

Mobiilirobotin navigointi UWB:n avulla ROS 2 -järjestelmässä

29.10.2024

Petteri Mäkelä, Orcid ID: 0000-0001-9394-1114, Seinäjoen ammattikorkeakoulu

Suositeltu viittaus: Mäkelä, P. (2024). Mobiilirobotin navigointi UWB:n avulla ROS 2 -järjestelmässä. *SeAMK Journal, 1*, artikkeli 2. https://urn.fi/URN:NBN:fi-fe2024102887561**Tiivistelmä:** Autonomisten mobiilirobottien (AMR) käyttö on lisääntynyt eri aloilla, kuten teollisuudessa, logistiikassa ja kiinteistönhoidossa. Tunnettuja esimerkkejä ovat kauppojen kuljetusrobotit ja robottiruohonleikkurit. Joustava tuotanto on lisännyt mobiilirobottien tarvetta, ja markkinoilla on laaja valikoima automaattisia työkoneita. Mobiilirobottien ohjaus tapahtuu usein valmistajakohtaisilla ohjelmistoilla, mutta kiinnostus avoimen lähdekoodin ROS 2 -järjestelmään on kasvussa. Sisätiloihin tarkoitetut mobiilirobotit käyttävät sijainnin estimoinnissa usein laserkeilaimen tuottamaa pistepilvidataa sekä inertiayksikön ja pyöräenkooderien tuottamia mittauksia. Joissakin olosuhteissa, kuten jatkuvasti muuttuvassa ympäristössä, nämä menetelmät eivät ole aina riittäviä. Paikannuksen tarkkuutta, saatavuutta ja luotettavuutta voidaan parantaa hyödyntämällä muita paikannusmenetelmiä, kuten esimerkiksi ultralaajakaistaiseen (UWB) radiosignaaliin perustuvaa paikannusta. Tässä artikkelissa tutkitaan kirjallisuuden perusteella, miten UWB-paikannus voidaan integroida ROS 2 -pohjaiseen mobiilirobotin paikannukseen.

Avainsanat: ROS 2, Navigation 2, UWB positioning

Abstract: The use of autonomous mobile robots (AMR) has increased in various sectors such as industry, logistics and property management. Well-known examples include shop delivery robots and robotic lawn mowers. Flexible production has increased the need for mobile robots, and a wide range of autonomous robots are available on the market. Mobile robots are often controlled by manufacturer’s proprietary software,

but interest in the open-source ROS 2 system is growing. Mobile robots for indoor use often point cloud data from a laser scanner and measurements from an inertial unit and wheel encoders for position estimation. In some circumstances, such as in a constantly changing environment, these methods are not always sufficient. The accuracy, availability and reliability of positioning can be improved by using other positioning methods, such as ultra-wideband (UWB) based positioning. In this paper, based on the literature, we investigate how UWB positioning can be integrated into ROS 2-based mobile robot positioning.

Key words: ROS 2, Navigation 2, UWB positioning

1. Johdanto

Autonomisten mobiilirobottien käyttö on yleistynyt viime vuosina teollisuudessa, logistiikassa ja kiinteistönhoidossa. Kauppojen kuljetusrobotit, siivousrobotit ja robottiruohonleikkurit ovat suurelle yleisölle tunnetuimpia esimerkkejä itsestään ohjautuvista mobiiliroboteista. Vaatimukset joustavalle tuotannolle ovat lisänneet myös autonomisten mobiilirobottien tarvetta myös teollisessa tuotannossa ja logistiikassa (Nortio, 2023). Joustavassa automatisoidussa piensarjatuotannossa materiaaleja siirretään tuotantosolujen välillä mobiilirobottien avulla kiinteiden kuljettimien sijaan. Markkinoilla on useita teollisuuden ja logistiikan tarpeisiin soveltuvia automaattisesti ohjautuvia työkoneita aina satamanostureista ja haarukkatrukeista mobiilirobotteihin saakka. Näitä autonomisia laitteita ohjataan yleensä valmistajakohtaisilla ohjelmistoilla, ja robottien navigoinnissa käytetään useita eri paikannusteknologioita (Aizat ym., 2023).

Mobiilirobottien ohjauksessa on herännyt viime vuosina kiinnostusta myös avoimen lähdekoodin ratkaisuihin (Macenski ym., 2022). Tunnetuin avoimen lähdekoodin ohjelmistokirjasto autonomisten mobiilirobottien ohjaamiseen on Robot Operating System 2 (ROS 2), jonka juuret ovat Stanfordin yliopistossa ja Willow Garage -tutkimuskeskuksessa (Wyrobek, 2017). ROS 2 -paketissa on useita ohjelmistokirjastoja, joita voidaan käyttää esimerkiksi reitin suunnitteluun, navigointiin, paikannukseen, robotin ohjaamiseen ja simulointiin (Macenski ym., 2020). ROS-järjestelmää on käytetty yleisesti tutkimuksessa, mutta sen hyödyntäminen on herättänyt kiinnostusta myös kaupallisten toimijoiden keskuudessa (GIM Robotics, 2022).

Mobiilirobottien paikannuksessa ja ympäristön havainnoinnissa käytetään useita erilaisia teknologioita (Aizat ym., 2023). Yleensä mobiilirobotin paikannus ei perustu mihinkään yksittäiseen teknologiaan, vaan sijainnin määrityksessä käytetään useita tekniikoita samaan aikaan fuusioimalla eri anturien tuottamat tiedot (Elsanhoury ym., 2022). Yleisimmät autonomisten mobiilirobottien paikannusanturit ovat laserkeilain, inertianavigointiyksikkö (inertial measurement unit, IMU) ja pyöräenkooderi, mutta myös muita paikannusmenetelmiä on käytössä (Huang ym., 2023). Ulkotiloissa käytetään usein myös satelliittinavigointia eli GNSS:ää (Global Navigation Satellite System), vaikkakin joissakin käyttäjäryhmissä pyritään välttämään satelliittinavigointia sen haavoittuvuuden takia (Kahlil, 2024).

Usein mobiilirobotit muodostavat kartan kohdealueesta SLAM-menetelmän (simultaneous localization and mapping) avulla, jossa paikannetaan ja kartoitetaan ympäristöä samaan aikaan käyttäen pistepilvimittauksia (Aizat ym., 2023). Nämä pistepilvet voidaan tuottaa esimerkiksi laserkeilauksen, syvyyskameran tai tutkan avulla. Kun kartoitus on tehty, mobiilirobotti paikantaa itsensä vertaamalla reaaliajassa kerättyjä pistepilvimittauksia aiemmin muodostettuun karttaan. Usein paikan estimointia parannetaan vielä fuusioimalla

mukaan myös muiden anturien tuottamia mittauksia. Pistepilvimittauksiin perustuva paikannus voi kuitenkin häiriintyä, jos ympäristössä tapahtuu paljon muutoksia kartan muodostamisen jälkeen tai ympäristössä on vain vähän toisistaan erottuvia piirteitä (Ngueyn ym., 2021). Paikannuksen suorituskykyä voidaan parantaa lisäämällä sijainnin estimointiin myös muita paikannusmenetelmiä. Ultralaajakaistaiseen (ultra-wideband, UWB) radiosignaaliin perustuva teknologia soveltuu sisätilapaikannukseen erinomaisesti hyvän tarkkuutensa takia (Xianjia ym., 2021).

