

Radioaktiivisten päästöjen mittaaminen ilmanvaihdon poistoilmasta

Anssi Nykänen



JYVÄSKYLÄN YLIOPISTO
FYSIKAN LAITOS

Pro gradu-tutkielma

Ohjaajat: Jaana Kumpulainen ja Aki Puurunen

17.10.2016

Tiivistelmä

Avolähdetyöskentelyssä radioaktiivisia aineita voi vapautua laboratorion poistoilmaan kaasumaisena tai aerosoleina. Säteilylain mukaan toiminnan harjoittajan tulee määrittää näin syntyvät päästöt ja niiden vaikutukset ympäristössä oleskelevien ihmisten säteilyannoksiin. Tässä tutkimuksessa mitattiin ilmanvaihdon kautta avolähdelaboratorion ulkopuolelle päätyvien radioaktiivisten päästöjen määrää. Mittaus toteutettiin MAP Medical Technologies Oy:n (MAP) vanhalla päästömittausjärjestelmällä tuottamalla siihen uusi laskentaohjelmisto. Tuloksia vertailtiin MAP:n uuden järjestelmän tuottamiin tuloksiin sekä erikoistyössä laskettuihin päästörajoihin. Vertailussa todettiin, että eri järjestelmillä lasketut ^{131}I -päästöt vastaavat hyvin toisiaan. Sen sijaan ^{123}I -päästöt erosivat toisistaan vakiokertoimella, minkä oletettiin johtuvan eroista kalibroinnissa. Alkuvuoden 2016 aikana mitatut päästöt alittavat selvästi päästörajat.

Sisällysluettelo

1.	Johdanto	1
2.	Ionisoiva säteily	2
2.1.	Radioaktiivinen hajoaminen ja puoliintumisaika	2
2.2.	Aktiivisuus.....	4
2.3.	Ionisoivan säteilyn lajit	4
2.4.	Vuorovaikutukset	6
2.4.1.	Alfahiukkaset	7
2.4.2.	Beetahiukkaset	7
2.4.3.	Gammakvantit	8
3.	Säteilyn mittaaminen	9
3.1.	Mittalaitteet	9
3.2.	Aktiivisuuden mittaaminen	9
3.3.	Tuikeilmaisimet	10
3.4.	Ilmaisimen kalibrointi	11
3.4.1.	Tehokkuuskalibrointi	12
3.4.2.	Energiakalibrointi.....	13
4.	Radioaktiiviset päästöt ja niiden suodattaminen.....	14
4.1.	Päästöjen olomuodot	14
4.2.	Suodattaminen	14
4.3.	Jodin isotoopit	15
5.	Radioaktiivisten aineiden mittaaminen ilmavirrasta.....	18
5.1.	Jatkuvasti mitattavan suodattimen menetelmä	19
5.1.1.	Näytteenotto	19

5.1.2.	Suodatinloukku ja sen mittaaminen.....	20
5.2.	Tietojenkäsittely jatkuvan näytteenoton menetelmässä.....	21
5.2.1.	Piikin pinta-alan määrittäminen.....	21
5.2.2.	Kertymän määrittäminen.....	23
5.2.3.	Tehokkuuskertoimet ja päästön määrittäminen.....	24
6.	Mittausjärjestelmien vertailu.....	25
6.1.	MAP Medical Technologies Oy:n toiminta Tikkakosken toimipisteessä.....	25
6.2.	Toimintaperiaatteiden erot.....	26
6.3.	Vertailun toteutus.....	27
6.4.	Radioaktiivisten päästöjen määrät.....	28
6.5.	Tuloksissa havaitut erot.....	28
7.	Johtopäätökset.....	31
8.	Lähteet.....	33
9.	Liitteet.....	35
9.1.	Liite 1: Päästöjen vaikutusarviot.....	35
9.2.	Liite 2: Laskentaohjelmisto.....	49

1. Johdanto

Tässä tutkielmassa perehdytään radioaktiivisten päästöjen mittaamiseen ilmanvaihdon poistoilmasta. Tutkielman pääpaino on mittausmenetelmässä, joka perustuu jatkuvasti kerättävän ilmanäytteen tarkkailuun, mutta myös muita menetelmiä tarkastellaan. Esimerkkitapauksena vertaillaan MAP Medical Technologies Oy:n (MAP) Tikkakosken toimipisteen käytöstä poistuvaa ja uutta ilmamonitorointijärjestelmää. Saatuja tietoja voidaan käyttää vanhalla järjestelmällä mitattujen päästöarvioiden tarkasteluun. Tutkielma jää yrityksen käyttöön dokumentaatioksi päästömonitoroinnin toimintaperiaatteesta. Liitteenä olevassa erikoistyössä määritetään mitattujen päästöjen vaikutus ympäröivään väestöön.

Radioaktiivisten aineiden kulkeutuminen ilmanvaihtoon aiheuttaa tarpeen säteilysuojelutoimenpiteille lähialueiden väestön suojelemiseksi [1]. Pääasialliset toimenpiteet ovat ilmakehään kohdistuvien päästöjen ehkäisy pitämällä lähteet mahdollisuuksien mukaan suljettuina, sekä poistoilman suodattaminen. Suodattimet eivät kuitenkaan kykene ilman täydelliseen puhdistamiseen ja lisäksi on olemassa mahdollisuus poikkeaviin tilanteisiin, joten on tarpeen myös tarkkailla suodatuksen läpi pääsevän radioaktiivisen aineen määrää. ST-ohje 6.2 [1] määrittää kaasumaisten radioaktiivisten päästöjen määrän sellaiseksi, että päästöt eivät aiheuta väestöä edustavalle henkilölle raja-arvoa 10 μSv suurempaa efektiivistä annosta vuodessa. Varsinaisen rajan lisäksi ohjeessa määrätään, että päästöjen määrässä on noudatettava niin kutsuttua ALARA (As Low As Reasonably Achievable) periaatetta, eli päästöt on rajasta riippumatta pidettävä niin matalina, kuin järkevin toimin on mahdollista. Tämän ohjeen perusteella kunkin toiminnan harjoittajan tulee määrittää rajat sille, kuinka paljon se voi päästää radioaktiivisia aineita ilmanvaihdon kautta ulos, sekä suorittaa päästöjen rajoittaminen ja mittaaminen.

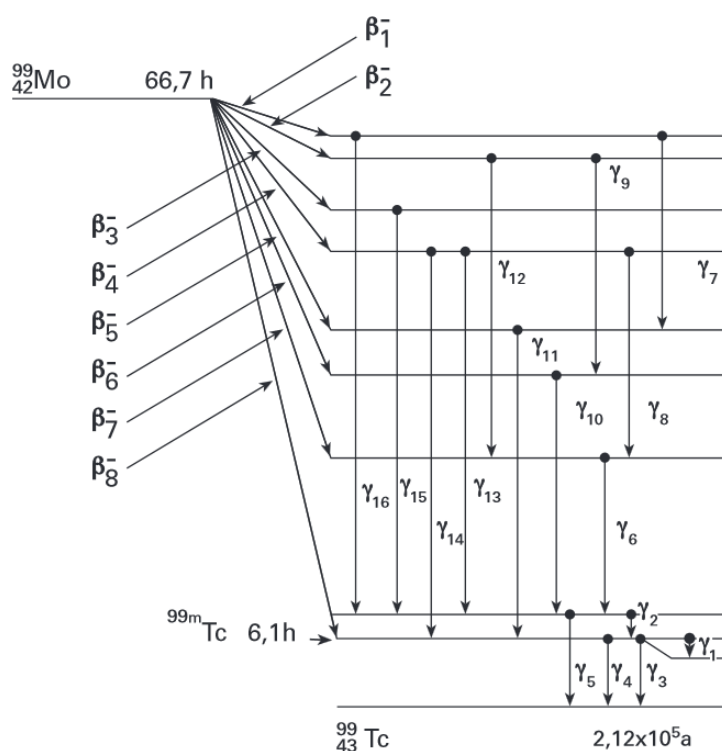
Tutkielma koostuu teoria- ja menetelmäosuuksista, joiden jälkeen on mittauslaitteistojen vertailu, sekä tulosten tarkastelu ja lopuksi johtopäätökset. Teoriaosuudessa kuvataan tutkielmassa käsitellyjä ilmiöitä ja pyritään luomaan lukijalle riittävä tietopohja lopun tekstin ymmärtämiseen. Menetelmäosiossa kuvataan erilaisia tapoja mitata radioaktiivisten päästöjen määrää ilmavirrasta ja pureudutaan tarkemmin yhteen niistä. Mittausjärjestelmien vertailuosiossa kuvataan MAP:n vanhan ja uuden päästömittausjärjestelmän tuottamien tulosten vertailua. Osio sisältää sekä vertailussa käytetyt menetelmät että saadut tulokset. Lopuksi johtopäätökset kokoavat tutkielman yhteen ja niissä esitetään myös ehdotuksia aiheeseen liittyvän tutkimuksen jatkolle.

2. Ionisoiva säteily

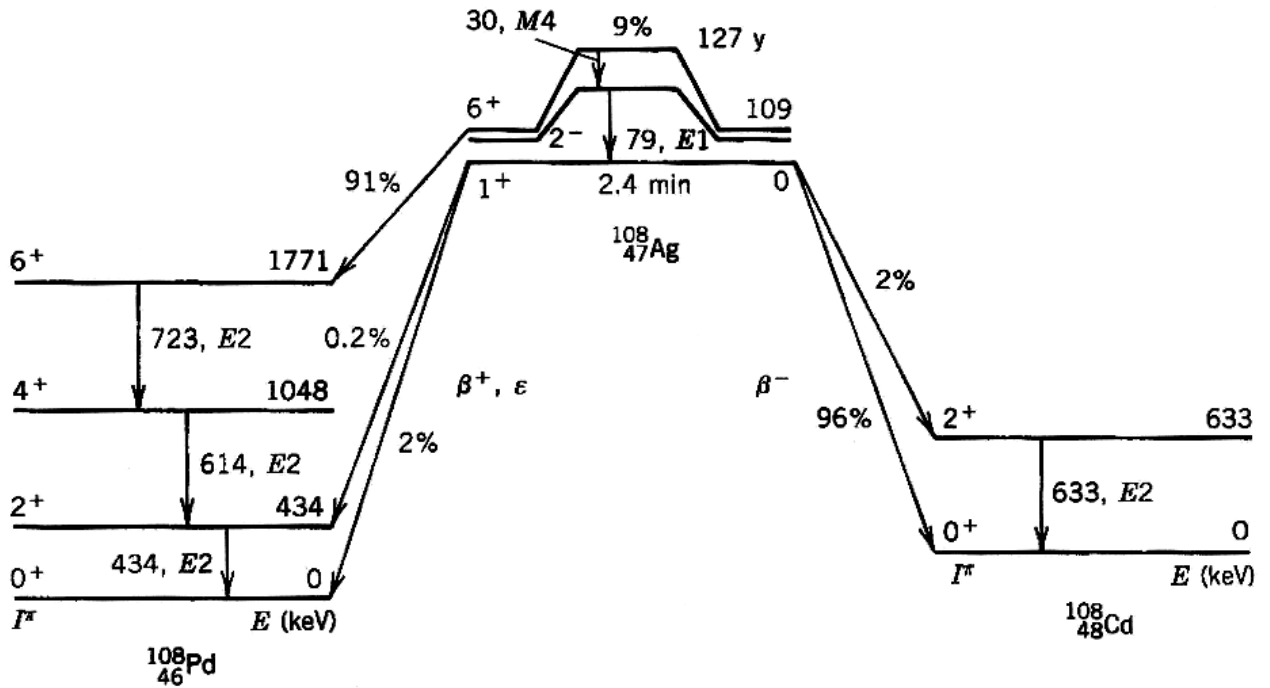
2.1. Radioaktiivinen hajoaminen ja puoliintumisaika

Radioaktiivinen hajoaminen on prosessi, jossa epävakaa ydin hajoaa spontaanisti kahdeksi tai useammaksi osaksi [2]. Tavallisimmat hajoamistavat ovat alfa- ja beetahajoaminen, sekä elektronisieppaus. Näiden lisäksi tunnetaan raskailla ytimillä esiintyvä hajoamistapa fissio, sekä hyvin protonirikkailla ytimillä protoniemissio ja neutronirikkailla neutroniemissio [3]. Hajoavaa ydintä kutsutaan emoytimeksi tai -nuklidiksi ja reaktiossa syntyvää uutta ydintä tytärnuklidiksi. Radioaktiivinen hajoaminen voi tapahtua spontaanisti vain silloin, kun hajoamisreaktion lopputulos on alemmalla energiatilalla kuin alkutila, eli reaktiossa vapautuu energiaa. Reaktiosta vapautuva energia päättyy yleensä pääasiassa reaktiotuotteiden liike-energiaksi.

Radioaktiivisessa hajoamisessa emonuklidin rakenne muuttuu, kun sieltä joko poistuu nukleoneja tai nukleonit muuttuvat toisikseen. Tämä rakennemuutos aiheuttaa yleensä sen, että syntynyt tytärnuklidi jää virittyneelle energiatilalle. Tällainen virittynyt tila voi toimia alkutilana uudelle hajoamiselle tai se voi purkautua yhden tai useamman gammakvantin tai konversioelektronin kautta. Virittyneellä tilalla voi olla useita mahdollisia purkautumisreittejä (Kuva 1), jotka voivat myös johtaa erilaisiin tytärnuklideihin (Kuva 2).



Kuva 1: ^{99}Mo :n hajoamiskaavio, jossa näkyy useita β -hajoamisreittejä ^{99}Tc :n eri energiatiloille, sekä niiden purkaantuminen erilaisten γ -hajoamisten kautta. Suurin osa tilojen purkautumisista tapahtuu metastabiilin tilan (^{99m}Tc) kautta. Kuva lähteestä [2].



Kuva 2: ^{108}Ag :n hajoaminen kahteen eri tytärnuklidiin. Kuva lähteestä [3].

Radioaktiivinen hajoamisprosessi on luonteeltaan satunnainen, eikä ole mahdollista ennustaa, milloin yksittäisen ytimen hajoaminen tapahtuu. Yksittäiselle ytimelle hajoamisen todennäköisyys tietyllä aikavälillä on kuitenkin vakio ytimen iästä riippumatta. Hajoamistodennäköisyyttä kuvaa hajoamisvakio λ , joka voidaan määrittellä suhteeksi

$$\lambda = -\frac{dN/dt}{N}, \quad (1)$$

jossa N on näytteessä olevien radionuklidien määrä ja dN/dt on hajoamisnopeus eli aktiivisuus [3]. Ratkaisemalla yhtälö (1) nuklidien määrän suhteen, saadaan radioaktiivista hajoamista kuvaava hajoamislaki

$$N(t) = N_0 e^{-\lambda t}, \quad (2)$$

jossa t on aika, N on näytteessä olevien radioaktiivisten nuklidien määrä, N_0 on määrä ajanhetkellä $t = 0$.

Hajoamislain eksponentiaalisesta luonteesta johtuen aika, jossa radioaktiivisten nuklidien määrä vähenee puoleen, on vakio riippumatta kyseisten nuklidien alkuperäisestä määrästä. Tätä aikaa kutsutaan kyseisen nuklidin puoliintumisajaksi $t_{1/2}$, joka saadaan ratkaistua sijoittamalla yhtälöön (2) arvo $N = N_0 / 2$. Tällöin puoliintumisajaksi saadaan

$$t_{1/2} = \frac{\ln 2}{\lambda}.$$

2.2. Aktiivisuus

Radioaktiivisen aineen määrää ilmaistessa käytetään usein suureena aktiivisuutta, koska se kertoo ainemäärää tai painoa paremmin, miten voimakkaasta säteilylähteestä on kyse. Aktiivisuuden yksikkö on Becquerel (Bq), joka vastaa yhtä radioaktiivista hajoamista sekunnissa [2]. Lähteen aktiivisuus on verrannollinen sen sisältämän radionuklidin määrään nuklidin hajoamisvakion kautta, joten aktiivisuus A vähenee ajan kuluessa samoin kuin radionuklidin määrä (yhtälö 3),

$$A(t) = \lambda N = A_0 e^{-\lambda t}. \quad (3)$$

Usein radioaktiivisessa hajoamisessa syntyvä tytärudin on myös radioaktiivinen. Tällaista useamman radioaktiivisen emo- ja tytärtyimien sarjaa kutsutaan hajoamissarjaksi. Näytteen aktiivisuutta tarkasteltaessa on huomioitava, että mitattu aktiivisuus voi koostua useamman samaan sarjaan kuuluvan nuklidin kontribuutiosta, eikä näytteen kokonaisaktiivisuus aina käyttäydy samoin, kuin yksittäistä nuklidia sisältävässä näytteessä. Useissa mittausjärjestelyissä eri nuklidit on kuitenkin mahdollista erottaa, jos niiden hajoamisista vapautuvalla säteilyllä on eri energiat.

2.3. Ionisoivan säteilyn lajit

Ionisoiva säteily on säteilyä, jonka energia on riittävän suuri ionisoimaan atomeja. Ionisoiva säteily voidaan jakaa sähkömagneettiseen säteilyyn ja hiukkassäteilyyn, joista hiukkassäteily voidaan edelleen jakaa varattuun ja varauksettomaan hiukkassäteilyyn. Varauksellista hiukkassäteilyä ovat raskaat varatut hiukkaset, kuten protonit ja alfahiukkaset, sekä nopeat elektronit. Varauksetonta hiukkassäteilyä on puolestaan esimerkiksi neutronisäteily. Sähkömagneettista ionisoivaa säteilyä ovat gamma- ja röntgensäteily [4]. Tässä kappaleessa käsitellään yleisimpiä säteilytyyppejä ja niiden ominaisuuksia.

Alfahiukkanen on kahdesta protonista ja kahdesta neutronista koostuva He-ydin, jolla on hyvin suuri sidosenergia. Alfahajoaminen johtuu Coulombin vuorovaikutuksesta, joka työntää emoytimessä muodostuneen alfahiukkasen irti emoytimeistä. Tällöin alfa-aktiivinen radionuklidi pääsee alemmalle energiatilalle irtautumalla alfahiukkasesta. Usein alfahajoamisen lopputuote jää hajoamisen jälkeen virittyneelle tilalle, jolta se jatkaa hajoamisprosessia alfa- tai beetahajoamalla tai luovuttamalla gammakvantin tai konversioelektronin. [3]

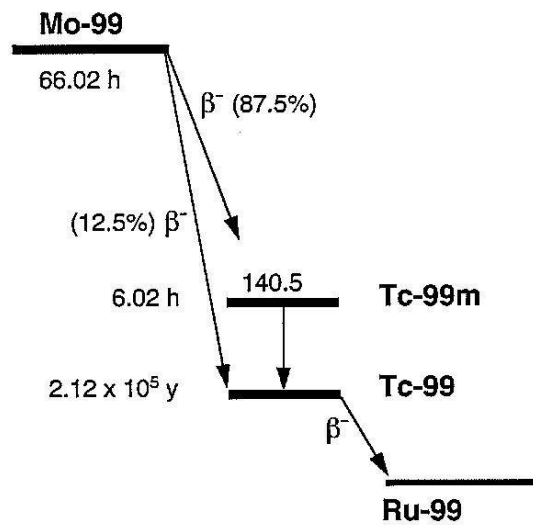
Beetahajoaminen tapahtuu ytimissä, joissa protonien ja neutronien suhde ei ole ytimen sidosenergian maksimoimisen kannalta optimaalinen. β^- -hajoamisessa neutroni muuttuu protoniksi ja ytimeistä emittoituu elektroni ja antineutriino. β^+ -hajoamisessa puolestaan protoni muuttuu neutroniksi, jolloin emittoituu positroni ja neutriino. Hajoamisprosessit on kuvattu yhtälöissä 4 ja 5. Prosessissa vapautuva energia päättyy lopputuotteiden liike-energiaksi, jolloin syntynyt elektroni tai positroni (eli beetahiukkanen), sekä neutriino tai antineutriino, sinkoutuvat suurella liike-energialla ulos ytimeistä.

Elektronisieppaus tapahtuu, kun ydin sieppaa sitä kiertävän elektronin. Tällöin yksi ytimen protoni muuttuu neutroniksi ja ytimeä emittoituu neutriino (yhtälö 6). Elektronisieppaus on mahdollinen, kun ytimessä on sen vakauden kannalta turhan paljon protoneja. Elektronisieppaus on β^+ -hajoamisen kanssa kilpaileva prosessi, mutta ytimissä, joissa hajoamisessa ei vapaudu tarpeeksi energiaa synnyttämään positronia, se on ainoa hajoamismuoto. Elektronisieppauksen jälkeen atomin elektroniverhossa on aukko useimmiten matalammilla elektronikuorilla. Tästä seuraa elektroniverhon uudelleenjärjestäytyminen, josta vapautuu röntgensäteitä tai Augerelektroneja. [3]



Virittyneelle tilalle päätyneet nuklidit siirtyvät alemmalle energiatilalle luovuttamalla tilojen välisen energian gammakvantin tai konversioelektronin muodossa. Sisäinen konversio on gammahajoamisen kanssa kilpaileva prosessi, jossa atomin elektroni saa tilojen välisen energian omaksi liike-energiäkseen ja poistuu atomista. Sisäinen konversio on yleinen silloin, kun virittyneen tilan energia on pieni ja tilan spin ja pariteetti eivät mahdollista yksittäisen gammakvantin emittoimista. Gammahajoaminen on useimmiten seurausta muista hajoamisprosesseista, joiden seurauksena ydin on päätyneet virittyneeseen tilaan. Näiden viritystilojen purkautuminen tapahtuu usein niin nopeasti, että sitä pidetään osana muita hajoamisprosesseja. [3]

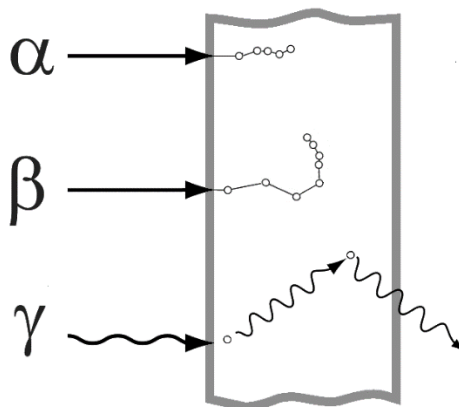
Joissain tapauksissa hajonnut ydin voi kuitenkin päätyä metastabiilille tilalle. Metastabiili tila on niin pitkäikäinen virittynyt tila, että sillä ajatellaan olevan erillinen puoliintumisaika. Yksi esimerkki tällaisesta on lääketieteessäkin käytetty ^{99m}Tc . Emodin ^{99}Mo β^- -hajoaa teknetiumiksi, mutta suurin osa hajoamisista johtaa metastabiilille tilalle (Kuva 3). Metastabiili ^{99m}Tc -tila puolestaan purkautuu n. 6 tunnin puoliintumisajalla perustilalle ^{99}Tc , joka puolestaan jatkaa hajoamissarjaa ^{99}Ru :ksi.



Kuva 3: Teknetiumin metastabiili tila ($^{99\text{m}}\text{Tc}$) ^{99}Mo :n hajoamisketjussa. Suuri osa hajonneista ^{99}Mo ytimistä päätyy metastabiilille tilalle, joka purkautuu pääasiassa 140,5 keV γ -hajoamisen kautta. Kuva lähteestä [5].

2.4. Vuorovaikutukset

Koska erilaisissa hajoamisissa syntyy hyvin erityyppistä säteilyä, myös niiden vuorovaikutukset ympäröivän materian kanssa eroavat toisistaan (Kuva 4). Säteilyä mitattaessa ja siltä suojauduttaessa on otettava huomioon erilaisten säteilytyyppien vuorovaikutustavat ja sen kautta myös niiden erityyppiset käytökset ja kantamat.



Kuva 4: Säteilyn eteneminen väliaineessa. Alfahiukkanen etenee lyhyen matkan kohtalaisen suoraan, vuorovaikuttaen tiheästi väliaineen kanssa. Beetahiukkanen vuorovaikuttaa harvemmin ja sen yksittäisillä vuorovaikutuksilla on suurempia vaikutuksia hiukkasen liike-energiaan. Gammakvantti läpäisee eniten väliainetta vuorovaikutusten välillä ja päätyykin tässä tapauksessa läpäisemään väliaineen kokonaan. Kuva muokattu lähteestä [6].

2.4.1. Alfahiukkaset

Raskaat alfahiukkaset vuorovaikuttavat coulombin vuorovaikutuksella väliaineen elektroneihin ja hidastuvat luovuttamalla niille energiaansa. Vuorovaikutusten välimatka on homogeenisessä aineessa lähes vakio ja vuorovaikutuksia tapahtuu tasaisesti joka suuntaan. Koska yksittäiset vuorovaikutukset vaikuttavat alfahiukkasen suuntaan ja nopeuteen vain vähän, on alfahiukkasten kantama tietyssä väliaineessa jotakuinkin vakio ja kulkusuunta pysyy miltei muuttumattomana. Hiukkasen pysähtyminen on siis monien törmäysten summa. Alfahiukkasen törmäykset ytimiin saavat aikaan voimakkaampia vuorovaikutuksia, mutta ne ovat hyvin harvinaisia. [4]

Alfasäteilyltä suojautumisessa tärkeintä on estää alfa-aktiivisen aineen päätyminen elimistöön. Elimistön ulkopuolelta tuleva alfasäteily pysähtyy nopeasti iholle, eikä kantama ilmassakaan ole kuin enimmillään muutamia senttimetrejä. Elimistön sisälle päästyään suurenergiset alfahiukkaset tekevät kuitenkin paljon vahinkoa [7]. Itse alfahiukkasilta suojautumisen lisäksi on kuitenkin otettava huomioon myös hajoamisen tytärytimet, mikä voi tarkoittaa varautumista esimerkiksi korkeaenergiseseen beeta- tai gammasäteilyyn. Alfahiukkasen ionisoimat väliaineen atomit voivat puolestaan tuottaa röntgensäteilyä, mikäli alfahiukkanen on poistanut atomista alhaisen kuoren elektronin ja muut elektronit järjestäytyvät uusille tiloille.

2.4.2. Beetahiukkaset

Myös beetahiukkaset, eli elektronit ja positronit, vuorovaikuttavat pääasiassa väliaineen elektronien kanssa. Koska vuorovaikuttavat hiukkaset ovat nyt samaa massaluokkaa, aiheuttavat vuorovaikutukset suuriakin muutoksia beetahiukkasen kulkusuuntaan. Tästä johtuen yksittäisen elektronin kantama väliaineessa vaihtelee paljon enemmän kuin alfahiukkasella, joka kulkee pääsääntöisesti suoraan.

Elektronien kanssa vuorovaikuttamisen lisäksi beetahiukkasen luovuttaa energiaa jarrutussäteilynä muuttaessaan kulkusuuntaansa. Jarrutussäteily on sähkömagneettista säteilyä, jota syntyy hiukkasen hidastuessa tai kiihtyessä nopeasti. Useimmiten beetahiukkasen tuottama jarrutussäteily on niin matalaenergistä, että suurin osa kvanteista absorboituu lähelle lähtöpistettään [4]. Mahdollisuus korkeaenergiseseen jarrutussäteilyyn on kuitenkin syytä ottaa huomioon säteilysojelua suunniteltaessa.

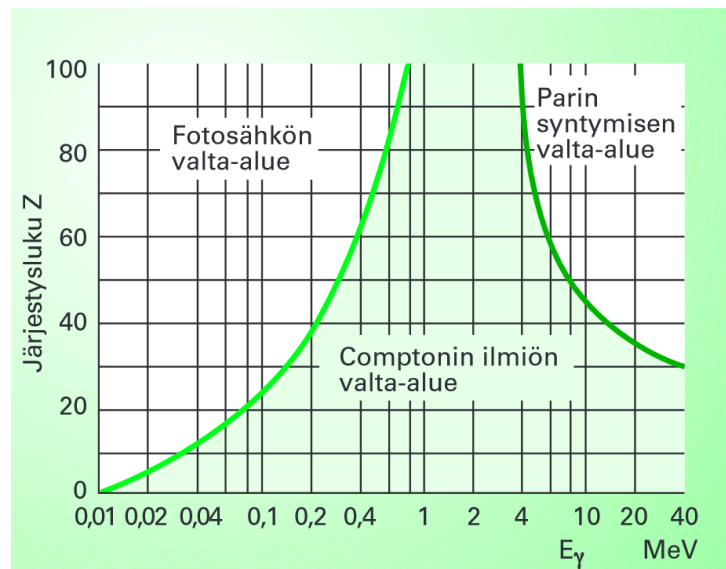
β^+ -hajoamisessa on myös huomioitava, että hajoamisessa syntyvä positroni annihiloituu ratansa päässä. Kun positroni hidastuu tarpeeksi, se annihiloituu väliaineen elektronin kanssa, jolloin elektroni ja positroni lakkaavat olemasta ja syntyy kaksi vastakkaisiin suuntiin etenevää 511 keV fotonia. Nämä fotonit ovat varsin läpäisykykyisiä, joten ne on syytä ottaa huomioon säteilysojelussa β^+ -aktiivisia aineita käsiteltäessä.

