigBee-koordinatoren, jo dårligere link får en med et statistisk høyere antall retransmisjoner.

To noder med maks avstand mellom seg innenfor ZigBee-koordinatorens POS vil høyst sannsynlig ikke høre hverandre ved hjelp av CCA i CSMA-CA algoritmen,. Problemet med *hidden terminal* vil da kunne oppstå, se kapittel 3.7.3, noe som vil medføre kollisjoner og ytterligere forringelse av ytelsen. Dette kan forhindres ved å begrense det definerte dekningsområdet til ZigBee-koordinatoren slik at alle nodene innenfor vil kunne høre hverandres utsendte pakker.

5.1.2 Effektforbruk

Stjernetopologi muliggjør applikasjoner med svært lavt effektforbruk i de tilkoblede nodene. ZigBee-koordinatoren antas det her er tilkoblet strømmettet. Alle tilkoblede enheter trenger bare bruke effekt på å sende og motta data til og fra seg selv, og trenger ikke bruke effekt på videresending av data eller delta i noen form for stifinng for andre noder.

Uttrykk 3.5 illustrerer det som vil være det gjennomsnittlige effektforbruket til en tilkoblet enhets transceiver. Den viser at en betydelig faktor for total levetid for en node som følge av effektforbruket vil avhenge av lengden på en *duty-cycle*, α . Enheter kan ligge i søvn, våkne opp, synkronisere seg mot superrammen ved hjelp av en *beacon*, sende ut data, for så å gå til søvn igjen. Parameteren α avhenger i stjernenettnettverk svært mye av intervallengden mellom utsendte *beacons*. Det som til sist uansett vil avgjøre effektforbruket til nodene vil være kravet til samplingshastighet og latenstid. Med lave krav til samplingshastighet og latenstid kan enhetene tillates å sove i lange perioder siden et lang avstand mellom *beacons* kan velges.

I [17] er totalt strømforbruk til en ZigBee RFD estimert til $3\mu\text{A}$ ved standby, 39mA i *receive mode* og 34mA i *transmit mode*. Tilsvarende verdier for en ZigBee FFD er $3,5\mu\text{A}$, $41,5\text{mA}$, $36,5\text{mA}$, samt for en ZigBee-koordinator $3,5\mu\text{A}$, 48mA , og 43mA . ZigBee selv anslår generelt $1\mu\text{A}$ i *sleep mode* og 15mA i *transmit* og *receive*. Uansett hva endelige enheter vil bruke av strøm er det iallfall ikke tvil om at forskjellen mellom *sleep-mode* og aktiv tilstand(Tx/Rx) er så stor at *duty-cycle* størrelsen vil være av stor betydning. Når en nå begynner

å få ferdig utviklede integrerte enheter, som blant annet norske Chipcon har bidratt med, vil mer reelle estimat kunne gjøres.

5.1.3 Sikkerhet for at pakker kommer frem

Sikkerheten for at pakken kommer frem i et stjernenettverk avhenger fullt og helt av linken mellom enheten selv og ZigBee koordinatoren. Linken avhenger av mottatt signalstyrke og dermed avstanden og de fysiske omgivelsene mellom enheten og ZigBee koordinatoren. Er enheten i fart vil også dopplereffekten kunne påvirke linkkvaliteten, men slike betraktninger går utover denne rapportens omfang. Om en link til ZigBee-koordinatoren detter ut har ikke enheten andre valg enn å prøve igjen helt til kvaliteten er god nok for å oppnå kontakt. Dette vil medføre sløsing av effekt som kunne vært brukt til sending og mottagning av data. Dermed er det ekstremt viktig i stjernenettverk å plassere nodene med en viss feilmargin slik at ikke normale endringer i omgivelsene vil kunne påvirke linken over lengre tid

Ved korte intervall mellom utsendte *beacons*, og med mange enheter(15) på en ZigBee-koordinator, vil faren for at CSMA-CA algoritmen ikke treffer en ledig tidsluke i løpet av en CAP øke. Dette vil være et mindre problem for større *beacon* intervall, men kan for noen applikasjoner, der mye data skal inn til ZigBee-koordinatoren, medføre en svekkelse av ytelsen, særlig i form av økt latenstid.

5.1.4 Throughput

Ved bruk av superrammestruktur synkronisert av *beacons* vil enheter ikke kunne sende mer enn en dataramme i hver CAP-periode. Ved 2,4 GHz er det et krav at hver enhet må kunne sende 250 kb/s av rådata, medregnet all den *overhead* som er generert fra alle lag. I tillegg vil noe *overhead* som følge av sending av kontrolldata, *acknowledge*-pakker og lignende ta opp båndbredde. En øvre grense for *throughput* vil for hver enhet være av avhengig av hvordan superrammen er strukturert og hvor lang den ikke-aktive perioden er, se figur 4.3.

Båndbredden W bit/sekund defineres som mulig datarate i form av rådata mellom en ZigBee-enhet og ZigBee-koordinatoren, hvor det sees vekk i fra annen trafikk som for eksempel *ack*-pakker og kontrolldata. Ved bruk av *beacons* og superramme er et øvre mål for *throughput*, G_{\max} , for en enhet inn til ZigBee-koordinatoren uttrykt i uttrykk 5.1.

$$G_{\max} = W \cdot \frac{\text{Maksimalt tidsrom innenfor en tidsluke enheten kan sende}}{\text{Tidsintervallet mellom beacons}(\text{beaconOrder})} \quad (5.1)$$

Varigheten til ei tidsluke kommer an på lengden på CAP og hvor mange like store tidsluker den er delt inn i, *default* er 16, der første luken er forbeholdt utsendt *beacon*. I tillegg ser en ut i fra uttrykk 5.1 at større *beacon*-interval naturlig nok gir mindre *throughput* om lengden på CAP-perioden og størrelsen på tidslukene forblir den samme. I tillegg vil det nødvendigvis måtte være et

guard-intervall mellom hver tidsluke for sikre at ikke data fra to tidsluker interferer med hverandre, noe som vil avgjøre maksimalt tidsrom innenfor en tidsluke sending kan foregå. Størrelsen på tidslukene vil avgjøre *throughput* i stjernenettverk, og må dermed nøye planlegges ut i fra antall sensorer og hvor mye data som blir generert av dem til en hver tid. Samtidig er det viktig å gjøre en avveining mot sannsynligheten for pakkekollisjoner, som vil øke for jo færre og større tidsluker man benytter seg av. Dynamisk løsninger for å variere disse parametrene i real-time vil kunne være hensiktsmessig for noen applikasjoner der antall tilkoblede enheter kan variere.