Tämän artikkelin tavoitteena on kartoittaa kirjallisuuden perusteella, miten UWB-paikannus voidaan integroida ROS 2 -pohjaiseen mobiilirobottiin. Artikkelin tarkoituksena on selittää myös ROS 2 -järjestelmän paikannuksen arkkitehtuuria, sillä se on dokumentoitu melko heikosti. ROS 2:n osalta tutkimukseen on valittu Navigation 2 -ohjelmistopaketin arkkitehtuuria selittäviä artikkeleita. Lisäksi paikannuksen ohjelmistoarkkitehtuuria käsitellään Navigation 2 -dokumentaation perusteella. UWB:n osalta tutkimukseen on valittu UWB-teknologiaa kuvaavia kirjallisuuskatsauksia (survey) sekä artikkeleita, jotka kuvaavat UWB:n käyttöä erityisesti ROS 2 -ympäristössä. Koska UWB:n rooli ROS 2:ssa on analoginen GNSS:n kanssa, on tutkimuksessa perehdytty myös artikkeleihin, jotka selittävät GNSS:n käyttöä ROS 2 -ohjelmistoarkkitehtuurissa. Artikkelin tarkoituksena ei ole kuitenkaan kartoittaa uusimpia paikannusmenetelmiä tai anturifuusion algoritmeja. UWB:n, laserkeilauksen ja IMU:n fuusiosta on kirjoitettu useita artikkeleita (Chikurtev ym., 2021; de Miguel, 2020; Wang ym., 2019; Nguyen, 2021), mutta näissä artikkeleissa ei ole yleensä selitetty, miten fuusio tehdään ROS 2 -järjestelmässä.

Artikkelissa esitellään ensin ROS 2:n paikannukseen liittyvät ohjelmistomoduulit sekä UWB-paikannuksen toimintaperiaatteet. Tämän jälkeen etsitään tutkimusartikkeleiden ja ROS 2 -dokumentaation pohjalta vaihtoehtoisia ratkaisuja, miten UWB-paikannus on fuusioitavissa ROS 2 -arkkitehtuuriin muiden paikannusmenetelmien mukaan. Artikkelissa esitetään uusi malli, jossa UWB on integroitu ROS 2:n kanssa samaan tapaan kuin GNSS. Toisena vaihtoehtona esitetään, että UWB-paikannus fuusioidaan ROS 2:n AMCL-moduulin kanssa. Fuusio pistepilvidatan kanssa voidaan tehdä joko sijaintitietotasolla tai raakamittaustasolla.

2. ROS 2

Robot Operating System 2 (ROS 2) on avoimen lähdekoodin ohjelmistokirjasto, joka on tarkoitettu autonomisesti kulkevien liikkuvien robottien ohjaamiseen (Open Robotics, 2024b). ROS 2 ei ole käyttöjärjestelmä, vaikka nimi antaakin näin ymmärtää, vaan se on pikemminkin väliohjelmisto (middleware). ROS 2 sisältää lukuisia laiteajureita ja ohjelmistokirjastoja, joita voidaan käyttää esimerkiksi reitin suunnitteluun, navigointiin, paikannukseen, robotin ohjaamiseen ja simulointiin. ROS 2 on hyvin laaja kokonaisuus, ja tässä artikkelissa käsitellään ainoastaan mobiilirobotin sijainnin estimointiin liittyviä moduuleja.

ROS 2 -järjestelmän ytimessä on ROS-graafi, jolla tarkoitetaan ROS-järjestelmän solmujen verkkoa ja solmujen välisiä yhteyksiä, joiden avulla solmut kommunikoivat keskenään (Open Robotics, 2024a). Kommunikointi ROS-verkossa tapahtuu julkaisija-tilaaja-arkkitehtuurin mukaisesti DDS-protokollalla (Direct Distribution Service). Solmut voivat julkaista nimettyjä viestejä (topics) muille solmuille tai tilata niitä. Myös

sijainnin estimointiin liittyvät moduulit käyttävät DDS-protokollaa tietojen välittämiseen.

2.1 NAV 2

ROS 2:n Navigation 2 -ohjelmistopaketti sisältää toimintoja muun muassa kartan luomiseen, ympäristön tunnistamiseen, paikantamiseen, reitin suunnitteluun, esteiden väistämiseen ja koordinaattimuunnoksiin (Macenski ym., 2020). Navigation 2 -paketin tarjoamien ohjelmistojen avulla robotti pystyy kulkemaan itsenäisesti kohteeseen mahdollisia esteitä väistellen. Kun robotille annetaan kartta ja tavoitekohde (sijainti ja asento), suunnittelee navigointijärjestelmä reitin alkupisteestä kohteeseen ja ohjaa robotin kulkemaan tätä reittiä. Robotin kulkiessa kohti tavoitekohdetta navigointijärjestelmä määrittää jatkuvasti robotin sijaintia ja havainnoi ympäristöään. Navigointijärjestelmä ottaa huomioon myös robotille määritellyt turvallisuusrajoitukset ja estää törmäykset esteisiin. Kohdatessaan esteen navigointijärjestelmä pyrkii löytämään reitin esteen ohi käyttäen apuna sijaintitietoa ja karttaa. Navigation 2 -paketti sisältää myös SLAM-algoritmin, jonka avulla voidaan kartoittaa kohdealue käyttäen robotin keräämää pistepilvidataa. Lisäksi paketin avulla voidaan toteuttaa monia sovelluksia, kuten automaattinen kohteen seuranta ja koko alueen kattava reitin suunnittelu, jota tarvitaan esimerkiksi siivousroboteissa.

Yksinkertaisimmatkin operaatiot, kuten navigointi pisteestä toiseen, vaativat monimutkaisen sarjan toimintoja. Käyttäytymispuiden (behavior trees, BT) avulla voidaan yhdistää monia erilaisia mobiilirobotin ohjaamisessa tarvittavia toimintoja (Macenski ym., 2023). Näitä toimintoja ovat muun muassa reitin suunnittelu, robotin ohjaus, esteiden väistäminen ja toipuminen virhetilanteista. Periaatteessa robotin tehtävien suorittamisessa tarvittavat käyttäytymismallit voitaisiin ohjelmoida myös hierarkkisten tilakoneiden avulla, mutta mobiilirobottien ohjaustehtävät ovat monimutkaisia, ja tilakoneen toteuttaminen olisi vaikeaa.