Beetasäteilijät ovat alfasäteilijöiden tapaan vaarallisimpia joutuessaan elimistöön, jolloin ne pystyvät tekemään lyhyen kantamansa puitteissa vahinkoa soluille [7]. Suurenergisimmät beetahiukkaset kykenevät myös läpäisemään muutamia millimetrejä vettä, eli ne pystyvät

läpäisemään ihon pintakerrokset ja aiheuttamaan vahinkoa eläville ihosoluille [2]. Tämän vuoksi beetasäteilijöistä johtuva ulkopuolelta tuleva altistus täytyy myös ottaa huomioon säteilysuojelussa. Ulkopuolelta tulevan beetasäteilyn pysäyttämiseen riittää kuitenkin huomattavasti kevyempi suojaus, kuin gammasäteilyn kohdalla. Lisäksi kevyen materiaalin käyttäminen suojauksessa vähentää jarrutussäteilyä.

2.4.3. Gammakvantit

Gammasäteillä, tai fotoneilla, on laaja kirjo erilaisia vuorovaikutuksia, mutta säteilyn mittaamisen kannalta olennaisimmat ovat valosähköinen ilmiö, Compton sironta ja parinmuodostus. Näistä valosähköisessä ilmiössä ja parinmuodostuksessa fotoni lakkaa kokonaan olemasta ja Compton sironnassa sen suunta ja energia muuttuvat usein merkittävästi. Vuorovaikutusten laji riippuu sekä fotonin energiasta että väliaineen järjestysluvusta (Kuva 5). Gammakvantit eivät kuitenkaan vuorovaikuta kaiken eteensä osuvan kanssa. Gammasäteilyn hyvä läpäisevyys perustuu siihen, että fotonien todennäköisyys vuorovaikuttaa väliaineen kanssa on pieni. [4]



Kuva 5: Fotonin vallitsevat vuorovaikutustavat väliaineen järjestysluvun ja fotonin energian funktiona. Kuva lähteestä [2]

Absorptiossa, eli valosähköisessä ilmiössä fotoni luovuttaa koko energiansa väliaineen atomille ja lakkaa olemasta [4]. Atomista energia poistuu elektronin liike-energiana, joka yleensä riittää irrottamaan kyseisen elektronin atomin vaikutuspiiristä. Koska irtoava elektroni on usein peräisin atomin sisemmiltä kuorilta, seuraa valosähköisen elektronin irtoamista sarja karakteristisia röntgensäteitä tai Augerelektroneja, jotka kuitenkin yleensä matalaenergisinä absorboituvat lähelle lähtöpistettään. [2, 4]

Compton sironnassa fotoni osuu elektroniin, mutta ei luovuta sille koko energiaansa. Tällöin seurauksena on fotonin suunnan muutos ja aallonpituuden kasvu, joka johtuu energian siirtymisestä elektroniin. Compton sironta on todennäköisempää suurienergisille fotoneille, kun

taas valosähköinen ilmiö on dominoiva pienemmillä energioilla (Kuva 5). Usein suurienergiset fotonit luovuttavatkin ensin osan energiastaan sironnalla, kunnes lopulta absorboituvat valosähköisesti. [2, 4]

Parinmuodostuksessa korkeaenerginen fotoni hajoaa elektroni-positroni-pariksi, ja luovuttaa parin luomisesta ylijääneen energian niiden liike-energiaksi. Parinmuodostus tulee mahdolliseksi vasta, kun fotonin energia ylittää kahden elektronin lepomassan (1,02MeV) ja on todennäköinen vasta kymmenien MeV:n energioissa ja raskailla alkuaineilla [2, 4].

Fotonin todennäköisyys pysähtyä tai muuttaa suuntaa jonkin vuorovaikutuksen seurauksena on vakio tasalaatuisessa väliaineessa. Tästä seuraa, että fotonille esteen läpäisytodennäköisyys esteen paksuuden funktiona, on eksponentiaalisesti laskeva käyrä.

3. Säteilyn mittaaminen

3.1. Mittalaitteet

Erilaisia säteilymittareita on paljon, joten on olennaista valita kuhunkin käyttötarkoitukseen parhaiten soveltuva. Mittarilta vaadittaviin ominaisuuksiin vaikuttaa esimerkiksi mitattavan säteilyn laatu ja määrä, sekä se, mitä säteilystä halutaan saada selville. Esimerkiksi korkea-aktiivisen lähteen aktiivisuuden määrittämiseen käytetään varsin erityyppistä mittaria kuin alueellisen taustasäteilyn mittaamiseen. Mittarin valinnassa huomioon otettavia ominaisuuksia voivat olla esimerkiksi energioresoluutio, kuollut aika ja tehokkuus [4]. Energiaroluutiolla tarkoitetaan sitä, kuinka paljon kahden eri säteilyenergian tulee erota, jotta mittari vielä pystyy erottamaan ne toisistaan. Kuollut aika puolestaan on aika, joka mittarilla kestää palautua havaitusta pulssista ennen kuin seuraava voidaan mitata. Tehokkuus taas kertoo, kuinka suuren osan ilmaisinta kohti tulevasta säteilystä mittari kykenee havaitsemaan. Näiden lisäksi useimmissa tapauksissa on otettava huomioon muitakin tekijöitä, kuten mittarin hinta, koko ja tarve säteilysuojaukselle.

3.2. Aktiivisuuden mittaaminen

Aktiivisuuden mittaaminen perustuu tarkasteltavasta säteilylähteestä tulevan säteilyn laskemiseen vakioidussa mittausasetelmassa. Kun tunnetaan mittausasetelman tehokkuus, eli kuinka suuren osan tunnetun lähteen aktiivisuudesta mittari havaitsee, voidaan tarkasteltavan lähteen aktiivisuus määrittää mittarin antaman tapahtumafrekvenssin ja tehokkuuskertoimen avulla.

Näytteen aktiivisuutta mitattaessa mittarin olennaisin ominaisuus on tehokkuus. Jos halutaan mitata suuria aktiivisuuksia, on syytä ottaa huomioon myös mittarin kuollut aika. Energiaroluutiota voidaan vaatia, mikäli on olennaista erottaa eri radionuklidien säteilyenergiat tarkasti toisistaan. Jos näytteessä olevien nuklidien energiat eroavat selkeästi toisistaan tai mitataan vain yksittäistä nuklidia, on heikkokin energiaroluutio riittävä.

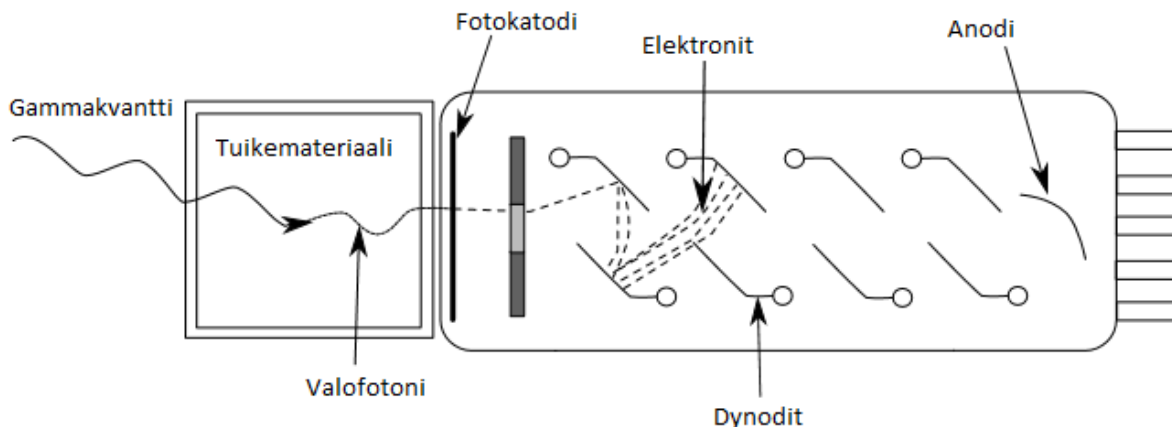
Mittarissa olevan ilmaisimen tehokkuuden lisäksi mittausjärjestelmän tehokkuuteen vaikuttaa mittausasetelman geometria. Esimerkiksi ionisaatiokammioon perustuvassa annoskalibraattorissa pyritään siihen, että ionisaatiokammio ympäröi näytteen mahdollisimman suurelta osin. Näin isotrooppisesti suuntautuvasta säteilystä mahdollisimman suuri osa saadaan kulkemaan kammion läpi. Tasomaisiin ilmaisimiin perustuvissa mittareissa mittausasetelman tehokkuutta saadaan lisättyä asettamalla näyte mahdollisimman lähelle ilmaisinta.

Näytteen aktiivisuutta määritettäessä tulee myös tuntea kyseessä oleva radionuklidi ja sen hajoamistavat. Useilla nuklideilla on monia hajoamistapoja, joista osa, esimerkiksi alfaemissio, voivat olla sellaisia, että käytettävä ilmaisim ei kykene niitä havaitsemaan. Jotta voitaisiin selvittää näytteen aktiivisuus, täytyy selvittää, mikä osuus näytteen hajoamisista pystytään havaitsemaan. Lisäksi tulee tuntea mittarin tehokkuus kyseiselle mittausasetelmalle, sekä ilmaisimen tehokkuus mitattavalle säteilylajille. Nämä voidaan saada selville erillisillä kalibrintimittauksilla, eli määrittämällä mittarin vaste tunnettuihin kalibrintilähteisiin. Tunnettujen tehokkuuskertoimien perusteella voidaan saadusta mittaustuloksesta laskea näytteen todellinen aktiivisuus.

3.3. Tuikeilmaisimet

Tuikeilmaisimet perustuvat materiaaleihin, jotka emittoivat valoa absorboituaan gammakvantin. Tuikemateriaali virittyy absorboidessaan gammakvantin ja viritystilan purkautuessa vapautuu kvantteja, joiden aallonpituus on näkyvän valon alueella. Tämä valontuikahdus voidaan havaita tuikemateriaaliin kiinnitettyllä valomonistinputkella, joka muuttaa valon sähköiseksi signaaliksi (*Kuva 6*). Valomonistinputken tuottama analoginen jännitepulssi vahvistetaan ja muutetaan sen jälkeen digitaaliseksi tiedoksi analogi-digitaalimuuntimella (ADC, Analog-to-Digital Converter). Digitaaliset tiedot pulssin korkeudesta tallennetaan monikanava-analysaattoriin (MCA, Multi Channel Analyzer), jonka tuottamia spektrejä voidaan tallentaa ja käsitellä jälkeenpäin PC:llä. [4]

Täydellinen tuikemateriaali muuttaisi absorboimansa säteilyn nopeasti ja tehokkaasti valontuikahdukseksi ja olisi lisäksi optisilta ominaisuuksiltaan sopiva ja helppo valmistaa. Koska kaikin puolin täydellistä materiaalia ei ole, täytyy kuhunkin mittaustarpeeseen valita ominaisuuksiensa puolesta sopivin. Yleisimmät tuikemateriaalit ovat epäorgaaniset kiteet, kuten NaI(Tl), sekä orgaaniset nesteet ja muovit. Epäorgaanisilla aineilla on useimmiten parempi valontuotto ja lineaarisuus, lisäksi niiden suurempi tiheys soveltuu hyvin gammaspektroskopiaan. Epäorgaaniset aineet puolestaan palautuvat tuikahduksista nopeammin ja soveltuvat paremmin beeta- ja neutronisäteilyn havaitsemiseen. [4]



Kuva 6: Tuikeilmaisimen rakenne ja toimintaperiaate. Gammakvantin aiheuttama tuikemateriaalin viritystila purkautuu näkyvän valon fotonina. Osuessaan valomonistinputken fotokatodille valokvantti irrottaa siitä yksittäisiä elektroneja, jotka ohjataan korkeajännitteellä sarjaan dynodeja. Jokaisella dynodilla kiihtynyt elektroni irrottaa dynodista useita elektroneja ja näin moninkertaistuva elektroniryöpäs muodostaa lopulta havaittavan sähköisen pulssin valomonistinputken ulostuloon. Kuva lähteestä [8], tekstit suomennettu.

Yksi yleisimmin käytetyistä tuikeilmaisimista on talliumilla rikastettuun natriumjodidikiteeseen (NaI[Tl]) perustuva ilmaisim. Natriumjodidi on yksi herkimmin valoa tuottavista tuikeilmaisimateriaaleista ja sen yleistymisen myötä hintakin on matala. Natriumjodidin huono puoli on suhteellisen hitaan tuikahduksen lisäksi kiteessä ilmenevä lievä jälkihehku. Nopeilla pulssitaajuuksilla hehku voi kertautua ja vaikuttaa mittaustuloksiin [4].

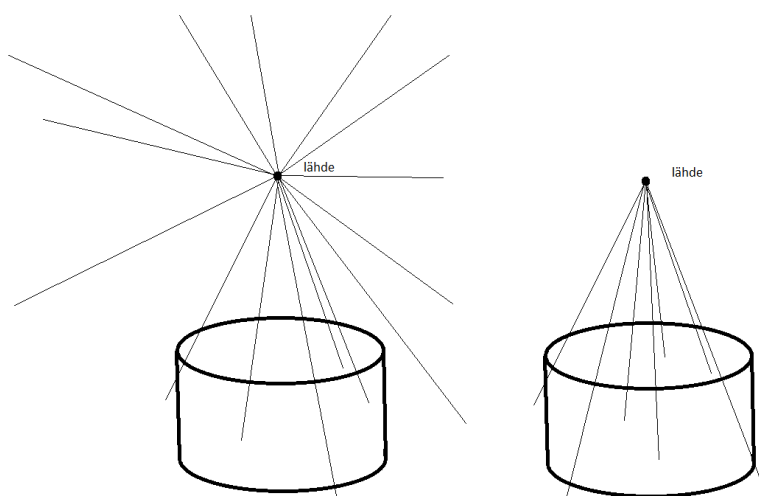
Tuikeilmaisimissa käytettävä tuikemateriaali pyritään valitsemaan siten, että sen tuottaman valotuikahduksen intensiteetti on verrannollinen sen absorboiman gammakvantin energiaan. Kun myös valomonistinputken tuottaman sähköisen signaalin amplitudi on verrannollinen sen havaitsemien fotonien määrään, on lopullisen signaalin amplitudi siis verrannollinen alkuperäisen gammakvantin energiaan. Selvittämällä kalibroitimittauksilla signaalin ja alkuperäisen kvantin energian välinen riippuvuus, voidaan tuikeilmaisimella siis mitata säteilyn energiaa. Ilmaisimen tuottama signaali syötetään usein MCA:lle, joka lajittelee ja tallentaa signaalit niiden voimakkuuden mukaan. Tällöin voidaan muodostaa mitatun säteilyn energiaspektri.

3.4. Ilmaisimen kalibrointi

Jotta säteilymittarilla voidaan saada mielekkäitä mittaustuloksia, se täytyy kalibroida. NaI-tuikeilmaisimelle täytyy tehdä tehokkuuskalibrointi ja energiakalibrointi. Tehokkuus määrittää sen, kuinka monta tapahtumaa mittari havaitsee suhteessa tulevan säteilyn määrään. Tehokkuuskerroin on tärkeä, kun halutaan selvittää mitattavan kohteen aktiivisuutta. Energiakalibroinnissa puolestaan etsitään kerroin ilmaisimen tuottaman signaalin voimakkuuden ja mitatun kvantin energian välille. Tällöin on mahdollista määrittää mitattavasta kohteesta tulevan säteilyn energia.

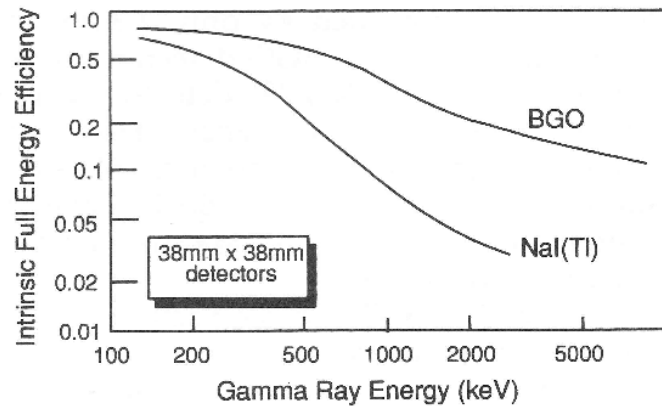
3.4.1. Tehokkuuskalibrointi

Ilmaisimen tehokkuus riippuu monista muuttujista, joten se voidaan määritellä eri tavoin. Yleensä ilmaisimen absoluuttinen tehokkuus määritellään mittarin havaitsemien tapahtumien ja mitatun lähteen tuottamien kvanttien määrien suhteeksi (Kuva 7). Mittarin sisäinen tehokkuus ottaa huomioon mittauksen geometrian, eli lähteen tuottamien kvanttien määrään lasketaan vain ne, jotka ylipäättään osuvat mittariin (Kuva 7) [4]. Gammaskpektrometriassa tärkein tehokkuuden määritelmä on täysenergiapiikin tehokkuus. Täysenergiapiikin tehokkuutta määritettäessä otetaan huomioon vain ne ilmaisimen havaitsemat tapahtumat, jossa kvantti luovuttaa koko energiansa ilmaisimelle [9]. Tämä määritelmä on tärkeä spektrometriassa, koska täyden energian luovuttaneiden kvanttien laskeminen on käytännössä ainoa tapa erotella tietyn radionuklidin tuottama osuus spektristä.



Kuva 7: Ilmaisimen tehokkuus ottaen huomioon kaikki lähteestä lähtevä aktiivisuus (vasen) tai ainoastaan ilmaisimeen osuva säteily (oikea).

Absoluuttinen tehokkuus riippuu mittarin rakenteellisten seikkojen lisäksi saapuvan säteilyn energiasta sekä mittausasetelman geometriasta. Geometriasta sikäli, että mitä kauempana mitattava lähde on ilmaisimesta, sitä pienempi osa säteilystä osuu ilmaisimeen. Tällöin havaittujen tapahtumien määräkin luonnollisesti pienenee. Ilmaisimen tehokkuus on myös riippuvainen mitattavan säteilyn energiasta. Esimerkiksi NaI-tuikeilmaisimen tehokkuus laskee mentäessä kohti suurienergistä säteilyä (Kuva 8).



Kuva 8: Kahden eri tuikemateriaalin sisäinen tehokkuus gammakvantin energian funktiona. Kuva lähteestä [9].

Tehokkuuskalibrointi voidaan suorittaa mittaamalla sarja tunnettuja erienergisä säteilylähteitä. Näin saadaan useita tehokkuuden arvoja eri energioille [4]. Optimaalista olisi, jos tehokkuuskalibrointi voitaisiin suorittaa sillä nuklidilla, jota mittarilla aiotaan mitata, mutta jos mitattujen pisteiden määrä on riittävä, on myös mahdollista interpoloida arvot niiden välille [9]. Tehokkuuskalibrointi voidaan toteuttaa myös laskennallisesti hyödyntäen tunnettuja vuorovaikutusten todennäköisyyksiä [4, 10]. Tehokkaiden tietokoneiden myötä Monte Carlo-menetelmät [11] ovat yleistyneet ilmaisimien laskennallisessa tehokkuuskalibroinnissa.

Tehokkuutta mitattaessa on syytä huomioida, että mittausasetelma voi vaikuttaa merkittävästi mittaustuloksiin. Esimerkiksi lähteen etäisyys ilmaisimesta, sekä lähteen muoto ja tiheys vaikuttavat tuloksiin. Näitä muutoksia on mahdollista korjata tekemällä sarja mittauksia muuttaen yksittäisen muuttujan arvoa ja sovittamalla tuloksiin käyrä. Tällöin ilmaisimen tehokkuutta tietylle mittausasetelmalle voidaan arvioida kalibrointikäyrästä avulla. Jos käytettäviä mittausasetelmia on vähän, on kuitenkin parasta määrittää tehokkuus nimenomaan käytettäville mittausasetelmille. Tällöin vältytään käyrien interpoloinnilta. [9]

3.4.2. Energiakalibrointi

NaI-tuikemateriaalilla voidaan mitata siihen pysähtyneen gammakvantin energia, koska kiteen tuottaman valon määrä on verrannollinen kvantin kiteelle luovuttaman energian määrään. Valotuikahdus puolestaan vahvistetaan valomonistinputkella, jonka vahvistuskerroin on verrannollinen putken jännitteeseen. Koska koko ketjun mallintaminen ja kvantin energian ja sähköimpulssin voimakkuuden välisen kertoimen määrittäminen laskennallisesti on haastavaa, kalibroidaan tuikeilmaisimien yleensä erillisillä kalibrointilähteillä. [9]

Energiakalibroinnissa mitataan useita tunnettuja nuklideja ja tarkastellaan nuklidien tuottaman säteilyn energian ja mittalaitteen tuottaman signaalin voimakkuuden suhdetta. Energian ja signaalin suhde ei ole kuitenkaan täysin lineaarinen, joten paras kalibrointi yksittäiselle energialle

saadaan käyttämällä samaa nuklidia, jota ilmaisimella aiotaan mitata [4]. Aina tämä ei kuitenkaan ole mahdollista ja tällöin kalibroinnissa joudutaan tyytymään kalibrointilähteisiin, joiden tuottaman säteilyn energia on lähellä mitattavan nuklidin säteilyn energiaa. Monilla ilmaisimilla, kuten NaI-tuikeilmaisimella, suhde on kuitenkin niin lähellä lineaarista, että kalibrointi voidaan suorittaa mittaamalla kaksi nuklidia, joiden tuottaman säteilyn energiat ovat lähellä toisiaan. Tällöin niiden väliin jäävät energiat voidaan interpoloida olettaen energian ja signaalin suhteen olevan lineaarinen [4]. Joissain sovelluksissa on tarpeen mitata useita eri energioita, jolloin mittausalue pyritään kattamaan muutamalla kalibrointilähteellä ja interpoloimalla niistä saatavien säteilyenergioiden väliin jäävät energiat.

4. Radioaktiiviset päästöt ja niiden suodattaminen

4.1. Päästöjen olomuodot

Radioaktiiviset aineet esiintyvät ilmvirrassa pääasiassa kaasuna, aerosoleina sekä hiukkasiin takertuneena. Eri esiintymismuotojen ominaisuudet tulee ottaa huomioon, kun suunnitellaan niiden suodattamista ilmvirrasta tai näytteenottoa ilmvirran päästöjen mittaamiseksi.

Aerosoli on ilmasta sekä pienistä kiinteistä hiukkasista tai nestepisaroista koostuva seos. Laboratorio-olosuhteissa aerosoleja voi muodostua esimerkiksi nesteitä siirrettäessä syntyvistä roiskeista. Aerosolit kulkeutuvat ilmvirran mukana ja päätyvät siten ilmanvaihtoon. Nesteiden lisäksi radioaktiiviset aineet voivat tarttua myös kiinteisiin ilmvirran mukana kulkeviin hiukkasiin. Puhdastilatoiminnassa hiukkasia on yleensä hyvin vähän, mutta ne on silti syytä ottaa huomioon päästöjen mittausta suunniteltaessa.

Jotkut radioaktiiviset aineet tai yhdisteet voivat myös esiintyä kaasumaisina, jolloin ne kulkeutuvat muun ilmvirran mukana ilmanvaihtoon. Kaasumaisia yhdisteitä voi haihtua radionuklideja sisältävästä liuksesta. Lisäksi ilmvirran mukana olevat kaasut voivat reagoida kemiallisesti ilmankosteuden tai muiden ilmvirran mukana kulkevien aineiden kanssa muodostaen uusia radioaktiivisia yhdisteitä.

4.2. Suodattaminen

Suodattamisella tarkoitetaan tässä työssä haluttujen aineiden erottamista ilmvirrasta ohjaamalla haluttu ilmavirta sopivan suodattimen läpi. Suodatinta valittaessa tulee ottaa huomioon, mitä ilmasta halutaan erottaa ja mitä muita vaatimuksia suodattimelle asetetaan.

Hiukkasiin tarttuneet radioaktiiviset aineet on suhteellisen helppo erottaa ilmvirrasta tehokkaalla hiukkassuodattimella (esimerkiksi HEPA- eli High Efficiency Particulate Air-suodatin). Tällöin radioaktiivinen aine jää useimmiten hiukkasen mukana suodattimeen. Myös aerosolipäästöjä voidaan suodattaa tehokkaalla hiukkassuodattimella. Hiukkassuodatin on syytä valita sen mukaan,

mitä siihen odotetaan päätyvän, ja mitä sen halutaan pidättävän. Suodattimen valinnassa tulee ottaa huomioon mm. suodattimen tehokkuus erikokoisille hiukkasille, suodattimen pinta-ala, hiukkasten ja suodatinmateriaalin huokosten koko sekä virtausvastus [12].

Kaasujen erottaminen ilmavirrasta on erilaista kuin kiinteiden hiukkasten tai aerosolien. Kaasuja suodatettaessa tulee tietää, mitä ilmavirrasta halutaan erottaa, ja millaisella menetelmällä se onnistuu. Radioaktiivisia päästöjä suodatettaessa yksi yleisimmistä kyseeseen tulevista aineista on jodi, jonka radioaktiiviset isotoopit muodostavat merkittävän osan ydinvoimaloiden ja myös monien muiden radionuklideja käsittelevien laboratorioden kaasumaisista päästöistä [13]. Alkuainemuotoisen jodikaasun (I_2) suodattamiseen soveltuu hyvin aktiivihilisuodatin. Osa jodin muodostamista kaasumaisista yhdisteistä tarttuu kuitenkin huonosti normaaliin aktiivihilleen. Tällöin tarvitaan esimerkiksi kaliumjodidi- tai trietyleenidiamiini-kyllästettyä (KI- tai TEDA-kyllästettyä) aktiivihiltä, joka sitoo nämä yhdisteet kemiallisesti suodattimeen [14]. Kyllästetty aktiivihili säilyttää myös paremmin ominaisuutensa kosteaa ilmaa suodatettaessa, koska kemiallinen sitomisprosessi ei ole herkkä kosteudelle. Tavallisen, pelkkään adsorptioon perustuvan, aktiivihilisuodattimen ominaisuudet kärsivät kosteuden vaikutuksesta, kun vesi tukkii hiilen huokosia ja vähentää näin käytössä olevaa pinta-alaa [14].

Perinteisin aktiivihilisuodatin sisältää rakeista aktiivihiltä, jonka teho perustuu sen suureen pinta-alaan. Aktiivihilen rakenne on erittäin huokoista, mikä mahdollistaa jopa useiden neliömetrien adsorptiopinta-alan alle 0.1mm kokoiselle aktiivihilirakeelle. Pelkkään hiileen perustuvien suodattimien lisäksi on kehitelty myös kuiturakenteisia suodattimia, jossa kuidut toimivat tukirakenteena heikommalle aktiivihillelle [15]. Myös muita huokoisia aineita, kuten zeoliitteja, silikageeliä ja alumiinioksidia [13], on käytetty vastaavanlaisiin adsorptioon perustuviin suodattimiin. Kuten aktiivihiltä, näitäkin aineita joudutaan kyllästäämään esimerkiksi hopealla tai kuparilla, jos halutaan saada kiinni huonosti adsorboituvia yhdisteitä.