I et stjernenettverk, der bare ZigBee-koordinatoren sender *beacons*, vil ikke en ikke-aktiv periode ha noe for seg i og med at ZigBee-koordinatoren i de fleste tilfeller vil være tilknyttet strømmettet. Det som derimot vil ta opp båndbredde er ugjennomtenkt mye bruk av GTSer, tidsluker for å garantere en tidsramme i hver superramme for viktige enheter. Simuleringer av IEEE 802.15.4 i ns-2 viser at bruk av GTSer en svært kostbar funksjonalitet og bør benyttes med varsomhet[13].

5.1.5 Latenstid

Latenstiden i et stjernenettverk er som for *throughput* sterkt avhengig av intervallet mellom *beacons*. En pakke kan kun sendes en gang for hvert intervall. Prosedyren for å sende data til en koordinator i et *beacon-enabled* nettverk består i å først synkronisere seg til neste utsendte *beacon*, for deretter sende i en tilfeldig valgt tidsluke ved hjelp av CSMA/CA. Dermed vil det verste tilfellet av latenstid oppstå når en enhet finner ut at den skal sende rett etter at ZigBee-koordinatoren har sendt ut en *beacon*, for deretter at enheten ikke får sende før i siste tidsluke i neste superramme. Dette forutsetter selvfølgelig det idealiserte tilfellet at enheten får sende i løpet av første mulige CAP-periode. Regner en derimot tidspunktet enheter ønsker å sende som tilfeldig, samt at tidslukene blir helt tilfeldig valgt av CSMA/CA algoritmen, så vil forventningsverdien for tiden det tar fra en enhet begynner sendeprosedyren til datene blir sendt til ZigBee-koordinatoren være på et *beacon*-intervall.

5.2 Ytelsen til ZigBeenettverk med *mesh*-topologi

Trådløse *mesh*-nettverk har vært utsatt for flere misforståtte ideer om at kapasiteten i form av *throughput* overstiger kapasiteten til en stjernetopologi. Det har blant annet blitt hevdet at kapasiteten til nettverket øker med antall noder. Argumentasjonen er begrunnet i den romlige utbredelsen som gjør at to noder på motsatte sider av nettverket vil kunne sende samtidig uten kollisjon. Da blir det faktum at store deler av transmisjonene i et multihoppnettverk består av videresending av data glemt, noe som vil spise opp vinningen en oppnår ved å kunne sende samtidig to steder i nettverket.

Mesh-topologi gir åpenbart spennende muligheter i forhold til nettverkets fleksibilitet, dets mulighet for å konfigurere seg selv, robusthet mot linkfeil, samt geografisk utbredelse. Begrensninger i forhold til *throughput*, latenstid og dets

kompleksitet som igjen kan gi en dramatisk økning i effektforbruk byr likevel på store utfordring som krever mange harde avveininger.

Det kreves mye av både PHY-, MAC- og nettverkslaget i et *mesh*-nettverk. Bruk av periodisk utsendte *beacons* og superrammestruktur er ikke lov som følge av topologiens utstrukturelle natur. Enheter sender når de ønsker og får aksess til kanalen med CSMA/CA uten tidsluker. Dermed er aksess til kanalen styrt av MAC-laget en avgjørende faktor for ytelsen. Nettverkslaget legger samtidig store begrensning på ytelsen om utstrakt bruk av stifinning som oversvømmer nettverket er nødvendig.

I dette kapitlet vil det bli lagt hovedvekt på de begrensninger deling på aksess til kanalen gir. Dette illustrer godt det som kan sees på som en øvre grense for *throughput* det er mulig å oppnå i en *mesh*-topologi med bare en sinknode, ZigBee-koordinatoren. Den øvre grensen vil tilsvare en stabil tidsperiode for nettverket der alle linkene fungerer som de skal, og hvor nettverkslaget ikke har satt i gang noen form for stifinning.

5.2.1 Begrensninger som følge av stifinningalgoritmen

ZigBee i *mesh*-topologi benytter seg som vist i kapittel 4.2.9 av AODV-algoritmen for ruting av pakker mellom enhetene. Denne algoritmen er mer kompleks og krever mer minne i enhetene med rutingkapasitet enn i tre-struktur. Måten den oversvømmer nettverket på når nye stier må finnes vil være en avgjørende negativ faktor for ytelsen. Riktig valg av rute, med bruk av akkumulert stikostnad med LQI som kvalitetsindikasjon, vil gi solide stier som vil minimalisere bruken av den kostbare stifinningsalgoritmen, noe som er bekreftet i [14, 21].

Simuleringer er en ofte benyttet metode for å analysere rutingalgoritmer. I [22, 21] blir AODV simulert og sammenlignet med blant annet DSR og noen andre ad hoc rutingalgoritmer. Hovedobservasjonen i [22] er at AODV utkonkurterer DSR i omgivelser med et stort antall noder hvor mye data blir generert. En av de største forskjellene til for eksempel DSR-algoritmen er at DSR lagrer multiple ruter i bufferet til en hver tid. Dette gjør at en linkfeil ikke nødvendigvis trenger medføre igangsettelse av en RREQ, slik det i AODV vil medføre. Likevel vil DSR, når den først blir satt i gang, være en mer omfattende prosedyre og kreve større minneplass, noe som er lite ønskelig i sensornettverk med batteridrevne noder. I [21] fokuseres det på å sammenligne energiforbruket til blant annet AODV og DSR, og det er klart at AODV-algoritmen vil fungere tilfredstillende i forholdsvis stabile omgivelser. Det henvises til kildehenvisningene for ytterligere detaljer.