2.2 Koordinaatistot

Robotin ohjauksessa tarvitaan jatkuvasti muunnoksia eri koordinaatistojen välillä. Maailmankoordinaatistossa esitetty sijaintitieto täytyy muuttaa robottiin kiinnitettyyn koordinaatistoon, että robottia voitaisiin ohjata (Foote, 2013). Tyypillisesti mobiilirobotiikassa tarvitaan useita koordinaatistoja ja muunnokset näiden välille. Mobiilirobotin kuhunkin anturiin ja toimilaitteeseen liittyy omat koordinaatistonsa. Jos mobiilirobotin mukana on esimerkiksi tarttujalla varustettu yhteistyörobotti, tarvitaan pitkä ketju koordinaattimuunnoksia, jotta tarttuja saadaan ohjattua konenäköjärjestelmän osoittamaan kohteeseen.

ROS 2:n TF2-kirjasto tarjoaa standardin tavan seurata sijaintitietoa eri koordinaattikehyksissä ilman, että yksittäisen komponentin käyttäjän tarvitsee tuntea kaikkia järjestelmän koordinaattikehyksiä (Foote, 2013). TF2-kirjasto mahdollistaa myös automaattiset muunnokset eri koordinaatistojen välillä hajautetussa järjestelmässä. Tieto tarvittavista koordinaattimuunnoksista esitetään puurakenteen avulla. Muunnospuussa määritellään tarvittavat siirrot ja kierrot eri koordinaattijärjestelmien välillä. Muunnettu data voidaan lähettää kaikille kuuntelijoille broadcaster-moduulin avulla. Kuuntelijat tilaavat tarvitsemansa muunnoksen tuloksen ja interpoloivat sen tiettyyn ajanhetkeen TF2-kirjaston avulla.

Dokumentissa REP 105 (Meeussen, 2010) on määritelty ROS 2:n vaatimat kolme välttämätöntä koordinaatistokehystä base_link, odom ja map. Kehys base_link on kiinnitetty mobiilirobottiin ja se liikkuu

robotin mukana. Odom-kehys on kiinnitetty maailmankoordinaatistoon. Odom-kehyksessä esitetyn robotin sijainti- ja asentotiedon täytyy olla jatkuvaa, eikä datassa saa olla nopeita muutoksia. Sijaintia päivitetään odom-kehyksessä usein sellaisten anturien perusteella, jotka tuottavat tarkkaa tietoa robotin sijainnin ja asennon muutoksista. Tyypillisiä antureita ovat matkamittari (pyöräenkooderi) ja IMU-yksikkö. Siirtymän laskennassa voidaan käyttää myös kamerakuvaan perustuvaa visuaalista odometriaa. Odom-kehys on hyödyllinen tarkkana lyhyen aikavälin paikallisena referenssinä, mutta virheiden kertymisen (drift) takia sitä ei voi käyttää pitkään. Map-kehys on kiinnitetty odom-kehyksen tavoin maailmankoordinaatistoon. Map-kehyksessä olevan sijaintitiedon ei tarvitse olla jatkuvaa, mutta sijainnin virhe ei saa kasvaa ajan mukana.

2.3 Globaali paikannusjärjestelmä ja AMCL

ROS 2 -käytäntöjen mukaan robotin navigointia varten täytyy toteuttaa vähintään muunnokset map-kehyksestä odom-kehykseen (map->odom) ja odom-kehyksestä base_link-kehykseen (odom->base) (Meeussen, 2010). Globaali paikannusjärjestelmä tuottaa absoluuttisen sijaintitiedon maailmankoordinaatistossa ja se vastaa map->odom -muunnoksesta. Muunnoksen odom->base_link toteuttaminen on taas lokaalin järjestelmän eli odometriajärjestelmän vastuulla. Odometriajärjestelmän tehtävänä on tuottaa tasaista ja jatkuvaa sijaintitietoa paikallisessa koordinaattikehyksessä. Koska sijaintia päivitetään merkintälaskentaperiaatteella (dead reckoning), kertyy odom-kehyksessä estimoituun sijaintiin virhettä. Globaali paikannusjärjestelmä päivittää map->odom -muunnoksen suhteessa globaaliin kehykseen odometriajärjestelmässä kertyneen virheen kompensoimiseksi.

Monissa sisätiloihin tarkoitetuissa mobiiliroboteissa globaali paikannus tehdään käyttäen Adaptive Monte Carlo Localization (AMCL) -menetelmää (Macenski, 2023). Navigation 2 -kirjaston AMCL-moduuli estimoi robotin sijainnin ja asennon annetulla kartalla käyttäen laserskannerin tuottamaa pistepilvidataa. Tila estimoidaan partikkelisuotimen avulla. Navigation 2 -kirjastossa on myös SLAM-algoritmi, jota käytetään samanaikaiseen paikannukseen ja kartan luomiseen. Sijaintitieto globaaliin paikannukseen voidaan saada myös GNSS:stä tai jotain absoluuttista paikkatietoa tarjoavasta paikannusjärjestelmästä, kuten UWB-paikannuksesta.

Pistepilvimittauksia ja karttaa käyttävä AMCL-algoritmi toimii hyvin silloin, kun ympäristössä on riittävästi tunnistettavia piirteitä, eikä ympäristössä tapahdu liikaa muutoksia (Ngueyn ym., 2021). AMCL saattaa kadottaa sijaintitiedon esimerkiksi suurissa avoimissa tiloissa tai käytävillä. Tätä tilaestimaatin katoamistilannetta kutsutaan kidnappaukseksi (Wang ym., 2018). AMCL saattaa tarvita myös alkutiedon robotin sijainnista ja asennosta. Nämä tiedot voidaan antaa käsin tai ne voidaan saada jostain muusta paikannusjärjestelmästä, kuten UWB:sta tai GNSS:stä (Wang ym., 2018; Wang ym., 2019).

2.4 Lokaali paikannusjärjestelmä, robot_localization ja GNSS-datan integrointi

ROS 2 -järjestelmässä lokaali paikannusjärjestelmä eli odometriajärjestelmä käyttää tilan estimointiin robot_localization-pakettia (Moore & Stouch, 2014; Macenski, 2023). Tässä paketissa tila estimoidaan joko laajennetun Kalman-suotimen (extended Kalman filter, EKF) tai hajuttoman Kalman-suotimen (unscented

Kalman filter, UKF) avulla fuusioimalla mittauksia useista eri antureista. Yksinkertaisissa sisätiloihin tarkoitetuissa mobiiliroboteissa robotin asento ja kulkusuunta saadaan IMU-yksiköstä ja kuljettu matka pyöräenkoodereista.