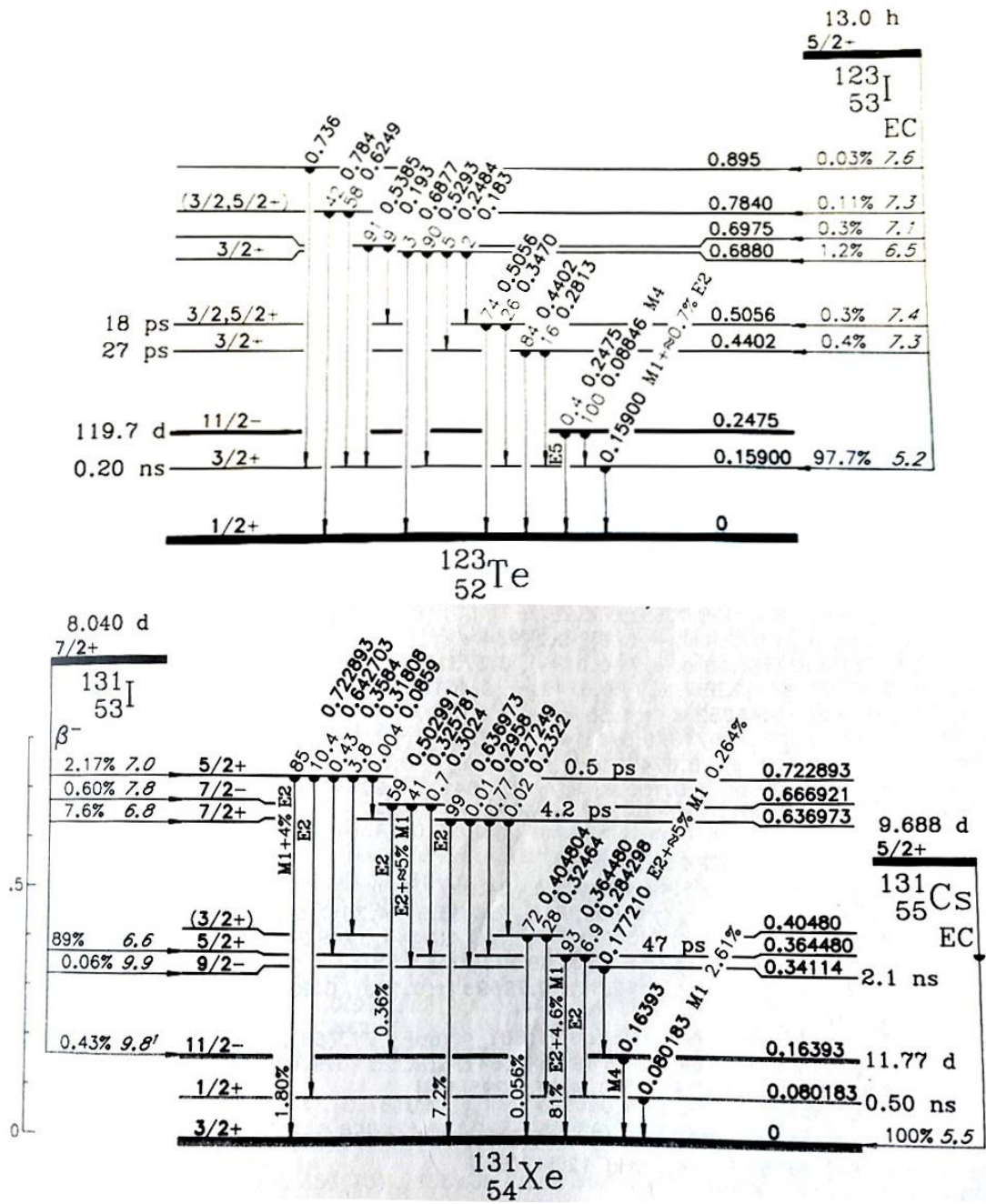
4.3. Jodin isotoopit

Jodin isotooppeja tunnetaan useita, mutta MAP käyttää niistä Tikkakosken toimipisteellä vain kahta (^{123}I ja ^{131}I). Näistä ^{123}I hajoaa elektronisieppauksella, josta seuraava virittynyt tila purkautuu useimmiten 159 keV gammaemission kautta (Kuva 9). Sen puoliintumisaika on noin 13,22 tuntia. Toinen käytetty isotooppi, ^{131}I , puolestaan hajoaa 8,02 päivän puoliintumisajalla useiden eri beetahajoamisten kautta (Kuva 9). Hajoamisreiteistä yleisin on korkeaenerginen 606 keV beeta. Beetahajoamiset jättävät ytimen virittyneeseen tilaan, jonka purkautuessa emittoituu eri energisiä gammakvantteja. Yleisin ^{131}I :n emittoimista gammakvanteista on energialtaan 364,5 keV ja sitä käytetäänkin usein ^{131}I :n aktiivisuuden mittaamiseen.

Jodin määrän monitoroiminen poistoilmasta on monin tavoin haasteellista. Jodia haihtuu liuoksista huoneenlämmössä [16], mutta se myös tiivistyy takaisin varsinkin viilleille pinnoille. Tämä voi aiheuttaa jodin kertymistä poistoputkistoon ja näytteenottoputkiin, sekä kulkeutumisen hidastumista toistuvan tiivistymisen ja haihtumisen seurauksena. Jodi voi myös tiivistyä ilmavirran

mukana kulkeviin aerosoleihin, jolloin syntyy radioaktiivisia aerosolipäästöjä. Jodin orgaaniset kaasumaiset yhdisteet, kuten metyylijodidi tarttuvat huonosti adsorptioon perustuvaan aktiivihiihluodattimeen, jonka vuoksi on tarpeen käyttää esimerkiksi KI- tai TEDA-kyllästettyä suodatinta. Tällöin kyllästeaineet sitovat jodiyhdisteet kemiallisesti ja parantavat täten suodattimen tehokkuutta. [14, 17]

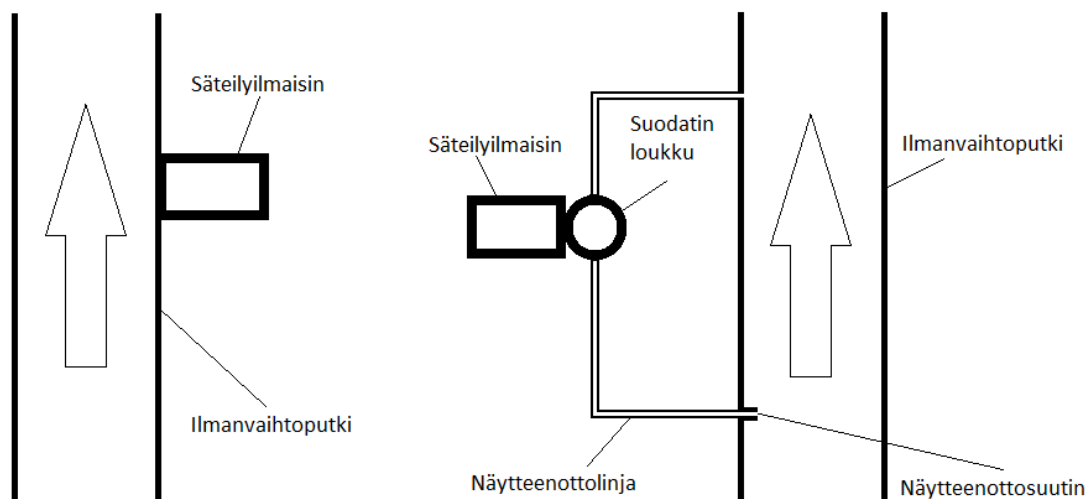
Jodin radioaktiiviset isotoopit ovat terveydelle erityisen haitallisia, koska kehoon joutuessaan ne kerääntyvät tehokkaasti kilpirauhaseen. Normaalisti toimivassa kilpirauhasessa noin 15 % injektiona annetusta jodista kerääntyy kilpirauhaseen 24 tunnin kuluessa [18]. Näin ollen iso osa kehoon saadun jodin emittoimasta säteilyenergiasta vapautuu pieneen osaan kehoa, toisin kuin monilla muilla aineilla, jotka poistuvat tehokkaammin virtsan ja hien mukana. Lisäksi jodin aiheuttama säteilyannos keskittyy pienelle alueelle, mikä lisää säteilyn haittavaikutusten riskiä kilpirauhasen ympäristössä pienilläkin annoksilla [19]. Käytetyistä jodin isotoopeista erityisesti ^{131}I :n korkeaenerginen beeta on hyvin tuhovoimainen kerääntyessään kudosten läheisyyteen. Lisäksi ^{131}I :n puoliintumisaika on huomattavasti pidempi kuin ^{123}I :n, joten vastaava aktiivisuus ehtii vaikuttaa pidempään kuin vastaava määrä ^{123}I :a.



Kuva 9: ^{123}I (yllä) ja ^{131}I (alla) hajoamiskaaviot, joissa näkyy primäärinen hajoamistapa (EC, eli elektronisieppaus 123-jodille, sekä beeta-miinus 131-jodille), sekä yleisimmät gammahajoamisten reitit tytärytimen perustilalle. Kuvat lähteestä [20]

5. Radioaktiivisten aineiden mittaaminen ilmvirrasta

Tässä kappaleessa esitellään erilaisia tapoja mitata radioaktiivisia päästöjä ilmanvaihtoputkessa kulkevasta ilmasta. Ensimmäiseksi käydään pintapuolisesti läpi erilaisten mittaustapojen hyötyjä ja haittoja ja lopuksi käydään perusteellisemmin läpi menetelmä, jota on käytetty työssä myöhemmin vertailtavissa monitorointilaitteistoissa.



Kuva 10: Kaaviokuva erilaisista mittausasetelmista. Vasemmalla suora ilmamittaus ilmvirran reunalta ja oikealla suodatinloukkuun kerättävä näyte, jota mitataan jatkuvasti.

Yksi tapa mitata radioaktiivisuutta ilmvirrassa on asettaa mittari suoraan ilmvirran reunalle (Kuva 10) tai keskelle. Tällöin mittauksesta saadaan reaaliaikaista palautetta putkessa liikkuvista radioaktiivisista aineista ja mittaus kattaa kaiken putkessa kulkevan ilman. Tässä mittaustavassa tulee kuitenkin helposti vastaan mittarin tehokkuus, eli tarvitaan epäkäytännöllisen suuri ilmaisin, jotta voidaan havaita tarpeeksi pienet määrät radioaktiivista ainetta suuresta nopeasti virtaavasta ilmamäärästä. Ongelmaa voidaan pienentää puristamalla mitattava ilma kulkemaan paineistettuna mittarin ohi, jolloin virtausnopeutta voidaan laskea ja ilma viettää pidemmän ajan mittarin havaitsemisalueella. [12]

Toinen vaihtoehto aktiivisuuden mittaamiselle ilmasta on kerätä ilma talteen esimerkiksi paineistettuun säiliöön mittauksen ajaksi [12]. Tämä on kuitenkin käytännöllistä vain riittävän pienillä ilman tilavuusvirroilla. Ilmasta on myös mahdollista ottaa talteen vain pieni näyte kerrallaan [12], mutta silloin tulee aiheelliseksi varmistua näytteen edustavuudesta. Ilman ottaminen säiliöön hankaloittaa mahdollisten hiukkaspäästöjen mittaamista, koska ne pyrkivät pysähtyneessä ilmassa laskeutumaan säiliön pohjalle.

Toinen tapa mitata ilmvirran aktiivisuutta on suodattaa radioaktiiviset aineet pois ilmasta ja mitata suodatinta. Tällöin ilmaisimen tehokkuudelta ei vaadita yhtä paljoa kuin suorassa ilmamittauksessa, mutta haasteeksi muodostuvat näytteenoton edustavuus, sekä suodattimen

tehokkuus radioaktiivisten päästöjen keräämisessä. Suodattimen mittaus voidaan toteuttaa joko jatkuvalla mittauksella (Kuva 10) tai erillisessä laboratoriossa [12]. Erillisessä laboratoriossa mitattaessa päästään eron näytteen keräyspisteessä mahdollisesti esiintyvistä taustasäteilyistä, mutta menetetään jatkuva tieto päästöistä. Lisäksi menetelmä ei sovi lyhytikäisille radionuklideille, koska niiden aktiivisuus ehtii vähentyä ennen varsinaista mittausta. Jatkuvan mittauksen menetelmää käytettiin tässä tutkielmassa vertailtavissa laitteissa, joten siitä kerrotaan tarkemmin seuraavassa kappaleessa.

5.1. Jatkuvasti mitattavan suodattimen menetelmä

Tämän tutkimuksen loppuosassa keskitytään menetelmään, jossa ilmanäytteestä kerätään radioaktiiviset aineet suodatinloukkuun, jonka aktiivisuutta mitataan jatkuvasti. Menetelmän etuna on jatkuva tieto päästöistä, sekä suoraa ilmamittausta pienemmät vaatimukset ilmaisimen tehokkuudelle. Haasteena on kuitenkin edustavan näytteen saaminen ja suodattimen tehokkuus esiintyville päästöille. Lisäksi täytyy huomioida muuttuva tausta, koska mittausta ei välttämättä saada suojattua tarpeeksi tai vietyä riittävän kauas laitoksen muista toiminnoista taustan eliminoimiseksi.

Jatkuvasti mitattavan näytteen menetelmä on myös laskennallisesti muita haastavampi. Mitattaessa aktiivisuutta suoraan ilmasta tai jälkeinpäin suodattimesta, saadaan mittauspisteen ohittanut aktiivisuus mittaustuloksesta kalibraatiokertoimen avulla. Jatkuvasti mitattavassa suodattimessa aktiivisuus kuitenkin kertyy ja puoliintuu jatkuvasti, jolloin mittauspisteen ohittanut aktiivisuus joudutaan selvittämään suodattimessa olevan aktiivisuuden muutosten perusteella.

5.1.1. Näytteenotto

Näytteenotto voidaan jakaa näytteenottopaikkaan, -tapaan ja kuljetuslinjaan. Kaikkien kohdalla on huomioitava, millaisia päästöjä laitoksessa on mahdollista tuottaa ja suunniteltava kukin osa-alue siten, että suodattimelle saadaan mahdollisimman edustava näyte ilmamassasta.

Näytteenottopaikka tulee valita siten, että näyte edustaa hyvin koko ilmavirran päästöjä. Sen valinnassa tulee huomioida virtauksen laminaarisuus näytteenottopaikassa, päästöjen riittävä sekoittuminen ilmavirtaan ennen virtauksen tasoittumista, sekä hiukkasten ja aerosolien kulkeutuminen näytteenottopisteen kohdalla. Jos mitataan pelkästään kaasumaisia päästöjä, ainoa vaatimus on päästöjen hyvä sekoittuminen. [21]

ISO-standardi radioaktiivisten päästöjen mittaamisesta ydinlaitoksen poistoilmasta asettaa viisi ehtoa hyvälle mittauspisteelle. Ilmavirran tulisi olla putken suuntaista ja sen nopeuden tasaista koko putken halkaisevalla tasolla. Lisäksi putkeen aikaisemmasta kohdasta päästetyn merkkikaasun ja merkkiaerosolin konsentraation variaatiokerroin ei saisi ylittää 20 % putken poikkipinta-alan kahden keskimmäisen kolmanneksen alueella eikä merkkikaasun konsentraatio saisi vaihdella yli 30 % poikkipinta-alan missään kohdassa. [12]

Hyvä näytteenottopaikka voidaan etsiä arvioimalla sopiva piste ja varmistamalla sen sopivuus tarkastamalla standardin määrittelemien ehtojen täyttyminen. Jos mittaaminen todellisella paikalla ei ole käytännöllistä, esimerkiksi laitoksen toiminnan häiriintymisen vuoksi tai koska laitos on vielä rakenteilla, on mittauspaikan etsiminen mahdollista myös pienoismallin tai laskennallisen mallin avulla [22].

Myös näytteenottotapa valitaan sen mukaan, onko tarpeen mitata aerosoleja ja hiukkasia. Jos päästöt ovat pelkästään kaasumaisessa muodossa ja ilmassa on täysin sekoittunutta, ei näytteenottoputken päähän tarvitse tehdä erillistä suutinta. Edustavan näytteen varmistamiseksi tai tilanteissa, joissa sekoittuminen ei ole täysin varmaa, voidaan käyttää useampaa suutinta mittauksen eri pisteissä. Jos halutaan saada edustava näyte myös aerosoleista, täytyy näytteenoton olla isokineettinen. Tämä tarkoittaa sitä, että suutin on suunniteltu ja suunnattu siten, ettei se aiheuta ilmapirtaan pyörteitä jotka voisivat ohjata aerosoleja ja hiukkasia sivuun tai näytteenottoputken seiniin. Lisäksi näytteenottonopeuden tulisi olla sellainen, että näytteeseen päätyneen ilman nopeus ei muutu sen tullessa näytteenottoputkeen. Jos ilmapirtan nopeus mitattavassa putkessa muuttuu merkittävästi kesken mittauksen, pitäisi näytteenoton imunopeutta muuttaa vastaavasti. [12]

Näytteenottolinja on pyrittävä suunnittelemaan sellaiseksi, että se minimoi radioaktiivisten päästöjen jäämisen linjaan. Erityistä huomiota on syytä kiinnittää hiukkasmuotoisten päästöjen hävikkiin. Hävikin minimoimiseksi linja tulisi suunnitella mahdollisimman lyhyeksi ja lisäksi linjassa tulisi olla mahdollisimman vähän kaarteita ja vaakatasossa kulkevia osioita [12, 21, 23]. Näytteenottolinjan tehokkuus voidaan määrittää kokeellisesti tai laskennallisesti [24]. Laskennallisen määrityksen avulla on mahdollista optimoida linja jo suunnitteluvaiheessa.

Myös näytteenottolinjan materiaalin valintaan tulisi kiinnittää huomiota. Esimerkiksi kuparia, PVC:tä ja nitrilikumia tulisi välttää linjassa, koska jodi reagoi niiden kanssa herkästi. Hyviä materiaaleja ovat esimerkiksi PTFE, PE, alumiini, hiiliteräs ja ruostumaton teräs [12]. Mikäli päästöissä tiedetään olevan reaktiivisia tai syövyttäviä aineita, tulisi materiaalivalinnassa ottaa huomioon nämä päästöjen erikoisominaisuudet.

5.1.2. Suodatinloukku ja sen mittaaminen

Ilmanäyte ohjataan läpi suodatinloukusta, jonka tarkoituksena on kerätä mahdollisimman tehokkaasti radioaktiiviset aineet ilmapirrasta ja pidättää ne suodattimessa. Suodattimen materiaali ja rakenne tulisi valita sellaiseksi, että sen suodatustehokkuus on optimaalinen niille aineille, joita laitoksessa pääsee poistoilmaan [12]. Esimerkiksi jodia suodatettaessa hyviä vaihtoehtoja ovat aktiivihiili tai zeoliitti [13]. Jos on kuitenkin tiedossa, että päästöilma on kostea tai päästöissä on myös jodin huonosti adsorboituvia yhdisteitä, kuten metyylijodidia, on tarpeen käyttää TEDA- tai KI-kyllästettyä suodatinmateriaalia [14]. Lisäksi suodatinloukku kannattaa suunnitella sellaiseksi, että siitä on mahdollisimman tehokasta mitata suodattimen aktiivisuus.

Ilmaisoin pitäisi saada sijoitettua lähelle suodatinta eikä suodattimen kannata olla liian suuri, jottei suodattimessa oleva aktiivisuus jää liian kauas ilmaisimesta.

Loukun mittaamiseen soveltuu esimerkiksi natriumjodidi-tuikeilmaisoin, jolla on hyvä herkkyys mm. jodin isotoopeille ^{123}I ja ^{131}I . NaI-ilmaisoin on myös suhteellisen halpa ja niitä on hyvin saatavilla. NaI-tuikeilmaisimen energioresoluutio on melko heikko, mutta mikäli päästöissä esiintyvien radionuklidien säteilyenergiat erottuvat hyvin toisistaan, ei huono energioresoluutio ole suuri ongelma. [4]

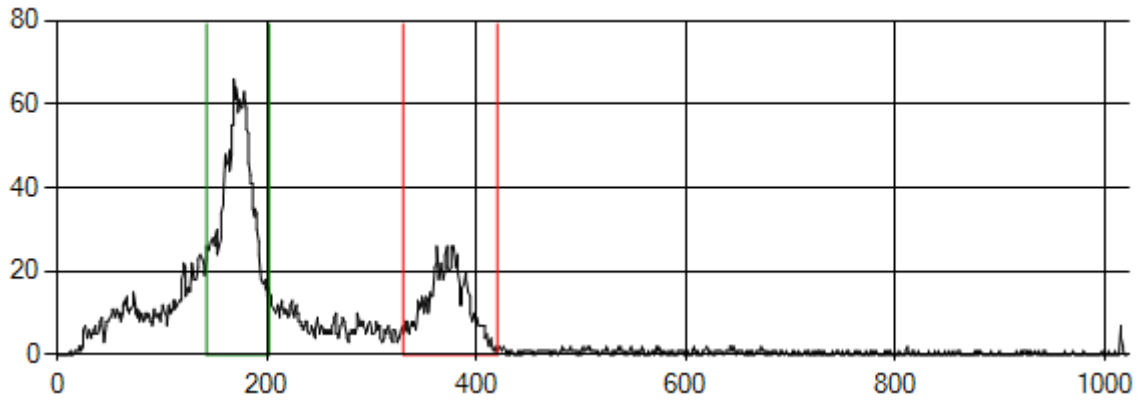
Ilmaisimeen liitetään analogi-digitaalimuunnin ja monikanava-analysaattori, sekä tallentava elektroniikka eli useimmiten PC. ADC-MCA-yhdistelmän tehtävänä on erotella ilmaisimelta tulevat pulssit niiden korkeuden mukaan. Koska valomonistinputken tuottaman pulssin korkeus on verrannollinen havaitun kvantin energiaan, muodostaa MCA käytännössä mitatun säteilyn energiaspektrin. Kun MCA:n tuottamat spektrit tallennetaan määräväleihin, saadaan tieto eri energisten kvanttien keskimääräisestä havaitsemisfrekvenssistä (CPS Counts per Second) kunkin mittausjakson aikana.

5.2. Tietojenkäsittely jatkuvan näytteenoton menetelmässä

Jotta edellä kuvatulla menetelmällä saatu mittaustulos suodatinloukkuun päätyneelle aktiivisuudelle voidaan muuttaa tiedoksi siitä, kuinka paljon ilmanvaihtoputkessa on kulkenut radioaktiivisia aineita, täytyy mittaustulosta käsitellä matemaattisesti. Tässä kappaleessa oletetaan, että mittausjärjestelmä tuottaa energiaspektrejä suhteellisen lyhyiltä mittausjaksoilta (esim. 100s). Spektrit voidaan esittää CxM-kokoisena matriisina, jonka alkio x_{cm} on MCA:n kanavaan c kertyneiden tapahtumien määrä mittausjaksolla m . Matriisin dimensiot ovat mittausjaksojen lukumäärä M ja MCA:n kanavien lukumäärä C .

5.2.1. Piikin pinta-alan määrittäminen

Jos mitattavassa näytteessä on jotain radionuklidia, kertyy spektriin nuklidin lähettämiä säteilyenergioita vastaavien kanavien ympäristöön enemmän tapahtumia. Näitä kohoumia spektrissä kutsutaan piikeiksi tai energiapiikeiksi. Piikin pinta-alan määrittystä varten, valitaan spektristä kiinnostava energia-alue (Region of Interest ROI), eli käytännössä kanavaväli, jolla haluttu piikki sijaitsee (Kuva 11). Alue voidaan valita manuaalisesti, mutta myös useita laskennallisia tapoja automaattiseen piikin etsimiseen on olemassa [9].

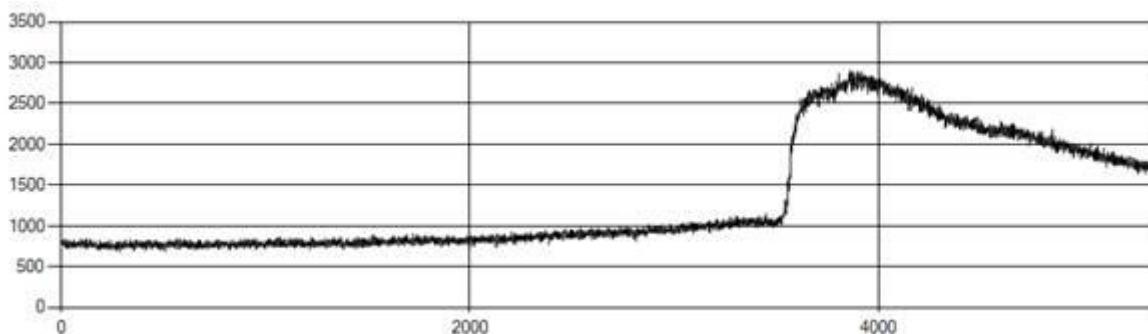


Kuva 11: Energiaspektri, jossa x -akselilla MCA:n kanavaluku ja y -akselilla tapahtumien määrä kussakin kanavassa. Spektristä on valittu kaksi ROI:ta, joista punainen kattaa ^{131}I :n täysenergiapiikin (364 keV) ja vihreä puolestaan ^{123}I :n täysenergiapiikin (159 keV). Vihreän ROI:n alle jää myös ^{131}I :n aiheuttama jatkumo.

Jos piikin ympäristössä ei ole muista lähteistä tulevia tapahtumia, on sen pinta-ala x_m yksinkertaisimmillaan spektrin integraali ROI:n yli. Jos merkitään ROI:n alarajaa a :lla ja ylärajaa b :llä, voidaan edellä mainittu integraali mittausjaksolle m kirjoittaa

$$x_m = \sum_{c=a}^b x_{cm} .$$

Esimerkki piikin pinta-alasta ajan funktiona on esitetty kuvassa 12.



Kuva 12: Piikin pinta-ala mittausjakson funktiona. 5000 sadan sekunnin mittausjaksoa on noin 6 päivää.

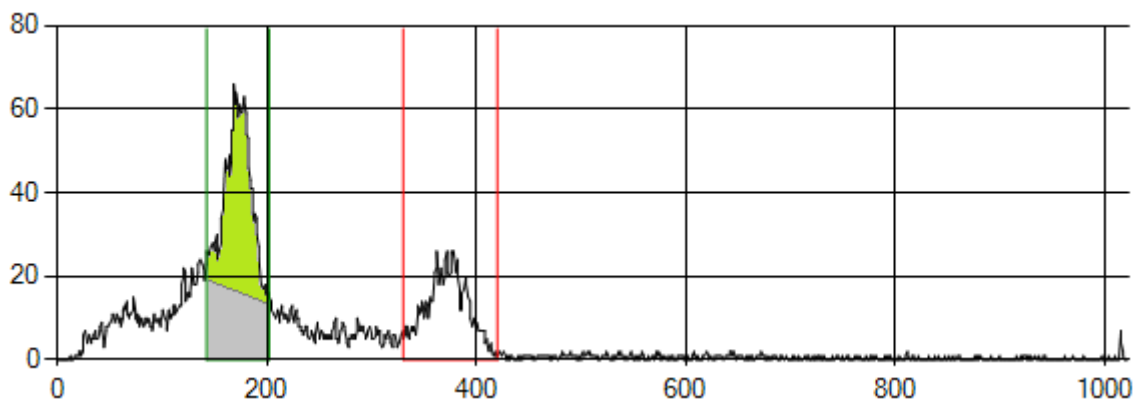
Useimmiten piikki sijaitsee kuitenkin jonkinlaisen muista lähteistä johtuvan perustason, taustan, päällä. Tällöin varsinaisen signaalin erottamiseksi täytyy edellä mainitusta integraalista vähentää taustan osuus. Yksinkertaisimmillaan tämä tapahtuu määrittämällä taustan taso piikin molemmilla puolilla ja olettamalla tausta lineaariseksi niiden välillä (Kuva 13). [4] Tällöin taustan osuudeksi B saadaan

$$B_m = \frac{(x_{(a-1)m} + x_{(b+1)m})}{2} \cdot (b - a).$$

Tällöin tausta B voidaan suoraan vähentää piikin pinta-alasta x_m seuraavasti

$$x_{m-B} = x_m - B_m.$$

Taustan tason määrittäminen yhdestä pisteestä ROI:n ulkopuolelta aiheuttaa vaihtelevan taustatason vuoksi virhettä. Tätä on kuitenkin mahdollista rajoittaa esimerkiksi laskemalla keskiarvo useammasta kanavasta ROI:n kummaltakin puolelta.



Kuva 13: Lineaarinen taustavähennys vähentää vihreillä viivoilla rajatusta ROI:sta harmaan osuuden taustana, jolloin piikin pinta-alaksi jää vihreällä merkitty pinta-ala

Jos halutaan määrittää pinta-ala sellaisesta piikistä, joka on osittain päällekkäin toisen piikin kanssa, joudutaan turvautumaan monimutkaisempiin menetelmiin. Tällöin on esimerkiksi mahdollista määrittää piikin muoto analyttisesti siitä osasta, johon päällekkäinen toinen piikki ei vaikuta. Piikin laskennallisesta muodosta voidaan tällöin arvioida piikin todellinen koko. [4]

5.2.2. Kertymän määrittäminen

Kahden peräkkäisen 100 sekunnin mittausjakson välinen ero selittyy useimmiten satunnaisella vaihtelulla. Koska radioaktiivinen hajoaminen on satunnaisilmiö, vaihtelee mittarin havaitsema pulssitaajuus sen mukaan, paljonko säteilyä lähde sattuu kullakin hetkellä tuottamaan ja kuinka suuri osuus siitä osuu mittariin. Satunnaisuutta voidaan tasoittaa laskemalla pidemmän aikavälin keskiarvo. Kymmenen pisteen keskiarvoja y_n saadaan laskemalla

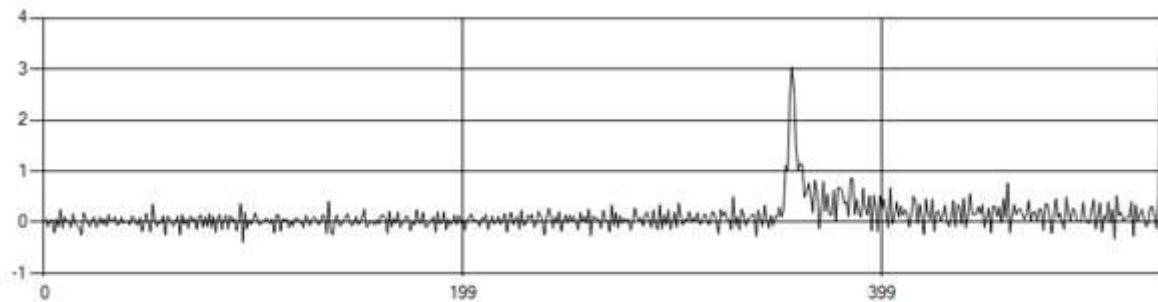
$$y_n = \frac{\sum_{i=10 \cdot n}^{10 \cdot (n+1)} x_{i-B}}{10}$$

Jos mittauksia on M kappaletta, saadaan näin $M/10$ keskiarvoa, joiden vaihtelu on raakoja mittaustuloksia vähäisempää.