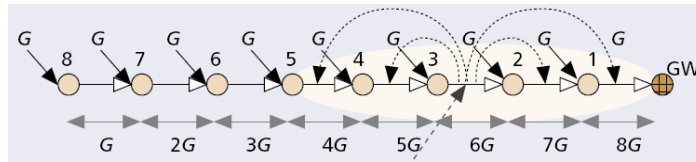
5.2.2 Throughput

Det aller meste av forskningsresultater på trådløse ad hoc nettverk fokuserer på det generelle tilfellet der trafikk flyter mellom tilfeldige noder i nettverket. Gupta og Kumar har bidratt med et klassisk resultat for slike nettverk [9], hvor det vises at *throughput* synker med $O(\frac{1}{\sqrt{n}})$ for store verdier av n , der n er antall noder. I et trådløst sensornettverk, og også de fleste andre typer nettverk, vil all trafikk være rettet mot en sinknode/gateway. For et slikt type nettverk

konkluderer Jun og Sichitiu [1] med at *throughput* synker med $O(\frac{1}{n})$ for hver node, noe som er et betydelig dårligere resultat enn det til Gupta og Kumar. I 3.7.3 ble kollisjonsområde og de begrensninger det medfører definert. Her brukes dette videre til å analysere større nettverkskonstellasjoner, og de begrensninger i MAC-*throughput* flaskehalskollisjonsområdet vil gi.

Maks *throughput* begrenset av kollisjonsområdet(MAC-laget)

Figure 5.1 viser en rekke av åtte noder som genererer og videresender trafikk til en *gateway*, GW. Det antas at hver node genererer samme mengde trafikk G og at hver node bare kan motta pakker fra nærmeste nabo.



Figur 5.1: En rekke med $n=8$ noder som genererer og videresender data til en *gateway*, GW, fra [1]

I første omgang så må den trafikken som må videresendes av hver link regnes ut. Det er tydelig fra figuren at noder som er nærmere *gateway*, sinknoden, vil måtte videresende mer trafikk enn nodene lenger vekk. Node 1 i figur 5.1 må videresende $n-1$ ganger trafikken den selv genererer, som tilsvarer at link 1-GW må kunne videresende en båndbredde på nG .

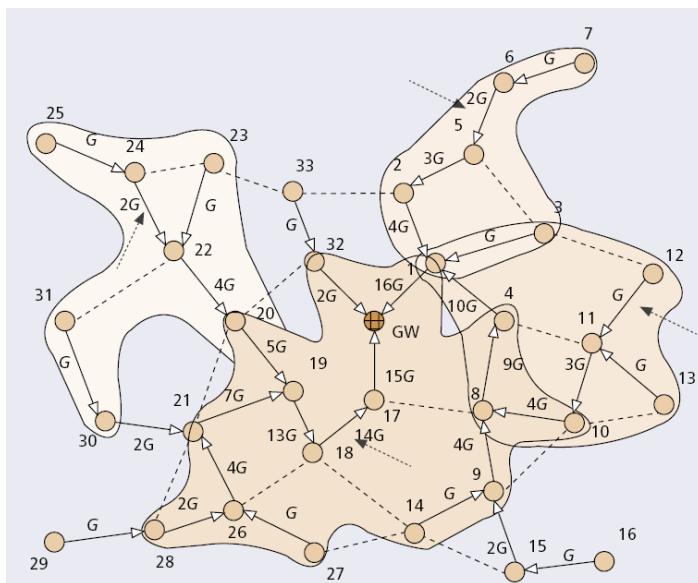
I andre omgang må *kollisjonsområdet* (*collision domain*), se kapittel 3.7.3, for hver link bestemmes. Forenklingene for CSMA/CA-algoritmen gjort i kapittel 3.7.3 antas heretter. De prikkede pilene i figur 5.1 viser de linkene hvor det ikke kan foregå transmisjon så lenge link 2-3 er aktiv. Da defineres og består *kollisjonsområdet* av linkene $\{2-3, GW-1, 1-2, 3-4, 4-5\}$. Hver link må ha kapasitet til å sende med en hastighet som tilsvarer summen av trafikken til hver link i sitt eget kollisjonsområde for at data ikke skal hope seg opp. Det vil si at link 2-3 må kunne videresende med en båndbredde på $4G + 5G + 6G + 7G + 8G = 30G$. Og hver link kan ikke videresende mer enn kapasiteten til MAC-laget B , definert i 3.7. Dermed vil det alltid finnes et flaskehals-*kollisjonsområde* som begrenser kapasiteten til hele nettverket.

Flaskehalskollisjonsområdet (*bottleneck collision domain*) defineres som det *kollisjonsområde* som må kunne videresende maksimalt med trafikk i nettverket.

For rekker, som vist i figur 5.1, med $n \geq 3$ kan det vises at *flaskehalskollisjonsområdet* alltid vil være link 2-3 fra eksempelet. Dermed, med $n=8$, vil tilgjengelig *throughput* for hver node i eksempelet, G_{\max} , være begrenset til $G_{\max} \leq B/30$. Genererer hver node en trafikk på G_{\max} må link 2-3 sende med en båndbredde på $30G_{\max}$, noe som må være mindre enn linkens makskapasitet på B .

Denne modellen kan enkelt utvides til to- eller tredimensjonale miljøer for å komme nærmere hvordan virkelige nettverk ser ut. I figur 5.2 er et mer

reellt trådløst *mesh*-nettverk vist. Den eneste forskjellen fra forrige eksempel er at *kollisjonsområdene* for hver link blir todimensionale. Hver node genererer trafikkmengden G , og må i tillegg videregjøre trafikk fra alle noder som er lenger vekk fra sinknoden, GW. Da gjenstår på samme måte som sist å identifisere linkene som er *flaskehals-kollisjonsområde*, som igjen vil avgjøre maksimalt tilgjengelig *throughput* for hver node, G_{\max} .



Figur 5.2: Et trådløst *mesh*-nettverk med vilkårlig plassering av noder, fra [1]

Det kan vises at linkene med *flaskehals-kollisjonsområdet* i figur 5.2 er 17-18. I figuren er tilhørende *kollisjonsområde* vist som det mørkere feltet link 17-18 befinner seg i midten av. Linkene *kollisjonsområde* består av er: (32-GW, 1-GW, 17-GW, 17-18, 18-19, 19-20, 19-21, 21-26, 26-28, 26-27, 9-14, 8-9, 4-8, 8-10). Teller en opp all trafikken som må transmitteres over *kollisjonsområdet* på samme måte som i forrige eksempel vil en få $97G$. Det vil si at maks tilgjengelig *throughput* G_{\max} for hver node er begrenset av $G_{\max} \leq B/97$. I et 2,4 GHz ZigBee *mesh* nettverk, kreves det at hver enhet skal kunne sende med en båndbredde på 250 kb/s. I eksempelet på figur 5.2 vil maksimalt utsendt båndbredde for hver enhet være på $250/97 = 2,58$ kb/s for at ikke nettverket skal gå i metning. Da er ikke *headere* og pakke-*overhead* fra retransmisjoner og AODV-algoritmen medregnet, noe som selvsagt vil begrense *throughput* enda mer. Nodeplassering vil dermed bli svært avgjørende for hvor nært opp til den maksimale båndbredden en kan komme.