Pääasiassa odometriajärjestelmään tarkoitettua robot_localization-ohjelmistoa voidaan käyttää myös globaalina paikannusjärjestelmänä esimerkiksi silloin, kun käytetään GNSS-vastaanotinta (Moore, 2016). Ohjelmisto robot_localization sisältää navsat_transform_node-nimisen moduulin, joka muuntaa GNSS-vastaanottimelta saadun sijainnin robotin käyttämään maailmankoordinaatistoon (Macenski, 2023; Moore, 2016). UWB-paikannus tuottaa absoluuttista paikkatietoa GNSS:n tavoin. GNSS-datan käsittely Nav 2 -arkkitehtuurissa on sikäli kiinnostavaa, että UWB:n integrointi Nav 2:een voidaan toteuttaa vastaavalla tavalla kuin GNSS-integrointi. Tästä ratkaisusta kerrotaan tarkemmin luvussa 4.

3 UWB

Ultralaajakaistaista (ultra-wideband, UWB) radiosignaalia käytetään yleisesti sisätilapaikannuksessa. UWB-paikannuksella päästään sisätiloissa parempaan paikannustarkkuuteen kuin kapeakaistaisilla radiosignaaleilla (Elsanhoury ym., 2022). Koska UWB-signaalin pulssi on erittäin lyhyt, voidaan signaalin saapumisaika mitata tarkasti. Yleisesti UWB-paikannuksen tarkkuudeksi ilmoitetaan 10–50 cm, mutta hyvissä olosuhteissa voidaan päästä jopa 2 cm tarkkuuteen. Lyhyen pulssin pituuden ansiosta suoraan saapunut UWB-signaali voidaan erottaa heijastuneesta signaalista helposti, mikä on tärkeä ominaisuus sisätiloissa, joissa esiintyy usein radiosignaalin monitie-etenemistä. Kapeakaistaisen signaalin tapauksessa heijastunut signaali sekoittuu suoraan saapuneeseen signaaliin, mikä huonontaa signaalin saapumisajan mittaustarkkuutta. UWB-paikannusjärjestelmä on myös melko edullinen ja suhteellisen helppo pystyttää. UWB:n heikkoutena on suuri virrankulutus ja riittämätön standardointi. Eri valmistajien UWB-laitteet eivät pysty useinkaan kommunikoimaan keskenään siten, että signaalin kulkuaika laitteiden välillä voitaisiin mitata. Kiinteästi asennettujen jatkuvasti toimivien UWB-lähettimien käyttö ulkotiloissa on luvanvaraista, mikä käytännössä estää UWB:n käytön paikannussovelluksissa ulkotiloissa.

3.1 Mittaussuureet

UWB-paikannuksessa käytetään samoja paikannusmenetelmiä kuin muissa radiosignaaliin perustuvissa paikannusmenetelmissä (Elsanhoury ym., 2022; Alarifi ym., 2016). Yleisimmät UWB-paikannuksessa käytettävät mittaussuureet ovat signaalin kulkuaika (Time of Flight, ToF) ja signaalin saapumisaikojen erotus (Time Difference of Arrival, TDoA). UWB-paikannus voidaan tehdä myös signaalin saapumiskulman perusteella (Angle of Arrival, AoA). Periaatteessa paikannus voidaan tehdä myös vastaanotetun signaalin voimakkuuden (Received Signal Strength Indicator, RSSI) perusteella, mutta signaalin voimakkuus riippuu ympäristöstä, ja paikannuksen tarkkuus on huomattavasti heikompi kuin kulkuaikamittauksilla.

ToF-menetelmässä kahden UWB-laitteen välinen etäisyys lasketaan mittaamalla laitteelta toiselle lähetetyn signaalin kulkuaika ja kertomalla tämä valonnopeudella (Elsanhoury ym., 2022). UWB-signaalin kulkuajan mittauksessa käytetään usein edestakaista kulkuaikamittausta, jolloin paikannettava laite ja tukiasema toimivat molemmat sekä lähettiminä että vastaanottimina. Edestakainen kulkuaikamittaus poistaa laitteiden

välisen kellovirheen, eikä laitteiden kelloja tarvitse siis synkronoida keskenään. Yleensä edestakainen kulkuaikamittaus tehdään vielä kaksipuoleisesti, jolloin myös kellojen aikaeron muutos (drift) saadaan kompensoitua. Tukiaseman tunnettu sijainti ja kulkuaikamittauksesta laskettu etäisyys määrittävät pallopinnan. Paikannettavan laitteen sijainti saadaan ratkaistua kolmen pallopinnan leikkauspisteestä.

TDoA-menetelmässä paikannettava laite toimii yleensä lähettimenä ja tukiasemat vastaanottimena (Cheng & Zhou, 2019; Elsanhoury ym., 2022). Tässä menetelmässä tukiasemien kellojen täytyy olla synkronoituja keskenään, mutta paikannettavan laitteen kellon ei tarvitse olla synkronoitu tukiasemien kelloihin. Yksi saapumisaikamittaus valitaan pivot-mittaukseksi, joka vähennetään muista saapumisaikamittauksista. Paikannettavan laitteen sijainti ratkaistaan laskemalla tunnettujen tukiasemien sijaintien sekä valonnopeudella kerrottujen saapumisaikojen erotuksien määrittelemien hyperboloidien leikkauspiste. Koska paikannettavan laitteen kellovirhe on tuntematon, tarvitaan kolmiulotteisen sijainnin määrittämiseen saapumisaikamittaukset vähintään neljästä tukiasemasta. TDoA-menetelmässä tukiasemien synkronointi voidaan tehdä joko langattomasti tai langallisesti (Leugner ym., 2016).

ToF-menetelmässä tagi toimii sekä lähettimenä että vastaanottimena edestakaisen kulkuaikamittauksen takia. Tämä nostaa paikannettavan laitteen virrankulutusta ja rajoittaa samalla alueella olevien tagien määrää (Xianjia ym, 2021). TDoA-menetelmässä tagi lähettää vain hyvin lyhyitä pulsseja, minkä ansioista tagin virrankulutus on suhteellisen pieni ja skaalautuvuus parempi kuin ToF-menetelmässä (Leugner ym., 2016). Toisaalta TDoA-menetelmän vaatima tukiasemien synkronointi on hankala toteuttaa ja ainakin langattomasti tehty synkronointi huonontaa jonkin verran tarkkuutta. ToF-menetelmä sopii yksinkertaisuutensa ja tarkkuutensa takia hyvin tutkimustarkoitukseen.

