

Toni Puhakka

## **5G NR UL/DL MATLAB -MALLIN INTEGROINTI JA AUTOMATISOINTI**

# **5G NR UL/DL MATLAB -MALLIN INTEGROINTI JA AUTOMATISOINTI**

Toni Puhakka  
Opinnäytetyö  
Syksy 2022  
Tietotekniikan tutkinto-ohjelma  
Oulun ammattikorkeakoulu

## TIIVISTELMÄ

Oulun ammattikorkeakoulu  
Tietotekniikan tutkinto-ohjelma, ohjelmistokehityksen suuntautumisvaihtoehto

---

Tekijä: Toni Puhakka

Opinnäytetyön nimi: 5G NR UL/DL Matlab -mallin integrointi ja automatisointi

Työn ohjaajat: Kirsti Simula, Teemu Korpela

Työn valmistumislukukausi ja -vuosi: Syksy 2022

Sivumäärä: 29

---

Tämän opinnäytetyön tavoitteena oli integroida olemassa olevan 5G NR UL/DL Matlab -malli Nokian kehittäjäympäristöön sekä automatisoida mallin toiminta. Tavoitteena oli yhdistää malli toimimaan automaattisesti olemassa olevien systeemin komponenttitestien kanssa sekä laajentaa mallin toimintaa tallentamaan vianmäärittystietoja komponenttien väliltä. Työn toimeksiantajana toimi Nokia Oyj.

Toteutus alkoi perehtymällä 5G NR:n toimintaan ja olemassa olevaan malliin. Opiskelun ohessa laajennettiin mallin toiminnallisuutta lisäämällä tukea uusille tiedostomuodoille. Tämän jälkeen aloitettiin Matlabin liittäminen Nokian kehittäjäympäristöön sekä mallin toiminnan automatisointi. Lopuksi malli automatisoitiin tuottamaan halutuille testitapauksille vertailudata ja luomaan vianmäärittystiedot käyttäjän niin halutessa.

Lopputuloksena Matlab saatiin liitettyä Nokian kehitysympäristöön sekä mallin toiminta saatiin automatisoitua sekä laajennettua malliin halutut lisätoiminnot.

---

Asiasanat: 5G, Tukiasemat, Matlab, Simulink, Mallipohjainen suunnittelu, Digitaalinen kaksonen

## ABSTRACT

Oulu University of Applied Sciences  
Degree Programme in Information Technology, Option of Software Development

---

Author: Toni Puhakka  
Title of thesis: 5G NR UL/DL Matlab -mallin integrointi ja automatisointi  
Supervisors: Kirsti Simula, Teemu Korpela  
Term and year when the thesis was submitted: Fall 2022  
Number of pages: 29

---

The goal of this thesis work was to integrate Matlab software to Nokia's developer environment and automatize the existing 5G NR UL/DL Matlab downlink part of the model to work in Linux environment without graphical interface. 5G NR UL/DL Matlab model is so called "Digital Twin", and it's part of Model Based Engineering, where exact copy on product is replicated virtually.

The thesis work did go as planned. As the result of this work, the Matlab software was integrated to Nokia's development environment, and the model was automated to work with only one terminal command. After this thesis work continues further development of the model and expanding its functionality to Uplink side.

---

Keywords: 5G, Base stations, Matlab, Simulink, MBSE, Digital twin

# SISÄLLYS

SANASTO.....	6
1 JOHDANTO.....	7
2 VIRTUAALISET MALLIT JA TUKIASEMAN TOIMINTA .....	8
2.1 Virtuaaliset mallit ja mallipohjainen suunnittelu .....	8
2.2 Tukiaseman toiminta .....	9
2.3 5G NR:n eri viestityypit.....	11
2.4 Keilanmuodostus.....	14
2.5 Systemin komponenttitestaus ja 5G NR UL/DL Matlab -malli.....	16
3 TYÖN VAIHEET .....	18
3.1 5G NR UL/DL Matlab -mallin vaatimukset.....	18
3.2 Työn aloitus ja mallin esittely.....	18
3.3 Matlabin asennus kehittäjäympäristöön.....	20
3.4 5G NR UL/DL Matlab -mallin automatisointi.....	21
3.5 5G NR UL/DL Matlab -mallin vianmääritys tietojenkeruu.....	24
4 YHTEENVETO .....	25
4.1 Ongelmakohtia .....	25
4.2 Yhteenveto .....	26
4.3 Jatkokehitys.....	27
LÄHTEET.....	28

## SANASTO

5G	Viidennen sukupolven radioverkkoteknologia.
5G NR	5G New Radio. 5G tukiasema.
DL	Downlink. Tukiasemalta matkapuhelimeen päin tapahtuva tiedonsiirto
Fronthaul	O-RU:n ja O-DU:n välinen yhteys(kaapeli).
IQ	In-phase and quadrature components. Tapa pakata ja lähettää tietoa radiotaajuuksilla.
O-CU	Oran Control Unit. Ohjainyksikkö.
O-DU	Oran Distributed Unit. Hajautettu yksikkö.
OFDMA	Orthogonal Frequency Division Multiplexing. Tiedonsiirtoa yhtäaikaisesti usealla eri taajuudella.
ORAN	Open Radio Access Network. Yhdistys, joka on vastuussa yhteisistä ORAN-standardeista.
O-RU	Oran-Radio Unit. Radioyksikkö
PRB	Physical Resource Blocks. Palanen aika- taajuusvaruudesta, käytetään tiedonsiirrossa.
SCT	System Component Test. Testit, jolla testataan yhtä tai useampaa komponenttia.
UL	Uplink. Matkapuhelimesta tukiasemalle päin tapahtuva tiedonsiirto

# 1 JOHDANTO

5G:n myötä myös tukiasemilta ja radioilta vaaditaan yhä enemmän toimintoja ja nopeutta, mikä tekee niiden toiminnasta aina vain monimutkaisempia. Tämä aiheuttaa haasteita tuotteen suunnittelussa, varsinkin kun vaatimukset ovat yhä enemmän hajautuneet monelle osa-alueelle ohjelmiston ja fyysisten komponenttien välille.

Tätä varten mallipohjainen suunnittelu on alkanut saamaan suosiota. Nokia on alkanut panostamaan mallipohjaiseen suunnitteluun apuna ohjelmistokehityksessä. Nokialla oli aloitettu tekemään mallia tukiaseman datan lähetyksen suuntaan jo ennen opinnäytetyön aloittamista. Tätä mallia kutsutaan tässä opinnäytetyössä 5G NR UL/DL Matlab -malliksi. 5G NR UL/DL Matlab -malli on ns. digitaalinen kaksonen (Digital Twin) eli tuotteen tarkka digitaalinen kopio 5G NR -tukiasemasta.

Tämän opinnäytetyön tarkoituksena on integroida 5G NR UL/DL Matlab -malli Nokian kehittäjäympäristöön sekä automatisoida malli toimimaan olemassa olevien systeemikomponenttien kanssa.

Opinnäytetyö toteutetaan Matlabilla ja siihen kuuluvalla Simulink-lisäosalla, joka on suosittu työkalu mallipohjaisessa suunnittelussa. Matlab on laskentaan erikoistunut ohjelma, joka on erityisesti suunniteltu matriisilaskentaan ja algoritmien toteuttamiseen (1). Simulink on Matlab-pohjainen lisäosa, joka mahdollistaa mallin tekemisen graafisella työympäristöllä, joka helpottaa hahmotusta eri osien ja niiden yhteyksien välillä (2).

## 2 VIRTUAALISET MALLIT JA TUKIASEMAN TOIMINTA

### 2.1 Virtuaaliset mallit ja mallipohjainen suunnittelu

Mallipohjaisessa suunnittelussa rakennetaan tarkka malli tuotteesta virtuaalisesti. Tämä tarkoittaa sitä, että suunniteltavasta järjestelmästä rakennetaan suoritettava malli, joka vastaa mahdollisimman paljon todellisuutta järjestelmälle asetettavien vaatimusten pohjalta. Tätä suoritettava virtuaalista mallia voidaan ajaa rakennettavaa järjestelmää vastaavassa virtuaaliympäristössä jo huomattavasti aikaisemmin kuin varsinaista järjestelmää on edes olemassa. (3.)

Tämä helpottaa tuotteen suunnittelua ja toteutusta sekä sillä saadaan järjestelmän monimutkaisuutta paremmin hallittavaksi. Mallia voidaan käyttää apuna hahmottamaan tuotteen eri osia ja niiden välisiä suhteita, kun se voidaan esittää visuaalisessa muodossa. Myös kontrollirakenteiden ja eri komponenttien välinen tiedon kulku pystytään visualisoimaan mallissa.