I [1] er denne oppførselen til et *mesh*-nettverks *throughput* verifisert av simuleringer. Total *throughput* fra hver node øker lineært som følge av generert trafikk G fra hver node, helt til G_{\max} nås. Deretter vil *throughput* forholde seg konstant på G_{\max} uansett hvor mye G økes for hver node. I virkeligheten vil opphopning av pakker bli enda mer dramatisk, noe som særlig vil gjøre seg utslag på latenstid.

5.2.3 Batterilevetid

Batterilevetiden til ZigBee-enheter i en *mesh*-topologi avhenger av flere faktorer. For ZigBee-enheter av typen RFD, som ikke vil delta i noen form for ruting, vil en i stor grad kunne bestemme effektforbruket ut i fra uttrykk 3.5. Kravet til samplingshastigheten avgjør dermed i stor grad effektforbruket til endeenheter. Det at *mesh*-nettverk ikke bruker superrammestruktur gjør at RFDer bare kan våkne opp når de vil, sende sine data, for deretter gå tilbake til søvn igjen. Dermed er det i stor grad applikasjonstypen som vil avgjøre effektforbruket til en RFD sensor i et *mesh*-nettverk.

Batterikapasiteten til ZigBee-enhetene som deltar i ruting vil derimot være vanskeligere å estimere. I tillegg til at de selv kan være sensorer som genererer data, vil de både måtte videresende data for andre sensorer, samt delta i stifinnsing hver gang stifinningsalgoritmen blir kjørt i gang fra en annen ZigBee-ruter. I det hele tatt er det vanskelig å kunne se at en ZigBee-ruter vil kunne ha mulighet til å ligge i søvn i det hele tatt. Den må stå i *receive* til en hver tid når den ikke sender, i og med at stifinningsalgoritmen kan bli satt i gang av andre rutere når som helst. Dermed får α verdien 1 og batterilevetiden avhenger av effektforbruket til den integrerte radiokretsen, samt sensoren.

5.2.4 Batterilevetid som følge av nodeplassering - en kuriositet

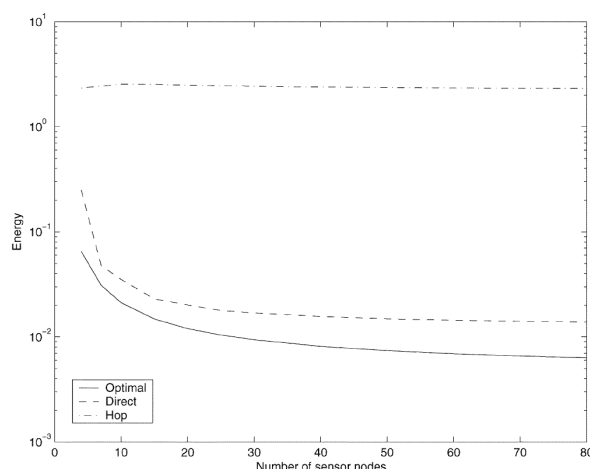
Med utgangspunkt i uttrykk 3.8 skal noen interessante resultater fra [15] presenteres. Resultatene baserer seg på en betraktning av trådløse sensornettverk der effektforbruket til hver node knyttes til effekten brukt til sensing, samt videresending av egen og andres trafikk. Teorien bak resultatene er betraktelig mer komplisert enn det som tas med her, og det vises til kildehenvisningen for ytterligere detaljer.

Modellen tar utgangspunkt i at utsendt effekt kan varieres for hver node, noe som innenfor regionale bestemmelser i utgangspunktet kan gjøres for ZigBee-enheter. I tillegg antas det at informasjonsraten kan varieres, noe som ikke er mulig i ZigBee foreløpig. Det er viktig å ha klart for seg denne forskjellen når en leser dette kapitlet. Dette er teoretiske resultat der informasjonsraten og transmisjonseffekten er gitt av Shannons kapasitetsuttrykk, og hvor begge parametre kan varieres for hver node. Dermed kan resultatet i utgangspunktet virke noe ulogisk å blande inn med ZigBee, der raten er fast og bestemt. Grunnen til at det likevel er tatt med er at det kan vise noe av potensialet for en videre utvikling av standarden. Dermed blir dette kapitlet mer en teoretisk kuriositet der det blir vist at det går an å finne en nedre grense for hvor lite energi det er mulig å bruke for å frakte en viss mengde informasjon fra et gitt sensornettverk.

Bakgrunnen for alle de grafene som her skal kommenteres er uttrykk 3.8, som kan optimaliseres ulinært for å finne optimalt og dermed minimalt energiforbruk med hensyn på informasjonsmengden som skal trekkes ut fra definerte nettverkskonstellasjoner og rutingmønstre.

Optimalt energiforbruk

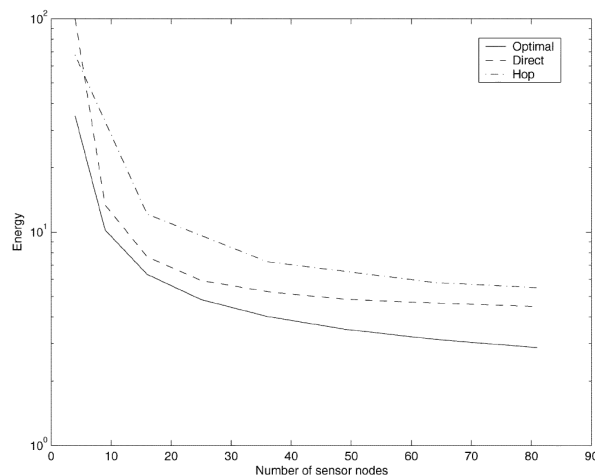
De første grafene, figur 5.3 og 5.4, har til hensikt å belyse konsekvensene to ulike og enkle måter å rute informasjon til sinknoden på, har for den totale energien brukt av nettverket for to ulike konstellasjoner. Disse to sammenlignes med et optimalt rutingmønster nodene seg i mellom, som gir minimalt totalt energiforbruk funnet ut i fra uttrykk 3.8. Den første teknikken, *direct*-transmisjon, tilater bare nodene å sende informasjonen direkte til sinknoden. Den andre, *hop*-transmisjon, ruter derimot all informasjon fra en node til nærmeste node i retning sinknoden, som vil si at videresending foregår. Den heltrukne linjen, *optimal*, representerer den optimale løsningen funnet i [15]. I dette eksempelet antas det at alle nodene genererer en like stor del av totalinformasjonen som skal inn til sinknoden.