3.2 Paikan estimointi

Koska sijainti ratkaistaan pallon tai hyperboloidien leikkauspisteestä, on ratkaistava yhtälöryhmä epälineaarinen. Sijainnin estimointiin voidaan käyttää esimerkiksi iteratiivista pienimmän neliösumman menetelmää (Gauss-Newton), Kalman-suodinta tai partikkelisuodinta (Cheng & Zhou, 2019; Ullah ym., 2020). ROS 2:n Navigation 2-paketti sisältää laajennetun Kalman-suotimen (extended Kalman filter, EKF) sekä hajuttoman Kalman-suotimen (unscented Kalman filter, UKF), joita voidaan hyödyntää myös UWB-signaalin kanssa (Moore, 2014). Kalman-suodin toimii navigointisovelluksissa paremmin kuin pienimmän neliösumman menetelmä, sillä se pystyy estimoimaan dynaamisen järjestelmän tilan aikaisempia mittaustuloksia hyödyntämällä, vaikka nämä olisivat epätarkkoja ja sisältäisivät kohinaa (Kaplan & Hegarty, 2006, s. 466–470). Kalman-suodinta käytetään paikannuksessa yleisesti myös erilaisten anturien fuusiossa (Feng ym., 2020). Periaatteessa sijainti voidaan laskea myös suljetussa muodossa ympyrän (2D) tai pallon (3D) yhtälöitä käyttäen, mutta tällöin ei hyödynnetä ylimääriteltyä yhtälöryhmää (Li, 2016; Cheng & Zhou, 2019).

3.3 NLOS-tilanne

Muiden radiosignaaleihin perustuvien sisätilapaikannusjärjestelmien tapaan myös UWB-paikannus kärsii niin sanotusta NLOS-tilanteesta (non-line-of-sight, NLOS) (Elsanhoury ym., 2022; Alarifi ym., 2016). NLOS-tilanteessa signaali ei saavu suoraan esteen takia, mutta vastaanotin havaitsee heijastuneen signaalin.

Heijastuneen signaalin kulkema matka on pidempi kuin tukiaseman ja paikannettavan laitteen välinen etäisyys, ja heijastuneen signaalin käyttö sijainnin laskemisessa aiheuttaa virheen tulokseen. Joissakin tapauksissa mahdollisesti heijastunut signaali voidaan tunnistaa sen ominaisuuksien perusteella, mutta useinkaan tämä ei ole mahdollista (Khodjaev ym., 2010).

Jos mittauksia on enemmän kuin ratkaistavia tuntemattomia yhtälöryhmässä, voidaan virheellinen mittaus tunnistaa lineaarialgebran keinoin laskemalla ylimääritellyn yhtälöryhmän ratkaisun jäännösarvo (residuaali) (Dani ym., 2022). Kolmen tuntemattoman muuttujan tapauksessa vaaditaan vähintään viisi mittausta, että yksi virheellinen mittaus voidaan tunnistaa ja poistaa laskennasta. Jos jäännösarvo ylittää ennalta määritellyn rajan, voidaan olettaa, että yhtälöryhmässä on mukana vähintään yksi virheellinen mittaus. Tämän jälkeen voidaan poistaa yhtälöryhmästä yksi mittaus kerrallaan ja ratkaista sijainti ja jäännösarvo kullekin viidelle nelikolle. Jos jonkin nelikon jäännösarvo jää alle kynnysrajan, valitaan tämä nelikko ratkaisuksi. Virheellisen mittauksen tunnistaminen ja poistaminen voidaan tehdä myös GNSS:ssä käytetyn RAIM-menetelmän (receiver autonomous integrity monitoring) avulla, joka on laskennallisesti tehokkaampi kuin edellä kuvattu kombinaatioiden laskeminen (European Space Agency (ESA), 2014).

Lineaarialgebraan perustuvat menetelmät virheellisten mittausten poistamiseksi eivät toimi kuitenkaan hyvin kaikissa NLOS-tilanteissa, koska mittauksia ei ole useinkaan riittävästi. Esimerkiksi partikkelisuodin toimii pienimmän neliösumman menetelmää ja Kalman-suotimia paremmin tilanteessa, jossa mittausvirheet eivät ole normaalijakauman mukaisia (Ullah ym., 2022). Partikkelisuotimeen perustuvan sijainnin estimoinnissa voidaan hyödyntää myös rakennuksen pohjapiirustusta (Davidson ym., 2011). ROS 2:ssa partikkelisuodinta hyödynnetään laserskannaukseen perustuvassa sijainnin määrityksessä (Macenski ym., 2023). NLOS-tilanteen ratkaisemiseen on käytetty myös Factor Graph -optimointia sekä koneoppimiseen perustuvia menetelmiä (Elsanhoury ym., 2022).

4 UWB-paikannuksen toteutus ROS 2 -arkkitehtuurissa

UWB-paikannus muistuttaa monessa suhteessa GNSS-paikannusta. Molemmat paikannusmenetelmät tuottavat absoluuttista sijaintitietoa, joka on estimoitu etäisyysmittauksista. ROS 2 -järjestelmässä GNSS on integroitu loosely-coupled-periaatteella (Moore, 2016). Tämä tarkoittaa sitä, että järjestelmä käyttää GNSS-vastaanottimen valmiiksi laskemaa sijaintitietoa ja integroi tämän muiden anturien (IMU ja odometria) tuottaman mittaustiedon kanssa. Fuusio voidaan tehdä myös tightly-coupled-periaatteella, jolloin sijainti estimoidaan käyttäen GNSS:n tuottamia raakamittauksia (koodi- ja kantoaaltovaiheet) yhdessä muiden mittausten kanssa (Nathan, 2024).

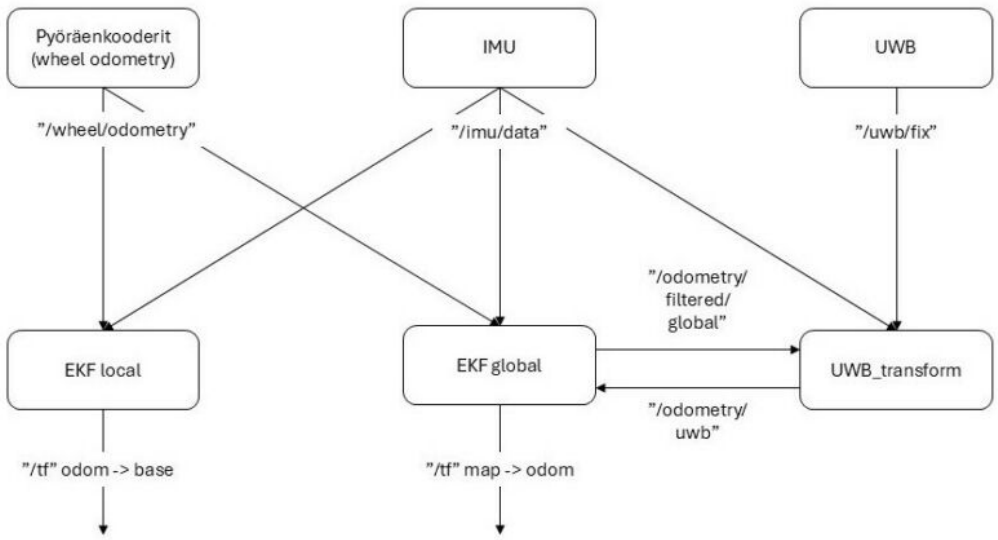
UWB:n tapauksessa voidaan toimia samalla tavoin kuin Mooren (2016) GNSS-ratkaisussa. Yksinkertaisin vaihtoehto on korvata robot_localization-paketin navsat_transform_node UWB:ta varten suunnitellulla ohjelmalla, joka toteuttaa tarvittavat koordinaattimuunnokset. Toinen tapa on fuusioida UWB:n tuottama paikkaestimaatti AMCL:n tuottaman paikkaestimaatin kanssa (Chikurtev ym., 2021; de Miguel ym., 2020; Wang ym. 2019). Kolmas vaihtoehto on fuusioida UWB:n tuottamat etäisyysmittaukset pistepilvidatan käsittelyn kanssa AMCL-laskennan sisällä tightly-coupled-periaatteella (Ngueyn, 2021). Viimeksi mainittu tapa

on monimutkaisin, mutta se toimii myös silloin, kun UWB-mittauksia on vähemmän kuin ratkaistavia muuttujia.