Mallipohjainen suunnittelu liittyy oleellisena osana Digital Twin -konseptia, joka on virtuaalinen malli ja jolla voidaan mallintaa tarkasti fyysisen laitteen toimintaa. Digital Twin eroaa simulaatiosta laajuudellaan. Siinä missä simulaatio keskittyy yleensä yhden toiminnan mallintamiseen, niin Digital Twin on virtuaalinen ympäristö, joka sisältää useita simulaatioita, jotka vaikuttavat toisiinsa. Virtuaalista mallia voidaan tutkia ja sen suorituskykyä voidaan mitata, jolloin tuotetta voidaan testata ja muuttaa suunnitelmia havaittujen tulosten perusteella. (4.)

Mallipohjaisella suunnittelulla ja virtuaalisen ympäristön avulla voidaan tutkia tuotteen toimintaa eri olosuhteissa ja kokeilla eri lähestymistapoja, ilman fyysistä tuotetta. Mallilla voidaan mahdollisesti huomata vikoja suunnittelussa ja korjata ne ennen varsinaisen kehitystyön aloittamista. Mallia voidaan käyttää apuna tuotteen testauksessa, jolloin verrataan mallin ja itse tuotteen tuloksia keskenään ja näin ollen vahvistetaan tuotteen toimivuus. Malli auttaa arvioimaan tuotteen tarvitsemää laskentatehoa, virrankulutusta ja ajoituksia. (3.)

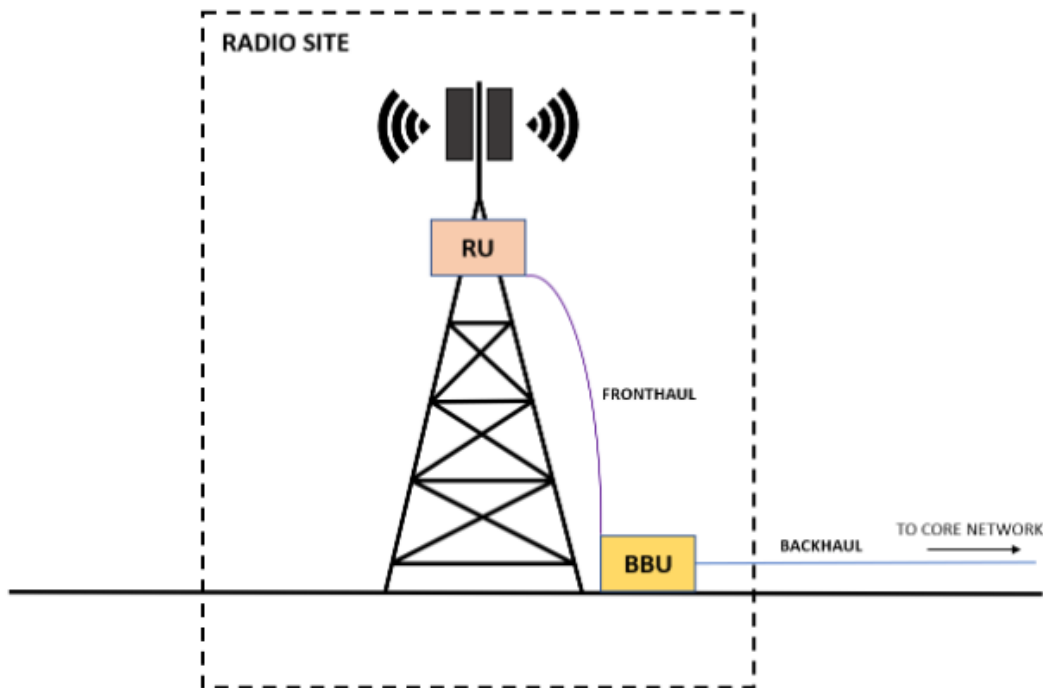
Mallista voidaan ottaa myös tiedot talteen jokaisen komponentin suorittaman operaation jälkeen, mikä auttaa vahvistamaan eri komponenttien toiminnan ja auttaa vikojen paikantamisessa. Mallilla



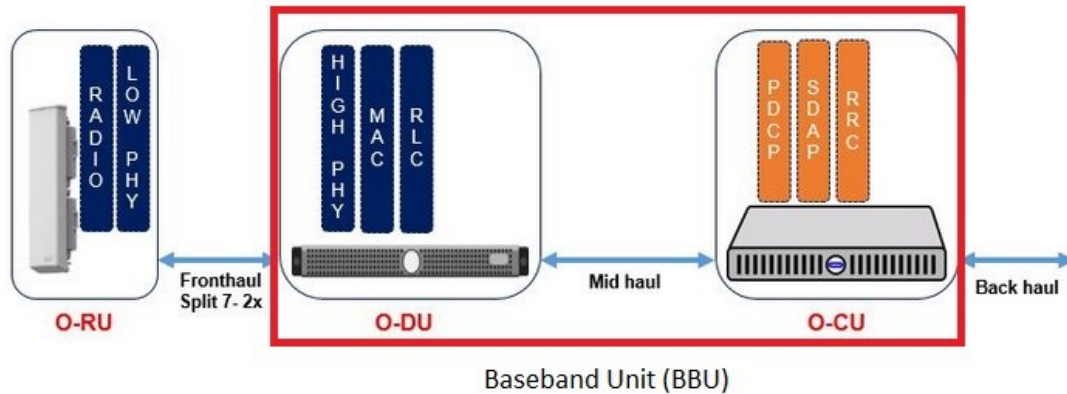
voidaan toistaa testejä nopeasti ja edullisesti. Lisäksi malli auttaa kehittäjätiimien ja muiden sidosryhmien välistä kommunikaatiota, koska mallin ajonaikainen simulaatio on havainnollinen ja helposti ymmärrettävissä ilman yksityiskohtaista tietämystä mallin kohteena olevasta järjestelmästä.

## 2.2 Tukiaseman toiminta

5G NR UL/DL Matlab -malli perustuu ORAN 7-2x -jakoon. ORAN-organisaatio pyrkii toiminnallaan tuottamaan yhteneväistä tapaa jaotella toiminnollisuuksia radioyksikön (O-RU) ja hajautetun yksikön (O-DU) välillä. 5G NR UL/DL Matlab -malli on virtuaalinen kopio kuvassa 1 näkyvässä RU-antenniyksiköstä. Kuva 2 tarkentaa O-DU:n ja O-CU:n suhdetta.



KUVA 1. Komponenttien sijainti tukiasemalla (5, s. 16)



KUVA 2. O-RU:n, O-DU:n ja O-CU:n välinen rakenne (6)

ORAN 7-2x:stä lähtien entistä enemmän toiminnollisuuksia siirtyy radioyksikölle, minkä takia laitteen arkkitehtuuri monimutkaistuu. Tätä monimutkaisuuden ymmärtämistä ja hallintaa varten 5G NR UL/DL Matlab -mallia kehitetään.

O-DU:n ja O-RU:n toiminnan jakamisessa on kaksi kilpailevaa intressiä:

1. On hyödyllistä pitää antenniyksikkö mahdollisimman yksinkertaisena, kevyenä ja virtaa säästävänä. Mitä enemmän toiminnallisuutta siirretään antenniin, sitä painavampi siitä tulee.
2. Mutta mitä enemmän toiminnallisuutta antenniyksikössä on, sitä vähemmän tarvitaan tiedonsiirtoa O-RU:n, O-DU:n ja muiden yksiköiden välillä.

Tämän ratkaisemiseksi ORAN 7-2x -jako määrittelee O-RU:lta vaaditun ulostulon, mutta sallii eri variaatioita O-RU:n toiminnallisuudesta. O-RU kategoria A:ssa järjestelmään tulevalle tiedolle ei tehdä erillistä esilaskentaa, vaan O-RU suorittaa vain saadun tiedon pohjalta digitaalisen ja analogisen keilanmuodostuksen. O-RU kategoria B:ssä O-RU hoitaa tiedon esilaskennan sekä keilanmuodostuksen. Tässä työssä esitelty 5G NR UL/DL Matlab -malli pohjautuu kategoria A:han (kuva 3).