Figur 5.3: Totalenergien brukt ved *optimal*-, *direct*- og *hop*-transmisjon i en lineær topologi som funksjon av antall sensornoder

Figur 5.3 viser resultatet fra en linær topologi, der nodene er plassert på ei rekke som i figur 5.1. 4 til 80 sensorernoder er plassert på en linje uniformt fordelt i en avstand 1 til 10 fra sinknoden. Energien brukt for hver av de tre måtene å sende trafikken på, *optimal*, *direct* og *hop*, til sinknoden er presentert som funksjon av antall noder. Den tydeligste observasjonen fra figur 5.3 er den dårlige ytelsen til *hop*-transmisjon. Årsaken er at i linjetopologi så vil all informasjon måtte rutes gjennom noden som er nærmest sinknoden. Denne noden blir da tvunget til å bruke en betydelig mengde energi for å sende denne informasjonen til sinknoden. Ved *direct* og *optimal* transmisjon vil transmisjonen være mer rettferdig fordelt mellom nodene siden ingen node må sende all informasjonen. Dette resultatet, sammenlignet med kapittel 3.4, der energibesparelser ved å unngå lange hopp blir diskutert, henger sammen med at det her genereres data i hver mellomliggende node i tillegg til variabel datarate nodene seg i mellom.

Figur 5.4 viser resultatet fra en kvadratisk topologi der 4 til 80 noder er uniformt fordelt på en rutenett $[0, 100] \times [0, 100]$ med sinknoden plassert i

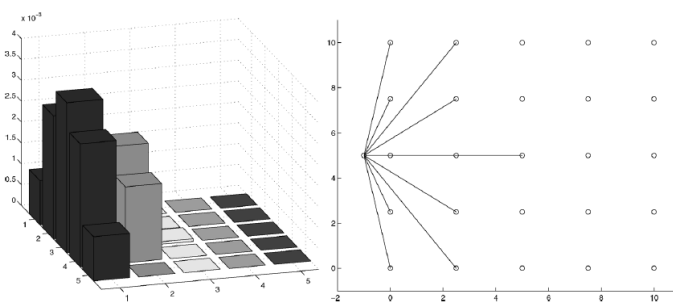


Figur 5.4: Totalenergien brukt ved *optimal*-, *direct*- og *hop*-transmisjon i en kvadratisk topologi som funksjon av antall sensornoder

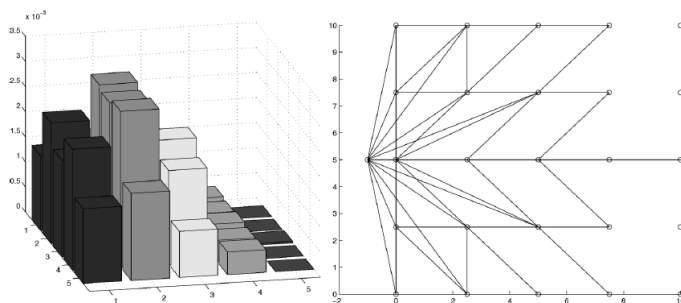
$[-30, 50]$. I dette eksempelet vil all informasjon måtte rutes til nodene på siden av kvadratet nærmest sinknoden og dermed unngår en til en viss grad flaskehalsproblemet for tilfellet med *hop*-transmisjon. Dermed yter *hop*-transmisjon bedre i dette tilfellet. Det er likevel interessant å se at *direct*-transmisjon yter bedre og ligger nærmere en optimal løsning i begge topologiene når variabel datarate er mulig.

Optimal energifordeling mellom nodene

Det andre resultatet har til hensikt å vise optimal energifordeling nodene seg i mellom i et *mesh*-nettverk, når hver node har en begrensning på hvor stor del av informasjonen de selv kan generere. Nettverket brukt i eksempelet består av 25 noder uniformt fordelt på et rutenett $[0, 10] \times [0, 10]$, der sinknoden er lokalisert i $(-3, 5)$. Det defineres en parameter α_i som definerer den andelen av total informasjon hver node i potensielt har lov til å generere.



Figur 5.5: Optimal energifordeling mellom nodene når hver node potensielt kan sende all informasjon som skal inn til sinknoden



Figur 5.6: Optimal energifordeling mellom nodene når hver node potensielt 5 % av total informasjon som skal inn til sinknoden

Figur 5.5 viser den optimale energifordeling når $\alpha_i = 1$ for alle i . Det vil si at hver node potensielt kan sende all informasjonen. Som figuren viser så er det optimalt å bruke flere noder, innenfor en gitt radius til sinknoden, til å sende hver sin forskjellige andel av informasjonsmengden. Det ville altså krevd mer om kun en node skulle sendt all informasjonen. I dette eksempelet er det optimalt at all informasjon rutet direkte til sinknoden, uten noen form for videresending.

Figur 5.6 viser den optimale energifordelingen når $\alpha_i = 0,05$. Det vil si at hver node bare kan sende maks 5% av den totale informasjonen som skal til sinknoden. Som kan sees på høyre halvdel av figuren har noe av informasjonen opphav så langt vekk fra sinknoden at det energimessig lønner seg å rute informasjonen gjennom andre noder.

Forklaringen på at det ikke lønner seg å sende all informasjon fra den nærmeste noden i første eksempel, samt at det lønner seg å rute informasjonen i eksempel to, kommer av at dette minimaliserer transmisjonskostnadene i den brukte modellen. Særlig innvirkning på om det lønner seg å rute informasjon versus å sende direkte har selvsagt frittromsdempningen som følge av avstanden til sinknoden.