Koska hiililoukku kerää näytettä jatkuvasti, ei edellisen mittausjakson aikana mitattu aktiivisuus poistu loukusta seuraavan mittausjakson alkaessa. Tämän vuoksi laskennassa on nimenomaan määritettävä se, paljonko loukussa oleva aktiivisuus on lisääntynyt. Kertymän määrittäminen tehdään vähentämällä kustakin mittaustuloksesta edellinen, jolloin erotus kertoo loukussa olevan aktiivisuuden muutoksen. Koska kyseessä on kuitenkin radioaktiivinen aine, täytyy myös huomioida aktiivisuuden väheneminen puoliintumisen myötä. Tämä tehdään kertomalla edellinen mittaustulos puoliintumiskertoimella ennen vähentämistä. Tapahtumamäärän muutos Δy (Kuva 14) saadaan laskemalla

$$\Delta y = y_i - y_{i-1} \cdot e^{-\ln 2 \cdot \frac{10t}{t_{1/2}}}$$

jossa $t_{1/2}$ on mitattavan radionuklidin puoliintumisaika ja t mittausjakson pituus. Puoliintumiskertoimessa käytetään aikana kymmentä mittausaikaa, koska laskussa käytetään kymmenen mittausjakson keskiarvoja y .



Kuva 14: Δy keskiarvoistuksien järjestysluvun funktiona. X-akselin arvo 400 vastaa mittaustulosten 4000-4010 ja 4011-4020 erotusta. Kuvassa 12 näkyvä nousu näkyy tässä kuvaajassa kohoumana.

Tapahtumamäärän muutoksesta voidaan laskea tapahtumanopeuden (cps) muutos

$$\Delta(\text{cps}) = \frac{\Delta y}{t},$$

jonka perusteella voidaan määrittää aktiivisuuden muutos, mikäli mittausasetelma on tunnettu.

5.2.3. Tehokkuuskertoimet ja päästön määrittäminen

Jos aktiivisuuden mittaamiseen käytetty mittari on tehokkuuskalibroitu oikein, sen tuottamasta cps-lukemasta voidaan määrittää sen mittaaman lähteen aktiivisuus. Koska aktiivisuus on kalibroitukertoimen kautta verrannollinen cps-lukemaan, voidaan myös aktiivisuuden muutos määrittää saman kertoimen kautta cps-lukeman muutoksesta

$$\Delta A = \Delta \text{CPS} \cdot \varepsilon_{\text{mittari}},$$

jossa $\varepsilon_{\text{mittari}}$ on mittalaitteen täysenergiapiikin tehokkuus.

Tässä mittausjärjestelyssä hiililoukkuun kertyneen aktiivisuuden muutoksesta määritetään otetun ilmanäytteen sisältämä aktiivisuus. Se saadaan laskettua hiililoukun aktiivisuuden muutoksesta ja loukun läpi kulkeneen ilman tilavuusvirrasta

$$C_{näyte} = \frac{\Delta A}{V_{näyte}},$$

jossa $C_{näyte}$ on näytteen aktiivisuuskonsentraatio, ΔA on hiililoukussa olevan aktiivisuuden muutos mittausjakson aikana ja $V_{näyte}$ on loukun läpi kulkeneen näyteilman tilavuus saman mittausjakson aikana. Ottamalla huomioon suodatinloukun ja näytteenottoputken tehokkuuskertoimet, eli kuinka suuri osa näytteestä todellisuudessa jää hiililoukkuun, saadaan arviota tarkennettua seuraavasti

$$C_{näyte} = \frac{\Delta A \cdot \varepsilon_{linja} \cdot \varepsilon_{loukku}}{V_{näyte}},$$

jossa ε_{linja} on siirtolinjan tehokkuus ja ε_{loukku} on suodatinloukun tehokkuus. Tuntemalla ilmanäytteen aktiivisuuskonsentraatio, saadaan ilmanvaihtoputken läpi kulkeva aktiivisuus A_{putki} laskettua yhtälöstä

$$A_{putki} = C_{näyte} \cdot V_{poisto},$$

jossa V_{poisto} on ilmanvaihtoputken läpi kulkeva ilmamäärä.

6. Mittausjärjestelmien vertailu

6.1. MAP Medical Technologies Oy:n toiminta Tikkakosken toimipisteessä

MAP on suomalainen lääkeyritys, joka valmistaa radioaktiivisia lääkkeitä Tikkakoskella ja Helsingissä. Yrityksen valmistamia lääkkeitä käytetään isotooppi- ja PET-kuvauksissa, sekä isotooppihoidoissa. Esimerkiksi ^{131}I -hoitoa käytetään kilpirauhasen liikatoiminnan ja kilpirauhassyövän hoitoon. [25]

Tässä tutkielmassa tarkastellaan Tikkakosken toimipisteen päästömonitorointijärjestelmää. Tikkakoskella MAP tuottaa $^{99}\text{Mo}/^{99}\text{Tc}$ -generaattoreita, ^{32}P -liuoksia sekä ^{123}I - ja ^{131}I -kapseleita ja liuoksia. Kaikkia näitä radionuklideja käsitellään laitoksella liuoksina, mutta niistä ainoastaan jodin isotoopit muodostavat merkittäviä määriä kaasumaisia yhdisteitä [16].

Päästömonitorointijärjestelmän tehtävänä on tarkkailla, että toimipisteen poistoilmassa olevien radioaktiivisten päästöjen määrä alittaa viranomaisten antamat raja-arvot. STUK määrittää säteilyturvallisuusohjeessaan (ST 6.2 [1]), että ympäristössä asuvan tai työskentelevän väestön vuosittainen säteilyannos täytyy pitää mahdollisimman pienenä. Lisäksi määrätään, että toiminnan harjoittajan tuottamat päästöt ilmakehään eivät saa aiheuttaa väestöä edustavalle henkilölle yli

10 μSv vuositasta annosta. Sallitun päästön määrän määrittävät siis kunkin radionuklidin biologiset haittavaikutukset ilmakehään päästettyä aktiivisuusyksikköä kohti, sekä laitoksen sijainti suhteessa ympäröivään väestöön.

Radioaktiivisten päästöjen aiheuttamat annokset on laskettu liitteenä 1 olevassa erikoistyössä. Päästöjen vaikutusten laskennassa on käytetty Gaussista päästöpilvimallia, jonka avulla on arvioitu ilmanvaihdesta ulos tulevien päästöjen vaikutus väestöä edustavalle henkilölle. Arvioinnissa on otettu huomioon annoksen kertyminen ulkoisesta altistuksesta päästöpilvestä ja laskeumasta, sekä hengityksen kautta elimistöön päätyvästä aktiivisuudesta. Lisäksi arvioidaan ravintoketjun kautta kertyvää annosta.

6.2. Toimintaperiaatteiden erot

MAP:lla on käytössä kaksi ilmanvaihdon radioaktiivisten päästöjen monitorointilaitteistoa. Uudempi järjestelmä (IONA) on Canberra Industriesin toimittama ja käyttöönottama kokonaisuus. Vanhemmassa järjestelmässä (Hidex) on puolestaan MAP:n rakentama näytteenotto, mutta Hidex Oy:n toimittama ilmaisinalaiteisto ja sen elektroniikka. Tässä työssä on tuotettu uusi päästönlaskenta-ohjelmisto vanhempaan laitteistoon.

Vanhemmassa järjestelmässä näytteenottopaikka on ennen ilmanvaihtokonetta karkeasuodattimien jälkeen, missä päästöt ovat ehtineet sekoittua ilmapirtaan. Ilmanäyte otetaan PVC-letkulla ja pumpataan aktiivihilisuodattimeen, joka kerää näytteestä jodin.

IONA kerää ilmanäytettä näytteenottosuuttimella, jossa näyte tulee suuttimeen 5 reiästä ilmanvaihtoputken koko leveydeltä. Näyte otetaan ilmanvaihtokoneen jälkeisestä osasta ilmastointiputkea, jotta päästöjen sekoittuminen olisi varmistettu. Suuttimelta näyte pumpataan sileää ruostumattomasta teräksestä valmistettua putkea pitkin HEPA-suodattimen kautta varsinaiseen suodatinpatruunaan. Virtausnopeutta tarkkaillaan jatkuvasti virtausmittarilla ja virtausnopeuden muutokset otetaan huomioon laskennassa.

Molemmat laitteistot ovat kuvan 10 (s.18) oikeanpuoleisen osan mukaisia. Molemmissa suodatinta mittaa jatkuvasti NaI(Tl)-tukeilmaisina, jossa on kaksituuminen kide. Ilmaisimen tuottama signaali vahvistetaan ADC:n kautta MCA:lle, jonka tuottama spektri puolestaan tallennetaan PC:lle. IONA laskee tuottamistaan spektreistä ja sille syötetystä poistoilman tilavuusvirrasta suoraan tuotetut päästöt. Hidexin järjestelmä tuottaa spektridatan tietokantatiedostoon, josta ne voidaan tuoda ulos analysoitavaksi. Hidexin järjestelmän tuottama spektridata käsitellään aikaisemmin kuvattuun laskentamenetelmään perustuvalla tietokoneohjelmalla päästön laskemiseksi. Ohjelmaa käsitellään tarkemmin liitteessä 2.

6.3. Vertailun toteutus

Järjestelmien vertailussa tarkasteltiin ^{123}I :n osalta kahden vuorokauden mittausjaksoja (tuotantopäivä + seuraava päivä). Tuotantopäivä valittiin jakson alkupisteeksi siksi, että ^{123}I -raaka-ainera saapuu MAP:lle kyseisenä päivänä aamuyöstä ja tuotanto tehdään samana aamuna. Aikaisemmasta päästöjen seurannasta on havaittu, että ^{123}I -päästöt ajoittuvat tuotantopäivälle, mutta seuraava päivä otettiin mukaan mahdollisten jälkipäästöjen varalta. Muina aikoina ^{123}I -päästöt ovat hyvin pieniä, koska ainetta on toimitiloissa vain pieniä määriä ja lyhyen puoliintumisaikansa vuoksi nekin vähenevät nopeasti.

Vertailuun otettiin 17 kappaletta tiistaina klo 0:00 alkavia ja keskiviikkona klo 23:59 loppuvia jaksoja, sekä yksi keskiviikkona klo 0:00 alkava ja torstaina klo 23:59 loppuva jakso. Yksittäinen keskiviikon ja torstain käsittävä jakso valittiin siksi, että kyseisellä viikolla raaka-aineen saapuminen ja tuotanto ajoittuivat keskiviikolle. Canberran IONA-järjestelmä tuotti jaksoille päästötiedot suoraan ja Hidexin järjestelmän tuottamista spektritiedoista laskettiin omalla tietokoneohjelmalla päästöt sekä taustavähennystä käyttäen että ilman taustavähennystä. Laskennassa käytettiin suodattimen tehokkuudelle arvoa 0,95 ja siirtolinjan tehokkuudelle arvoa 0,75. Mittalaitteen täysenergiapiikin tehokkuutena käytettiin aikaisemmista kalibrointimittauksista saatuja arvoja 0,0523 (^{123}I) ja 0,0281 (^{131}I).

Myös ^{131}I :n osalta vertailuun otettiin samat kahden päivän jaksot. IONA-järjestelmästä tulokset saatiin jälleen suoraan ja Hidexin järjestelmästä päästöt laskettiin itse. ^{131}I -päästöjen laskennassa ei käytetty taustavähennystä, koska ^{131}I :n täysenergiapiikki sijaitsee sellaisessa kohdassa spektriä, jossa taustan vaikutus on pieni (ks. kuva 13, s. 23).

Vertailussa eri mittaustapojen tuottamista tuloksista laskettiin korrelaatiokertoimet, jotta saatiin määritettyä tulossarjojen vastaavuus ottamatta kantaa niiden välillä mahdollisesti olevaan suuruuseroon. Korrelaatiokertoimen lisäksi määritettiin tulosten keskimääräinen prosentuaalinen kokoero. Tämä laskettiin kunkin mittausjakson kohdalla vähentämällä vanhan järjestelmän tuottama tulos IONA:n tuottamasta tuloksesta ja jakamalla erotus IONA:n tuloksella. Näin saatiin vanhan järjestelmän tuloksen prosentuaalinen ero IONA:n tulokseen verrattuna. Näistä eroista laskettiin lopuksi keskiarvo.

Eri järjestelmillä mitattiin myös kokonaispäästö ajalle, jonka molemmat järjestelmät ovat olleet yhtäaikaisesti käytössä. Tämän päästön perusteella laskettiin arvio koko vuoden kokonaispäästölle, mikäli päästöprofiili pysyisi koko vuoden samanlaisena. Kokonaispäästö saatiin IONA:n ohjelmistosta suoraan ja Hidexin järjestelmästä saatujen mittaustulosten perusteella ne laskettiin itse.

6.4. Radioaktiivisten päästöjen määrät

Päästömittauksista saatuja tuloksia on koottu taulukkoon 1. Tarkastelluista kahden päivän mittausjaksojen päästöistä on mainittu keskiarvo ja suurin mittaustulos. Tarkastelujakson kokonaispäästössä on otettu huomioon myös muut ajat, kuin edellä mainitut kahden päivän jaksot. Koko vuoden 2016 päästöille on laskettu arvio olettamalla, että päästöt ovat koko vuoden keskimäärin yhtä suuret, kuin välillä 8.1.-31.7. Vertailun vuoksi voidaan todeta, että vuoden 2015 aikana toimipisteelle tuotiin ¹³¹I raaka-ainetta 7250 GBq ja ¹²³I raaka-ainetta 706 GBq. Vuodelle 2016 arvioidut päästöt olisivat siis ¹³¹I:n osalta 0,0017% ja ¹²³I:n osalta 0,045% vuoden 2015 raaka-aineen määrästä.

Taulukko 1: Päästömittauksista saatuja tuloksia

	IONA	Hidex
Keskimääräinen ¹²³ I-päästö 2 päivän jaksolla (MBq)	4,24	2,06
Suurin ¹²³ I-päästö 2 päivän jaksolla (MBq)	7,48	3,39
¹²³ I-kokonaispäästö tarkastelujaksolla 8.1.-31.7. (MBq)	178	89
¹³¹ I-kokonaispäästö tarkastelujaksolla 8.1.-31.7. (MBq)	70	61
Arvioitu ¹²³ I-kokonaispäästö koko vuodelle (MBq)	316	158
Arvioitu ¹³¹ I-kokonaispäästö koko vuodelle (MBq)	124	108

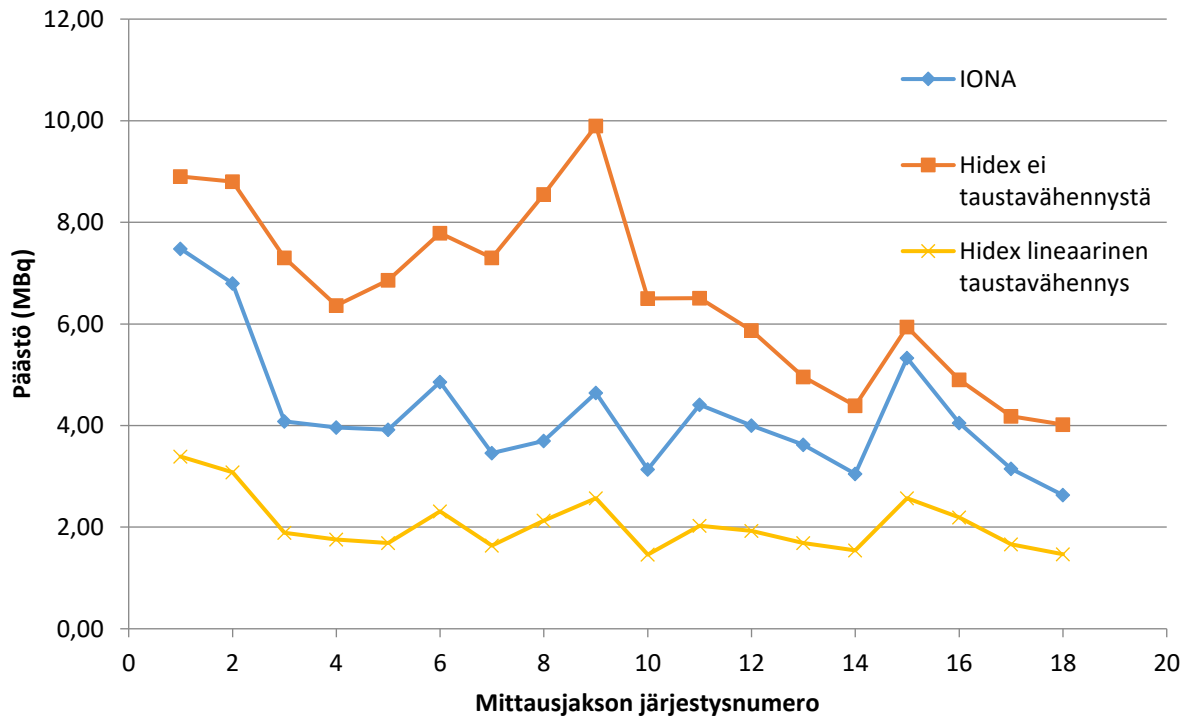
6.5. Tuloksissa havaitut erot

Suurimmat päästömäärät ¹²³I:lle saatiin taustakorjaamattomalla laskumenetelmällä Hidexin järjestelmän mittauksista ja pienimmät Hidexin järjestelmästä käyttäen lineaarista taustavähennysalgoritmia. IONA:n tuottamat tulokset jäivät systemaattisesti näiden väliin (kuvaaja 1). IONA:n ja Hidexin taustavähennetyin tulossarjan korrelaatiokerroin oli 0,959, mutta Hidexin tulokset olivat keskimäärin 48 % pienempiä. Taustavähentämättömien Hidexin tulosten ja IONA:n tulosten korrelaatio oli puolestaan 0,640 ja Hidexin tulokset olivat keskimäärin 63 % IONA:n tuloksia suurempia.

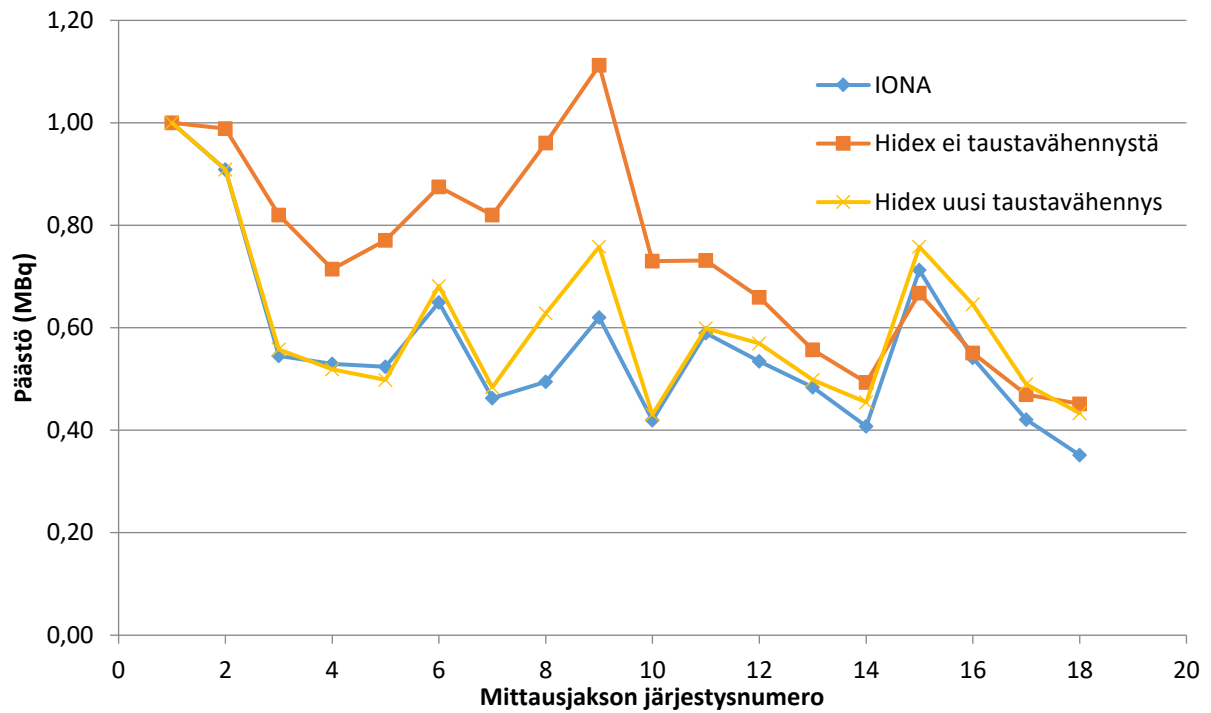
Tulossarjat myös normalisoitiin jakamalla kaikki ¹²³I-tulospisteet kyseisen tulossarjan ensimmäisellä mittaustuloksella. Näin saaduissa tulossarjoissa nähdään hyvin IONA:n ja Hidexin taustavähennetyin sarjan vastaavuus (kuvaaja 2). Toisin sanoen järjestelmät tuottavat hyvin samanmuotoista

raakadataa, mutta niiden laskennat on kalibroitu siten, että lopulliset mittaustulokset eroavat kertoimella 2.

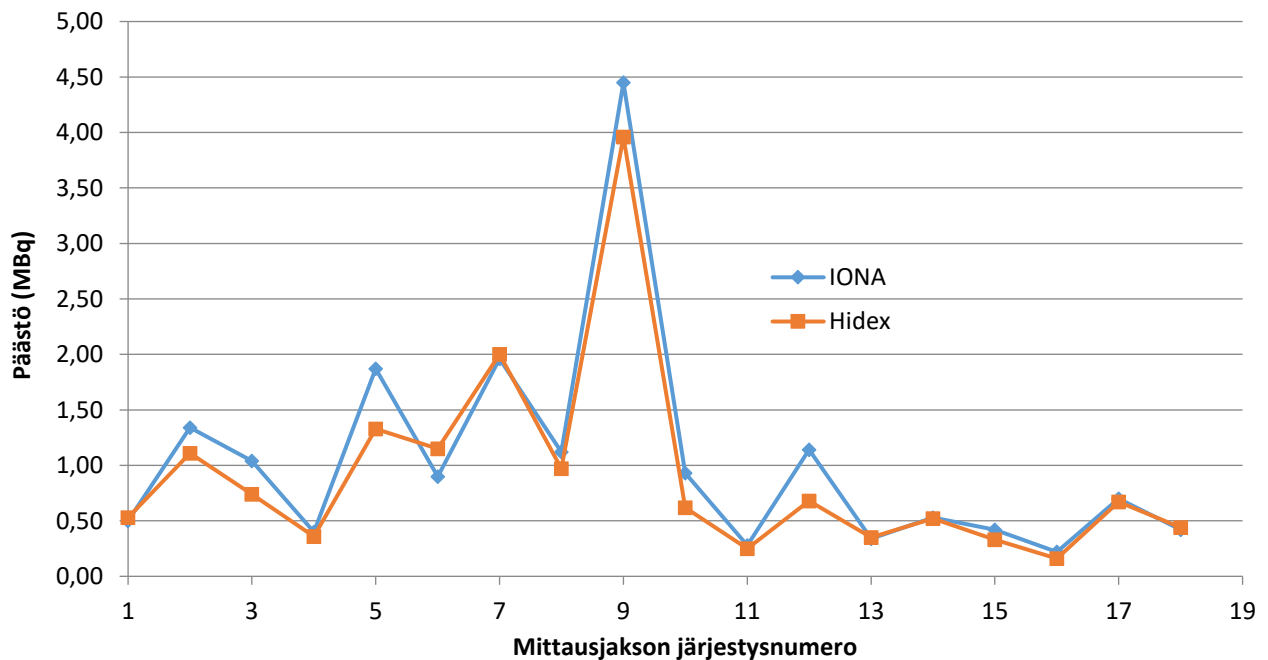
Molempien järjestelmien ^{131}I tulossarjat olivat puolestaan hyvin lähellä toisiaan (kuvaaja 3). Tulossarjojen välinen korrelaatio oli 0,981 ja IONA:n tulokset olivat keskimäärin 11% suurempia.



kuvaaja 1: ^{123}I päästöt eri mittausjärjestelyillä kahden päivän mittausjaksoilla.



kuvaaja 2: ^{123}I normalisoidut päästöt eri mittausjärjestelyillä kahden päivän mittausjaksoilla.



kuvaaja 3: ^{131}I päästöt eri mittausjärjestelyillä kahden päivän mittausjaksoilla.

7. Johtopäätökset

IONA käyttää kaikissa mittauksissaan lineaarista taustavähennysmenetelmää, mutta MAP:n vanhassa laitteistossa sitä käytetään vain ^{123}I -laskennassa. Molemmat laitteistot antavat kuitenkin hyvin samanlaiset tulokset ^{131}I -päästöille, mikä vahvistaa oletusta siitä, että taustan merkitys ^{131}I -mittauksessa on hyvin pieni. Sen sijaan ^{123}I -mittauksessa taustavähennyksellä on merkittävä rooli. Vanhan järjestelmän taustavähentämättömän tuloksen ja IONA:n taustavähennetyin tuloksen välinen korrelaatio on melko huono, kun taas vanhan järjestelmän taustavähennetyin tuloksen ja IONA:n välinen korrelaatio on hyvä.

Taustavähentämättömän ja taustavähennetyin ^{123}I -mittauksen erilaisuus johtuu suurelta osin ^{123}I -piikin kohdalle osuvasta ^{131}I :n Compton-jatkumosta. Vaikutusta voi olla myös ^{99}Mo :n 181 keV piikillä, joka osuu lähelle ^{123}I -ROI:ta. Toimipisteellä käsitellään ajoittain suurehkoja määriä molybdeeniä ja on mahdollista, että sen aiheuttama taustasäteily vaikuttaa mittaustuloksiin. Koska laskenta olettaa ^{123}I :n ROI:ssa näkyvän aktiivisuuden vähenevän ^{123}I :n puoliintumisajalla, hitaammin puoliintuvat ^{131}I - ja ^{99}Mo -komponentit tulkitaan virheellisesti ^{123}I :n jatkuvaksi kertymiseksi suodatinloukkuun. ^{131}I :n ROI:ssa ei ole tässä mittauksessa vastaavaa muista radionuklideista johtuvaa jatkumoa ja muu tausta on saatu mittaussäätelyillä niin matalaksi, että taustavähennyksen merkitys jää pieneksi.

IONA:n antama tulos ^{123}I -mittauksissa oli noin 2,1-kertainen vanhan järjestelmän taustavähennettyin tulokseen verrattuna. Hyvä korrelaatio mittaussarjojen välillä viittaa kuitenkin siihen, että ero johtuu laskennassa käytettävien kertoimien eroavaisuudesta. Käytännössä ero voisi olla korjattavissa kalibroimalla laitteet tarkasti ja määrittämällä siirtolinjan ja suodattimen tehokkuudet tarkemmin.

On mahdollista, että mittaustuloksia voitaisiin tarkentaa käyttämällä kehittyneempää taustavähennysalgoritmia. Käytetty lineaarinen taustavähennys olettaa, että piikin alla oleva tausta on jotakuinkin lineaarinen, mutta tämän tutkimuksen mittauksissa ^{123}I -piikin alla oleva tausta on muodoltaan enemmänkin kumpu. Lisäksi ^{123}I -ROI:n alueelle osuu laitoksen taustasäteilyssä esiintyvä ^{99}Mo :n 181 keV piikki, jota on ROI:ta muokkaamalla mahdoton erottaa ^{123}I -piikistä NaI-tuokeilmaisimen tuottamassa spektrissä. Paremman taustavähennysalgoritmin tuottamaa tarkkuuden lisäystä pidettiin kuitenkin niin vähäisenä, että monimutkaisempien algoritmien käyttöönottoon ei haluttu ryhtyä.

Toinen tapa piikkien yhdistymisen välttämiseksi ja taustavähennyksen helpottamiseksi olisi vaihtaa käyttöön sellainen mittaussäätely, jossa olisi parempi energioresoluutio. Ilmaisintyyppiä vaihdettaessa täytyisi kuitenkin tehdä kompromisseja muiden ominaisuuksien, kuten tehokkuuden tai hinnan osalta.

Päästöjen mittaamisen tarkoitus MAP:lla on varmentaa, ettei päästöistä koidu määräyksiä suurempaa annosta väestölle. Yllä mainitut virhelähteet johtavat tämän tutkimuksen mittauksessa

lopputuloksena saadun päästön yliarviointiin ja koska MAP:n päästöt ovat mittausten yliarvioinnista huolimatta reilusti viranomaisten asettamien rajojen alapuolella, arvion optimointia alaspäin ei koettu tarpeelliseksi. Lisäksi päästöjen tuottaman annoksen arviointiin käytetty menetelmä teki paljon konservatiivisia oletuksia, eikä varsinaisesti pyrkinytkään tarkkaan tulokseen. Tällöin erityisen tarkan päästölukeman määrittäminen annoslaskennan lähtöarvoksi ei ollut mielekästä.