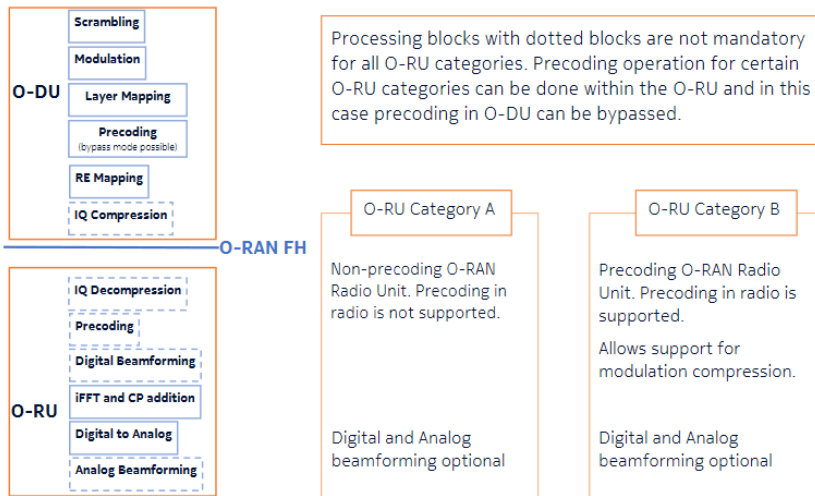


Figure 2-2 : Split Point and Category A and Category B O-RAN Radio Units

### KUVA 3. ORAN 7-2x -kategoria A:n ja B:n eroavaisuudet (7, s, 20)

ORAN 7-2x -jaon tarkoitus on vähentää tietoliikennettä antenniyksikön ja ohjainyksikön välillä siirtämällä osan ohjainyksikön toiminnasta antenniin. Tämä tekee antenniyksiköstä painavamman, koska se sisältää enemmän elektroniikkaa mutta samalla vähentää yksiköiden välille tarvittavaa kaapelointia.

## 2.3 5G NR:n eri viestityypit

Tämä ja seuraava kappale toimivat teoreettisena johdantona tämän opinnäytetyön varsinaiseen työn osuuteen ja työn kohteena olevaan 5G NR UL/DL Matlab -malliin.

5G NR:ssä kulkee neljä erilaista viestityyppiä.

- **S-Plane** = Synchronisation Plane = Synkronointiviestit. Näillä viesteillä synkronoidaan O-RU:n ja O-DU:n toiminta keskenään (7, s, 186).
- **M-Plane** = Management Plane = Hallinnointiviestit. Nämä ovat ei-reaaliaikaisia viestejä, joilla ylläpidetään O-RU:n toimintaa. Tähän liittyvät laitteen alustukset, vikatilanteet, suorituskyvyn ja turvallisuuden hallinta (7, s, 268).
- **C-Plane** = Control Plane = Ohjausviestit. Ohjausviestit sisältävät tarvittavat tiedot käyttäjäviestien käsittelyyn. Näihin kuuluvat viestien ajoitustiedot ja keilanmuodostus käskyt (7, s, 66).

- **U-Plane** = User Plane = Käyttäjaviestit. Käyttäjaviestit sisältävät itse datan, joka kulkee käyttäjältä systeemimoduulille ja toisinpäin (7, s, 159).

Tässä opinnäytetyössä keskitytään käyttäjä- ja ohjaustiedon mallintamiseen. Edellä esitetyt User Plane- ja Control Plane -viestit sisältävät kaksiosaisen tunnisteosan. Ensimmäinen osa sisältää eCPRI- tai IEEE 1914.3 -tunnisteen, jolla voidaan määrittää viestin tyyppi. Toinen osa sisältää tarvittavat tiedot viestin kontrollointiin ja synkronointiin.

User Plane- ja Control Plane -viestit lähetetään vakiolla Ethernet-kehyksellä, jonka kuva 4 havainnollistaa.



Figure 3-1 : Native Ethernet frame with VLAN

KUVA 4. Ethernet-kehiksen rakenne (7, s. 52)

Käyttäjaviestit lähetetään pakattuna IQ-datana (In-phase and Quadrature) (8). Tämän takia data joudutaan purkamaan ennen sen käyttöä. Käyttäjaviestin suurin pakettikoko on 1500 tavua tavallisella kehyksellä tai 9000 tavua jumbokehyksellä. Näin ollen yhden symbolin IQ-data voidaan joutua lähettämään useassa Ethernet-kehyksessä.

IQ-data lähetetään fyysisinä resurssilohkoina (Physical Resource Block). Resurssilohko koostuu 14:stä OFDMA-symbolista, joissa jokaisessa on 12 apukantoaaltoa, joita käytetään lähetysten ja vastaanoton aikataulutukseen. Resurssilohkoja käytetään varsinaisen tiedon lähettämiseen ja vastaanottamiseen. Resurssilohkon bittimäärä voi vaihdella, mutta jos IQ-data on lyhyempi kuin resurssilohko, niin täytetään loppuosa 0-bitteinä, kunnes resurssilohkon vaatima bittimäärä on saavutettu.

Resurssilohko on palanen aika-taajuusvaruudesta, joka on havainnollistettu kuvassa 5. Kuvassa 5 näkyvä Frame on 5G:ssä aina 10 millisekuntia ja Slot on aina 1 millisekuntia (9). Resurssilohkojen määrä symbolissa vaihtelee taajuuden mukaan.



KUVA 5. Yhden radiokehiksen rakenne

```

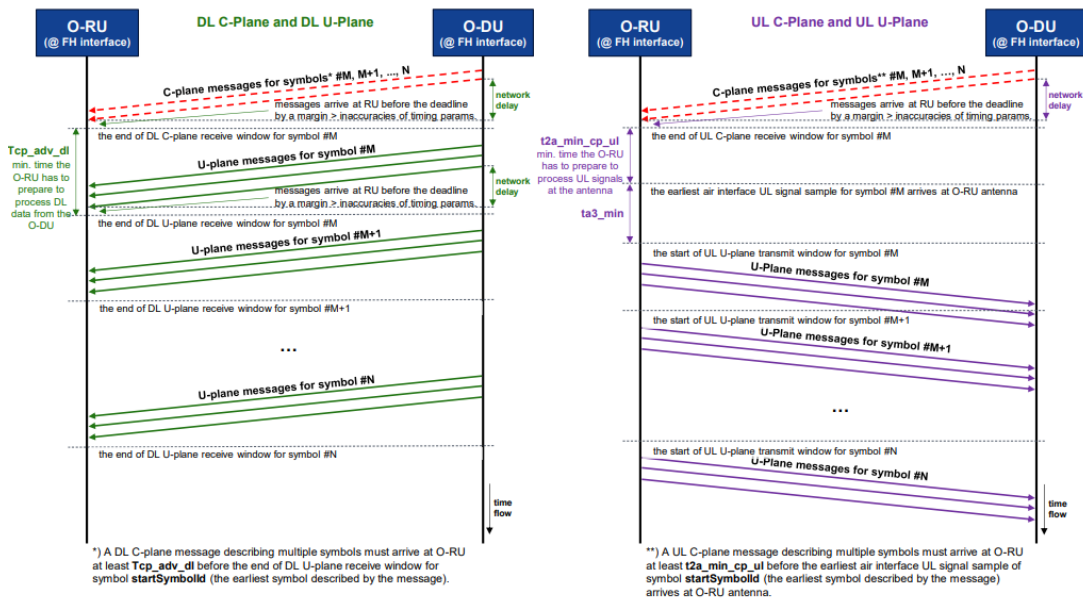
0000  0f 95 68 f1 5e 5f ec cd 65 55 07 d9 ae fe 10 00
0010  05 80 00 85 29 80 90 cd 00 00 20 00 00 32 04 00
0020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0030  00 00 00 00 00 00 00 00 00 00 04 01 01 01 01
0040  01 01 01 01 01 01 01 01 01 01 01 01 01 01 01
0050  01 01 01 01 01 01 04 02 02 02 02 02 02 02 02
0060  02 02 02 02 02 02 02 02 02 02 02 02 02 02 02
0070  02 02 04 03 03 03 03 03 03 03 03 03 03 03 03
0080  03 03 03 03 03 03 03 03 03 03 03 03 03 04 04
0090  04 04 04 04 04 04 04 04 04 04 04 04 04 04 04
00A0  04 04 04 04 04 04 04 04 04 04 05 05 05 05 05
00B0  05 05 05 05 05 05 05 05 05 05 05 05 05 05 05
00C0  05 05 05 05 05 05 04 06 06 06 06 06 06 06 06
00D0  06 06 06 06 06 06 06 06 06 06 06 06 06 06 06
00E0  06 06 04 07 07 07 07 07 07 07 07 07 07 07 07
00F0  07 07 07 07 07 07 07 07 07 07 07 07 07 04 08
0100  08 08 08 08 08 08 08 08 08 08 08 08 08 08 08

```

KUVA 6. Esimerkkikuva SCT-testeille annetusta IQ-datasta. Kuva otettu yhdestä SCT-testille annettavasta käyttäjätieto-tekstitiedostosta

Kuvassa 6 näkyy esimerkki testissä käytetystä käyttäjäviestien IQ-datasta tekstimuodossa. Alussa sinisellä merkattu on tunnisteosa, jonka jälkeen alkaa itse IQ-data. Punaisella on merkattu esimerkki yhdestä resurssilohkosta. Kuvassa on vain yhdelle symbolille lähetettävää dataa.

Ohjaus- ja käyttäjädataviestit eivät kulje samassa Ethernet-kehyksessä. Ohjausviesti lähetetään ensin, jonka jälkeen käyttäjädataviestit saapuvat symboli järjestyksessä (Kuva 7).