I tillegg til å være en kuriositet kan disse eksemplene være enda en bekreftelse på betydningen av nodeplassering i et *mesh*-nettverk. Skal en oppnå minimalt effektforbruk i større *mesh*-nettverk vil en lite gjennomtenkt plassering av noder kunne få fatale konsekvenser med tanke på batterikapasiteten. Ut i fra figur 5.5 og 5.6, og som også er naturlig, vil det være mest hensiktsmessig å plassere noder som genererer mye informasjon nærmest mulig sinknoden. En node som genererer mye informasjon som må sendes over mange hopp mot sinknoden vil gi ringvirkninger for hele nettverkets ytelse.

5.2.5 Forventet latenstid

Latenstid i et ZigBee *mesh*-nettverk er meget vanskelig å forutsi, men det er klart at det ikke kan gis noen garanti for lav latenstid. Det er også klart at latenstiden er sterkt avhengig av størrelsen på nettverket og nodeplasseringen seg i mellom. Høy kollisjonsrate og dårlig kvalitet på linkene vil kunne øke latenstiden dramatisk som følge av stifinning og retransmisjoner, noe som vil

oppta tidsmessige ressurser som igjen vil gå ut over latenstiden.

En latenstid på $d \cdot h$ for h hopp med forsinkelsen d vil være ubrukelig i de fleste *mesh*-nettverk. I mindre lavratenettverk, med stabile omgivelser og linker, vil en gjennomsnittlig d likevel kunne gi mening, og latenstiden vil kunne approksimeres som $d \cdot h$. Antall hopp h vil også kunne variere i *mesh* og gjennomsnittsverdier vil da måtte brukes. I større nettverk vil $d \cdot h$ heller kunne fungere som et nedre mål for latenstid, om man måler parametrene under stabile omgivelser uten at retransmisjoner og stifinningsalgoritmer foregår.

5.2.6 Robusthet mot linkfeil

Økt sikkerhet for at pakker kommer frem er hovedmotivasjonen bak strukturer som *mesh*. Redundante stier mellom to noder i nettverket gjør det robust om en link skulle falle ut. Ved mange hopp mellom en ZigBee-enhet og ZigBee-koordinatoren øker sannsynligheten for at totallinken faller ut, som vist i uttrykk 3.7. Redundante stier gjør store geografisk utbredte nettverk robuste mot problemet med akkumulert feilsannsynlighet. Dette skjer på bekostning av økt kompleksitet, lavere *throughput*, høyere gjennomsnittlig latenstid og økt effektivforbruk.

5.2.7 Beacon requests i mesh

Det har i den senere tid, etterhvert som ZigBee-standarder er forsøkt implementert, blitt klart at *beacon requests* vil kunne bli et større bidrag av total trafikk i *mesh*-nettverk. Dette er ikke ønskelig på grunn av det ikkeinformative innholdet. *Beacon request* blir generert under tilkobling (*association*) ved aktivt skan for å finne passende ZigBee-ruter å koble seg på. Alle rutene innenfor en enhets rekkevidde vil motta denne forespørselen og sende en *beacon* tilbake med nødvendig informasjon om seg selv. I nettverk med mange enheter som kobler seg på og av i perioder vil denne trafikken kunne begrense ytelsen ytterligere.

5.3 ZigBeenettverk med tretopologi

I IEEE 802.15.4-spesifikasjonen [3] blir det hevdet at et *beacon-enabled* nettverk brukes for å støtte applikasjoner som krever lav latenstid, som for eksempel pc-tilbehør. For noen applikasjoner vil et *beacon enabled* trenettverk i det minste kunne gi mer forutsigbar latenstid enn ett *mesh*-nettverk. Men om latenstiden generelt sett vil bli kortere vil være avhengig av en rekke faktorer. I utgangspunktet er det ikke et krav i ZigBee-spesifikasjonen at trenettverk skal være *beacon-enabled*, men de to egenskapene blir stort sett nevnt i samme åndedrag. Dermed tas det her utgangspunkt i at all transmisjon er synkronisert og strukturert av *beacons* og superrammestruktur på samme måte som et stjernenettverk. Dette vil kreve nøye innbyrdes planlegging rutene seg i mellom for å unngå *beacon*-kollisjoner.

Superrammestruktur gjør både transmisjon og ruting forholdsvis forutsigbart i en trestruktur. I et trenettverk i perfekt harmoni, god nok linkkvalitet og perfekt planlegging av utsendte *beacons*, vil både *throughput* og latenstid

i stor grad være gitt av *beacon*-intervallengden og størrelsen på tidslukene. I dette kapitlet gis en kort gjennomgang av de viktigste ytelsesbegrensningene til trenettverk, samt noen vage forestillinger om forventet ytelse innenfor visse kriterier til omgivelsene.

5.3.1 Throughput

Som sagt vil *throughput* være avhengig av størrelsen på et *beacon*-intervall, samt størrelsen på tidslukene. Uttrykk 5.1 beskriver en øvre grense for *throughput* for et stjernenettverk. Dette uttrykker maks *throughput* inn til nærmeste ZigBee-ruter fra en enhet. I et perfekt dimensjonert trenettverk, uten retransmisjoner og kollisjoner, vil dette også være et uttrykk for *throughput* for et trenettverk. Dette fordi ZigBee-ruteren, som i et trenettverk kan betegnes som enhetens forelder, igjen vil kunne sende videre dataene med samme datarate til sin forelder igjen, helt til ZigBee-koordinatoren på toppen av hierarkiet nås.

Perfekt dimensjonering av et trenettverk er derimot et vanskelig kriterie å oppfylle. På samme måte som i et *mesh*-nettverk vil det også her kunne oppstå kollisjonsområder som vil fungere som flaskehalser. For ZigBee-rutere innenfor hverandres dekningsområde vil en transmisjon innenfor den enes tidsluke oppfattes som en opptatt kanal for linker inn til de andre rutere av CCA-funksjonen. Dermed vil maks *throughput* også i trenettverk kunne begrenses på samme måte som *mesh*-nettverk.

Samtidig er det også klart at superrammen definert av ZigBee-koordinatoren vil avgjøre maks kapasitet sinknoden klarer å betjene. En superramme inndelt i 15 tidsluker vil ikke klare å betjene mer enn 15 noder som sender med en datarate som krever en tidsluke per superramme om ikke data skal hope seg opp i nettverket. Dermed er det klart at om nettverket skal inneholde flere enn 15 noder må dataraten innrette seg deretter. Med 30 noder i et trenettverk vil maks datarate i forhold til uttrykk 5.1 for et stjernenettverk, der superrammen har 15 tidsluker, måtte deles på to. Det tas da utgangspunkt i det idealiserte tilfellet at hver node sender like mye data. Dette fordi samme antall tidsluker inn til ZigBee-koordinatoren må deles av dobbelt så mange noder, som i praksis vil si at transmisjonen må fordeles over to *beacon*-intervall. Med kollisjonsområder til stede vil selvsagt maks *throughput* begrenses ytterligere.