4.1 UWB-paikannuksen integrointi robot_localization-paketissa

UWB-paikannus on helpointa integroida ROS 2 -järjestelmään tekemällä se samalla tavoin kuin GNSS:n tapauksessa. Paketissa robot_localization on mukana navsat_transform_node-niminen moduuli, joka muuntaa GNSS:n tuottaman sijaintitiedon map-koordinaatistoon (Moore, 2016). Moduuli navsat_transform_node on vastuussa myös asentotietoon tarvittavasta koordinaattimuunnoksesta, vaikka sitä ei saada aina suoraan GNSS-vastaanottimelta. Asentotieto voidaan antaa manuaalisesti tai se voidaan tuoda IMU-yksiköltä. Asento voidaan määrittää myös käyttämällä kahta GNSS-antennia ja -vastaanotinta sekä kantoaallon vaihemittauksiin perustuvaa laskentaa.

Mooren (2016) esittämässä ratkaisussa GNSS-datalle käytetään tilan estimointiin kahta robot_localization-instanssia, joista ensimmäinen huolehtii lokaalista paikannuksesta ja toinen globaalista paikannuksesta. UWB-paikannuksen integrointi voidaan toteuttaa samalla tavalla kuin Mooren ratkaisussa ja tätä on havainnollistettu kuviossa 1.



Kuvio 1. UWB-laskennan integrointi robot_localization-paketissa Mooren (2016) mallin mukaisesti.

Estimaattori *EKF Local* estimoi tilan käyttäen ainoastaan IMU- ja odometriamittauksia, jotta estimoitu tilatieto on tasaista ja jatkuvaa. Tila estimoidaan käyttäen laajennettua Kalman-suodinta (EKF). EKF Local tuottaa odom->base -muunnoksen. Kehykseen base_link muunnettua sijainti- ja asentotietoa käytetään mobiilirobotin paikallisessa reitin suunnittelussa ja liikkeiden ohjaamisessa. Odom-koordinaattikehyksessä estimoituun sijaintiin kertyy virhettä ajan myötä. UWB-paikannuksen käyttöönotto ei vaikuta EKF Local -estimaattorin toimintaan.

Estimaattori *EKF Global* käyttää IMU-, odometria- ja UWB-mittauksia tilan estimointiin. Esitetty UWB-ratkaisu noudattaa Mooren (2016) dokumentoimaa GNSS:lle tehtyä arkkitehtuuria. UWB-mittaukset tuodaan moduulin

UWB-transform kautta, joka tekee tarvittavat muunnoksen UWB-paikannusjärjestelmän koordinaatistosta robotin maailmankoordinaatistoon (map). EKF Global tuottaa map->odom muunnoksen, mikä poistaa odom-kehyksessä estimoituun sijaintiin kertyneen kumulatiivisen virheen.

Moduuli UWB_transform voidaan toteuttaa mukaillen GNSS:lle tehtyä navsat_transform_node-moduulin toteutusta (Moore, 2014; Moore, 2016). Ratkaisu on monimutkainen kehäriippuvuuksineen, kuten on myös alkuperäinen GNSS-versio. UWB_transform-moduulin täytyy tuottaa sijainnin lisäksi myös robotin asentotietoa. Alkuasento saadaan IMU-yksiköltä (/imu/data). UWB_transform muuntaa UWB-järjestelmän tuottaman paikkatiedon (/uwb/fix) UWB-koordinaatistosta robotin maailmankoordinaatistoon. Toisin kuin GNSS:n tapauksessa, molemmat koordinaatistot ovat suorakulmaisia, ja tarvitaan vain koordinaatiston siirto ja kierto. UWB_transform tarvitsee tiedon robotin sijainnista ja asennosta maailmankoordinaatistossa koordinaattimuunnosta varten. Tätä varten UWB_transform tarvitsee viestin /odometry/filtered/global EKF Global -moduulilta.

Alkuperäinen GNSS-ratkaisu on tekijöidensäkin mielestä monimutkainen ja sitä voisi yksinkertaistaa (Zhang, 2020; Moore, 2017). Mooren mukaan navsat_transform_node-moduulissa oleva logiikka voitaisiin periaatteessa siirtää EKF-moduuliin.

4.2 UWB-paikannuksen fuusio lidarin ja AMCL-algoritmin kanssa

UWB- tai GNSS-paikannus voidaan fuusioida myös pistepilvidataa käyttävän AMCL:n kanssa. Chikurtev ym. (2021) esittävät ROS-pohjaisen ratkaisun, jossa lidar- ja GNSS-paikannus on integroitu loosely-coupled-periaatteella. Tässä järjestelmässä AMCL estimoi normaaliin tapaan robotin sijainnin ja asennon käyttäen lidarin tuottamaa pistepilvidataa ja karttaa. GNSS-paikannusta käytetään silloin, kun AMCL:n sijainnin estimointi ei ole riittävän luotettava. de Miguel ym. (2020) esittävät menetelmän, jossa AMCL:n partikkelisuotimen painokertoimet asetetaan GNSS-ratkaisun avulla. Periaatteessa GNSS-ratkaisu on korvattavissa UWB-ratkaisulla. Menetelmän avulla voidaan poistaa epätodennäköisimmät partikkelit ja vähentää kidnappaustilanteen mahdollisuutta.