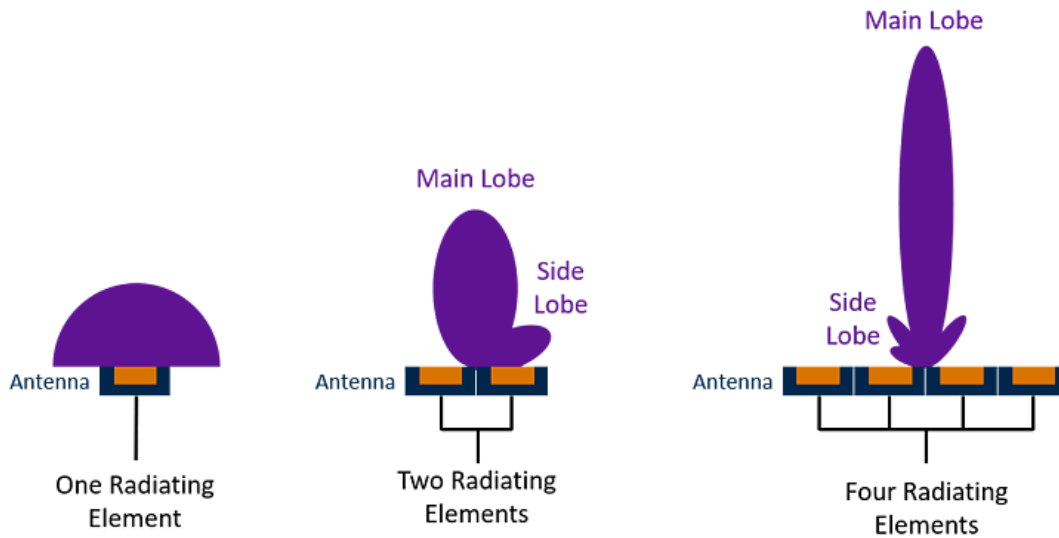


KUVA 7. Ohjaus- ja käyttäjäviestien vastaanottojärjestys (7, s. 166)

## 2.4 Keilanmuodostus

MIMO (Multiple Input Multiple Output) -tekniikka on ollut käytössä jo 4G -tekniikassa, jossa käytetään useaa antennia lähettämään ja vastaanottamaan dataa yhtäaikaaisesti. Pienemmät antennit mahdollistavat useiden kymmenien antennien ryhmän asentamisen yhteen antenniyksikköön, joka taas mahdollistaa MU-MIMO:n (Multiple User MIMO) - ja keilanmuodostuksen (Beamforming). MU-MIMO:ssa hyödynnetään useita antenneja lähettämällä yhtäaikaisesti dataa usealle eri käyttäjälle samalla ajanhetkellä, jolloin lähetystehokkuus kasvaa (10). Jotta lähetys usealle eri käyttäjälle onnistuu samanaikaisesti, tarvitaan siihen avuksi keilanmuodostusta.

Keilanmuodostus vaatii usean antennin ryhmän, jolloin niiden lähettämä signaali voidaan keskittää yhdeksi keilaksi. Mitä enemmän antenneja on käytössä, sitä tarkempi keila saadaan muodostettua (kuva 8).



KUVA 8. Keilanmuodostus eri antennimäärillä (10)

Keilanmuodostuksella voidaan voimistaa signaalia yhdelle käyttäjälle tarkemmin tai vaihtoehtoisesti lähettää dataa samalla ajanhetkellä usealla käyttäjälle eri suuntiin (kuva 9). Keilanmuodostus toimii muuttamalla signaalin vahvuutta ja vaihdetta, jolloin signaali voidaan lähettää haluttuun suuntaan.

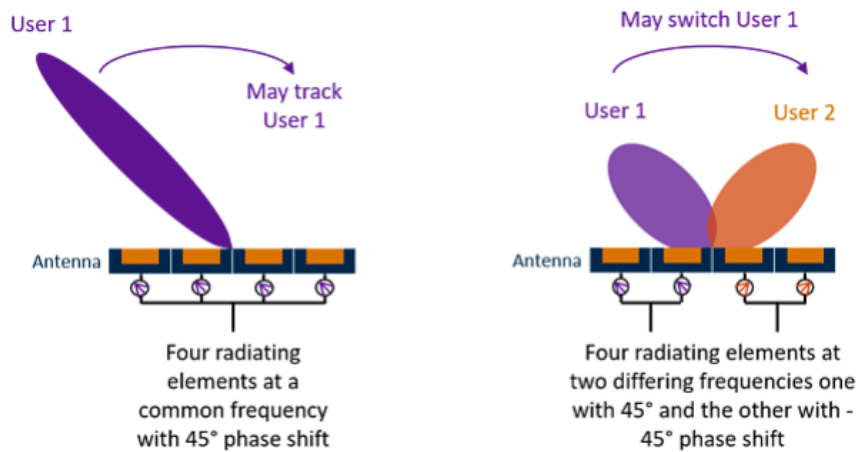


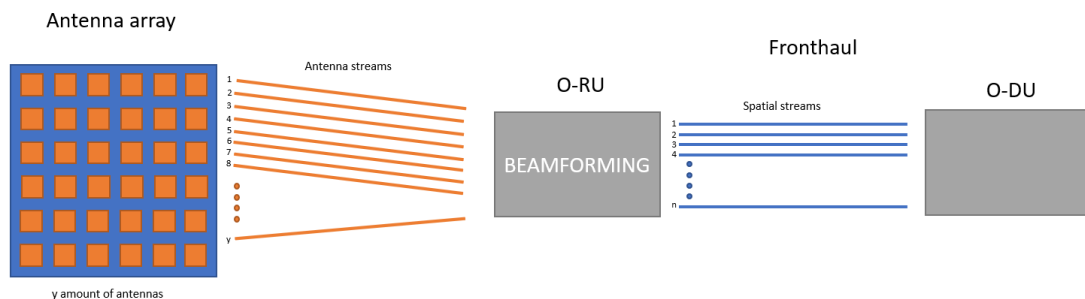
Figure 2: Beam steering and beam switching

KUVA 9. Keilanmuodostuksen ohjaus ja keilan vaihto (10)

Digitaalisessa keilanmuodostuksessa signaali esilasketaan ennen sen lähettämistä antennille. O-RU-kategoria A:ssa tämä tapahtuu O-DU:ssa kuvan 3 mukaisesti.

O-DU:n ja O-RU:n välisessä signaloinnissa keilanmuodostus hoitaa datan muokkaamisen niiden välille seuraavilla tavoilla:

- Downlinkissä keilanmuodostus laskee matriisilaskentaa hyödyntäen O-DU:lta saadusta  $n$  määrästä signaaleja signaalin jokaiselle RU:n antennille (kuva 10).
- Uplinkissä taas keilanmuodostus muuttaa RU:n antennilta saadut  $y$  määrä signaaleja  $n$  määräksi signaaleja, jotka sitten lähetetään O-DU:lle (kuva 10).



KUVA 10. Keilanmuodostuksen periaate yksinkertaistettuna.

## 2.5 Systemin komponenttitestaus ja 5G NR UL/DL Matlab -malli

Nokiassa valvotaan kehitettävän koodin eheyttä useilla eri testeillä, ennen kuin yksittäinen kehittäjä voi liittää tekemänsä uuden koodin tai olemassa olevan koodin muutokset versionhallinnan päähaaraan. Näin ollen päähaara on aina toimiva versio koodista.

Tukiasemakoodia säilytetään Nokiassa Git-versionhallinnassa (11). Versionhallinnan päähaara on aina toimiva ja uusien versio koodista. Jotta päähaara pysyy toimivana, täytyy jokainen siihen liitettävä muutos aina testata ennen sen liittämistä päähaaraan. Tätä varten Nokiassa on käytössä useita testejä, mutta tähän opinnäytetyöhön liittyen tärkeimpänä ovat systeemin komponenttitestit (tästä lähtien käytetään termiä SCT-testit). SCT-testit testaavat nimensä mukaisesti yhtä osa-aluetta eli komponenttia koodista. Siinä missä Unit-testit testaavat koodin yksittäistä funktiota tai sen osaa, testaavat SCT-testit monesta eri luokasta ja funktioista koostuvaa kokonaisuutta.



Tässä opinnäytetyössä keskeisimpinä SCT-testeinä ovat testit, jotka testaavat tukiasemassa kontrolli- ja käyttäjätiedon kulkua eri komponenttien välillä. Tukiasemakoodia testataan mahdollisimman monella eri tavalla, jotta voidaan todeta komponenttien kontrolliin- ja niiden käyttämien tiedon eheys ja se, että käsiteltävään tietoon kohdistuvat toimenpiteet vastaavat vaatimuksia. Kun SCT-testit tehdään mahdollisimman kattavaksi, varsinkin kaikkien reuna- ja poikkeustilanteiden osalta, voidaan varmistua tukiaseman toiminnasta sekä koodin eheydestä.