5.3.2 Batterilevetid

Som i en *mesh*-topologi vil ZigBee-rutere ha et mye høyere effektforbruk enn endeenhetene, også kalt løvnoder i en trestruktur. Til forskjell har ZigBee-rutere i et *beacon enabled* trenettverk mulighet for å bruke ikke-aktive perioder for dermed å oppnå en lavere *duty-cycle*. Igjen blir superrammestrukturen og *beacon*-intervallet svært avgjørende. Jo kortere superrammelengde i forhold til *beacon*-intervall, jo lengre ikke-aktiv periode og dermed lavere *duty-cycle*. Men lavere *duty-cycle* er ikke gratis. I uttrykk 5.1 ser en at effekten er lavere *throughput*, så her må det tas en nøye avveining.

Et trenettverk fungerer som et hierarki ut i fra ZigBee-ruteren. Det er klart at jo høyere opp i hierarkiet en ZigBee-ruter befinner seg, jo mer trafikk må

den kunne videresende. Siden *beacon*-intervallet og lengden på superrammen vil være lik for hele nettverket må de bestemmes ut i fra trafikkmengden ZigBee-koordinatoren må kunne håndtere. Det er ikke slik at den ZigBee-ruteren med maks dybde i nettverket kan velge å kjøre sin egen struktur på superrammen på grunn av at den bare skal håndtere trafikken fra en løvnode. Dermed vil det for store nettverk være meget begrenset hvor stor ikke-aktiv periode som kan tillates, selv ved forholdsvis lave datarater. Konsekvensen er at ZigBee-rutere også i trenettverk vil få så stort batteriforbruk på grunn av stor *duty-cycle* at batteridrift i mange tilfeller vil være lite hensiktsmessig. For spesielle applikasjoner med svært lave datarater, som for eksempel svitsjer som skal skrus av og på et par ganger i døgnet, kan ikke-aktiv periode utnyttes for trenettverk.

Endenoder av typen RFD vil i trenettverk, som vist i kapittel 4.2.10, kunne oppnå lavere *duty-cycle* enn i et *mesh*nettverk. Dette på grunn av oppvåkning, synkronisering og sending av en datapakke, og deretter gå til søvn igjen, vil kunne skje raskere. Igjen forutsetter dette at *beacon*-intervallet settes til et minimum, noe som vil føre til høyere effektforbruk i ZigBee-rutene.

5.3.3 Forventet latenstid

En latenstid på $d \cdot h$ for h hopp med forsinkelsen d vil under visse forhold være et brukende resultat for trenettverk. Det er under forutsetning om at nettverket er i en ideell tilstand, der alle linker fungerer og minimalt med kollisjoner oppstår, samt at det er dimensjonert riktig slik at det ikke foregår opphopning av pakker noe sted. Et trenettverks struktur av transmisjoner, samt forutsigbarhet med tanke på hvor pakker rutes, gjør latenstiden mer forutsigbar enn i *mesh*.

For å bestemme forsinkelsen d for hvert hopp er igjen *beacon*-intervallet avgjørende. Lengden på et *beacon*-intervall vil, som i et stjernenettverk, utgjøre et gjennomsnittlig minstemål for latenstiden mellom to enheter. Med *beacon*-intervall på over 250 sekund sier det seg selv at lang latenstid på påregnes. Her blir det igjen en avveining mellom effektforbruk og latenstid. Fordelen i trenettverk er at en har muligheten til å gjøre denne avveiningen og dermed ha en viss kontroll.

5.3.4 Lite robust mot linkfeil

Et trenettverk er minimalt robust mot oppståtte linkfeil. Store deler av trafikken inn til ZigBee-koordinatoren går over linkene øverst i hierarkiet. Skulle noen av disse dette ut vil dette påvirke en stor prosentdel av total trafikk. Dette vil kunne gi katastrofale konsekvenser for applikasjoner innen for eksempel overvåkning og kontroll. Dermed er det svært viktig at nodeplasseringen nær sinknoden blir gjort med omhu. Det bør stilles strenge krav til robuste linker jo nærmere sinknoden en kommer.

Kapittel 6

Konklusjon og videre arbeid

6.1 Konklusjon

ZigBee-standarder tilbyr tre ulike nettverkstopologier med hver sine fordeler og ulemper. For små enkle nettverksformasjoner som stjerne vil ytelsen være forholdsvis enkel å forutse. For større topologier av typen tre eller *mesh* er det derimot svært vanskelig å forutse endelig ytelse før en har ferdig utviklede produkt som kan testes. I denne rapporten er det forsøkt å gi et bilde av hvordan de viktigste teoretiske begrensningene påvirker trådløse sensornettverk for de tre ulike topologitypene. Det er klart at for ønskede applikasjoner må det gjøres en rekke viktige avveininger særlig med tanke på batterilevetid versus ytelse, i form av *throughput* og latenstid.

Den klareste begrensningen for tre og *mesh* oppstår i MAC-laget fordi all transmisjon foregår over samme frekvens. Denne rapporten viser at ytelsen, som følge av dette, i stor grad vil være avgjort av datamengden nodene genererer, antall noder, nodeplassering og nodetetthet. Det kan ikke foregå transmisjon over linker i nærheten av hverandre samtidig, noe som medfører *flaskehalskollisjonsområder* som setter en øvre grense for *throughput* for hver enhet. Dette begrenser særlig hvor tett en kan plassere nodene. En annen stor begrensning kommer av at enheter nær sinknoden vil måtte videresende all trafikk generert lenger ute i nettverket, noe som kan gi flaskehalser og dermed sette begrensninger på antall noder.

AODV-algoritmen vil begrense ytelsen til *mesh*-nettverk ytterligere. Algoritmen vil ta opp store ressurser både i form av *throughput*, latenstid og batterikapasitet avhengig av kvaliteten på linkene. Igjen blir plassering av nodene sentralt. Med for mange hopp mellom en enhet og ZigBee-koordinatoren vil høy aggregert feilsannsynlighet øke sjansen for at rutingalgoritmen blir igangsatt.