Kuvaajassa on havaittavissa kahden mittausjakson pituinen eroavaisuus IONA:n ja Hidexin taustavähennetyt ¹²³I-mittauksen välillä. Mittauksien ajoitusta tarkasteltaessa huomattiin, että IONA:n suodatinpatruuna vaihdettiin ennen näitä kahta mittausjaksoa ja niiden jälkeen. On mahdollista, että kyseisen suodatinpatruunan suodatustehokkuus oli huonompi tai se oli asennettu väärin, minkä vuoksi IONA:n mittaustulos olisi muita mittausjaksoja pienempi.

Hidexin laitteistossa aktiivihilisuodattimessa oleva hiili on vanhaa, joten sen suodatustehokkuus on todennäköisesti heikentynyt ilmankosteuden ja aktiivihiihen täyttymisen myötä. Laskentamenetelmässä suodattimen tehokkuutta käytetään kuitenkin parametrina, joten syöttämällä laskentaan oikeat kalibrointikertoimet, voidaan vanhan hiilen vaikutus saada vähäisemmäksi. Jos suodattimen tehokkuus kuitenkin laskee tai muuttuu jatkuvasti (esimerkiksi suodatinpatruunan vaihtojen yhteydessä), vaihtelevat myös mittaustulokset, mikäli laitetta ei kalibroida jokaisen muutoksen jälkeen uudelleen.

IONA:n näytteenotto ja näytteen kuljetuslinja vaikuttavat lähteiden suositusten perusteella olevan huomattavasti paremmin toteutettu kuin Hidexin järjestelmässä, mutta tulokset vastaavat silti melko hyvin toisiaan. Tämä johtunee siitä, että vaikka näytteenoton tehokkuus olisi toisessa laitteessa huonompi, ero kompensoidaan laskennassa käytetyillä tehokkuus- ja kalibrointikertoimilla. Eroja voisi kuitenkin syntyä, jos päästöt vaihtelisivat aerosolien ja kaasumaisten välillä. Tällöin näytelinjan mahdollisesti erilainen tehokkuus erityyppisille päästöille voisi tulla esille.

Yhteenvetona voidaan todeta, että laitteiden tulokset vastaavat ¹²³I-mittauksessa esiintyvää eroavaisuutta lukuun ottamatta hyvin toisiaan. Eroja voisi syntyä, jos mitattavana olisi myös aerosoleja, mutta mitattujen päästöjen perusteella käytössä on kaksi käyttökelpoista järjestelmää. Erikoistyössä laskettujen päästöjen vaikutusten perusteella MAP:n Tikkakosken toimipisteen vuoden 2015 päästöt alittavat viranomaisten asettamat päästörajat. Näiden mittausten perusteella vuoden 2016 alun päästöt ovat selkeästi pienemmät, joten päästöraja alittunee myös vuonna 2016.

8. Lähteet

- [1] Säteilyturvakeskus. ST 6.2: Avolähteiden käytöstä syntyvät radioaktiiviset jätteet ja päästöt. Grano Oy, Helsinki, 2014.
- [2] T. K. Ikäheimonen (toim.) Säteily ja sen havaitseminen. Karisto Oy, Hämeenlinna, 2002.
- [3] K. S. Krane, Introductory Nuclear Physics. John Wiley & Sons, 1988.
- [4] G. F. Knoll, Radiation Detection and Measurements, 4th edition. John Wiley & Sons, 2010.
- [5] Delft University of Technology -verkkosivu, Molybdeeniin hajoamiskaavio, saantitapa: http://www.tnw.tudelft.nl/fileadmin/Faculteit/TNW/Over_de_faculteit/Afdelingen/Radiation_Radionuclides_Reactors/Research/Research_Groups/RIH/Research/img/decay-Mo.jpg (Viitattu 12.8.2016).
- [6] Wikimedia Commons -verkkosivu, Types of radiation, saantitapa: https://commons.wikimedia.org/wiki/File:Types_of_radiation.svg (Viitattu 25.5.2016).
- [7] R. Mustonen & A. Salo, Säteily ja solu. Teoksessa W. Paile (toim.) Säteilyn terveysvaikutukset. Karisto Oy, Hämeenlinna 2002.
- [8] University of California, Davis: Chemwiki -verkkosivu, Valomonistinputken kaaviokuva, saantitapa: http://chemwiki.ucdavis.edu/@api/deki/files/317/=Photomultiplier_Tube_Schematic.png?revision=1. (Viitattu 13.5.2016).
- [9] G. Gilmore & J. Hemingway, Practical Gamma Ray Spectroscopy, John Wiley & Sons Ltd, 1995.
- [10] A. M. El-Khatib, M. M. Gouda, M. S. Badawi, S. S. Nafee & E. A. El-Mallah, New analytical approach to calibrate the NaI (TI) detectors using spherical radioactive sources. Radiation Protection Dosimetry 156 [1], 2013 s. 109-117.
- [11] R. Casanovas, J. J. Morant & M. Salvadó, Energy and resolution calibration of NaI(Tl) and LaBr₃(Ce) scintillators and validation of an EGS5 Monte Carlo user code for efficiency calculations. Nuclear Instruments and Methods in Physics Research A, 675, 2012, s. 78-83.
- [12] ISO 2889: Sampling airborne radioactive materials from the stacks and ducts of nuclear facilities. Sveitsi, 2010.
- [13] S. Kulyukhin, A. Kamenskaya, N. Mikheev, I. Melikhov, N. Konovalova & I. Rumer, Basic and applied aspects of the chemistry of radioactive iodine in a gas medium. Radiochemistry, 50 [1], 2008, s. 1-20.

- [14] S. W. Park, H. S. Park, W. K. Lee & H. Moon, Effect of vapor on adsorption of methyl iodide to triethylnydiamine-impregnated activated carbons. *Separations technology*, 5 [1], 1995 , s. 35-44.
- [15] V. G. Kritskii, N. I. Ampelogova, V. Krupennikova, V. N. Kornienki, V. Ivanov & N. I. Rybkin, Comparative tests of the efficiency of sorption-filtering materials for removing radioactive iodine from gaseous emissions. *Atomic Energy*, 97 [6], 2004, s. 851-857.
- [16] J. E. F. Reynolds (toim.). *Martindale: The Extra Pharmacopoeia*, 30th edition. The Pharmaceutical press, Lontoo, 1993.
- [17] N. B. Borisov, Investigation of gaseous fractions of radioactive iodine. *Atomic Energy*, 97 [5], 2004, s. 761-766.
- [18] H. Korpela, *Isotooppiläätiede*. Teoksessa O. Pukkila (toim.) *Säteilyn käyttö*. Karisto Oy, Hämeenlinna, 2004.
- [19] D. Delacroix, J. P. Guerre, P. Leblanc & C. Hickman, *Radionuclide and Radiation Protection Data Handbook 2002*, *Radiation Protection Dosimetry*, 98 [1], 2002, s. 1-168.
- [20] C. M. Lederer & V. S. Shirley, *Table of Isotopes seventh edition*. John Wiley & Sons, Inc., New York, 1978.
- [21] J. M. Barnett, *Concepts for Environmental Radioactive Air Sampling and Monitoring*, Teoksessa E. O. Ekundayo (toim.) *Environmental Monitoring*, 2011. saantitapa: <http://cdn.intechopen.com/pdfs-wm/22748.pdf> (Viitattu 19.2.2016).
- [22] K. P. Recknagle, S. T. Yokuda, M. Y. Ballinger & M. J. Barnett, Scaled tests and modeling of effluent stack sampling location mixing. *Health Physics*, 96[2], 2009, s. 164-173.
- [23] N. Anand, A. McFarland, K. Kihm & F. Wong, Optimization of Aerosol Penetration through Transport Lines. *Aerosol Science and Technology*, 16 [2], 1992, s. 105-112.
- [24] M. Hogue, M. Thompson, E. Farfan & D. Hadlock, Hand calculations for transport of radioactive aerosols through sampling systems. *The Radiation Safety Journal*, 106 [5], 2014, s. S78-87.
- [25] Tuotteet, Mapmedical - verkkosivut. saantitapa: <http://www.mapmedical.fi/en/products/radiopharmaceuticals>. (Viitattu 28.5.2016).

9. Liitteet

9.1. Liite 1: Päästöjen vaikutusarviot

Johdanto

Säteilyturvakeskus määrittää ST-ohjeissaan [L1.1] ilmaan päästettävien radioaktiivisten päästöjen ylärajaksi, että päästöt eivät saa aiheuttaa väestöä edustavalle henkilölle yli 10 μSv vuosittaista annosta. Tätä annosta vastaavan päästön suuruus jätetään toiminnan harjoittajan määritettäväksi. Yksi tapa määrittää poistoilmassa olevien päästöjen aiheuttama annos väestön edustajalle on Gaussinen päästöpilvimalli. Mallin käyttö esitetään IAEA:n raportissa [L1.2]. Raportin malli on suunniteltu kuvaamaan radioaktiivisia aineita käsittelevästä laitoksesta vapautuvien pitkäaikaisten päästöjen vaikutusta ympäristössä asuviin ihmisiin. Siinä arvioidaan mm. päästöpilvestä, pilvessä hengittämisestä, laskeumasta, sekä saastuneesta ruuasta aiheutuvat säteilyannokset.

Tässä työssä käytetään IAEA:n raportissa [L1.2] esiteltyä päästöpilvimallia MAP Medical Technologies Oy:n (MAP) toiminnasta väestölle aiheutuvan säteilyannoksen määrittämiseen. Mallin avulla lasketaan arvio toimipisteen ympärillä sijaitsevista pisteissä oleskeleville kuvitteellisille henkilöille kertyville annoksille. Tikkakosken toimipistettä lähimmät asuinrakennukset sijaitsevat 250 m ja 350 m päässä toimipisteen katolla sijaitsevasta ilmanvaihdon poistoputkesta itään sekä lounaaseen. Lähimmät rakennukset ovat Tasowheel Tikka Oy:n rakennus 75 m päässä poistoputkesta lounaaseen, sekä puolustusvoimien rakennus 100 m etelään. Lähin pelto sijaitsee n. 1 km kaakkoon ja lähimmät järvet n. 500 m länteen.

MAP Medical Technologies Oy:n Tikkakosken toimipisteessä käsitellyistä aineista ainoastaan ¹²³I ja ¹³¹I päätyvät merkittävässä määrin ulos ilmanvaihdon kautta. Kussakin tarkastelupisteessä oleskelevälle väestön edustajalle lasketaan kummankin jodin isotoopin aiheuttama annos, sekä yhteisannos. Näin saadaan kattava käsitys aktiivisten ilmapäästöjen vaikutuksista päästöpiirteen lähistöllä.

Säteilyannos ja sen määrittäminen

Säteilyannoksen kuvaamiseen käytetään ekvivalenttiannosta ja efektiivistä annosta, jotka ovat laskennallisia keskimääräissuureita. Ekvivalenttiannos kuvaa elimeen kertyneen annoksen vaikutusta ottaen huomioon säteilyn laatutekijät. Efektiivinen annos puolestaan on kehon kaikkiin osiin kohdistuneiden painotettujen ekvivalenttiannosten painotettu summa. [L1.3]

Efektiivisen annoksen mittaaminen suoraan ei ole mahdollista, koska se vaatisi säteilystä absorboituvan annoksen tuntemista kehon jokaisessa pisteessä. Tämän vuoksi säteilytyössä saatua annosta mitataan usein henkilökohtaisella annosmittarilla. Mittariin absorboituneesta annoksesta voidaan jälkikäteen laskennallisesti arvioida mittaria käyttäneen henkilön saama efektiivinen annos. Tämä menetelmä ei kuitenkaan ole mahdollinen, kun määritetään laajemman väestön

saamaa annosta yksittäisestä lähteestä. Sen sijaan esimerkiksi ilmapäästöistä aiheutuva annos voidaan arvioida laskennallisten mallien avulla.

Yksinkertaistettu malli

Yksinkertaisin tapa määrittää suurin mahdollinen päästöstä aiheutuva annos on olettaa, ettei päästö laimene lainkaan matkalla päästöpuolesta mittauspuoleeseen. Tämä on sama, kuin että laitos tuottaisi ympärilleen päästöpuolesta, jonka aktiivisuuskonsentraatio vastaa ilmanvaihdon poistoputken aktiivisuuskonsentraatiota. Tätä mallia voidaan käyttää sulkemaan pois tarkemman tarkastelun tarve toiminnoissa, jotka tuottavat vain minimaalisia päästöjä.

IAEA:n raportti [L1.2] antaa tällaisella mallilla lasketut annoskertoimet yleisimmin käytetyille radionuklideille. Esimerkkihenkilön annos saadaan laskusta

$$E_i = C_i \cdot k_i,$$

jossa E_i on radionuklidista i aiheutuva efektiivinen annos, C_i on kyseisen nuklidin keskimääräinen päästökonsentraatio (Bq/m^3) ja k_i on kyseisen nuklidin annoskerroin. MAP:n vuoden 2015 päästötietojen ja IAEA:n raportin antamien annoskertoimien perusteella saadaan ^{123}I :n aiheuttamaksi vuosiansioksi

$$136,52 \text{ Bq}/\text{m}^3 \cdot 1,2 \cdot 10^{-5} (\text{Sv}/\text{a})/(\text{Bq}/\text{m}^3) = 0,00163824 \text{ Sv}/\text{a} = 1638,24 \mu\text{Sv}/\text{a}$$

ja ^{131}I :n aiheuttamaksi annokseksi

$$26,99 \text{ Bq}/\text{m}^3 \cdot 3,7 \cdot 10^{-2} (\text{Sv}/\text{a})/(\text{Bq}/\text{m}^3) = 0,99863 \text{ Sv}/\text{a} = 998630 \mu\text{Sv}/\text{a},$$

jotka molemmat ylittävät viranomaisien salliman $10 \mu\text{Sv}$ vuosiansioksen reilusti. Koska sallittu annos ylittyy tällä erittäin konservatiivisella mallilla, on syytä käyttää tarkempaa päästöpuolesta mallia.

Gaussinen päästöpuolesta malli

Gaussinen päästöpuolesta malli (Gaussian Plume Model) on matemaattinen malli, joka arvio ilmassa olevan aineen konsentraation jakautumista tietyissä sääolosuhteissa. Teoriassa malli pätee vain suhteellisen vakaisissa olosuhteissa sään ja ympäristön osalta, mutta sitä on testattu onnistuneesti varsin monimuotoisiin dispersio-ongelmiin [L1.2].

IAEA:n raportissa [L1.2] mallia on sovellettu radioaktiivisen päästön aktiivisuuskonsentraation määrittämiseen tietyn etäisyyden päässä päästön lähteestä. Käytetyt laskut olettavat, että päästö on suhteellisen tasaista ja pitkäaikaista, jolloin maaperään laskeutuva aktiivisuus saavuttaa muutaman puoliintumisaikansa kuluessa jotakuinkin tasapainotilan.

Malli on tarkoitettu antamaan konservatiivinen arvio maksimaalisesta annoksesta väestön edustajaan. Tämä tulee kyseeseen, kun halutaan osoittaa vaadittujen päästöpuolesta rajojen alittaminen laitoksessa, jossa päästöpuolesta rajoja ei lähestytä. Suuremman skaalan toiminnassa, jossa päästöpuolesta rajoja lähestytään, on syytä käyttää tarkempaa mallia. Mallin suunnittelussa on lähdetty siitä, että se

minimoi todennäköisyyden aliarvioida yksilön saama annos alle kymmenesosaan. Tämän vuoksi suositellaan tarkemman mallin käyttöä, mikäli päästöpilvimallin tuottama annosarvio ylittää 10% viranomaisten antamasta annosrajasta. [L1.2]

Aktiivisuuskonsentraatio

Päästöpilvimalli ottaa huomioon ilmvirran pyörteisyyden päästöpuoleen lähistöllä. Malli on siis erilainen riippuen siitä, sijaitseeko tarkasteltava mittauspiste lähellä suurta rakennusta vai onko se vapaasti virtaavassa ilmassa. Jos päästöpuoleen korkeus H on yli 2,5-kertainen suurimman lähellä sijaitsevan rakennuksen korkeuteen H_B verrattuna, katsotaan ilmvirran päästöpuoleen ympärillä olevan lähes häiriintymätön. Jos päästöpuoleen korkeus on tätä matalampi, mutta mittauspiste etäisyys päästöpuoleesta (x) on kauempana kuin 2,5 kertaa suurimman lähellä sijaitsevan rakennuksen seinän pinta-alan (A_B) neliöjuuri, katsotaan mittauspiste olevan rakennuksen vaikutuspiirissä (wake zone). Jos päästöpuole on matalalla ja tarkastelupiste lähellä rakennusta, katsotaan mittauspiste olevan rakennuksen takana olevassa pyörteessä (cavity zone). [L1.2]

MAP:n Tikkakosken toimipisteellä ilmanvaihdon poistoputken korkeus maanpinnasta (H) on noin 8 m ja lähimmäksi seinäksi voidaan katsoa rakennuksen takana oleva metsä, jonka pinta-ala on noin 600 m² (korkeus 12 m, leveys 50 m). Tällöin vapaan ilmvirran ehto ($H \geq 2,5 \cdot H_B$) ei täyty, koska

$$8\text{m} \leq 2,5 \cdot 12\text{m},$$

eikä jälkipyörrealueen ehto ($x \leq 2,5 \cdot \sqrt{A_B}$) täyty, koska etäisyys lähimpään mittauspisteeseen

$$75\text{m} \geq 2,5 \cdot \sqrt{600\text{m}^2}.$$

Tällöin kaikki tarkastelupisteet ovat metsäseinämän vaikutuspiirissä (wake zone). Wake zone alueella ilman aktiivisuuskonsentraatio C_A saadaan yhtälöstä

$$C_A = \frac{P_p B Q_i}{u_a},$$

missä Q_i on radionuklidin i keskimääräinen päästönopeus (Bq/s), P_p on osuus ajasta, jonka tuuli puhaltaa mittauspistettä kohti, B on diffuusiokerroin ja u_a on tuulen keskimääräinen nopeus (m/s). Diffuusiokerroin riippuu päästöpuoleen geometriasta seuraavasti

$$B = \frac{12}{\sqrt{2\pi^3}} \cdot \frac{1}{x \sum_z},$$

missä σ_z on vertikaalinen hajontaparametri ja

$$\sum_z = \left(\sigma_z^2 + \frac{A_B}{\pi} \right)^{0,5}.$$

Alle 45m päästökorkeuksille σ_z saadaan yhtälöstä

$$\sigma_z = \frac{(0,06)(x)}{\sqrt{1 + (0,0015)(x)}},$$

missä x on mittauspisteen etäisyys päästöpisteestä. Tarkasteltavien pisteiden etäisyydet arvioitiin kartan avulla ja tuulitiedot saatiin ilmatieteenlaitoksen tilastosta [L1.4]. Esimerkkitapauksena aktiivisuuskonsentraatio 250 metrin päässä idässä sijaitsevalle omakotitalolle saatiin ^{131}I -aktiivisuuskonsentraatio seuraavasti:

$$\sigma_z = \frac{0,06 \cdot 250}{\sqrt{1 + 0,0015 \cdot 250}} = 12,79 \dots,$$

jolle IAEA:n raportti antaa yksiköksi m. Tästä saadaan laskettua

$$\sum_z = \left((12,79 \dots m)^2 + \frac{600m^2}{\pi} \right)^{0,5} = 18,83 \dots m,$$

jolloin dispersiovakioksi saadaan

$$B = \frac{12}{\sqrt{2\pi^3}} \cdot \frac{1}{250m \cdot 18,83 \dots m} = 3,236 \dots \cdot 10^{-4} m^{-2}$$

ja aktiivisuuskonsentraatioksi saadaan

$$C_A = \frac{0,06 \cdot 3,236 \dots \cdot 10^{-4} m^{-2} \cdot 26,99 \text{ Bq/s}}{3 \text{ m/s}} = 1,747 \dots \cdot 10^{-4} \text{ Bq/m}^3.$$

Laskeuma

Ilman mukana kulkeutuva radioaktiivinen aines laskeutuu myös maahan, mistä käsin se aiheuttaa ulkoista annosta alueella oleskeleville henkilöille. Ilman aktiivisuuskonsentraation perusteella voidaan arvioida maahan kertyväksi laskeumanopeudeksi

$$\dot{d} = V_{tot} C_A,$$

jossa V_{tot} on kokonaislaskeumakerroin. Kokonaislaskeumakerroimen arvona suositellaan jodin tapauksessa käytettävän arvoa 1000 m/d, joka vastaa ^{131}I - ja ^{137}Cs -laskeumaa Tšernobylin onnettomuuden jälkeen [L1.2].

Esimerkkinä käytetyn omakotitalon kohdalla laskeumanopeudeksi saadaan

$$\dot{d} = 1000 \text{ m/d} \cdot 1,747 \dots \cdot 10^{-4} \text{ Bq/m}^3 = 1,747 \dots \cdot 10^{-1} \text{ Bq/m}^2 \text{ d}.$$

Veden aktiivisuuskonsentraatio

Radioaktiivinen laskeuma voi kertyä myös vesivarantoihin ja päätyä sitä kautta aiheuttamaan annosta väestölle. Vettä käytetään mm. viljelykasvien kastelemiseen, kalastamiseen sekä karjan ja ihmisen juomavetenä. Lisäksi veden aktiivisuus voi aiheuttaa annosta esimerkiksi uimareille.

Järviveden aktiivisuuskonsentraatio voidaan laskea järveen kertymän laskeuman perusteella. Jos tarkasteltavan radionuklidin puoliintumisaika on lyhyt tai järven veden vaihtuvuus jokien takia on suurta, voidaan järviveden aktiivisuuskonsentraatiota pitää vakiona. IAEA:n raportti [L1.2] antaa tälle ehdoksi

$$\left(\frac{q_r}{V} + \lambda_i\right) > 10^{-8} \text{s}^{-1},$$

missä q_r on jokien tilavuusvirta m^3/s , V on järven tilavuus m^3 ja λ_i on tutkittavan radionuklidin hajoamisvakio $1/\text{s}$. Sekä ^{123}I , että ^{131}I täyttävät tämän ehdon riippumatta järven parametreista, joten pitkäaikaisen päästön tapauksessa molempien konsentraatiota järvivedessä voidaan pitää vakiona. Aktiivisuuden kertymänopeus järvessä (Q'_i) on suoraan järveen lasketun aktiivisuuden ja ilmasta järveen laskeutuvan aktiivisuuden summa. Tarkasteltavassa tapauksessa suoraan vesistöihin ei kuitenkaan päästetä radioaktiivisia jätteitä, joten ne voidaan jättää huomiotta. Laskeuman osalta voidaan olettaa, että järveen kertyy valumavesiä noin 100 kertaa järven kokoiselta alueelta ja että valumavesien mukana järveen kulkeutuu noin 2% maahan päätyneestä aktiivisuudesta [L1.2]. Näin järven vaikuttava pinta-ala on kolminkertainen järven todelliseen pinta-alaan verrattuna, eli kertymänopeus on

$$Q'_i = \frac{3dA_l}{86400},$$

missä d on laskeumanopeus ja A_l on järven pinta-ala.

Järviveden aktiivisuuskonsentraatio (C_w) lyhytikäisen radionuklidin tapauksessa on IAEA:n raportin mukaan

$$C_w = \frac{Q'_i}{q_r + \lambda_i V}.$$

Järven pohjasedimenttiin laskeutuneen aktiivisuuden aktiivisuuskonsentraatio $C_{s,s}$ voidaan määrittää yhtälöstä

$$C_{s,s} = \frac{0,1 \cdot 0,001 K_d \cdot 60 C_w}{1 + 0,001 S_s K_d} \cdot \frac{1 - e^{-\lambda_i T_e}}{\lambda_i T_e},$$

jossa K_d on jakautumiskerroin (L/Kg), joka kuvaa aktiivisuuden jakautumista pohjasedimenttien ja järviveden välillä. S_s on järviveden mukana kulkevan sedimentin osuus (Kg/m^3) ja T_e on aktiivisuuden kertymäaika sekunteina. K_d arvona käytettiin IAEA:n raportin taulukoitua lukua 10

L/Kq, S_s arvona käytettiin lukua $5 \cdot 10^{-2} \text{ kg/m}^3$ (raportin suositus järville ja joille) ja T_e arvona käytettiin yhtä vuotta (jolla oletetaan saatavan konservatiivinen arvio).

Päästöpuoleen lähin järvi sijaitsee n. 500m lounaaseen. järven kohdalla laskeumanopeudeksi ^{131}I :lle saatiin $9,73 \cdot 10^{-2} \text{ Bq/m}^2\text{d}$, jolloin kertymänopeus

$$Q'_i = \frac{3 \cdot 9,73 \cdot 10^{-2} \text{ Bq/m}^2\text{d} \cdot 10^5 \text{ m}^2}{86400 \text{ s/d}} = 0,3378 \dots \text{ Bq/s},$$

järviveden aktiivisuuskonsentraatio

$$C_w = \frac{0,3378 \dots \text{ Bq/s}}{10 \text{ m}^3/\text{s} + 9,98 \cdot 10^{-7} \text{ 1/s} \cdot 5 \cdot 10^5 \text{ m}^3} = 3,217 \dots \cdot 10^{-2} \text{ Bq/m}^3$$

ja pohjasedimentin aktiivisuuskonsentraatio

$$C_{s,s} = \frac{0,1 \cdot 10^{-3} \text{ m}^3/\text{L} \cdot 10 \text{ L/Kg} \cdot 60 \text{ Kg/m}^3 \cdot 3,217 \cdot 10^{-2} \text{ Bq/m}^3}{1 + 10^{-3} \text{ m}^3/\text{L} \cdot 5 \cdot 10^{-2} \text{ kg/m}^3 \cdot 10 \text{ L/Kg}} \cdot \frac{1 - e^{-9,98 \cdot 10^{-7} \text{ 1/s} \cdot 3,15 \cdot 10^7 \text{ s}}}{9,98 \cdot 10^{-7} \text{ 1/s} \cdot 3,15 \cdot 10^7 \text{ s}}$$

$$= 6,141 \dots \cdot 10^{-5} \text{ Bq/m}^3.$$

Kasvien aktiivisuuskonsentraatio

Maahan päätyneestä laskeumasta osa päätyy lopulta myös kasveihin. Kasveja puolestaan käytetään ravinnoksi sekä ihmisille että karjalle, josta se puolestaan voi päätyä ihmisravinnoksi mm. maitotuotteiden tai lihan muodossa. Radioaktiivisia aineita kertyy kasveihin pääasiassa juurien kautta maaperästä ja kasvin pinnan kautta pinnalle jääneestä laskeumasta.

Pitkäaikaisessa altistuksessa kasvissa ja sen pinnalla olevan radioaktiivisen aineen konsentraatio $C_{v,i,1}$ (Bq/Kg) kasvin syötävissä osissa voidaan laskea yhtälöstä

$$C_{v,i,1} = \frac{d_i \alpha [1 - \exp(-\lambda_{E_i^v} t_e)]}{\lambda_{E_i^v}},$$

jossa α on osuus aktiivisuudesta, joka imeytyy kasvin syötävään osaan, t_e on aika, jonka kasvit altistuvat aktiivisuudelle kasvukauden aikana ja $\lambda_{E_i^v}$ on aktiivisuuden vähenemisvakio, joka on summa radionuklidin hajoamisvakioista ja ei-radioaktiivisesta vähenemisvakioista λ_w . Ei-radioaktiivinen vähenemisvakio kuvaa aktiivisuuden vähenemistä kasvin pinnalta muilla tavoilla kuin radioaktiivisen hajoamisen kautta. IAEA:n raportti antaa ei-radioaktiiviseksi vähenemisvakioksi

kaikille kasvipinnoille 0,05 1/d ja ruokakasvien altistumisajaksi kasvukaudella 60 päivää. Syötävään osaan kertyvän aktiivisuuden osuudeksi annetaan 0,3 m²/Kg.

Kasviin maan kautta imeytyvän aktiivisuuden määrä $C_{v,i,2}$ voidaan laskea yhtälöstä

$$C_{v,i,2} = F_v \cdot C_{s,i} ,$$

missä $C_{s,i}$ on isotoopin i aktiivisuus maaperässä ja F_v on kertymäkerroin, eli kuinka paljon kasvin syötäviin osiin kertyy aktiivisuutta suhteessa maaperän aktiivisuuteen. Maaperän aktiivisuus voidaan laskea yhtälöstä

$$C_{s,i} = \frac{\dot{d}_i [1 - \exp(-\lambda_{E_i^s} t_b)]}{\rho \lambda_{E_i^s}}$$

jossa ρ on juurakon alueen pintatiheys (kg/m²) ja $\lambda_{E_i^s}$ on aktiivisuuden ei-radioaktiivinen vähenemisvakio kasvin juurakon ympäristössä. $\lambda_{E_i^s}$ on radionuklidin hajoamisvakion ja ei-radioaktiivisista prosesseista koostuvan vähenemisvakion λ_s summa. IAEA:n raportti antaa pintatiheydelle arvion 260 Kg/m² ja ei-aktiiviseksi vähenemisvakioksi jodille arvon 0 1/d.