Ongelmaksi tässä muodostuikin se, että käytössä ei ole valmiita ohjaus- sekä käyttäjäviestejä ja niille vastaavia oikeaksi todettuja vertailutietoja, jolla voitaisiin varmistaa tukiasemakoodin toimivuus. Tukiasemassa liikkuva tieto on myös hyvin monimutkaista, eikä juurikaan ihmissilmin suoraan luettavaa, joten tiedon tarkistaminen manuaalisesti ei ole mahdollista.

Tätä ongelmaa 5G NR UL/DL Matlab -malli auttaa ratkaisemaan. Mallille voidaan antaa täysin samat ohjaus- ja käyttäjädataviestit kuin SCT-testeille, jolloin malli muokkaa annetut tiedot oikeaan muotoon ja saadaan vertailutiedot SCT-testille. Tällä tavalla voidaan vahvistaa koodin toimivuus vertaamalla yksinkertaisesti koodin tuottamaa tietoa mallin tuottamaan tietoon. Kuvassa 11 on esimerkki mallin tuottamasta vertailutiedosta. Tätä verrataan SCT-testistä saatuun tietoon, ja niiden täytyy täsmätä täysin toisiinsa tai muuten testi on hylätty.

```

0000 aa bb cc dd ee 01 aa bb cc dd ee ff 81 00 f0 00 .....
0010 ae fe 10 00 10 d8 00 00 00 80 90 cd 00 00 00 00 .....
0020 00 64 00 00 00 00 00 00 00 00 00 00 00 00 00 .d.....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....
0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 80 .....
0050 04 00 20 01 00 08 00 40 02 00 30 10 00 80 04 00 .....@..0.....
0060 20 01 00 08 00 40 02 00 30 10 00 80 04 00 20 01 .....@..0.....
0070 00 08 00 40 02 00 30 10 00 01 00 08 00 40 02 00 ...@..0.....@..
0080 10 00 80 0c 04 00 20 01 00 08 00 40 02 00 10 00 .....@.....
0090 80 0c 04 00 20 01 00 08 00 40 02 00 10 00 80 0c .....@.....
00a0 04 00 20 00 01 80 0c 00 60 03 00 18 00 c0 0e 04 .....
00b0 30 30 01 80 0c 00 60 03 00 18 00 c0 0e 04 30 30 00.....00
00c0 01 80 0c 00 60 03 00 18 00 c0 0e 04 30 30 00 02 .....00..
00d0 00 10 00 80 04 00 20 03 01 00 08 00 40 02 00 10 .....@...
00e0 00 80 04 00 20 03 01 00 08 00 40 02 00 10 00 80 .....@.....
00f0 04 00 20 03 01 00 08 00 40 00 02 80 14 00 a0 05 .....@.....
0100 00 28 03 41 02 08 30 50 02 80 14 00 a0 05 00 28 .(.A..0P.....(

```

KUVA 11. Esimerkki mallin ja SCT testin tuottamasta vertailudatasta tekstitiedostosta.

## 3 TYÖN VAIHEET

### 3.1 5G NR UL/DL Matlab -mallin vaatimukset

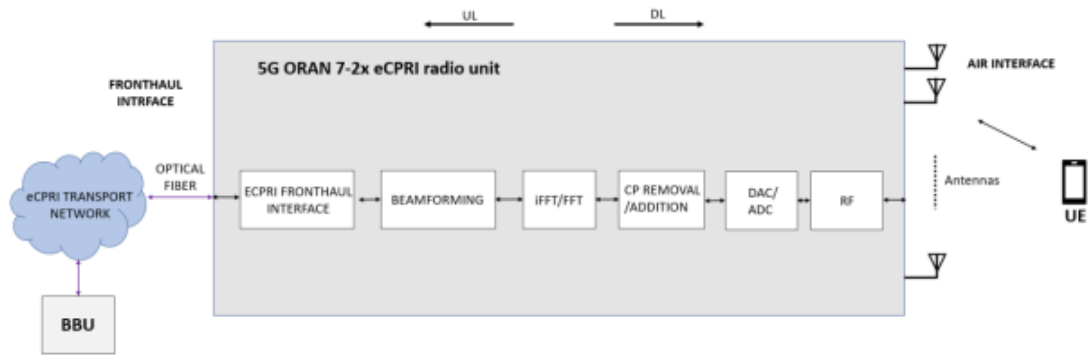
Tämän opinnäytetyön kannalta tärkein vaatimus on saada 5G NR UL/DL Matlab -malli integroitua toimimaan itsenäisesti Nokian Linux-pohjaisessa kehittäjäympäristössä ilman graafista käyttöliittymää sekä automatisoida 5G NR UL/DL Matlab -malli toimimaan automaattisesti olemassa olevien SCT-testien kanssa. 5G NR UL/DL Matlab -mallin täytyy osata automaattisesti hakea oikeat käyttäjä- ja kontrolliviestit annetun tiedostopolun perusteella. 5G NR UL/DL Matlab -mallin täytyy käyttäjän halutessa myös tallentaa välitulokset mallin komponenttien väliltä vikatilanteiden selvittämistä varten.

5G NR UL/DL Matlab -mallin vaatimus on, että sillä tulisi pystyä mallintamaan kontrolli- sekä käyttäjä viestien prosessointi ilmarajapinnan ja systeemimoduulin välillä. 5G NR UL/DL Matlab -mallille täytyy pystyä määrittelemään kontrolli- ja signaalipolkuun liittyviä parametreja eri käyttötarkoituksia varten. Signaalipolulla tarkoitetaan viestintää eri komponenttien välillä, jotka omalta osaltaan osallistuvat 5G NR UL/DL -tukiaseman kontrollointiin ja tiedon käsittelyyn. Mallin täytyy tukea kontrolli- ja signaalointi dataa eri formaateissa.

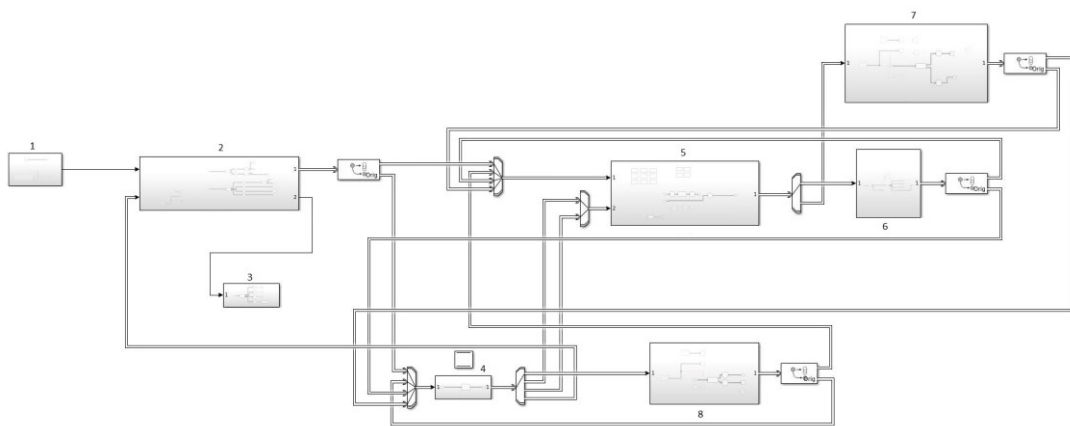
### 3.2 Työn aloitus ja mallin esittely

Opinnäytetyön tekeminen alkoi tutustumalla Matlab- ja Simulink ohjelmiin ja opiskelemalla 5G NR- sekä ORAN 7-2x -dokumentaatiota. Opiskelussa keskeisenä oli myös jo olemassa oleva 5G NR UL/DL Matlab -mallin Downlink-osuus.

Malli vastaanottaa luvussa 2.3 esiteltyjä eCPRI-paketteja ilmarajapinnasta. eCPRI-paketit ovat pakatussa formaatissa, joten ne täytyy ensin purkaa. Purettu data ohjataan edelleen prosessoitavaksi, kuten esimerkiksi keilanmuodostukseen. Prosessoinnin jälkeen data pakataan uudelleen ja lähetetään optista kuitukaapelia pitkin O-DU:lle, joka esitetään kuvassa 12.



KUVA 12. O-RU:n toimintaperiaate (5, s. 37)



KUVA 13. Esimerkkikuva 5G NR UL/DL mallista Simulink-ympäristössä

Kuvassa 13 esiintyvä 5G NR UL/DL -malli simuloi datan käsittelyn O-RU:ssa Downlinkissä seuraavassa järjestyksessä:

1. Toiminnallisuus 1 simuloi Fronhaultia eli O-RU:n ja O-DU:n välistä kaapelointia. Fronthaul vastaanottaa O-DU:lta saadut viestit (Ohjaus, käyttäjä, synkronointi ja hallinnointi).
2. Toiminnallisuudet 2 ja 8 vastaanottavat ja purkavat saadut Ethernet-paketit.
3. Toiminnallisuus 6 toteuttaa keilanmuodostuksen.
4. Toiminnallisuus 7 pakkaa muokatun tiedon eteenpäin lähetettäväksi.
5. Toiminnallisuudet 2 ja 3 hoitavat datan tiedon lähettämisen eteenpäin.