Wang ym. (2019) esittävät algoritmin, jossa UWB on fuusioitu AMCL:n kanssa. Sijainti estimoidaan ensin AMCL-algoritmin avulla käyttäen lidarin pistepilvidataa sekä odometriamittauksia. AMCL:n tuottama tilaestimaatti fuusioidaan UWB:n kanssa EKF:n kanssa. Jos AMCL:n tuottama tilaestimaatti ei ole luotettava, käytetään UWB:n perusteella laskettua tilaestimaattia. Muuten käytetään UWB-ratkaisulla korjattua AMCL:n tilaestimaattia. Ehdotettu menetelmä parantaa paikannuksen suorituskykyä silloin, kun ympäristössä ei ole riittävästi erottuvia piirteitä AMCL:n tunnistettavaksi. UWB:lta saadun alkusijainnin käyttäminen nopeuttaa myös AMCL:n toimintaa alkutilanteessa ja silloin, kun AMCL kadottaa paikannuskyvyn (kidnappaustilanne). Nguyen ym. (2021) ratkaisussa IMU-, lidar- ja UWB-mittaukset on integroitu tightly-coupled-periaatteella, mikä mahdollistaa UWB:n hyödyntämisen myös silloin, kun mittausten lukumäärä ei mahdollista itsenäistä UWB-paikannusta.

5 Tulokset ja johtopäätökset

Tässä artikkelissa tutkittiin kirjallisuuden perusteella, miten UWB-paikannus voidaan integroida mobiilirobottiin käyttäen standardeja ROS 2 (Navigation 2) -ohjelmistomoduuleja. Artikkelissa kuvattiin ensin ROS 2 -järjestelmän ohjelmistoarkkitehtuuri sijainnin ja asennon määrityksen osalta sekä UWB-paikannuksen periaatteet. Tämän jälkeen etsittiin tutkimusartikkeleiden ja ROS 2 -dokumentaation pohjalta vaihtoehtoisia ratkaisuja, miten UWB-paikannus olisi fuusioitavissa ROS 2 -arkkitehtuuriin.

Artikkelissa esitettiin uusi malli, jossa UWB integroidaan ROS 2 -järjestelmään samalla tavoin kuin Moore (2016) on esittänyt GNSS:n tapauksessa. Tässä ratkaisussa käytetään kahta robot_localization-estimaattoria, joista toinen vastaa lokaalista ja toinen globaalista paikannuksesta. Globaalin estimaattorin GNSS-muunnoksesta vastaava ohjelmistomoduuli navsat_transform_node korvataan uudella ohjelmistomoduulilla, joka tekee tarvittavat koordinaatistomuunnokset UWB-datalle. Tämä ratkaisu on kuitenkin monimutkainen viestien kehäriippuvuuksien takia. Ohjelmistoarkkitehtuuria voidaan yksinkertaistaa viemällä UWB-datan käsittely koordinaatistomuunnoksineen globaalin estimaattorin sisälle. Kirjallisuudesta ei löydy kuitenkaan selitystä, missä roolissa AMCL on sen jälkeen, kun globaali robot_localization-estimaattori vastaa map->odom-muunnoksesta.

Toinen vaihtoehto on integroida UWB-paikannus pistepilvidataa käsittelevän AMCL:n kanssa. Tämä fuusio voidaan tehdä joko loosely-coupled- tai tightly-coupled-periaatteella. Loosely-coupled-ratkaisussa UWB- ja pistepilvipaikannus toimivat itsenäisesti toisistaan riippumattomasti ja paikannusratkaisut integroidaan. Tightly-coupled-ratkaisussa sijainnin ja asennon estimointi tapahtuu käyttäen UWB:n etäisyysmittauksia yhdessä pistepilvi- ja odometriadatan kanssa. Tämä mahdollistaa paikannuksen myös silloin, kun havaintoja ei ole riittävästi paikannukseen yhdellä paikannusmenetelmällä.

Kirjallisuuskatsauksen ja ROS 2 -järjestelmään tutustumisen perusteella voidaan todeta, että ROS 2 -järjestelmä on huonosti dokumentoitu paikannukseen liittyvien ohjelmistomoduulien osalta. ROS 2 voisi yleistyä kaupallisessa ja tutkimuskäytössä, jos sen arkkitehtuurista ja käyttötapauksista julkaistaisiin selittäviä artikkeleita kommentoituine lähdekoodeineen.

Kirjallisuuden (Wang ym., 2019; de Miguel ym. (2020) perusteella voidaan myös todeta, että UWB-paikannuksen lisääminen mobiilirobotin paikannukseen vähentää kidnappaustilanteen todennäköisyyttä ja parantaa paikannuksen luotettavuutta. Jatkotutkimuksena on tarkoitus toteuttaa Mooren (2016) mallin mukainen UWB-paikannuksen integraatio ROS 2 -järjestelmään ja julkaista testitulosten lisäksi myös toteutettu ohjelmistoratkaisu. Toinen jatkotutkimuksen aihe on testata NLOS-tilanteen takia syntyneiden virheellisten mittausten poistamista residuaali- ja RAIM-menetelmillä ROS 2:een toteutetussa UWB-paikannuksessa.

Kirjoittaja haluaa kiittää Töysän säästöpankkisäätiön rahoittamaa Autonomiset ajoneuvot -selvityshanketta sekä maakunnallisella kehittämisrahoituksella (AKKE) rahoitettua hanketta Avoimen lähdekoodin ratkaisut autonomisessa työkoneissa (OpenUGV).

Lähteet

Aizat, M., Azmin, A., & Rahiman, W. (2023). A survey on navigation approaches for automated guided vehicle robots in dynamic surrounding. *IEEE access, 11,* 33934–33955. https://doi.org/10.1109/ACCESS.2023.3263734

Alarifi, A., Al-Salman, A., Alsaleh, M., Alnafessah, A., Al-Hadhrami, S., Al-Ammar, M. A., & Al-Khalifa, H. S. (2016). Ultra wideband indoor positioning technologies: Analysis and recent advances. *Sensors, 16*(5), artikkeli 707. https://doi.org/10.3390/s16050707

Cheng, Y., & Zhou, T. (2019). UWB indoor positioning algorithm based on TDOA technology. *2019 10th International conference on information technology in medicine and education (ITME*), Qingdao, China, 777–782. https://doi.org/10.1109/ITME.2019.00177

Chikurtev, D., Chivarov, N., Chivarov, S., & Chikurteva, A. (2021). Mobile robot localization and navigation using LIDAR and indoor GPS. *IFAC-PapersOnLine*, *54*(13), 351–356. https://doi.org/10.1016/j.ifacol.2021.10.472

Davidson, P., Collin, J., & Takala, J. (2011). Application of particle filters to a map-matching algorithm. *Gyroscopy and navigation, 2*(4), 285–292. https://doi.org/10.1134/S2075108711040067

Dani, Y., Gunawan, A. Y., & Indratno, S. W. (2022). Detecting online outlier for data streams using recursive residual. 2022 s*eventh international conference on informatics and computing (ICIC), Denpasar, Bali, Indonesia*, 1–7. https://doi.org/10.1109/ICIC56845.2022.10006916

de Miguel, M. Á., García, F., & Armingol, J. M. (2020). Improved LiDAR probabilistic localization for autonomous vehicles using GNSS. *Sensors, 20*(11), artikkeli 3145. https://doi.org/10.3390/s20113145