Yhdistämällä kasvin pinnalta ja maaperästä kertyvät aktiivisuudet, saadaan kasvien kokonaisaktiivisuuskonsentraatio käyttöhetkellä

$$C_{v,i} = (C_{v,i,1} + C_{v,i,2}) \exp(-\lambda_i t_h),$$

missä t_h on viive kasvin keräämisen ja käyttämisen välillä.

Esimerkkinä käytetyn omakotitalon lähistöllä saadaan pinnan kautta imeytyväksi ¹³¹I-aktiivisuudeksi

$$C_{v,i,1} = \frac{0,175 \text{ Bq/m}^2 \text{d} \cdot 0,3 \text{ m}^2/\text{Kg} [1 - \exp(-0,136 \text{ 1/d} \cdot 60 \text{d})]}{0,136 \text{ 1/d}} = 0,3859 \dots \text{ Bq/Kg},$$

maaperän aktiivisuudeksi

$$C_{s,i} = \frac{0,175 \text{ Bq/m}^2 \text{d} [1 - \exp(-0,0862 \text{ 1/d} \cdot 11000 \text{d})]}{260 \text{ Kg/m}^2 \cdot 0,0862 \text{ 1/d}} = 7,808 \dots \cdot 10^{-3} \text{ Bq/Kg}.$$

ja juurakon kautta imeytyväksi aktiivisuudeksi

$$C_{v,i,2} = 7,808 \dots \cdot 10^{-3} \text{ Bq/Kg} \cdot 0,02 = 1,562 \dots \cdot 10^{-4} \text{ Bq/Kg}$$

Kasvien kokonaisannoskonsentraatioksi saadaan

$$C_{v,i} = \left(0,3859 \dots \frac{Bq}{Kg} + 1,562 \dots \cdot 10^{-4} \frac{Bq}{Kg}\right) \exp\left(-8,62 \cdot 10^{-2} 1/d \cdot 14d\right) \\ = 0,1155 \frac{Bq}{Kg}.$$

Annokset

Kun tunnetaan ilman, veden ja kasvien aktiivisuuskonsentraatiot, sekä päästöistä aiheutuneet laskeumat, voidaan määrittää päästöistä aiheutuneet annokset. Ulkoista annosta kertyy aktiivisesta ilmasta ja vedestä sekä maassa olevan laskeuman läheisyydessä oleskelusta. Sisäistä annosta puolestaan kertyy aktiivisen ilman hengittämisestä ja saastuneen ruuan syömisestä.

Ulkoisen säteily

Ulkoisen säteilyn annos, E_{im} , joka aiheutuu päästöpilvessä oleskelusta, voidaan laskea yhtälöstä

$$E_{im} = C_A DF_{im} O_f,$$

jossa C_A on ilman aktiivisuuskonsentraatio, DF_{im} on annoskerroin immersiolle ja O_f on osuus vuodesta, joka päästöpilvessä oleskellaan. Annoskertoimet saatiin IAEA:n raportista [L1.2], joka listaa niitä useimmin käytetyille radionuklideille. Oleskeluajan osuudelle käytettiin arvoa 1,0 asuinrakennuksille ja 0,5 työpaikoille. Työpaikkojen arvo lienee hieman konservatiivinen keskimääräisen työajan (35–40 viikkotyötuntia) ollessa alle 0,25.

Jotkin radionuklidit tuottavat myös beeta-säteilyä, joka aiheuttaa säteilyannosta iholle päästöpilvessä oleskelun aikana. IAEA:n raportti [L1.2] antaa ihoannokselle annoskertoimen DF_s , jonka avulla saadaan laskettua ihoannos E_s seuraavasti,

$$E_s = C_A DF_s O_f.$$

Esimerkkinä käytetyn omakotitalon kohdalla annosnopeuksiksi saadaan ulkoiselle annokselle

$$E_{im} = 1,7472 \dots \cdot 10^{-4} \frac{Bq}{m^3} \cdot 5,80 \dots \cdot 10^{-7} \frac{Sv/a}{Bq/m^3} \cdot 1 = 1,013 \dots \cdot 10^{-4} \mu Sv/a$$

ja beeta-säteilyn annosnopeudeksi

$$E_s = 1,7472 \dots \cdot 10^{-4} \frac{Bq}{m^3} \cdot 9,40 \cdot 10^{-7} \frac{Sv/a}{Bq/m^3} \cdot 1 = 1,642 \dots \cdot 10^{-4} \mu Sv/a.$$

Myös maassa olevasta laskeumasta aiheutuva annos saadaan laskettua vastaavalla annoskertoimella DF_{gr} , mutta nyt konsentraation sijaan käytetään maaperän laskeumatiheyttä C_{gr} . Laskeumatiheys saadaan yhtälöstä

$$C_{gr} = \frac{\dot{d}_i [1 - \exp(-\lambda_E t_b)]}{\lambda_E},$$

jossa \dot{d}_i on laskeumanopeus ($\text{Bq}/\text{m}^2\text{d}$), λ_E on kokonaisvähennysvakio, joka maahan päätyneen jodin tapauksessa on yhtä kuin hajoamisvakio λ_i [L1.2] ja t_b on jatkuvan päästön kesto aika päivinä. Jatkuvasa toiminnassa päästön kesto aikana IAEAN raportti suosittelee käytettävän 30 vuotta, mutta eksponentiaalinen puoliintumistermi päättyy molempien tarkasteltavien jodin isotooppien puoliintumisajoilla nollaan jo alle vuodessa. Varsinainen laskeuman aiheuttama annos saadaan yhtälöstä

$$E_{gr} = C_{gr} DF_{gr} O_f$$

Esimerkkitapauksessa laskeumatiheydeksi saadaan

$$C_{gr} = \frac{0,175 \text{ Bq}/\text{m}^2\text{d} \cdot [1 - 0]}{8,62 \cdot 10^{-2} \text{ 1}/\text{d}} = 2,030 \dots \text{ Bq}/\text{m}^2$$

ja laskeuman aiheuttamaksi annokseksi

$$E_{gr} = 2,030 \dots \text{ Bq}/\text{m}^2 \cdot 1,20 \cdot 10^{-8} \frac{\text{Sv}/\text{a}}{\text{Bq}/\text{m}^2} \cdot 1 = 2,436 \dots \cdot 10^{-2} \mu\text{Sv}/\text{a}.$$

Ulkoista annosta voi kertyä myös järvisedimentin lähistöllä oleskelusta. Sedimentistä aiheutuva annos E_m saadaan yhtälöstä

$$E_m = C_{s,s} DF_{gr} O_f,$$

jossa $C_{s,s}$ on järvisedimenttiin kertynyt aktiivisuustiheys (Bq/m^2). Päästöapistettä lähimmässä järvestä järvisedimentin aktiivisuuden aiheuttama annosnopeus on

$$E_m = 6,1415 \dots \cdot 10^{-5} \text{ Bq}/\text{m}^3 \cdot 1,20 \cdot 10^{-8} \frac{\text{Sv}/\text{a}}{\text{Bq}/\text{m}^2} \cdot 1 = 7,370 \dots \cdot 10^{-7} \mu\text{Sv}/\text{a},$$

olettaen, että henkilö olisi järvestä jatkuvasti. Järven ja pohjasedimentin aktiivisuuskonsentraatiot ovat niin pieniä, ettei niistä kerry merkittävää annosta järven lähellä oleskeville tai siellä uiville. Näin pienillä järvilla ei myöskään ole laajamittaista kalastuskäyttöä.

Sisäinen annos

Päästöpilvessä hengittämisestä saatavalle sisäiselle annokselle IAEA:n raportti antaa annoskertoimen DF_{inh} . Hengittämisestä kertyvä annos voidaan siis laskea yhtälöstä

$$E_{inh} = C_A R_{inh} DF_{inh},$$

jossa R_{inh} on hengitysnopeus (m^3/a). Esimerkkitapauksessa annosnopeudeksi saadaan

$$E_{inh} = 1,7472 \dots \cdot 10^{-4} Bq/m^3 \cdot 1400 m^3/a \cdot 7,20 \cdot 10^{-8} Sv/Bq = 1,761 \dots \cdot 10^{-2} \mu Sv/a.$$

Sisäiseksi annokseksi määritellään myös kontaminoituneen ruoan syömisestä saatava annos. Tämä voidaan laskea yhtälöstä

$$E_{ing} = C_{p,i} H_p D F_{ing},$$

jossa $C_{p,i}$ on ruoka-aineen p sisältämä aktiivisuuskonsentraatio radionuklidia i , H_p on kyseisen ruoka-aineen kulutus vuodessa ja $D F_{ing}$ on annoskerroin kyseisen radionuklidin nielemiselle. Jos oletetaan, että esimerkkinä käytetyn omakotitalon pihalla kasvatettaisiin kaikki siellä asuvan yksivuotiaan kasviperäinen ravinto, saadaan annokseksi

$$E_{ing} = 0,1155 \dots Bq/Kg \cdot 150 kg/a \cdot 1,80 \cdot 10^{-7} Sv/Bq = 3,119 \dots \mu Sv/a.$$

Koska oletamus kaiken kasviperäisen ruoan tuotannosta omakotitalon pihalla on melko konservatiivinen, annos laskettiin myös olettamalla, että päästöpuoleen lähistöllä asuvat ihmiset tuottaisivat itse keskimäärin 10% kasviperäisestä ravinnostaan. Tällöin kasviperäisen ravinnon nielemisestä kertyvä annos on

$$E_{ing,10} = E_{ing} \cdot 0,1$$

Eli esimerkkitapauksessa

$$E_{ing,10} = 3,119 \dots \mu Sv/a \cdot 0,1 = 0,3119 \dots \mu Sv/a$$

Kokonaisannos

Tarkasteltavassa pisteessä oleskelevan henkilön päästöistä johtuva kokonaisannos saadaan muuttamalla ihon beeta-annos efektiiviseksi annokseksi ja summaamalla kaikki edellä käsitellyt annoksen kertymämekanismit yhteen. Tällöin

$$E_{tot} = E_{im} + 0,01 \cdot E_s + E_{gr} + E_{inh} + E_{ing}.$$

Esimerkkitapauksena on laskettu annos 250m päässä päästöpuolesta itään sijaitsevassa omakotitalossa asuvan 1-vuotiaan lapsen annos, joka on

$$E_{tot} = 1,013 \dots \cdot 10^{-4} \mu Sv/a + 1,642 \dots \cdot 10^{-4} \mu Sv/a + 2,436 \dots \cdot 10^{-2} \mu Sv/a + 1,761 \dots \cdot 10^{-2} \mu Sv/a + 3,119 \dots \mu Sv/a = 3,161 \dots \mu Sv/a.$$

Taulukko 1: Vuosiannoslaskujen tulokset eri tarkastelupisteille

	työpaikka		asuinrakennus			
	lounas 75 m aikuinen	etelä 100 m aikuinen	itä 250 m aikuinen	lapsi	lounas 350 m aikuinen	lapsi
¹²³ I-annos (μSv)	1,76E-02	2,05E-02	4,47E-03	4,90E-03	4,43E-03	4,85E-03
¹³¹ I-annos (μSv)	1,62E-01	1,88E-01	1,07E+00	3,15E+00	1,06E+00	3,12E+00
kokonaisannos (μSv)	1,79E-01	2,08E-01	1,08E+00	3,16E+00	1,07E+00	3,12E+00
kokonaisannos 10% kasviannoksella (μSv)	-	-	1,44E-01	3,58E-01	1,42E-01	3,54E-01

aikuinen		Lapsi			
asuinrakennus lounas		asuinrakennus lounas			
I-123	I-131	I-123	I-131		
1,46E-05	9,98E-07	1,46E-05	9,98E-07	λ_i	Hajoamisvakio (1/s)
1,26E+00	8,62E-02	1,26E+00	8,62E-02	λ_d	Hajoamisvakio (1/d)
3,50E+02	3,50E+02	3,50E+02	3,50E+02	x	Etäisyys päästöstä
6,00E+02	6,00E+02	6,00E+02	6,00E+02	A_b	Seinän pinta-ala (m ²)
8,00E+00	8,00E+00	8,00E+00	8,00E+00	H	Päästöpuiteen korkeus
1,70E+01	1,70E+01	1,70E+01	1,70E+01	σ	Pystysuuntainen haajaantumiskerroin (vertical diffusion)
2,19E+01	2,19E+01	2,19E+01	2,19E+01	Σ_z	
1,99E-04	1,99E-04	1,99E-04	1,99E-04	B	Dispersiovakio etäisyydelle x (1/m ²)
1,00E-01	1,00E-01	1,00E-01	1,00E-01	P_p	Osuus ajasta jonka tuuli puhaltaa kohti mittauspistettä
3,10E+00	3,10E+00	3,10E+00	3,10E+00	u_a	Tuulen keskimääräinen nopeus (m/s)
1,37E+02	2,70E+01	1,37E+02	2,70E+01	Q_i	Keskimääräinen päästönopeus (Bq/s)
8,75E-04	1,73E-04	8,75E-04	1,73E-04	A_c	Ilman aktiivisuuskonsentraatio (Bq/m ³)
1,00E+03	1,00E+03	1,00E+03	1,00E+03	V_{tot}	Laskeumakerroin (oletus 1000 m/d)
8,75E-01	1,73E-01	8,75E-01	1,73E-01	d_i	Laskeumanopeus (Bq/(m ² d))
2,30E-07	5,80E-07	2,30E-07	5,80E-07	DF_{imm}	Immersion annoskerroin (Sv/a per Bq/m ³)
1,00E+00	1,00E+00	1,00E+00	1,00E+00	O_f	Osuus ajasta, joka pilvessä oleskellaan
2,01E-04	1,00E-04	2,01E-04	1,00E-04	E_{im}	Immersion aiheuttama annos (μSv/a)
3,00E-07	9,40E-07	3,00E-07	9,40E-07	DF_s	Ihon annoskerroin immersiossa (Sv/a per Bq/m ³)
2,63E-04	1,63E-04	2,63E-04	1,63E-04	$E_{im,s}$	Ihoannos betasta immersiossa (μSv/a).
1,10E+04	1,10E+04	1,10E+04	1,10E+04	t_b	Aika joka vastaa laitoksen toiminta-aikaa (30v)
0,00E+00	0,00E+00	0,00E+00	0,00E+00	λ_s	"Hajoamisvakio" muille kuin radioaktiivisille prosesseille (1/s)
6,94E-01	2,01E+00	6,94E-01	2,01E+00	C_{gr}	Laskeumatiheys (Bq/m ²)
5,30E-09	1,20E-08	5,30E-09	1,20E-08	DF_{gr}	Laskeuman annoskerroin (Sv/a per Bq/m ²)
3,68E-03	2,41E-02	3,68E-03	2,41E-02	E_{gr}	Laskeuman aiheuttama ulkoinen annos (μSv/a)
8,40E+03	8,40E+03	1,40E+03	1,40E+03	R_{inh}	Hengitysmäärä (m ³ /a)
7,40E-11	7,40E-09	7,90E-10	7,20E-08	DF_{inh}	Hengityksen annoskerroin (Sv/Bq)
5,44E-04	1,08E-02	9,68E-04	1,74E-02	E_{inh}	Annos, joka kertyy hengittämällä päästöpuiteessa (μSv/a)
3,00E-01	3,00E-01	3,00E-01	3,00E-01	α	Aktiivisuuskertymän osuus syötävässä osassa kasvia (m ² /Kg)
6,00E+01	6,00E+01	6,00E+01	6,00E+01	t_e	Aika jonka kasvi altistuu kasvukaudella (d)
5,00E-02	5,00E-02	5,00E-02	5,00E-02	λ_w	Ei-radioaktiivinen vähenemisvakio kasvin pinnoille
1,31E+00	1,36E-01	1,31E+00	1,36E-01	λ_{ev}	Efektiiivinen vähenemisvakio (1/d)
2,00E-01	3,81E-01	2,00E-01	3,81E-01	$C_{v,i,1}$	Aktiivisuuskonsentraatio kasvin päältä (Bq/Kg)
2,00E-02	2,00E-02	2,00E-02	2,00E-02	F_v	Kerroin kasvin aktiivisuudelle, jota kasvin syötävät osat keräävät
2,60E+02	2,60E+02	2,60E+02	2,60E+02	ρ	Maaperän pintatiheys
1,10E+04	1,10E+04	1,10E+04	1,10E+04	t_b	Aktiivisuuden kertymäaika maaperään (oletus 30 vuotta päivinä)
1,40E+01	1,40E+01	1,40E+01	1,40E+01	t_{in}	Aika sadonkorjuun ja kulutuksen välillä (oletus 14 päivää)
0,00E+00	0,00E+00	0,00E+00	0,00E+00	λ_s	Ei-radioaktiivinen vähenemisvakio juurakon aktiivisuudelle
1,26E+00	8,62E-02	1,26E+00	8,62E-02	λ_{ei}	Efektiiivinen vähenemisvakio (1/d)
5,34E-05	1,54E-04	5,34E-05	1,54E-04	$C_{v,i,2}$	Aktiivisuuskonsentraatio juurien kautta (Bq/Kg)
4,38E-09	1,14E-01	4,38E-09	1,14E-01	$C_{v,i}$	Yhteenlaskettu aktiivisuuskonsentraatio kasvissa (Bq/Kg)
4,10E+02	4,10E+02	1,50E+02	1,50E+02	H_p	Ruoka-aiheen kulutus vuodessa (Kg/a)
2,10E-10	2,20E-08	1,90E-09	1,80E-07	DF_{ing}	Nielemisen annoskerroin
3,77E-10	1,03E+00	1,25E-09	3,08E+00	E_{ing}	Syömisestä kertyvä annos (μSv/a)
4,43E-03	1,06E+00	4,85E-03	3,12E+00	$E_{tot,i}$	Kokonaisannos yksittäisestä nuklidista (μSv/a)
0,00	1,06	0,00	3,12		
1,07		3,12		E_{tot}	Kokonaisannos (μSv/a)

Kuva 1: Osa laskennassa käytetystä Excel-tilusta, jossa näkyy tulokset 350 m päästöpuiteesta lounaseen sijoittuvalle asuinalueelle.

Pohdintaa

Ehdottomasti suurimmat annosarviot saatiin radioaktiivisia aineita sisältävän ruoan syömisestä. Laskenta tehtiin vain kasviperäiselle ravinnolle, koska sitä voidaan kasvattaa päästöpiirteen lähialueilla. Muita ruoka-aineita, kuten kalaa, lihaa tai maitoa, ei tuoteta niin lähellä päästöpiirettä, että ne voisivat aiheuttaa merkittävää annosta. Ruoasta kertyvä annos laskettiin ensin oletuksella, että tarkasteltavan piirteen lähistöllä kasvatettaisiin kaikki kyseisessä talossa asuvien henkilöiden kasviperäinen ravinto. Tulos on suurempi kuin viranomaisrajan kymmenesosa sekä aikuiselle että 1-vuotiaalle. IAEA:n raportin mukaan tällöin olisi syytä käyttää tarkempaa mallia päästön vaikutusten määrittämiseksi. Tulosta arvioitaessa on kuitenkin syytä huomioida, että annosarvio olettaa kyseisen yksivuotiaan kuluttavan 150kg ja aikuisen 410kg omalla pihalla tuotettuja kasviksia ja vihanneksia vuodessa. On melko epätodennäköistä, että päästöpiirteen lähistöllä sijaitsevien omakotitalojen pihalla kasvatettaisiin koko perheen kaikki kasviperäinen ravinto. Jos oletetaan, että päästöpiirteen lähistöllä asuvat ihmiset tuottaisivat keskimäärin 10 % kasviperäisestä ravinnostaan omalla tontillaan, saataisiin laskuissa esimerkkinä käytetyssä omakotitalossa asuvalle aikuiselle vuosiansioksi 0,14 μSv ja 1-vuotiaalle lapselle 0,36 μSv . Myös oletus 10 % omavaraisuudesta taajaman omakotitaloalueella lienee varsin konservatiivinen, erityisesti puhuttaessa alueella asuvan väestön keskimääräisestä omavaraisuudesta.

Voidaan myös pohtia, onko yksittäinen rajatapaus säteilysuojelullisesti olennainen. Viranomaisraja 10 μSv vuodessa on melko pieni verrattuna esimerkiksi Jyväskylän alueen ulkoisesta luonnonsäteilystä johtuvaan vuosittaiseen annokseen (1180 μSv [L1.5]) tai suomalaisen keskimääräiseen vuosittaiseen säteilyannokseen (3230 μSv [L1.6]). Tiukka raja on ymmärrettävästi tarpeen, jos vuosiansiosta kertyy suurelle väkijoukolle, koska tällöin haittavaikutusten riski kasvaa väestötasolla. Kun täyden kasvisruokaomavaraisuuden aiheuttama kohonnut annos koskee mahdollisesti yksittäisiä henkilöitä, on 3 μSv lisäys vuositasolla merkityksentöntä. Esimerkiksi ulkoisesta luonnonsäteilystä johtuvan annoksen ero Jyväskylän ja Kotkan mittauspisteiden välillä on 700 $\mu\text{Sv}/\text{vuosi}$ [L1.5].

Jos aktiivisuutta sisältävän ruoan syömistä ei oteta huomioon, suurimmat annoksen lähteet ovat laskeuma ja hengittäminen. Lähimpien talojen pihalla ^{131}I -laskeumaa kertyy laskennassa tehtyjen oletusten mukaan yli 10 Bq/m^2 , mikä aiheuttaa jatkuvaa ulkoista annosta. Lisäksi hengittäessä jodi kertyy keuhkojen kautta elimistöön, jolloin se pääsee vaikuttamaan tehokkaammin kuin ulkoisen säteilyn kautta. Nämä annoksen kertymismekanismit jäävät konservatiivisista oletuksista huolimatta promilleluokkaan viranomaisen määrittämästä annosrajasta.

Tämän arvion perusteella MAP:n ympäröivälle väestölle aiheuttama annos on viranomaisrajojen puitteissa. Mahdollisimman konservatiivisilla oletuksilla gaussisen päästöpiivilmallin tuottamat annosarviot ylittävät rajan, jonka ylittyessä mallin kehittäjät suosittelevat käyttämään tarkempaa mallinnusta päästöarvion pienentämiseksi. Kohtuullistamalla oletuksia ruoan kotituotannon määrässä, päästöarvio kuitenkin alittaa mainitun rajan. Yleisesti ottaen MAP:n päästöjen vaikutus on pientä verrattuna esimerkiksi asuinpaikkakohtaiseen vaihteluun luonnonsäteilyn määrässä.

Lähteet

[L1.1] Säteilyturvakeskus. ST 6.2: *Avolähteiden käytöstä syntyvät radioaktiiviset jätteet ja päästöt*. Grano Oy, Helsinki, 2014.

[L1.2] IAEA *Safety Reports Series no. 19: Generic Models for Use in Assessing the Impact of Discharges of Radioactive Substances to the Environment*. IAEA, Vienna, 2001.

[L1.3] Marttila O. J. *Suureet ja yksiköt*. Teoksessa Ikäheimonen, T. K. (toim.), *Säteily ja sen havaitseminen*. Karisto Oy, Hämeenlinna 2002.

[L1.4] Pirinen P, Simola H, Aalto J, Kaukoranta J-P, Karlsson P ja Ruuhela R, *Tilastoja Suomen Ilmastosta 1981-2010*. Ilmatieteen laitos, Helsinki, 2012.

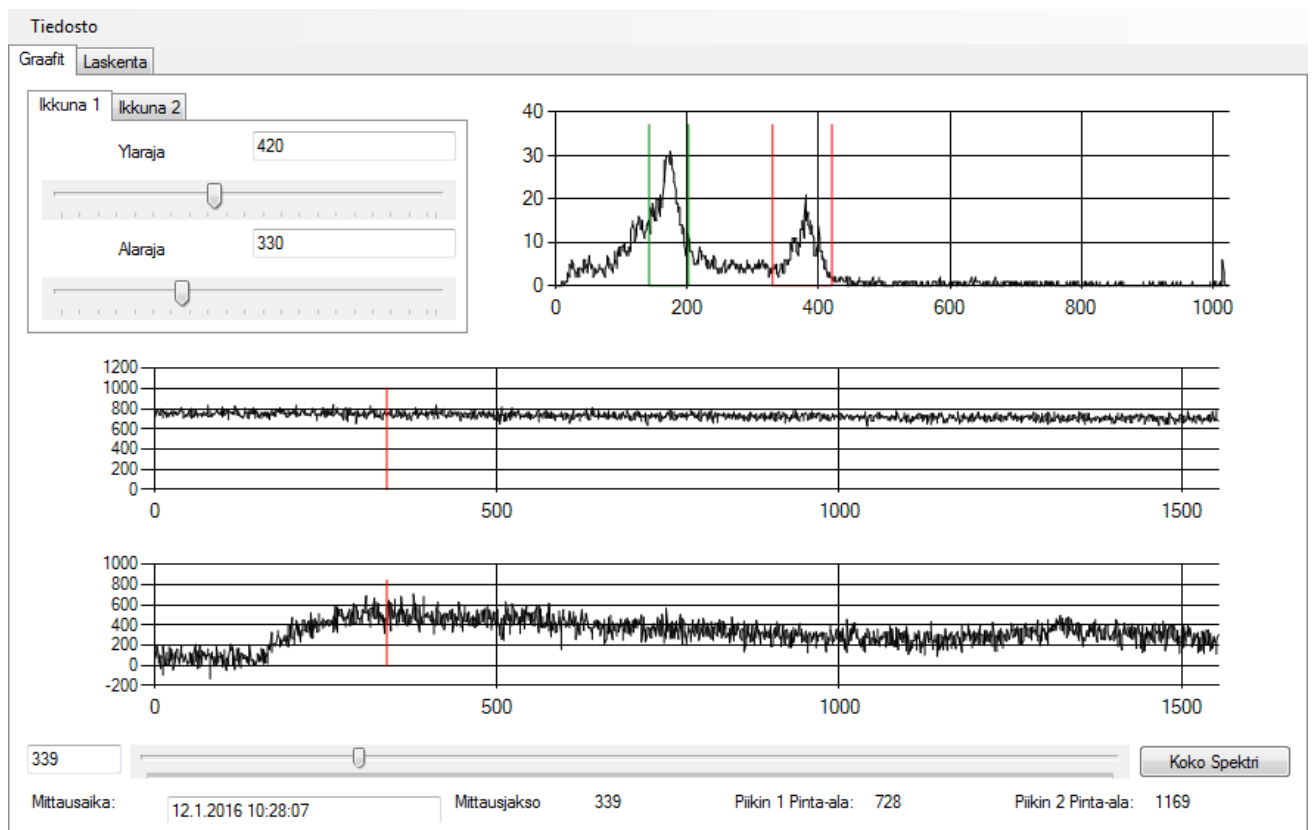
[L1.5] STUK, *Säteilytilanne tänään*-verkkosivu. Saatavilla: <http://www.stuk.fi/aiheet/sateily-ymparistossa/sateilytilanne-tanaan>. Viitattu 27.7.2016.

[L1.6] STUK, *Suomalaisen keskimääräinen säteilyannos*-verkkosivu. Saatavilla: <http://www.stuk.fi/aiheet/mita-sateily-on/ihmisen-radioaktiivisuus/suomalaisen-keskimaarainen-sateilyannos>. Viitattu 18.8.2016.