Opiskelun ohessa ensimmäisenä tehtävänä oli laajentaa 5G NR UL/DL Matlab -mallia tukemaan .pcap-tiedostomuotoisia ohjaus- ja käyttäjätietoviestejä. Matlabiin oli juuri lisätty versiossa 2021b funktio .pcap-tiedostojen lukuun ja sen liittäminen malliin olikin hyvin yksinkertaista (12). Mallin opiskelu ja .pcap-tiedostojen lukutoiminnon kehittäminen tapahtui Nokian Windows pohjaisella työ-kannettavalla.

### 3.3 Matlabin asennus kehittäjäympäristöön

Tähän mennessä mallia oli kehitetty työntekijöille jaetuilla kannettavilla tietokoneilla, jotka ovat Windows pohjaisia. Jokaisella työntekijällä saattoi myös olla eri versio Matlabista, joten tätä varten täytyi kehittää ratkaisu, jotta 5G NR UL/DL Matlab -mallin yhteensopivuus taattaisiin.

Nokialla on käytössä Linux-pohjainen konttitekniikkaa käyttävä kehitysympäristö (13). Konttitekniikalla saadaan jokaiselle kehittäjälle yhtenäinen kehitysympäristö, jossa on kaikki tarvittavat kehitystyökalut ja ohjelmistot ja joissa on kaikissa sama versio. Näin ollen kehitystyössä ei tule ongelmia versioiden yhteensopivuuksien kanssa ja kehittäjän ei tarvitse itse alkaa asentamaan tarvitsemaansa ohjelmistoja. Matlab ei ole tähän mennessä ollut osa kehitysympäristöä, joten sen liittäminen konttiympäristöön oli seuraava tehtävä.

Nokialla on oma tiimi, joka vastaa konttiympäristöstä ja sen ohjelmistoista. Matlabin asennusta keikeltiin ensin perinteisellä tavalla lataamalla Mathworksin sivuilta Linux asennuspaketti ja asentamalla sen konttiympäristöön (14). Asennus toimi ongelmitta ja näin päästiin käyttämään Matlabia kehittäjäympäristössä tarvittavine lisäosineen. Koska asennuspaketin koko on suuri ja asennus vaadittaisiin jokaiseen konttiympäristöön, päädyttiin etsimään vaihtoehtoisia keinoja.

Nokian kehitysympäristöt toimivat siten, että yhdelle palvelimelle yhdistyy kymmeniä sovelluskehittäjiä, jotka sitten luovat oman konttiympäristönsä kyseiselle palvelimelle. Näin ollen päätettiin pyytää IT-tukea asentamaan Matlabin uusimman version palvelimelle, jotta sitä voitaisiin käyttää konttiympäristössä ilman tarvetta asentaa Matlabia jokaiseen konttiympäristöön erikseen.

IT-tuki asensi Matlabin kaikille käytössä olleille palvelimille, mutta sitten ongelmaksi muodostui, että serverille asennettua Matlabia ei päässyt kuitenkaan käyttämään konttiympäristössä, koska konttiympäristön käynnistysohjeisiin ei ollut määritelty Matlabin asennuspolun liittämistä konttiin.

Tätä varten seuraavaksi tehtäväksi muodostuikin konttiympäristöjen sekä Nokialla käytössä olevien konttiympäristön käynnistämiseen kirjoitettujen käynnistyskomentojen opiskelu. Lopulta käynnistyskomennosta löytyi tiedostopolkujen liittäminen palvelimelta konttiympäristöön. Asennuspolun lisääminen käynnistyskomentoihin mahdollisti Matlab asennuskansion käytön konttiympäristössä, jolloin Matlab pystyttiin käynnistämään konttiympäristössä. Näin ollen jokaisella serverin käyttäjällä on sama versio Matlabista käytössä, eikä kehittäjän tarvitse itse asentaa Matlabia ja haaskata levytilaa palvelimelta.

Ratkaisu ei vielä ollut aivan täydellinen, koska jokainen Matlabia tarvitseva kehittäjä joutuu lisäämään kontin käynnistyskomentoon rivin koodia ja käynnistämään konttiympäristönsä uudelleen. Jatkokehityksenä on tarkoitus liittää Matlab pysyväksi työkaluksi osaksi konttiympäristöä.

### **3.4 5G NR UL/DL Matlab -mallin automatisointi**

Tähän mennessä 5G NR UL/DL Matlab -mallia on ajettu vain Matlabin graafisella käyttöliittymällä. 5G NR UL/DL -mallille täytyi ajaa käsin pätkä kerrallaan, jolloin käyttöliittymä kysyi halutut ohjaus- ja käyttäjäviestit. Koska tavoitteena oli saada malli toimimaan automaattisesti SCT-testien ohella, oli seuraava tehtävä automatisoida mallin alustus ja ajo toimimaan ilman graafista käyttöliittymää.

5G NR UL/DL -mallin automatisointi alkoi tutkimalla Matlabin ajamista pelkillä terminaalikomennolla. Matlab onkin toteutettu niin, että ajo terminaalista on tuettuna vakiona, joten sen suhteen ei ollut muuta ongelmaa kuin opetella oikeat käynnistyskomennot. Seuraavaksi täytyikin tehdä päätös, miten aikaisempi manuaalinen tiedostojen syöttötapa toteutettaisiin terminaalilla automaattisesti.

SCT-testeissä on tallennettu jokaisen testin kansioon ohjaus- ja käyttäjäviestit erikseen SCT-testeille sekä Matlabille, koska 5G NR UL/DL Matlab -malli vaatii datan eri formaatissa kuin SCT-testeissä käytetty ohjelma. Matlabille tarkoitetut tiedostot alkavat sanalla "matlab\_" joten päätettiin tehdä ohjelma, joka lukee annetulla tiedostopolulla kaikki sen sisältämät tiedostot ja tallentaa kaikki "matlab\_" - sanan sisältävät tiedostot listaan. Tämän jälkeen listasta seulotaan uusin erillisiin listoihin ohjaus- ja käyttäjäviestit. Tämä sen takia, että 5G NR UL/DL Matlab -mallin alustuksessa käyttäjä ja ohjausviestit alustetaan eri komponenteissa.

Koska itselläni on eniten kokemusta Python ohjelmointikielestä, niin päätettiin suorittaa tiedostojen lukeminen ja seulominen Pythonilla (15). Aluksi piti kuitenkin automatisoida kaikki olemassa olevat komentosarjat toimimaan automaattisesti, jotta niitä voidaan käyttää automatisoidussa versiossa. Näitä komentosarjoja olivat mm. tietojen luku ja muuntaminen formaatista toiseen sekä mallin alustus ja saatujen tulosten muokkaaminen ja tallennus oikeaan muotoon.

Sen jälkeen tehtävänä oli kirjoittaa 5G NR UL/DL Matlab -mallin pääkomentosarja, joka vaatii käynnistysparametriksi listan ohjaus- ja käyttäjäviestien tiedostopoluista. Tarkoitus oli, että Python ohjelma kutsuu seulonnan jälkeen 5G NR UL/DL Matlab -mallin pääkomentosarjan, antaen samalla käynnistyskomentona seulotut listat viesteistä.

Tässä ongelmaksi muodostuikin viestilistojen siirtäminen Pythonista, kun kutsuttiin Matlab komentosarjaa. Koska listat täytyi antaa String-tiedostomuotoisena listana Matlabin komentosarjalle, oli listan muodostaminen hyvin altis pienillekin virheille. Tehtiin päätös jättää Python ohjelma kokonaan pois ja siirtää myös tiedostojen lukeminen ja seulominen kokonaan Matlabin komentosarjoihin.

Tähän mennessä tietojen luku tiedostosta Matlabin työalueelle sekä tiedostojen muokkaaminen 5G NR UL/DL Matlab -mallille haluttuun formaattiin tehtiin jokainen omassa komentosarjassa. Päätettiin että kirjoitetaan yksi pääkomentosarja, joka sitten kutsuu muut alikomentosarjat tarvittaessa ja oikeassa järjestyksessä. Pääkomentosarja tarvitsee käynnistyksen yhteydessä vain testin tiedostopolun, jonka jälkeen komentosarja toimii automaattisesti seuraavassa järjestyksessä:

1. Komentosarja hakee käyttäjä- ja ohjausviestit annetulla tiedostopolulla automaattisesti ja tallentaa ne omiin listoihinsa.
2. Komentosarja kutsuu tarvittavia alikomentosarjoja, jotta saadaan tiedot mallille oikeassa formaatissa. Viestit voivat olla teksti- tai .pcap-muodossa, jolloin tunnistetaan, kumpi formaatti on kyseessä ja kutsutaan oikea alikomentosarja muokkaamaan data mallin vaatimaan .mat formaattiin.
3. Alustetaan käyttäjä- ja ohjausviestit 5G NR UL/DL Matlab -mallin komponentteihin.
4. Ajetaan 5G NR UL/DL Matlab -malli.