For trådløse sensornettverk innen kontroll og overvåking av prosessanlegg, hvor ABB ser for seg mulige anvendelser, vil en *mesh*-topologi likevel være interessant. Særlig med tanke på robusthet mot linkfeil. Et hovedproblem som må løses er strømtilførsel. Det er klart at enheter som deltar i ruting vil måtte stå i *receive mode* til en hver tid, noe som medfører ett høyt effektforbruk. Med mange noder vil vedlikehold av et slikt nettverk bli for stort. En fornuftig løsning vil være å la alle enheter med rutingkapasitet være tilknyttet strømmettet

og dermed utgjøre et kjernenettverk som definerer et dekningsområde. Innenfor dekningsområdet kan batteridrevne RFD-enheter med sensorer settes inn hvor som helst. Dette vil gi en noe mer tidkrevende installasjon med høyere kostnader, men vedlikeholdskostnadene vil reduseres betraktelig noe som vil bli en svært viktig faktor.

Det er ikke tvil om at ZigBee er en svært lovende standard som åpner for helt nye muligheter innen trådløse applikasjoner. Dens suksess vil være avhengig av at gode produkter kommer raskt ut på markedet og at forventninger innfris. Denne rapporten går langt i å antyde at forventningene kan være satt for høyt. ZigBee alliansen kan gjøre seg tjent med en forventningsavklaring og ikke presentere tall som beskriver nettverk med opptil 65000 noder og batterilevetid på oppmot 10 år i samme åndedrag, slik en gjerne ser i flere av deres presentasjoner. Dette kan gi falske forventninger og umiddelbare skuffelser når produkter nå etter hvert kommer på markedet.

6.2 Videre arbeid

Dette prosjektet har på mange måter funnet som et forprosjekt for et videre arbeid som skal gjøres for ABB til våren 2005 i form av en masteroppgave. Der vil hovedfokus være på avveininger mellom batterilevetid og latenstid på et nettverk med ca. 200 noder. Dette arbeidet vil blant annet bestå av simuleringer i NS-2, med forhåpninger om å i samarbeid med ABB få testet resultatene i en reell testbed som er under oppsetning i Sverige.

Bibliografi

- [1] J. Jun og M. L. Sichitiu. 'The Nominal Capacity of Wireless Networks'. *IEEE Wireless Communication*, Oktober 2003.
- [2] Venkat Bahl. 'ZigBee Overview'. <http://www.zigbee.org/documents/ZigBeeOverview4.pdf>, september 2002.
- [3] 802.15.4 IEEE Standard for Information Technology-Part 15.4:. 'Wireless Medium Access Control(MAC) and Physical Layer(PHY) specifications for Low Rate Wireless Personal Area Networks(LR-WPANS)'. *IEEE Standard*, 2003.
- [4] William C. Craig. 'Zigbee: Wireless Controll That Simply Works'. http://www.zigbee.org/resources/documents/2004_ZigBee_CDC-P810_Craig_Paper.pdf.
- [5] Jr. Edgar H. Callaway. '*Wireless Sensor Networks: Architectures and Protocols*'. CRC Press LLC, Boca Raton, Florida, 2004.
- [6] Anna Hac. '*Wireless Sensor Network Designs*'. John Wiley & Sons Ltd, The Atrium, West Sussex, England, 2003.
- [7] 'The Zigbee Alliance'. <http://www.zigbee.org>.
- [8] ZigBee Alliance. 'Network Specification Revision 9'. Draft Version 0,92, 2004.
- [9] P. Gupta og P. R. Kumar. 'The Capacity of Ad Hoc Wireless Networks'. *IEEE Trans. Info. Theory*, vol. 46, Mars 2000.
- [10] U. Bilstrup, K. Sjøberg, B. Svensson, P. A. Wiberg. 'Capacity Limitations in Wireless Sensor Networks'. *Emerging Technologies and Factory Automation. EFTA '03. IEEE Conference*, 2003.
- [11] USC Information Sciences Institute, Marina del Rey, CA. 'The Network Simulator - NS2'. <http://www.isi.edu/nsnam/ns>.
- [12] M. L. Sichitiu J.Jun, P.Peddabachagari. 'Theoretical Maximum Throughput of IEEE 802.11 and Its Applications'. *Proc. 2nd IEEE Int'l. Symp. Net. Comp. and Applications*, April 2003.
- [13] Jianliang Zheng and Myung J. Lee. 'A Comprehensive Performance Study of IEEE 802.15.4'. *IEEE Press Book*, 2004.

- [14] D. S. J. De Couto, D. Aguayo, B. A. Chambers, R. Morris. 'Performance of Multihop Wireless Networks: Shortest Path is Not Enough'. *M.I.T. Laboratory for Computer Science*, (*fdecouto, aguayo, bac, rtmg*)@lcs.mit.edu.
- [15] B. Krishnamachari F. Ordóñez. 'Optimal Information Extraction in Energy-Limited Wireless Sensor Networks'. *Selected Areas in Communication, IEEE Journal*, 2004.
- [16] ISO/IEC 7498-1. Open systems interconnection - Basic reference model: The basic model, 1994.
- [17] Roke Mansor Research. 'ZigBee: Technical Information'. http://www.roke.co.uk/download/datasheets/zigbee_tech.pdf.
- [18] C. E. Perkins og E. M. Royer. 'Ad Hoc On-demand Distance Vector Routing'. *Proc.2nd IEEE Wksp. Mobile Comp. Sys. and Apps.*, February 1999.
- [19] C. E. Perkins, E. M. Royer, S. R. Das. 'Ad Hoc on Demand Distance Vector (AODV) Routing'. <http://www.ietf.org/internet-drafts/draft-ietf-manet-aodv-06.txt>, July 2000.
- [20] J. Broch, D. Johnson, D. Maltz. 'The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks'. <http://www.ietf.org/internet-drafts/draft-ietf-manet-dsr-03.txt>, October 1999.
- [21] A. Misra S. Bansal, R. Shorey. 'Comparing the Routing Energy Overheads of Ad-Hoc Routing Protocols'. *IEEE Personal Communication*, 2003.
- [22] S. R. Das M. K. Marina C. E. Perkins, E. M. Royer. 'Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks'. *IEEE Personal Communication*, February 2001.