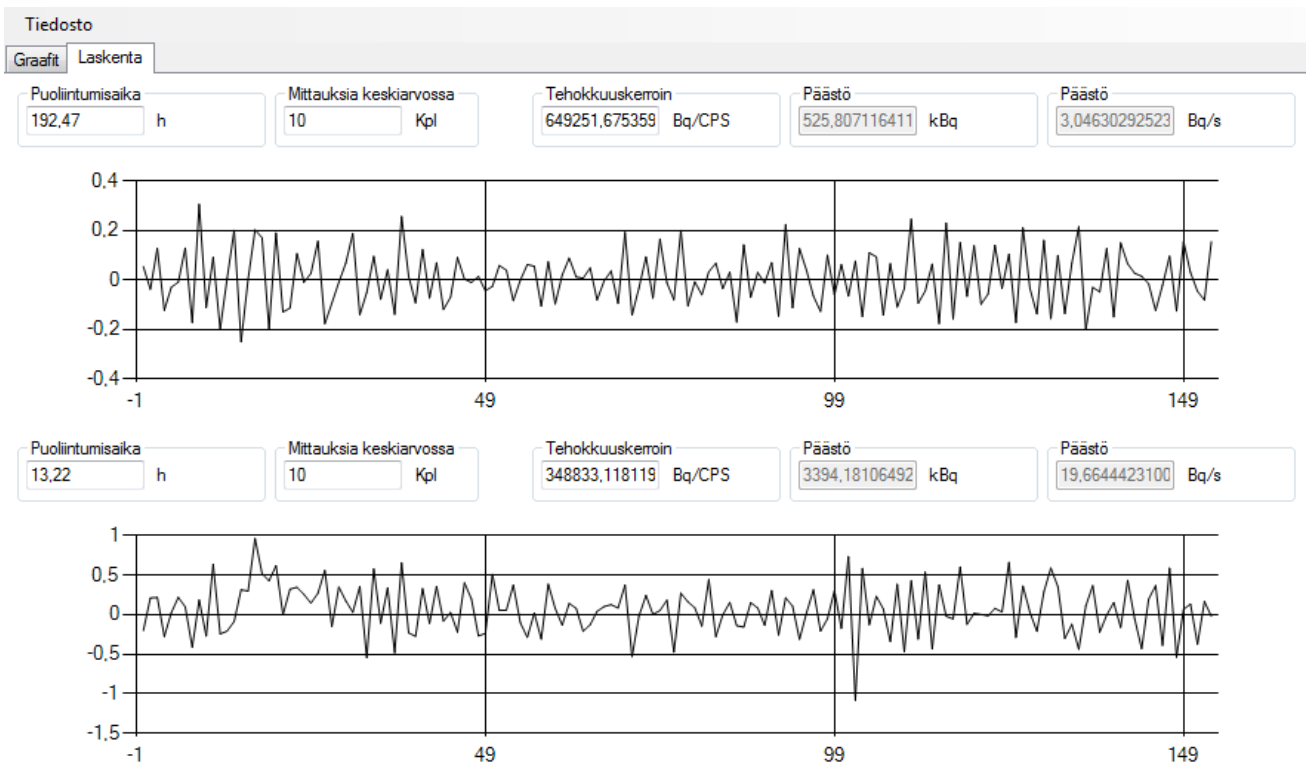
9.2. Liite 2: Laskentaohjelmisto

Päästöjen laskemiseen Hidexin järjestelmän tuottamasta spektridatasta käytettiin C#-kielistä Microsoft Visual Studio-ohjelmistolla tuotettua tietokoneohjelmaa. Tehdyssä ohjelmassa oli graafinen käyttöliittymä, jossa käytettiin Windows Forms-kirjastoa. Ohjelma koostui pääasiassa kolmesta luokasta. Form1.cs toimi pääluokkana ja sen tehtävänä oli graafisen käyttöliittymän sekä muiden luokkien ylläpito. DataHandler.cs laski matriisimuotoisesta spektridatasta halutun ROI:n pinta-aloja ajan funktiona. DataCalculator.cs määritteli DataHandler-luokan laskemasta datasta varsinaisen päästön. Nämä kolme luokkaa ovat tämän liitteen lopussa. Näiden lisäksi ohjelmassa oli Visual Studion automaattisesti luomat luokat, joilla käyttöliittymän ulkoasu määriteltiin.

Ohjelman käyttöliittymä koostuu kahdesta välilehdestä. Graafit-välilehdellä (kuva L2.1) ohjelma esittää halutun spektrin, sekä kahden eri "ikkunan", eli ROI:n, pinta-alan mittausjakson funktiona. Käyttäjä voi muuttaa liukukytkimiä tai tekstikenttiä käyttäen mittausarvojen ROI:ta, sekä sitä, mistä kohdasta mittaus spektri esitetään. Lisäksi "Koko Spektri"-painikkeella saa näkyviin koko mittausjakson summaspektrin. Laskenta-välilehdellä (kuva L2.2) käyttäjä voi syöttää tekstikenttiin valitsemassaan ROI:ssa esiintyvän radionuklidin puoliintumisaajan, halutun mittausmäärän keskiarvoistuksessa, sekä mittausjärjestelmän tehokkuuskertoimen kyseiselle nuklidille. Tämän jälkeen ohjelma esittää kyseisestä ROI:sta lasketun päästön kuvaajana ja numeroina.



Kuva L2.1: Päästölaskentaohjelmiston graafisen käyttöliittymän "Graafit"-sivu, jossa käyttäjä voi valita ROI:t (punainen ja vihreä) yläreunan energiaspektristä ja nähdä ROI:n pinta-alan mittausjakson funktiona alemmissa kuvaajissa (punainen on ylempi kuvaaja ja vihreä alempi, alemmassa kuvaajassa on käytetty lineaarista taustavähennystä). Näytettäväksi spektri on valittu mittausjakson 339 spektri, jossa erottuu suodatinloukkuun kertynyt hiljalleen puoliintuva ^{131}I -piikki (punainen ROI), sekä tiistain tuotannosta kertynyt ^{123}I -piikki ^{131}I -jatkumon päällä (vihreä ROI). "Koko spektri" painikkeesta sai näkyviin kaikkien mittausjaksojen spektrien summan, jolloin tapahtumien määrä oli suurempi ja spektri tasaisempi.



Kuva L2.2: Päästölaskuriohjelmiston "Laskenta"-välilehti, jossa nähdään kuvaajamuodossa keskiarvoistettujen päästölukemien erotukset (laskentamenetelmässä Δy). Käyttäjä voi syöttää kummallekin ROI:lle oman puoliintumisaikan ja tehokkuuskertoimen, sekä määrittää, kuinka monta mittausta otetaan yhteen keskiarvoon. Kuvassa L2.1 nähtävä alimman kuvaajan nousu mittausjaksoilla 200-300 nähdään tämän kuvan alemmassa kuvaajassa keskiarvoistuksissa 20-30. Kuvaajan pysyminen nollan yläpuolella muodostaa laskurin antaman päästön. Kaikki muu käyrän heilahtelu nollan molemmin puolin on statistista virhettä, ja se kumoutuu käyrää integroitaessa.

Form1.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Windows.Forms.DataVisualization.Charting;

namespace PaastoLaskuriForm
{
    /// <summary>
    /// Ohjelman pääluokka. Hallitsee ohjelman graafista käyttöliittymää ja
    /// ottaa vastaan käyttäjän syötteet. Lisäksi luo ja hallitsee datan
    /// käsittelyyn käytettäviä dataHandler- ja DataCalculator-olioita.
    /// </summary>
    public partial class Form1 : Form
    {
        private bool loopLock = false;
        private int spektrinKeskus = -1;

        private DataHandler data;
        private DataClaculator calculator;

        /// <summary>
        /// alustetaan ohjelma
        /// </summary>
        public Form1()
        {
            InitializeComponent();
        }

        /// <summary>
        /// Kysytään käyttäjältä avattava tiedosto ja
        /// luetaan koko tiedosto muistiin.
        /// </summary>
        private void readFile()
        {
            OpenFileDialog openFileDialog = new OpenFileDialog();

            openFileDialog.InitialDirectory = "c:\\";
            openFileDialog.Filter = "txt files (*.txt)|*.txt|All files (*.*)|*.*";
            openFileDialog.FilterIndex = 2;
            openFileDialog.RestoreDirectory = true;
            string[] lines;

            if (openFileDialog.ShowDialog() == System.Windows.Forms.DialogResult.OK)
            {
                try
                {
                    lines = System.IO.File.ReadAllLines(@openFileDialog.FileName);
                    data = new DataHandler(lines);
                    data.spektriLaskettu += data_spektriLaskettu;
                    data.timelineLaskettu1 += data_timelineLaskettu1;
                    data.timelineLaskettu2 += data_timelineLaskettu2;
                }
            }
        }
    }
}
```



```

        catch (Exception ex)
        {
            System.Windows.Forms.MessageBox.Show(ex.ToString());
        }
    }

}

/// <summary>
/// piirretään spektri kuvaajaelementtiin
/// </summary>
/// <param name="myosGraafi">true piirtää koko spektrin, false päivittää vain
rajamerkinntä</param>
private void plotSpectrum(bool myosGraafi)
{
    //jos päivitetään koko spektri
    if (myosGraafi)
    {
        //ensin poistetaan vanha spektri
        chartSpektri.Series["testi"].Points.Clear();
        //i käy läpi kaikki kanavat ja piirretään niitä vastaavat datapisteet
        for (int i = 0; i < 1024; i++)
        {
            chartSpektri.Series["testi"].Points.AddXY(i, data.spektri[i]);
        }
        //asetetaan kuvaajan tyyppi
        chartSpektri.Series["testi"].ChartType = SeriesChartType.FastLine;
        chartSpektri.Series["testi"].Color = Color.Black;
    }

    //nollataan rajat
    chartSpektri.Series["rajat"].Points.Clear();

    //piirretään rajat pisteinä
    chartSpektri.Series["rajat"].Points.AddXY(trackBarAlaraja.Value,
data.spektriMaxArvo * 1.2);
    chartSpektri.Series["rajat"].Points.AddXY(trackBarAlaraja.Value, 0);
    chartSpektri.Series["rajat"].Points.AddXY(trackBarYlaraja.Value, 0);
    chartSpektri.Series["rajat"].Points.AddXY(trackBarYlaraja.Value,
data.spektriMaxArvo * 1.2);

    //asetetaan rajakuvaajan tyypit
    chartSpektri.Series["rajat"].ChartType = SeriesChartType.FastLine;
    chartSpektri.Series["rajat"].Color = Color.Red;

    //nollataan toiset rajat
    chartSpektri.Series["rajat2"].Points.Clear();

    //piirretään rajat pisteinä
    chartSpektri.Series["rajat2"].Points.AddXY(trackBarAlaraja2.Value,
data.spektriMaxArvo * 1.2);
    chartSpektri.Series["rajat2"].Points.AddXY(trackBarAlaraja2.Value, 0);
    chartSpektri.Series["rajat2"].Points.AddXY(trackBarYlaraja2.Value, 0);
    chartSpektri.Series["rajat2"].Points.AddXY(trackBarYlaraja2.Value,
data.spektriMaxArvo * 1.2);

    //asetetaan rajakuvaajan tyypit
    chartSpektri.Series["rajat2"].ChartType = SeriesChartType.FastLine;
    chartSpektri.Series["rajat2"].Color = Color.Green;

    //asetetaan kuvaaja-alueen x-akselille minimiarvo
    chartSpektri.ChartAreas["ChartArea1"].AxisX.Minimum = 0;

```

```

//lasketaan spektrin pinta-ala näkyviin
if (data.spektri != null && spektrinKeskus != -1)
{
    int Piikin1PintaAla = 0;
    int Piikin2PintaAla = 0;

    //i käy alarajalta ylärajalle
    for (int i = int.Parse(textBoxAlaraja.Text); i <
int.Parse(textBoxYlaraja.Text); i++)
    {
        //summataan datapisteet
        Piikin1PintaAla += data.spektri[i];
    }
    //i käy toisen piikin alarajalta ylärajalle
    for (int i = int.Parse(textBoxAlaraja2.Text); i <
int.Parse(textBoxYlaraja2.Text); i++)
    {
        //summataan datapisteet
        Piikin2PintaAla += data.spektri[i];
    }
    //päivitetään tiedot näkyville
    labelPiikin2PintaAla.Text = Piikin2PintaAla.ToString();
    labelPiikin1PintaAla.Text = Piikin1PintaAla.ToString();
    if (spektrinKeskus < data.ajat.Length)
        textBoxMittausAika.Text = data.ajat[spektrinKeskus];
    labelMittausJakso.Text = spektrinKeskus.ToString();
}
}

/// <summary>
/// piirretään timeline kuvaelementtiin
/// </summary>
/// <param name="myosGraafi">true piirtää koko graafin uusiksi, false päivittää vain
spektrin keskuksen sijainnin</param>
private void plotTimeline1(bool myosGraafi)
{
    //jos true, päivitetään koko graafi uusiksi
    if (myosGraafi)
    {
        //nollataan graafi
        chartTimeline1.Series["data"].Points.Clear();
        //i käy datan läpi koko pituudelta
        for (int i = 0; i < data.dataLength; i++)
        {
            //asetetaan datapiste elementtiin
            if (data.timeline1 != null)
                chartTimeline1.Series["data"].Points.AddXY(i, data.timeline1[i]);
        }
        //asetetaan kuvaajan tyyppi
        chartTimeline1.Series["data"].ChartType = SeriesChartType.FastLine;
        chartTimeline1.Series["data"].Color = Color.Black;
    }

    //nollataan spektrin keskus
    chartTimeline1.Series["raja"].Points.Clear();

    //lisätään spektrin keskus kuvaajaan
    chartTimeline1.Series["raja"].Points.AddXY(spektrinKeskus, 0);
    chartTimeline1.Series["raja"].Points.AddXY(spektrinKeskus, data.timeline1MaxArvo
* 1.2);

```

```

//päivitetään kuvaajan tyyppi spektrin keskukselle
chartTimeline1.Series["raja"].ChartType = SeriesChartType.FastLine;
chartTimeline1.Series["raja"].Color = Color.Red;
chartTimeline1.ChartAreas["ChartArea1"].AxisX.Minimum = 0;

// jos spektrin keskus on muualla kuin koko spektrissä
if (data.spektri != null && spektrinKeskus != -1)
{
    int Piikin1PintaAla = 0;
    //i käy alarajalta ylärajalle
    for (int i = int.Parse(textBoxAlaraja.Text); i <
int.Parse(textBoxYlaraja.Text); i++)
    {
        //summataan piikin pinta-alaa
        Piikin1PintaAla += data.spektri[i];
    }
    //päivitetään näkyvät tekstit
    labelPiikin1PintaAla.Text = Piikin1PintaAla.ToString();
    if (spektrinKeskus < data.ajat.Length)
        textBoxMittausAika.Text = data.ajat[spektrinKeskus];
    labelMittausJakso.Text = spektrinKeskus.ToString();
}
}

/// <summary>
/// piirretään timeline kuvaelementtiin
/// </summary>
/// <param name="myosGraafi">true piirtää koko graafin uusiksi, false päivittää vain
spektrin keskuksen sijainnin</param>
private void plotTimeline2(bool myosGraafi)
{
    //jos true, päivitetään koko graafi uusiksi
    if (myosGraafi)
    {
        //nollataan graafi
        chartTimeline2.Series["data"].Points.Clear();
        //i käy datan läpi koko pituudelta
        for (int i = 0; i < data.dataLength; i++)
        {
            //asetetaan datapiste elementtiin
            if (data.timeline2 != null)
                chartTimeline2.Series["data"].Points.AddXY(i, data.timeline2[i]);
        }
        //asetetaan kuvaajan tyyppi
        chartTimeline2.Series["data"].ChartType = SeriesChartType.FastLine;
        chartTimeline2.Series["data"].Color = Color.Black;
    }
    //nollataan spektrin keskus
    chartTimeline2.Series["raja"].Points.Clear();

    //asetetaan spektrin keskus näkyviin
    chartTimeline2.Series["raja"].Points.AddXY(spektrinKeskus, 0);
    chartTimeline2.Series["raja"].Points.AddXY(spektrinKeskus, data.timeline2MaxArvo
* 1.2);

    //asetetaan kuvaajan tyyppi
    chartTimeline2.Series["raja"].ChartType = SeriesChartType.FastLine;
    chartTimeline2.Series["raja"].Color = Color.Red;

    //asetetaan x-akselille nollakohta
    chartTimeline2.ChartAreas["ChartArea1"].AxisX.Minimum = 0;

```

```

// jos spektrin keskus on muualla kuin koko spektrissä
if (data.spektri != null && spektrinKeskus != -1)
{
    int Piikin2PintaAla = 0;
    //i käy alarajalta ylärajalle
    for (int i = int.Parse(textBoxAlaraja2.Text); i <
int.Parse(textBoxYlaraja2.Text); i++)
    {
        //summataan piikin pinta-ala
        Piikin2PintaAla += data.spektri[i];
    }
    //päivitetään näkyvät tekstit
    labelPiikin2PintaAla.Text = Piikin2PintaAla.ToString();
    if (spektrinKeskus < data.ajat.Length)
        textBoxMittausAika.Text = data.ajat[spektrinKeskus];
    labelMittausJakso.Text = spektrinKeskus.ToString();
}
}

/// <summary>
/// menun file->open painike
/// Luetaan data, alustetaan sliderit ja piirretään ensimmäisen kerran
/// </summary>
private void openToolStripMenuItem_Click(object sender, EventArgs e)
{
    readFile();
    calculator = new DataClaculator(data);
    textBoxYlaraja.Text = "420";
    textBoxAlaraja.Text = "330";
    textBoxYlaraja2.Text = "202";
    textBoxAlaraja2.Text = "142";
    trackBarAikajana.Minimum = 0;
    trackBarAikajana.Maximum = data.dataLength;
    textBoxAikajana.Text = "-1";
    data.setSpektrinKeskus(spektrinKeskus);
    textBoxPuoliintumisaika1.Text = "192,47";
    textBoxMittauksiaKeskiarvossa1.Text = "10";
    textBoxTehokkuuskerroin1.Text = "649251,675359";
    textBoxPuoliintumisaika2.Text = "13,22";
    textBoxMittauksiaKeskiarvossa2.Text = "10";
    textBoxTehokkuuskerroin2.Text = "348833,118119";
    calculator.paasto1Laskettu += calculator_paasto1Laskettu;
    calculator.paasto2Laskettu += calculator_paasto2Laskettu;
}

/// <summary>
/// kun calculator antaa eventin että päästö on laskettu,
/// päivitetään päästön tiedot.
/// </summary>
void calculator_paasto1Laskettu(object sender, resultEventArgs e)
{
    textBoxpaastokBq1.Text = e.paastokBq.ToString();
    textBoxPaastoBqPerSec1.Text = e.paastoPerSecBq.ToString();
    textBoxPaastoCts1.Text = e.paastoCts.ToString();
    chartPaasto1.Series["data"].Points.Clear();
    for (int i = 0; i < e.data.Length; i++)
    {
        chartPaasto1.Series["data"].Points.AddXY(i, e.data[i]);
    }
    chartPaasto1.Series["data"].ChartType = SeriesChartType.FastLine;
    chartPaasto1.Series["data"].Color = Color.Black;
}

```

```

}

/// <summary>
/// kun calculator antaa eventin että päästö on laskettu,
/// päivitetään päästön tiedot
/// </summary>
void calculator_paasto2Laskettu(object sender, resultEventArgs e)
{
    textBoxpaastokBq2.Text = e.paastokBq.ToString();
    textBoxPaastoBqPerSec2.Text = e.paastoPerSecBq.ToString();
    textBoxPaastoCts2.Text = e.paastoCts.ToString();
    chartPaasto2.Series["data"].Points.Clear();
    for (int i = 0; i < e.data.Length; i++)
    {
        chartPaasto2.Series["data"].Points.AddXY(i, e.data[i]);
    }
    chartPaasto2.Series["data"].ChartType = SeriesChartType.FastLine;
    chartPaasto2.Series["data"].Color = Color.Black;
}

/// <summary>
/// kun data-olio antaa eventin että mittaustulosten aikajana on laskettu,
/// päivitetään aikajana.
/// </summary>
void data_timelineLaskettu1(object sender, dataEventArgs e)
{
    plotTimeline1(true);
}

/// <summary>
/// kun data-olio antaa eventin että mittaustulosten aikajana on laskettu,
/// päivitetään aikajana.
/// </summary>
void data_timelineLaskettu2(object sender, dataEventArgs e)
{
    plotTimeline2(true);
}

/// <summary>
/// kun data-olio antaa eventin että spektri on laskettu,
/// päivitetään spektri.
/// </summary>
void data_spektriLaskettu(object sender, dataEventArgs e)
{
    plotSpectrum(true);
}

#region Graafit
/// <summary>
/// Jos käyttäjä muuttaa alarajaa tekstikentästä, päivitetään näkymä
/// </summary>
private void textBoxAlaraja1_TextChanged(object sender, EventArgs e)
{
    int luku;
    //yritetään lukea luku tekstikentästä, jos onnistuu, jatketaan,
    //jos ei onnistu, muutetaan kenttä punaiseksi
    if (int.TryParse(textBoxAlaraja.Text, out luku))
    {
        //jos luku on rajoissa, hyväksytään, jos ei, muutetaan kenttä punaiseksi
        if (luku >= 0 && luku <= trackBarYlaraja.Value)

```

```

    {
        //asetetaan annettu luku ja värjätään kenttä valkoiseksi
        //hyväksymisen merkiksi.
        trackBarAlaraja.Value = luku;
        textBoxAlaraja.BackColor = Color.White;

        //päivitetään spektri timelinen rajat
        plotSpectrum(false);
        data.setTimelineRajat1(luku, trackBarYlaraja.Value);
    }
    else
    {
        textBoxAlaraja.BackColor = Color.Red;
    }
}
else
{
    textBoxAlaraja.BackColor = Color.Red;
}
}

/// <summary>
/// Jos käyttäjä muuttaa ylärajaa tekstikentästä, päivitetään näkymä
/// </summary>
private void textBoxYlaraja1_TextChanged(object sender, EventArgs e)
{
    int luku;
    //yritetään lukea luku tekstikentästä, jos onnistuu, jatketaan,
    //jos ei onnistu, muutetaan kenttä punaiseksi
    if (int.TryParse(textBoxYlaraja.Text, out luku))
    {
        //jos luku on rajoissa, hyväksytään, jos ei, muutetaan kenttä punaiseksi
        if (luku <= 1024 && luku >= trackBarAlaraja.Value)
        {
            //asetetaan annettu luku ja värjätään kenttä valkoiseksi
            //hyväksymisen merkiksi.
            trackBarYlaraja.Value = luku;
            textBoxYlaraja.BackColor = Color.White;

            //päivitetään spektri timelinen rajat
            plotSpectrum(false);
            data.setTimelineRajat1(trackBarAlaraja.Value, luku);
        }
        else
        {
            textBoxYlaraja.BackColor = Color.Red;
        }
    }
    else
    {
        textBoxYlaraja.BackColor = Color.Red;
    }
}

/// <summary>
/// Jos käyttäjä muuttaa alarajaa trackbarista, päivitetään näkymä
/// </summary>
private void trackBarAlaraja1_Scroll(object sender, EventArgs e)
{
    //trackbarin siirtäminen aiheuttaa tekstikentän päivityksen,
    //joka aiheuttaa trackbarin siirtämisen. Jos koko homma aiheutui

```

```

//trackbarin päivittämisestä, katkaistaan kierre.
if (loopLock)
{
    loopLock = false;
    return;
}
//laitetaan kierteen estäjä päälle
loopLock = true;

//jos trackbar vedetään ylärajan yläpuolelle, asetetaan arvo ylärajaksi
if (trackBarYlaraja.Value < trackBarAlaraja.Value)
    trackBarYlaraja.Value = trackBarAlaraja.Value;
//päivitetään tekstikentät
textBoxAlaraja.Text = trackBarAlaraja.Value.ToString();
textBoxYlaraja.Text = trackBarYlaraja.Value.ToString();

//päivitetään spektri ja timelinen rajat
plotSpectrum(false);
data.setTimelineRajat1(trackBarAlaraja.Value, trackBarYlaraja.Value);
}

/// <summary>
/// Jos käyttäjä muuttaa ylärajaa trackbarista, päivitetään näkymä
/// </summary>
private void trackBarYlaraja1_Scroll(object sender, EventArgs e)
{
    //jos muutos johtui toisesta muutoksesta, ei muuteta
    if (loopLock)
    {
        loopLock = false;
        return;
    }
    //laitetaan kierteen estäjä päälle
    loopLock = true;

    //jos trackbar vedetään alarajan alapuolelle, asetetaan arvo ylärajaksi
    if (trackBarAlaraja.Value > trackBarYlaraja.Value)
        trackBarAlaraja.Value = trackBarYlaraja.Value;
    //päivitetään tekstikentät
    textBoxYlaraja.Text = trackBarYlaraja.Value.ToString();
    textBoxAlaraja.Text = trackBarAlaraja.Value.ToString();

    //päivitetään spektri ja timelinen rajat
    plotSpectrum(false);
    data.setTimelineRajat1(trackBarAlaraja.Value, trackBarYlaraja.Value);
}

/// <summary>
/// Jos käyttäjä muuttaa aikajanan kursoria trackbarista, päivitetään näkymä
/// </summary>
private void trackBarAikajana_Scroll(object sender, EventArgs e)
{
    //jos muutos johtui toisesta muutoksesta, ei muuteta
    if (loopLock)
    {
        loopLock = false;
        return;
    }
    //laitetaan kierteen estäjä päälle
    loopLock = true;

    //asetetaan spektrin kuvaamaan valittua kohtaa aikajanalta

```

```

    spektrinKeskus = trackBarAikajana.Value;
    //päivitetään tekstikentän luku
    textBoxAikajana.Text = spektrinKeskus.ToString();

    //päivitetään aikajanat näyttämään kursori oikeassa kohdassa
    data.setSpektrinKeskus(spektrinKeskus);
    plotTimeline1(false);
    plotTimeline2(false);
}

/// <summary>
/// Jos käyttäjä muuttaa aikajanahan kursoria tekstikentästä, päivitetään näkyvä
/// </summary>
private void textBoxAikajana_TextChanged(object sender, EventArgs e)
{
    int luku;
    //yritetään lukea luku tekstikentästä, jos onnistuu, jatketaan,
    //jos ei onnistu, muutetaan kenttä punaiseksi
    if (int.TryParse(textBoxAikajana.Text, out luku))
    {
        //jos luku on rajoissa, hyväksytään, jos ei, muutetaan kenttä punaiseksi
        if (luku >= -1 && luku <= data.dataLength)
        {
            //jos luku on -1, se tarkoittaa koko spektriä
            if (luku < 0)
                trackBarAikajana.Value = 0;
            else
                trackBarAikajana.Value = luku;

            //tallennetaan tieto muuttujaan ja hyväksytään värjäämällä valkeaksi
            spektrinKeskus = luku;
            textBoxAikajana.BackColor = Color.White;

            //päivitetään koko spektri ja kursorit timelineistä
            data.setSpektrinKeskus(spektrinKeskus);
            plotTimeline1(false);
            plotTimeline2(false);
        }
        else
        {
            textBoxAikajana.BackColor = Color.Red;
        }
    }
    else
    {
        textBoxAikajana.BackColor = Color.Red;
    }
}

/// <summary>
/// Jos käyttäjä klikkaa "koko spektri" painiketta, laitetaan
/// koko mittausjakson summaspektri näkyviin.
/// </summary>
private void buttonAikajana_Click(object sender, EventArgs e)
{
    //asetetaan tieto spektrinKeskukseen
    spektrinKeskus = -1;
    //nollataan aikajanahan kursori
    trackBarAikajana.Value = 0;
    //päivitetään tekstikenttä
    textBoxAikajana.Text = "-1";
    //päivitetään aikajanoihin kursorit

```



```

        plotTimeline1(false);
        plotTimeline2(false);
        //päivitetään spektri
        data.setSpektrinKeskus(spektrinKeskus);
    }

    /// <summary>
    /// Jos käyttäjä muuttaa alarajaa trackbarista, päivitetään näkymä
    /// </summary>
    private void trackBarYlaraja2_Scroll(object sender, EventArgs e)
    {
        //jos muutos johtui toisesta muutoksesta, ei muuteta
        if (loopLock)
        {
            loopLock = false;
            return;
        }
        //laitetaan kierteen estäjä päälle
        loopLock = true;

        //jos trackbar vedetään ylärajan yläpuolelle, asetetaan arvo ylärajaksi
        if (trackBarAlaraja2.Value > trackBarYlaraja2.Value)
            trackBarAlaraja2.Value = trackBarYlaraja2.Value;
        textBoxYlaraja2.Text = trackBarYlaraja2.Value.ToString();
        textBoxAlaraja2.Text = trackBarAlaraja2.Value.ToString();

        //päivitetään spektri ja timelinen rajat
        plotSpectrum(false);
        data.setTimelineRajat2(trackBarAlaraja2.Value, trackBarYlaraja2.Value);
    }

    /// <summary>
    /// Jos käyttäjä muuttaa ylärajaa trackbarista, päivitetään näkymä
    /// </summary>
    private void trackBarAlaraja2_Scroll(object sender, EventArgs e)
    {
        //jos muutos johtui toisesta muutoksesta, ei muuteta
        if (loopLock)
        {
            loopLock = false;
            return;
        }
        //laitetaan kierteen estäjä päälle
        loopLock = true;

        //jos trackbar vedetään alarajan alapuolelle, asetetaan arvo ylärajaksi
        if (trackBarYlaraja2.Value < trackBarAlaraja2.Value)
            trackBarYlaraja2.Value = trackBarAlaraja2.Value;
        textBoxAlaraja2.Text = trackBarAlaraja2.Value.ToString();
        textBoxYlaraja2.Text = trackBarYlaraja2.Value.ToString();

        //päivitetään spektri ja timelinen rajat
        plotSpectrum(false);
        data.setTimelineRajat2(trackBarAlaraja2.Value, trackBarYlaraja2.Value);
    }

    /// <summary>
    /// Jos käyttäjä muuttaa ylärajaa tekstikentästä, päivitetään näkymä
    /// </summary>
    private void textBoxYlaraja2_TextChanged(object sender, EventArgs e)
    {
        int luku;

```

```

//yritetään lukea luku tekstikentästä, jos onnistuu, jatketaan,
//jos ei onnistu, muutetaan kenttä punaiseksi
if (int.TryParse(textBoxYlaraja2.Text, out luku))
{
    //jos luku on rajoissa, hyväksytään, jos ei, muutetaan kenttä punaiseksi
    if (luku <= 1024 && luku >= trackBarAlaraja2.Value)
    {
        //asetetaan annettu luku ja värjätään kenttä valkoiseksi
        //hyväksymisen merkiksi.
        trackBarYlaraja2.Value = luku;
        textBoxYlaraja2.BackColor = Color.White;