5. 5G NR UL/DL Matlab -mallilta saadut tulokset ovat .mat-formaatissa, joten muutetaan ne tekstitiedostoksi ja lisätään myös datan ASCII-muutos. ASCII-muutoksesta esimerkki kuvan 11 oikeassa reunassa. Tämä siksi, koska SCT-testien tulokseen kyseinen ASCII-muutos lisätään automaattisesti.
6. Tutkitaan, onko annetussa SCT-testin tiedostokansiossa jo valmiiksi kansiota 5G NR UL/DL Matlab -mallin vertailutuloksille. Mikäli ei ole niin luodaan sellainen ja tallennetaan viestit sinne. Mikäli kansio löytyy, ylikirjoitetaan olemassa olevat tiedostot.
7. Mikäli "debug\_flag"- käynnistyskomento on asetettu arvoon "True", tarkistetaan, onko SCT-testin tiedostopolussa kansiota komponenttien välitiedoille. Mikäli ei ole niin luodaan sellainen ja tallennetaan välitiedot kyseiseen kansioon. Tästä lisää luvussa 3.5.
8. Tyhjennetään Matlabin työympäristö kaikesta datasta uuden testin ajoa varten.

Kun 5G NR UL/DL Matlab -mallin ajo saatiin toimimaan yhdellä komentosarjalla ja sille syötetyllä tiedostopolulla ilman ongelmia, olikin seuraavaksi päätettävä, miten mallin ajo yhdistettäisiin toimimaan olemassa oleviin SCT-testien kanssa automaattisesti.

Mallin automatisointiin SCT-testien kanssa pohdittiin kolmea eri vaihtoehtoa:

1. 5G NR UL/DL -malli yhdistettäisiin olemassa oleviin SCT-testeihin siten, että aina kun testit ajetaan niin ajettaisiin 5G NR UL/DL Matlab -mallilla niille joka kerta uudet vertailutiedot. Tällöin saataisiin aina ajankohtaiset vertailutiedot, mikäli testissä olevia käyttäjä- ja ohjausviestejä muutettaisiin. Ongelmaksi tässä toteutuksessa muodostuisi suuri työn määrä sekä olemassa olevien testien muokkaaminen ja niiden suorituksen kesto mallin ajon keston takia.
2. Kopioitaisiin olemassa olevat testit omaan testiryhmäänsä ja integroidaan 5G NR UL/DL -malli toteutuksen 1 mukaisesti. Tällöin voitaisiin pitää olemassa olevat testit erillään, jolloin testien pitkä ajoaika ei haittaisi. Näitä testejä voitaisiin ajaa harvemmin kestopensa takia ja todeta siten koodin eheys. Huonona puolena tässä toteutuksessa on työn määrä sekä kahden eri testitavan ylläpito.
3. Koska SCT-testit käyttävät vertailussa niihin tallennettuja vertailutietoja, niin voitaisiin kirjoittaa ohjelma, joka ajettaessa tallentaisi määritellyille testeille uudet vertailutiedot. Tällöin ohjelma voitaisiin ajaa ennen SCT-testejä, jolloin varmistetaan vertailutietojen paikkaansa pitävyys. Tämä olisi myös helpoin toteuttaa, koska 5G NR UL/DL -mallin ajo oli automatisoitu aikaisempaan tehtävänä. Tätä toteutusta voitaisiin myös käyttää apuna, jos päätetään tulevaisuudessa laajentaa toimintaa ensimmäiseen tai toiseen vaihtoehtoon.

Päätimme valita vaihtoehdon 3, koska sen työmäärä oli vähäisin muihin toteutuksiin nähden ja koska se toimii myös samalla pohjana ensimmäiselle ja toiselle toteutukselle.

Koska malli ei pysty tuottamaan vertailutietoa kaikille olemassa oleville SCT-testeille, kuten esimerkiksi poikkeustapaus testeille, piti ratkaista, miten automaattiselle ajolle kerrotaan tuetut testitapaukset. Päätettiin että luodaan tekstitiedosto, johon listataan kaikki tuetut testitapaukset kansio-  
polkuineen. Koska aikaisempaan tehtävänä oli automatisoitu malli toimimaan pelkän kansiopolon avulla, saatiin hyvin pienellä vaivalla kirjoitettua ohjelma, joka lukee tekstitiedostoon määritellyt kansiot polut ja ajaa mallin jokaiselle testille erikseen.

### **3.5 5G NR UL/DL Matlab -mallin vianmääritys tietojenkeruu**

Yksi 5G NR UL/DL Matlab -mallin vaatimuksista oli lisätä toiminto, jolloin malli ajaessaan tallentaisi myös tiedot ennen ja jälkeen komponenttien. Tätä tietoa voitaisiin käyttää apuna vikatilanteiden selvittämisessä, kun tarvitsee todeta komponentin toimivuus. 5G NR UL/DL Matlab -malliin päätettiin lisätä välitietojen tallennus tiedostoon ennen ja jälkeen kolmen eri komponentin kohdalle. Kommentosarjan käynnistyskomenttoon lisättiin parametri, jolla määritetään, halutaanko tallentaa ajettavan testin välitiedot.

Tiedon tallennus toteutettiin lisäämällä mallissa kulkevaan signaaliin Entity Replicator(16), joka kopioi signaalin ja ohjaa sen To File-toiminnallisuuteen(17), joka tallentaa kopioidun tiedon .mat-tiedostoon.

Kun halutut tiedot komponenttien välistä saatiin tallennettua omiin .mat-tiedostoihinsa, muokattiin komentosarjaa, joka muuttaa vertailu-.mat-tiedostot tekstitiedostoon käsittelemään myös uudet välitiedot ja tallentamaan ne omaan vianmääritys kansioonsa.

Näin ollen kehittäjät voivat asettaa halutessaan vianmääritystilan päälle komentosarjan käynnistykseessä, jolloin mallin ajo tallentaa samalla tiedot tärkeiden komponenttien väliltä, mikä helpottaa vikojen etsintää.



## 4 YHTEENVETO

### 4.1 Ongelmakohtia

Ensimmäisenä ongelmana opinnäytetyössä oli Matlabin asentaminen kehitysympäristössä. Vaikkakin Matlab saatiin lopulta asennettua Nokian servereille ja ohjeet sen käyttöönottoon kirjoitettiin Nokian ohjesivuille, jäi se vielä tulevaisuuden päivitettäväksi asentaa pysyväksi osaksi kehitysympäristön työkaluja.

Ensimmäiseksi ongelmaksi mallin automatisoinnissa muodostui Pythonin ja Matlab-komentosarjan välinen tiedon siirtäminen. Tarkoituksena oli, että Python-ohjelma seuloi tarvittavat tiedostot ja käynnistäisi sitten Matlabin ja välittäisi tarvittavat tiedot käynnistyskomennoissa. Koska Matlabin terminaalikäynnistyksen yhteydessä annettavat tiedot täytyi antaa hyvin tarkassa muodossa, muodostui sen muodostaminen Pythonilla ongelmaksi. Koska kyse oli niin triviaalista ongelmasta, päätettiin ongelma ratkaista jättämällä Python-osan kokonaan pois ja siirtää kaiken toiminnallisuuden Matlab-komentosarjaan. Tämä osoittautui lopulta hyväksi ratkaisuksi, kun saatiin kaikki toiminta Matlab-komentosarjoihin, jolloin ei tullut turhaa Python-ohjelmaa sen rinnalle.

Toisena pienenä ongelmana opinnäytetyössä oli Matlabin tarkkuus tietojen yhdistämisessä. Kun oli tarve yhdistää taulukkoja yhteen, täytyi niiden olla täysin samankokoiset, jotta yhdistäminen onnistuu. Tämän ongelman ratkaisemiseksi täytyi osaa tiedosta muokata samankokoiseksi yhdistettävän taulukon kanssa, jotta yhdistäminen onnistui. Tämä oli hieman outoa itselleni, koska muut ohjelmointikielet eivät ole olleet niin tarkkoja asian suhteen.