Elsanhoury, M., Mäkelä, P., Koljonen, J., Välisuo, P., Shamsuzzoha, A., Mantere, T., Elmusrati, M., & Kuusniemi, H. (2022). Precision positioning for smart logistics using ultra-wideband technology-based indoor navigation: A review. *IEEE Access: Practical Innovations, Open Solutions, 10*, 44413–44445. https://doi.org/10.1109/ACCESS.2022.3169267

European Space Agency (ESA). (2014). RAIM. *ESA navipedia*. https://gssc.esa.int/navipedia/index.php/RAIM

Feng, D., Wang, C., He, C., Zhuang, Y., & Xia, X.-G. (2020). Kalman-filter-based integration of IMU and UWB for high-accuracy indoor positioning and navigation. *IEEE internet of things journal, 7*(4), 3133–3146. https://doi.org/10.1109/JIOT.2020.2965115

Foote, T. (2013). tf: The transform library*. 2013 IEEE conference on technologies for practical robot applications (TePRA), Woburn, MA, USA,* 1–6. https://doi.org/10.1109/TePRA.2013.6556373

GIM Robotics. (2022). *Playing the long game in the open-source framework/middleware league*. https://gimrobotics.fi/the-long-road-towards-ros2/

Huang, J., Junginger, S., Liu, H., & Thurow, K. (2023). Indoor positioning systems of mobile robots: A review.

Robotics, 12(2), 47. https://doi.org/10.3390/robotics12020047

Kahlil, J. (16.7.2024). KrattWorks awarded $6M for GNSS-free navigation. *GPS world*, https://www.gpsworld.com/krattworks-awarded-6m-for-gnss-free-navigation/

Kaplan, E., & Hegarty, C. (toim.). (2006). *Understanding GPS* (2. p.). Artech House.

Khodjaev, J., Park, Y., & Saeed Malik, A. (2010). Survey of NLOS identification and error mitigation problems in UWB-based positioning algorithms for dense environments. *Annals of telecommunications, 65*(5–6), 301–311. https://doi.org/10.1007/s12243-009-0124-z

Leugner, S., Pelka, M., & Hellbruck, H. (2016). Comparison of wired and wireless synchronization with clock drift compensation suited for U-TDoA localization. *2016 13th workshop on positioning, navigation and communications (WPNC), Bremen, Germany,* 1–4. https://doi.org/10.1109/WPNC.2016.7822846

Li, X., Deng, Z. D., Rauchenstein, L. T., & Carlson, T. J. (2016). Contributed review: Source-localization algorithms and applications using time of arrival and time difference of arrival measurements. *The review of scientific instruments*, *87*(4), artikkeli 041502. https://doi.org/10.1063/1.4947001

Macenski, S., Martín, F., White, R., & Clavero, J. G. (2020). The Marathon 2: A navigation system. *2020 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, 2718–2725. https://doi.org/10.1109/IROS45743.2020.9341207

Macenski, S., Foote, T., Gerkey, B., Lalancette, C. & Woodall, W. (2022). Robot operating system 2: Design, architecture, and uses in the wild. *Science robotics, 66*(7). https://doi.org/10.1126/scirobotics.abm6074

Macenski, S., Moore, T., Lu, D. V., Merzlyakov, A., & Ferguson, M. (2023). From the desks of ROS maintainers: A survey of modern & capable mobile robotics algorithms in the robot operating system 2. *Robotics and autonomous systems, 168*, artikkeli 104493. https://doi.org/10.1016/j.robot.2023.104493

Meeussen, W. (2010). REP 105: Coordinate frames for mobile platforms. *Open robotics*. https://www.ros.org/reps/rep-0105.html

Moore, T., & Stouch, D. (2014). A generalized extended Kalman filter implementation for the robot operating system. *Intelligent autonomous systems 13: Proceedings of the 13th international conference on intelligent autonomous systems IAS-13, 302*, 335–348. https://doi.org/10.1007/978-3-319-08338-4_25

Moore, T. (2016). Integrating GPS Data. *robot_localization wiki*. https://docs.ros.org/en/melodic/api/robot_localization/html/integrating_gps.html

Moore, T. (6.8.2017). Allow direct integration of GPS data #377. *GitHub*. https://github.com/cra-ros-pkg/robot_localization/issues/377

Nathan, A. (1.6.2024). Loose vs. tight coupling GNSS. *Point One Navigation*. https://pointonenav.com/news/loose-vs-tight-coupling-gnss/

Nguyen, T.-M., Cao, M., Yuan, S., Lyu, Y., Nguyen, T. H., & Xie, L. (2021). LIRO: Tightly coupled lidar-inertia-

ranging odometry. 2021 *IEEE international conference on robotics and automation (ICRA), Xi'an, China,* 14484–14490. https://doi.org/10.1109/ICRA48506.2021.9560954

Nortio, J. (2023). Verkkokauppa lisää logistiikan automaatiotarvetta. *Automaatioväylä*, (6), 34–36. https://www.automaatiovayla.fi/site/assets/files/6207/automaatiovayla_6_2023.pdf

Open Robotics. (2024a). *Basic concepts*. https://docs.ros.org/en/jazzy/Concepts/Basic.html

Open Robotics. (2024b). *ROS 2 documentation*. https://docs.ros.org/en/jazzy/index.html

Ullah, I., Shen, Y., Su, X., Esposito, C., & Choi, C. (2020). A localization based on unscented Kalman filter and particle filter localization algorithms. *IEEE Access, 8*, 2233–2246. https://doi.org/10.1109/ACCESS.2019.2961740

Wang, P., Chen, N., Zha, F., Guo, W., & Li, M. (2018). Research on adaptive Monte Carlo location algorithm aided by ultra-wideband array. *2018 13th world congress on intelligent control and automation (WCICA), Changsha, China,* 566–571. https://doi.org/10.1109/WCICA.2018.8630520

Wang, Y., Zhang, W., Li, F., Shi, Y., Chen, Z., Nie, F., Zhu, C., & Huang, Q. (2019). An improved adaptive Monte Carlo localization algorithm fused with ultra wideband sensor. *2019 IEEE international conference on advanced robotics and its social impacts (ARSO), Beijing, China*, 421–426. https://doi.org/10.1109/ARSO46408.2019.8948809

Wyrobek, K. (31.10.2017). The origin story of ROS, the Linux of Robotics. *IEEE spectrum*, https://spectrum.ieee.org/the-origin-story-of-ros-the-linux-of-robotics

Xianjia, Y., Qingqing, L., Queralta, J. P., Heikkonen, J., & Westerlund, T. (2021). Applications of UWB networks and positioning to autonomous robots and industrial systems. *2021 10th Mediterranean conference on embedded computing (MECO),Budva, Montenegro*, 1–6. https://doi.org/10.1109/MECO52532.2021.9460266

Zhang, M. (31.3.2020). Feature request, navsat_transform documentation diagram #550. *GitHub*. https://github.com/cra-ros-pkg/robot_localization/issues/550