        //päivitetään spektri timelinen rajat
        plotSpectrum(false);
        data.setTimelineRajat2(trackBarAlaraja2.Value, luku);
    }
    else
    {
        textBoxYlaraja2.BackColor = Color.Red;
    }
}
else
{
    textBoxYlaraja2.BackColor = Color.Red;
}
}

/// <summary>
/// Jos käyttäjä muuttaa alarajaa tekstikentästä, päivitetään näkymä
/// </summary>
private void textBoxAlaraja2_TextChanged(object sender, EventArgs e)
{
    int luku;
    //yritetään lukea luku tekstikentästä, jos onnistuu, jatketaan,
    //jos ei onnistu, muutetaan kenttä punaiseksi
    if (int.TryParse(textBoxAlaraja2.Text, out luku))
    {
        //jos luku on rajoissa, hyväksytään, jos ei, muutetaan kenttä punaiseksi
        if (luku >= 0 && luku <= trackBarYlaraja2.Value)
        {
            //asetetaan annettu luku ja värjätään kenttä valkoiseksi
            //hyväksymisen merkiksi.
            trackBarAlaraja2.Value = luku;
            textBoxAlaraja2.BackColor = Color.White;

            //päivitetään spektri timelinen rajat
            plotSpectrum(false);
            data.setTimelineRajat2(luku, trackBarYlaraja2.Value);
        }
        else
        {
            textBoxAlaraja2.BackColor = Color.Red;
        }
    }
    else
    {
        textBoxAlaraja2.BackColor = Color.Red;
    }
}
}

#endregion

```

```

/// <summary>
/// jos käyttäjä muuttaa puoliintumisaikaa, päivitetään muuttuja ja käsketään
/// data-olion laskea data uudelleen
/// </summary>
private void textBoxPuoliintumisaika1_TextChanged(object sender, EventArgs e)
{
    calculator.puoliintumisaikaH1 = parsiluku(textBoxPuoliintumisaika1);
    data.lahetaData1();
}

/// <summary>
/// jos käyttäjä muuttaa keskiarvon mittausten määrää, päivitetään muuttuja ja
/// käsketään data-olion laskea data uudelleen
/// </summary>
private void textBoxMittauksiaKeskiarvossa1_TextChanged(object sender, EventArgs e)
{
    calculator.mittauksiaKeskiarvossa1 =
(int)parsiluku(textBoxMittauksiaKeskiarvossa1);
    data.lahetaData1();
}

/// <summary>
/// jos käyttäjä muuttaa tehokkuuskerrointa, päivitetään muuttuja ja käsketään
/// data-olion laskea data uudelleen
/// </summary>
private void textBoxTehokkuuskerroin1_TextChanged(object sender, EventArgs e)
{
    calculator.kerroinBqPerCPS1 = parsiluku(textBoxTehokkuuskerroin1);
    data.lahetaData1();
}

/// <summary>
/// jos käyttäjä muuttaa puoliintumisaikaa, päivitetään muuttuja ja käsketään
/// data-olion laskea data uudelleen
/// </summary>
private void textBoxPuoliintumisaika2_TextChanged(object sender, EventArgs e)
{
    calculator.puoliintumisaikaH2 = parsiluku(textBoxPuoliintumisaika2);
    data.lahetaData2();
}

/// <summary>
/// jos käyttäjä muuttaa keskiarvon mittausten määrää, päivitetään muuttuja ja
/// käsketään data-olion laskea data uudelleen
/// </summary>
private void textBoxMittauksiaKeskiarvossa2_TextChanged(object sender, EventArgs e)
{
    calculator.mittauksiaKeskiarvossa2 =
(int)parsiluku(textBoxMittauksiaKeskiarvossa2);
    data.lahetaData2();
}

/// <summary>
/// jos käyttäjä muuttaa tehokkuuskerrointa, päivitetään muuttuja ja käsketään
/// data-olion laskea data uudelleen
/// </summary>
private void textBoxTehokkuuskerroin2_TextChanged(object sender, EventArgs e)
{
    calculator.kerroinBqPerCPS2 = parsiluku(textBoxTehokkuuskerroin2);
    data.lahetaData2();
}

/// <summary>
/// Metodi poimii luvun tekstikentästä ja yrittää tehdä siitä liukulukumuuttujan.

```

```

/// Jos muunnos ei onnistu, kentän taustaväri muutetaan punaiseksi, jos muunnos
/// onnistuu väri palautetaan valkoiseksi
/// </summary>
/// <param name="textBox">tekstikenttä, josta luku yritetään parsia</param>
/// <returns>luku kaksinkertaisen tarkkuuden liukulukuna</returns>
private double parsiluku(TextBox textBox)
{
    double luku;
    if (double.TryParse(textBox.Text, out luku))
    {
        textBox.BackColor = Color.White;
        return luku;
    }
    else
    {
        textBox.BackColor = Color.Red;
        return 0;
    }
}
}
}

```

DataHandler.cs

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;
using System.Threading;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace PaastoLaskuriForm
{
    /// <summary>
    /// Luokka, jonka tehtävänä on käsitellä dataa. Luokka laskee
    /// matriisimuotoisesta spektridatasta halutulla ROI:lla
    /// aikajanan, eli listan mittaustuloksista. Laskeminen tehdään
    /// erillisessä threadissa, jotta käyttöliittymä ei jumiudu
    /// laskun ajaksi.
    /// </summary>
    class DataHandler
    {
        public event EventHandler<dataEventArgs> spektriLaskettu;
        public event EventHandler<dataEventArgs> timelineLaskettu1;
        public event EventHandler<dataEventArgs> timelineLaskettu2;

        private int[,] data;
        private int spektrinValinta = 1;
        public int[] spektri;
        public int[] timeline1;
        public int[] timeline2;
        public string[] ajat;
        public int spektriMaxArvo;
        public int spektriMinArvo;
        public int timeline1MaxArvo;
        public int timeline2MaxArvo;
    }
}

```

```

private int spektrinKeskus = -1;
private int alaraja1 = 0;
private int ylaraja1 = 1023;
private int alaraja2 = 0;
private int ylaraja2 = 1023;
public int dataLength;
private bool spektriLaskuKaynnissa = false;
private bool timelineLaskuKaynnissa1 = false;
private bool timelineLaskuKaynnissa2 = false;
private bool dataLadattu = false;

private BackgroundWorker spektriLaskuri;
private BackgroundWorker timelineLaskuri1;
private BackgroundWorker timelineLaskuri2;

/// <summary>
/// constructori, joka parsii tiedot annetusta string-listasta
/// </summary>
/// <param name="lines">lista datariveistä</param>
public DataHandler(String[] lines)
{
    parseData(lines);
}

/// <summary>
/// asettaa spektrin keskuksen haluttuun paikkaan. -1 tarkoittaa koko spektriä.
/// ja laskee kuvaajan uusiksi
/// </summary>
/// <param name="keskus">Mihin mittaukseen spektrin keskus tulisi asettaa</param>
public void setSpektrinKeskus(int keskus)
{
    //tallennetaan valinta
    spektrinKeskus = keskus;

    // jos dataa ei ole ladattu, ei yritetä laskea sillä mitään
    if (!dataLadattu)
        return;

    //jos spektrin laskeminen ei ole käynnissä, aloitetaan se
    if (!spektriLaskuKaynnissa)
    {
        spektriLaskuKaynnissa = true;
        spektriLaskuri = new BackgroundWorker();
        spektriLaskuri.WorkerReportsProgress = true;
        spektriLaskuri.WorkerSupportsCancellation = true;
        spektriLaskuri.ProgressChanged += spektriLaskuri_ProgressChanged;
        spektriLaskuri.DoWork += spektriLaskuri_DoWork;
        spektriLaskuri.RunWorkerAsync();
    }
    else
    {
        //muuten keskeytetään lasku, jolloin se alkaa automaattisesti alusta
        spektriLaskuri.CancelAsync();
    }
}

/// <summary>
/// asettaa timeline ykköselle rajat
/// ja laskee kuvaajan uusiksi
/// </summary>
/// <param name="AlaRaja">haluttu alaraja</param>

```

```

/// <param name="YlaRaja">haluttu yläraja</param>
public void setTimelineRajat1(int AlaRaja, int YlaRaja)
{
    //tallennetaan asetukset
    alaraja1 = AlaRaja;
    ylaraja1 = YlaRaja;

    //jos dataa ei ole ladattu, ei yritetä laskea sillä mitään
    if (!dataLadattu)
        return;

    //jos lasku ei ole käynnissä, käynnistetään se
    if (!timelineLaskuKaynnissa1)
    {
        timelineLaskuKaynnissa1 = true;
        timelineLaskuri1 = new BackgroundWorker();
        timelineLaskuri1.WorkerReportsProgress = true;
        timelineLaskuri1.WorkerSupportsCancellation = true;
        timelineLaskuri1.ProgressChanged += timelineLaskuri1_ProgressChanged;
        timelineLaskuri1.DoWork += timelineLaskuri1_DoWork;
        timelineLaskuri1.RunWorkerAsync();
    }
    else
    {
        //jos lasku on käynnissä, keskeytetään, jolloin se käynnistyy automaattisesti
        uudelleen
        timelineLaskuri1.CancelAsync();
    }
}

/// <summary>
/// asettaa timeline kakkoselle rajat
/// ja laskee kuvaajan uusiksi
/// </summary>
/// <param name="AlaRaja">haluttu alaraja</param>
/// <param name="YlaRaja">haluttu yläraja</param>
public void setTimelineRajat2(int AlaRaja, int YlaRaja)
{
    //tallennetaan valinnat
    alaraja2 = AlaRaja;
    ylaraja2 = YlaRaja;

    //jos dataa ei ole ladattu, ei yritetä laskea sillä mitään
    if (!dataLadattu)
        return;

    //jos lasku ei ole käynnissä, aloitetaan se
    if (!timelineLaskuKaynnissa2)
    {
        timelineLaskuKaynnissa2 = true;
        timelineLaskuri2 = new BackgroundWorker();
        timelineLaskuri2.WorkerReportsProgress = true;
        timelineLaskuri2.WorkerSupportsCancellation = true;
        timelineLaskuri2.ProgressChanged += timelineLaskuri2_ProgressChanged;
        timelineLaskuri2.DoWork += timelineLaskuri2_DoWork;
        timelineLaskuri2.RunWorkerAsync();
    }
    else
    {
        //jos lasku on käynnissä, keskeytetään, jolloin se alkaa automaattisesti
        uudelleen
        timelineLaskuri2.CancelAsync();
    }
}

```

```

    }
}

/// <summary>
/// aiheuttaa dataCalculatorissa laskun päivittymisen.
/// </summary>
public void lahetaData1()
{
    if (timelineLaskettu1 != null)
        timelineLaskettu1(this, new dataEventArgs(timeline1));
}

/// <summary>
/// aiheuttaa dataCalculatorissa laskun päivittymisen.
/// </summary>
public void lahetaData2()
{
    if (timelineLaskettu2 != null)
        timelineLaskettu2(this, new dataEventArgs(timeline2));
}

/// <summary>
/// asynkroninen metodi, jossa lasketaan omassa threadissa timeline
/// </summary>
void timelineLaskuri1_DoWork(object sender, DoWorkEventArgs e)
{
    //otetaan lähettäjä talteen
    BackgroundWorker worker = sender as BackgroundWorker;
    //alustetaan muuttujat
    int timelineMax = 0;
    int[] timeline = new int[dataLength];

    //i käy läpi datan koko pituuden
    for (int i = 0; i < dataLength; i++)
    {
        //tarkistetaan välillä, onko threadin käsketty pysähtyä
        if (worker.CancellationPending == false)
        {
            timeline[i] = 0;
            //j käy yhden mittauksen datan alarajalta ylärajalle
            for (int j = alaraja1; j <= ylaraja1; j++)
            {
                //summataan timelineen
                timeline[i] += data[j, i];
            }

            //jos oli tähän mennessä suurin arvo, otetaan talteen jotta saadaan
            lopuksi suurin arvo selville
            if (timeline[i] > timelineMax)
                timelineMax = timeline[i];
        }
        else // jos oli pydetty keskeyttämään, lähetetään ilmoitus keskeytyksestä ja
        lopetetaan
        {
            worker.ReportProgress(0);
            e.Cancel = true;
            break;
        }
    }
    //ilmoitetaan valmistumisesta
    worker.ReportProgress(1, new TimelineData(timeline, timelineMax));
    //poistetaan worker
}

```

```

        worker.Dispose();
    }

    /// <summary>
    /// Metodi, jota kutsutaan kun worker raportoi tilamuutoksista
    /// </summary>
    void timelineLaskuri1_ProgressChanged(object sender, ProgressChangedEventArgs e)
    {
        //jos prosentti == 1 homma on saatu loppuun ja heitetään eventti
        if (e.ProgressPercentage == 1)
        {
            //puretaan ilmoituksen mukana tullut data
            TimelineData data = (TimelineData)e.UserState;
            timeline1 = data.timeline;
            timeline1MaxArvo = data.timelineMax;
            //lähetetään eteenpäin tieto, että data on laskettu
            if (timelineLaskettu1 != null)
                timelineLaskettu1(this, new dataEventArgs(timeline1));
            BackgroundWorker worker = sender as BackgroundWorker;
            worker.Dispose();
            timelineLaskuKayynnissa1 = false;
        }
        else // jos prosentti == 0 laskuri on keskeytetty, luodaan uusi
        {
            timelineLaskuri1 = new BackgroundWorker();
            timelineLaskuri1.WorkerReportsProgress = true;
            timelineLaskuri1.WorkerSupportsCancellation = true;
            timelineLaskuri1.ProgressChanged += timelineLaskuri1_ProgressChanged;
            timelineLaskuri1.DoWork += timelineLaskuri1_DoWork;
            timelineLaskuri1.RunWorkerAsync();
        }
    }

    /// <summary>
    /// asynkroninen metodi, jossa lasketaan omassa threadissa timeline
    /// </summary>
    void timelineLaskuri2_DoWork(object sender, DoWorkEventArgs e)
    {
        //otetaan lähettäjä talteen
        BackgroundWorker worker = sender as BackgroundWorker;
        //alustetaan muuttujat
        int timelineMax = 0;
        int[] timeline = new int[dataLength];

        //i käy läpi datan koko pituuden
        for (int i = 0; i < dataLength; i++)
        {
            //välillä tarkistetaan onko threadin käsketty keskeytyä
            if (worker.CancellationPending == false)
            {
                timeline[i] = 0;
                //j käy läpi yhden mittausjakson alarajalta ylärajalle
                for (int j = alaraja2; j <= ylaraja2; j++)
                {
                    //summataan arvo talteen
                    timeline[i] += data[j, i];
                }

                float yla = (data[ylaraja2 + 1, i] + data[ylaraja2 + 2, i] +
data[ylaraja2 + 3, i]) / 3;
                float ala = (data[alaraja2 - 1, i] + data[alaraja2 - 2, i] +
data[alaraja2 - 3, i]) / 3;
            }
        }
    }

```



```

        timeline[i] -= (int)(((yla + ala) / 2)*(ylaraja2-alaraja2));

        //jos oli tähän mennessä suurin arvo, otetaan talteen jotta saadaan
lopuksi suurin arvo selville
        if (timeline[i] > timelineMax)
            timelineMax = timeline[i];
    }
    else // jos käskettiin keskeyttää, ilmoitetaan keskeytyksestä ja katkaistaan
    {
        worker.ReportProgress(0);
        e.Cancel = true;
        break;
    }
}
// ilmoitetaan valmistumisesta
worker.ReportProgress(1, new TimelineData(timeline, timelineMax));
//poistetaan worker
worker.Dispose();
}

/// <summary>
/// Metodi, jota kutsutaan kun worker raportoii tilamuutoksista
/// </summary>
void timelineLaskuri2_ProgressChanged(object sender, ProgressChangedEventArgs e)
{
    //jos prosentti == 1 homma on saatu loppuun ja heitetään eventti
    if (e.ProgressPercentage == 1)
    {
        //puretaan ilmoituksen mukana tullut data
        TimelineData data = (TimelineData)e.UserState;
        timeline2 = data.timeline;
        timeline2MaxArvo = data.timelineMax;
        //ilmoitetaan laskun valmistumisesta eteenpäin
        if (timelineLaskettu2 != null)
            timelineLaskettu2(this, new dataEventArgs(timeline2));
        BackgroundWorker worker = sender as BackgroundWorker;
        worker.Dispose();
        timelineLaskuKaynnissa2 = false;
    }
    else // jos prosentti == 0 laskuri on keskeytetty, luodaan uusi
    {
        timelineLaskuri2 = new BackgroundWorker();
        timelineLaskuri2.WorkerReportsProgress = true;
        timelineLaskuri2.WorkerSupportsCancellation = true;
        timelineLaskuri2.ProgressChanged += timelineLaskuri2_ProgressChanged;
        timelineLaskuri2.DoWork += timelineLaskuri2_DoWork;
        timelineLaskuri2.RunWorkerAsync();
    }
}

/// <summary>
/// asynkroninen metodi, jossa lasketaan omassa threadissa spektri
/// </summary>
void spektriLaskuri_DoWork(object sender, DoWorkEventArgs e)
{
    //poimitaan lähettäjä talteen
    BackgroundWorker worker = sender as BackgroundWorker;

    //alustetaan muuttujia
    int spektriMax = 0;
    int spektriMin = int.MaxValue;

```

```

int[] Spektri = new int[1024];

//määritetään spektrin alku ja loppu
int alku = 0;
int loppu = dataLength;
//jos spektrinkeskus == -1, alue on koko spektri
if (spektrinKeskus != -1)
{
    //jos spektrinkeskuselle on annettu arvo, asetetaan spektri näyttämään vain
    alku = spektrinKeskus;
    loppu = spektrinKeskus + 1;
    if (alku < 0)
        alku = 0;
    if (loppu > dataLength)
        loppu = dataLength;
}

//i käy läpi kaikki kanavat
for (int i = 0; i < 1024; i++)
{
    //tarkistetaan onko keskeytyspyyntöä
    if (worker.CancellationPending == false)
        //j käy läpi mittausjaksot, jotka on määritetty otettavaksi mukaan
        for (int j = alku; j < loppu; j++)
        {
            //summataan arvo talteen
            Spektri[i] += data[i, j];
        }
    else // jos halutaan keskeytys, ilmoitetaan keskeytyksestä ja lopetetaan
    {
        worker.ReportProgress(0);
        e.Cancel = true;
        break;
    }

    //seurataan spektrin maksimi ja minimiarvoa
    if (Spektri[i] > spektriMax)
        spektriMax = Spektri[i];
    if (Spektri[i] < spektriMin)
        spektriMin = Spektri[i];
}

//lähetetään tiedot
worker.ReportProgress(1, new spektridata(Spektri, spektriMax, spektriMin));
}

/// <summary>
/// Metodi, jota kutsutaan kun worker raportoi tilamuutoksista
/// </summary>
void spektriLaskuri_ProgressChanged(object sender, ProgressChangedEventArgs e)
{
    //jos prosentti == 1 homma on saatu loppuun ja heitetään eventti
    if (e.ProgressPercentage == 1)
    {
        //puretaan ilmoituksen mukana tullut data
        spektridata data = (spektridata)e.UserState;
        spektri = data.spektri;
        spektriMaxArvo = data.spektriMax;
        spektriMinArvo = data.spektriMin;
        //ilmoitetaan laskun valmistumisesta eteenpäin
    }
}

```

```

        if (spektriLaskettu != null)
            spektriLaskettu(this, new dataEventArgs(spektri));
        BackgroundWorker worker = sender as BackgroundWorker;
        worker.Dispose();
        spektriLaskuKaynnissa = false;
    }
    else //jos lasku keskeytettiin, aloitetaan se uudelleen
    {
        spektriLaskuKaynnissa = true;
        spektriLaskuri = new BackgroundWorker();
        spektriLaskuri.WorkerReportsProgress = true;
        spektriLaskuri.WorkerSupportsCancellation = true;
        spektriLaskuri.ProgressChanged += spektriLaskuri_ProgressChanged;
        spektriLaskuri.DoWork += spektriLaskuri_DoWork;
        spektriLaskuri.RunWorkerAsync();
    }
}

/// <summary>
/// Parsitaan luetuista riveistä yksittäiset mittaustiedot dataan
/// </summary>
private void parseData(String[] lines)
{
    dataLength = 0;

    //lasketaan pilkkujen määrä datassa
    foreach (char c in lines[3])
        if (c == ',') dataLength++;
    dataLength--;

    //alustetaan taulukko datalle
    data = new int[1024, dataLength];

    //alustetaan taulukko mittausajoille
    ajat = new string[dataLength];

    //ensin parsitaan mittauksien ajankohdat
    int iter = 10; //aloitetaan vasta "channel"-sanon jälkeen
    for (int i = 0; i < dataLength; i++)
    {
        //määritetään pituus iteraattorista seuraavaan pilkkuun
        int length = lines[0].IndexOf(',', iter) - iter;
        //tallennetaan pätkä
        ajat[i] = lines[0].Substring(iter, length);
        //siirretään iteraattoria seuraavan pilkun yli
        iter += length + 1;
    }

    //varsinaisen datan erottelu
    for (int i = 0; i < 1024; i++)
    {
        //aloitetaan datan parsiminen vasta kanavanumeron jälkeen
        iter = 2;
        if (i > 9) // lines[i+1] kanavanumero on kaksinumeroinen + pilkku
            iter++;
        if (i > 99)
            iter++;
        if (i > 999)
            iter++;

        //käydään data läpi
        for (int j = 0; j < dataLength; j++)

```

```

        {
            //määritetään seuraavan numeron pituus
            int length = lines[i + 1].IndexOf(',', iter) - iter;
            //luetaan luku merkkijonoon
            string luku = lines[i + 1].Substring(iter, length);
            //parsitaan merkkijonosta luku ja tallennetaan se
            data[i, j] = int.Parse(luku);
            //siirretään iteraattoria
            iter += length + 1;
        }
    }

    //tallennetaan tieto, että data on luettu
    dataLadattu = true;
}

/// <summary>
/// luokka, johon on tallennettu int-taulukollinen dataa
/// </summary>
public class dataEventArgs
{
    public int[] data;

    public dataEventArgs(int[] Data)
    {
        data = Data;
    }
}

/// <summary>
/// luokka, johon on tallennettu spektri ja sen minimi ja maksimi
/// </summary>
public class spektridata
{
    public int[] spektri;
    public int spektriMax;
    public int spektriMin;

    public spektridata(int[] Spektri, int SpektriMax, int SpektriMin)
    {
        spektri = Spektri;
        spektriMax = SpektriMax;
        spektriMin = SpektriMin;
    }
}

/// <summary>
/// luokka, johon on tallennettu timeline data ja sen maksimi
/// </summary>
public class TimelineData
{
    public int[] timeline;
    public int timelineMax;

    public TimelineData(int[] Timeline, int TimelineMax)
    {
        timeline = Timeline;
        timelineMax = TimelineMax;
    }
}
}

```

DataCalculator.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace PaastoLaskuriForm
{
    /// <summary>
    /// Luokka jonka tehtävänä on laskea varsinainen päästö timelinestä.
    /// </summary>
    class DataClaculator
    {
        public event EventHandler<resultEventArgs> paasto1Laskettu;
        public event EventHandler<resultEventArgs> paasto2Laskettu;

        public int mittauksiaKeskiarvossa1;
        public double kerroinBqPerCPS1;
        public double puoliintumisaikaH1;

        public int mittauksiaKeskiarvossa2;
        public double kerroinBqPerCPS2;
        public double puoliintumisaikaH2;

        private readonly double mittauksenPituus = 100; //mittauksen pituus sekunteina
        private readonly double mittausValiAika = 11; //mittauksien välinen kuollut aika
        sekunteina

        private DataHandler data;

        /// <summary>
        /// Constructori, alustaa olion
        /// </summary>
        /// <param name="Data">Datahandler, jossa käytettävä data on tallennettu</param>
        public DataClaculator(DataHandler Data)
        {
            data = Data;
            //asetetaan kuuntelijat datahandlerin eventeille
            data.timelineLaskettu1 += data_timelineLaskettu1;
            data.timelineLaskettu2 += data_timelineLaskettu2;
        }

        /// <summary>
        /// Kun DataHandler on laskenut timelinen, lasketaan päästöt ja lähetetään tieto
        eteenpäin
        /// </summary>
        /// <param name="sender">ei käytetä</param>
        /// <param name="e">dataEventArgs, joka sisältää timelinen</param>
        void data_timelineLaskettu1(object sender, dataEventArgs e)
        {
            if (paasto1Laskettu != null)
                paasto1Laskettu(this, laskedata(e, mittauksiaKeskiarvossa1,
                puoliintumisaikaH1, kerroinBqPerCPS1, false));
        }

        /// <summary>
        /// Kun DataHandler on laskenut timelinen, lasketaan päästöt ja lähetetään tieto
        eteenpäin
        /// </summary>
    }
}
```

```

/// <param name="sender">ei käytetä</param>
/// <param name="e">dataEventArgs, joka sisältää timelinen</param>
void data_timelineLaskettu2(object sender, dataEventArgs e)
{
    if (paasto2Laskettu != null)
        paasto2Laskettu(this, laskedata(e, mittauksiaKeskiarvossa2,
puoliintumisaikaH2, kerroinBqPerCPS2, true));
}

/// <summary>
/// metodi, joka hoitaa varsinaisen päästölaskennan
/// </summary>
/// <param name="e">eventArgs-olio, johon on tallennettu mittaustulokset</param>
/// <param name="mittauksiaKeskiarvossa">mittauksien määrä
keskiarvoistuksessa</param>
/// <param name="puoliintumisaikaH">puoliintumisaika tunteina</param>
/// <param name="kerroinBqPerCPS">kalibraatiokerroin</param>
/// <param name="kakkonen">tehdäänkö kakkos-timelinen vähennys</param>
/// <returns></returns>
private resultEventArgs laskedata(dataEventArgs e, int mittauksiaKeskiarvossa, double
puoliintumisaikaH, double kerroinBqPerCPS, bool kakkonen)
{
    //jos jompikumpi näistä syöttöarvoista on nolla, palautetaan pelkkää nollaa.
    if (mittauksiaKeskiarvossa == 0 || puolintumisaikaH == 0)
        return new resultEventArgs(new double[] { 0 }, 0, 0, 0);

    //alustetaan muuttujat
    int i;
    double CPSkeskiarvo;
    double edellinenkeskiarvo = 0;
    double totalSumma = 0;
    double paasto = 0;
    double[] paastoData = new double[e.data.Length / mittauksiaKeskiarvossa - 1];

    //i käy läpi kaikki keskiarvoistukset
    for (i = 0; i < data.dataLength / mittauksiaKeskiarvossa; i++)
    {
        CPSkeskiarvo = 0;
        //j käy läpi kaikki yksittäiseen keskiarvoistukseen kuuluvat mittausjaksot
        for (int j = i * mittauksiaKeskiarvossa; j < (i + 1) *
mittauksiaKeskiarvossa; j++)
        {
            //summataan keskiarvoistukseen kuuluvat countsit
            CPSkeskiarvo += e.data[j];
        }
        //jaetaan summa koko keskiarvoistuksen mittausajalla, jolloin saadaan CPS-
keskiarvo
        CPSkeskiarvo = CPSkeskiarvo / (mittauksiaKeskiarvossa * mittauksenPituus);
        //ensimmäisellä (nollannella) keskiarvoistuksella ei voida vähentää
edellisestä
        if (i > 0)
        {
            //lasketaan puolintumiskerroin
            double aika = mittauksiaKeskiarvossa * (mittauksenPituus +
mittausValiAika);
            double kerroin = Math.Pow(2, ((-aika) / (puoliintumisaikaH * 3600)));
            //lasketaan päästö kahden peräkkäisen keskiarvoistuksen välillä
            paasto = CPSkeskiarvo - (edellinenkeskiarvo * kerroin);
            //tallennetaan päästö
            paastoData[i - 1] = paasto;
        }
        //siirretään nykyinen keskiarvo muistiin edelliseksi
    }
}

```

```

        edellinenkeskiarvo = CPSkeskiarvo;
        //summataan kaikki päästöt kokonaispäästökksi
        totalSumma += paasto;
    }
    //lasketaan päästö eri formaatteihin
    double paastoBq = totalSumma * kerroinBqPerCPS;
    double paastoBqPerS = paastoBq / (data.dataLength * (mittauksenPituus +
mittausValiAika));
    //palautetaan tiedot
    return new resultEventArgs(paastoData, totalSumma, paastoBq / 1000,
paastoBqPerS);
    }
}

/// <summary>
/// Event-args-luokka jolla toimitetaan päästötiedot eteenpäin
/// </summary>
class resultEventArgs
{
    public double[] data;
    public double paastoCts;
    public double paastokBq;
    public double paastoPerSecBq;

    public resultEventArgs(double[] Data, double PaastoCts, double PaastokBq, double
PaastoPerSecBq)
    {
        data = Data;
        paastoCts = PaastoCts;
        paastokBq = PaastokBq;
        paastoPerSecBq = PaastoPerSecBq;
    }
}
}

```