Kolmanneksi ongelmaksi muodostui 5G NR UL/DL -mallin hidas käynnistyminen ja sulkeminen ajettaessa terminaalissa. Mallin käynnistymiseen kuluu aikaa n. 1,5 minuuttia ja sen sulkemiseen n. 2 minuuttia. Itse datan generoinnissa ei kestä kuin muutamia sekunteja. Tälle ei löytynyt tekohekkellä ratkaisua, joten asia täytyi jättää sikseen. Kuitenkin useiden testien yhtäaikaisessa ajossa ei suljeta mallia välissä, joten useamman testin ajon kesto ei kasva juurikaan. Tätä varten osoitettiin kuitenkin hyväksi ratkaisuksi olla lisäämättä generointia olemassa oleviin SCT-testeihin, koska niiden kesto olisi kasvanut huomattavasti.

Neljäs ongelma ilmeni, kun pohdittiin, millä nimellä 5G NR UL/DL -malli tallentaisi vertailutiedot SCT-testikansioon. Kun selattiin olemassa olevia käsin tallennettuja vertailutietoja, huomattiin, että jokainen oli nimetty hieman eri tavalla. Tämän ongelman ratkaisemiseksi päätettiin kaikkien olemassa olevien SCT-testien vertailudatojen nimeämisen yhtenäistäminen, jotta mallin automaattinen ajo ei riko SCT-testejä. Tämä ei kuitenkaan kuulunut opinnäytetyön piiriin, koska muutettavia vertailutietoja ja testitapauksia oli huomattava määrä.

## 4.2 Yhteenveto

Opinnäytetyö onnistui suunnitellusti. Työ alkoi opiskelemalla olemassa olevan 5G NR UL/DL Matlab -mallin toimintaa ja siihen liittyvää Nokian sisäistä sekä julkista ORAN 7-2x -jaon dokumentaatiota. Ensikosketuksena 5G NR UL/DL Matlab -malliin toimi sen toiminnan laajentaminen muille tiedostotyypeille.

Kun 5G NR UL/DL Matlab -malli saatiin toimimaan uusilla tiedostotyypeillä, oli seuraavaksi vuorossa mallin integrointi Nokian kehittäjäympäristöön. Tämä vaihe vei hieman odotettua enemmän aikaa, kun käytiin keskustelua kehitysympäristöstä vastaavan tiimin kanssa. Matlab saatiin asennettua Nokian palvelimille ja hieman konttiympäristön käynnistyskomentoja muokaten myös jokaiselle kehittäjälle.

Kun saimme Matlabin käyttöön Nokian kehitysympäristössä, oli seuraava tehtävä muokata malli toimimaan Linux-pohjaisessa käyttöjärjestelmässä ilman graafista käyttöliittymää. Malli toimikin suoraan ilman muutoksia Linux-käyttöjärjestelmällä ajettuna, joten voitiin aloittaa suoraan mallin automatisointi. Ensimmäiseksi kirjoitettiin uusiksi komentosarja, jolla 5G NR UL/DL Matlab -malli alustetaan toimintaan sekä mallille annettavien tietojen ja mallilta saatujen tietojen tiedostotyyppien muutto- ja niiden tallennukseen liittyvät komentosarjat.

Kun olemassa olevat komentosarjat saatiin automatisoitua, oli tehtävä kirjoittaa pääkomentosarja, joka poistaa ihmisen tarpeen mallin ajamisesta. Pääkomentosarja hakee annetulla tiedostopolulla tarvittavat ohjaus- ja käyttäjäviesti tiedostot ja kutsuu tarvittavat komentosarjat tietojen formaatin muokkaukseen, mallin alustukseen ja ajoon sekä lopputietojen tallentamiseen.

Kun yhden SCT-testitapauksen ajo 5G NR UL/DL Matlab -mallilla saatiin automatisoitua, oli tehtävä vielä automatisoida useamman testin ajo. Kirjoitettiin vielä yksi komentosarja, joka lukee tekstitiedostosta määritellyt kansiot ja kutsuu aikaisemmin kirjoitetun pääkomentosarjan jokaiselle määritellylle tiedostopolulle.

Lopputuloksena saatiin 5G NR UL/DL Matlab -malli integroitua Nokian kehittäjäympäristöön sekä automatisoitua mallin ajo. 5G NR UL/DL Matlab -mallia voidaan ajaa joko kaikille tuetuille SCT-testitapauksille tai sitten voidaan ajaa yksittäinen testitapaus ja ottaa käyttöön vianmääritys ominaisuus.

### **4.3 Jatkokehitys**

5G NR UL/DL -mallin kehitystyö jatkuu opinnäytetyön jälkeen ja seuraavaksi alkaakin Uplink-osan kehittäminen. 5G NR UL/DL -malli toimii opinnäytetyön kirjoitushetkellä automatisoiduilla komentosarjoilla, mutta kehitystyön jatkuessa myös todennäköisesti tarvitsevat kyseiset komentosarjat myös ylläpitoa.

Matlabin asennus kehitys ja konttiympäristöön jää myös tulevaisuudessa päivitettäväksi. Tavoitteena olisi, ettei yksittäisen kehittäjän tarvitse tehdä muutoksia kehitysympäristön käynnistyskomentoihin saadakseen Matlabin käyttöön, vaan se olisi valmiiksi asennettuna jokaisessa ympäristössä.

Suurin ja selvin 5G NR UL/DL -mallin päivitykseen liittyvä työ on muuttaa olemassa olevien SCT-testien vertailutietojen nimeämisten yhtenäistäminen, jotta mallin tallentamat vertailudatat toimivat suoraan SCT-testien kanssa. Selvitettäväksi myös jää, voidaanko mallin ajon kestoa vähentää ajettaessa terminaalissa.

## LÄHTEET

1. MathWorks 2022. What Is MATLAB? Hakupäivä 7.2.2022. <https://se.mathworks.com/discovery/what-is-matlab.html>.
2. Mathworks 2022. Simulink. Hakupäivä 24.5.2022. <https://se.mathworks.com/products/simulink.html>.
3. Ahola Ville 2019. Miksi mallipohjainen suunnittelu? Simulointi ja testaus osaksi vaativaa ohjelmistokehitystä. Hakupäivä 3.2.2022. <https://devecto.com/miksi-mallipohjainen-suunnittelu/>.
4. IBM 2022. What is a digital twin? Hakupäivä: 7.2.2022. <https://www.ibm.com/topics/what-is-a-digital-twin>.
5. Hämäläinen Harri 2020. Uplink data measurement and analysis for 5G eCPRI radio unit. Oulun yliopisto. Tieto- ja sähkötekniikan tiedekunta, elektroniikan ja tietoliikennetekniikan tutkinto- ohjelma. Opinnäytetyö. Hakupäivä 2.5.2022. <http://jultika.oulu.fi/files/nbnfioulu-202011273208.pdf>.
6. Techplayon 2020. O-RAN Fronthaul Split Option 7-2x. Hakupäivä 21.02.2022. <https://www.techplayon.com/o-ran-fronthaul-split-option-7-2x/>
7. O-RAN Alliance 2021. O-RAN Specifications. Hakupäivä 21.02.2022 <https://www.o-ran.org/specifications>.
8. Wikipedia 2022. In-phase and quadrature components. Hakupäivä 10.5.2022 [https://en.wikipedia.org/wiki/In-phase\\_and\\_quadrature\\_components](https://en.wikipedia.org/wiki/In-phase_and_quadrature_components).
9. ShareTechnote. 5G/NR - Frame Structure. Hakupäivä 2.5.2022: [https://www.sharetechnote.com/html/5G/5G\\_FrameStructure.html#OFDM\\_Symbol\\_Duration](https://www.sharetechnote.com/html/5G/5G_FrameStructure.html#OFDM_Symbol_Duration).
10. Metaswitch Networks 2022. What is 5G beamforming, beam steering and beam switching with massive MIMO. Hakupäivä 10.5.2022. <https://www.metaswitch.com/knowledge-center/reference/what-is-beamforming-beam-steering-and-beam-switching-with-massive-mimo>.
11. Git 2022. Git --distributed-even-if-your-workflow-isnt. Hakupäivä 6.6.2022. <https://git-scm.com/>.
12. Mathworks 2022. pcapReader. Hakupäivä 26.4.2022. <https://se.mathworks.com/help/5g/ref/pcapreader.html>.
13. Docker 2022. Use containers to Build, Share and Run your applications. Hakupäivä 10.5.2022. <https://www.docker.com/resources/what-container/>.

14. Mathworks 2022. Downloads. Hakupäivä 26.4.2022. [https://se.mathworks.com/downloads/web\\_downloads/](https://se.mathworks.com/downloads/web_downloads/).
15. Python 2022. Python. Hakupäivä 10.5.2022. <https://www.python.org/>.
16. Mathworks 2022. Entity Replicator. Hakupäivä 1.6.2022. <https://se.mathworks.com/help/simevents/ref/entityreplicator.html>.
17. Mathworks 2022. To File. Hakupäivä 1.6.2022. <https://se.mathworks.com/help/simulink/slref/tofile.html>.