

Eemil Elomaa

KONEOPPIMISEN SOVELLUKSET LÄM- PÖVOIMALAITOSTEN KÄYTÖNAIKAI- Sessa OPTIMOINNISSA

Kandidaatintyö
Tekniikan ja luonnontieteiden tiedekunta
Tarkastaja: Aleksi Sivonen
Toukokuu 2025

TIIVISTELMÄ

Eemil Elomaa: Koneoppimisen sovellukset lämpövoimalaitosten käytönaikaisessa optimoinnissa

Machine learning applications in online optimization of thermal power plants

Kandidaatintyö

Tampereen yliopisto

Tekniikan ja luonnontieteiden kandidaattiohjelma, ympäristö- ja energiatekniikan opintosuunta
Toukokuu 2025

Tässä kandidaatintyössä tutkittiin koneoppimisen hyödyntämistä lämpövoimalaitosten käytönaikaisessa optimoinnissa. Työn tavoitteena oli selvittää, mitä optimointikohteita lämpövoimalaitoksissa on, millaisia koneoppimisalgoritmeja on olemassa ja miten koneoppimista voidaan hyödyntää lämpövoimalaitosten käytönaikaisessa optimoinnissa. Tutkimusmenetelmänä työssä käytettiin kirjallisuuskatsausta aiheen kirjallisuuteen ja tutkimuksiin perustuen.

Koneoppiminen voitiin jaotella ohjattuun, ohjaamattomaan ja vahvistettuun oppimiseen. Ohjatussa oppimisessa oppiminen perustuu merkittyyntä dataan eli syötteille on tiedossa tietyt tunnistet. Yleisiä ohjatun oppimisen algoritmeja ovat esimerkiksi päätöspuut, satunnaismetsä, k-lähin naapuri, tukivektorikone ja Naiivi Bayes. Ohjaamattomassa oppimisessa data on taas merkittömätöntä eli syötteille ei ole tiettyjä tunnisteita. Ohjaamattoman oppimisen yleisiä algoritmeja ovat esimerkiksi k-keskiarvot ja DBSCAN. Vahvistusoppimisen oppiminen taas perustuu ympäristön ja agentin, eli oppijan, vuorovaikutukseen. Yksi yleinen vahvistusoppimisen algoritmi on Q-oppiminen.

Keskeisinä lämpövoimalaitosten optimointikohteina todettiin prosessien tehokkuus sekä ennakoiva kunnossapito ja käyttövarmuus. Prosessien tehokkuuden kannalta koneoppimista voitiin hyödyntää muun muassa palamisprosessien mallintamiseen ja ennustamiseen. Koneoppimista hyödyntävillä malleilla voitiin havaita esimerkiksi voimalaitoskattilan lämpötehokkuuden 1 %:n nousu sekä NO_x-päästöjen pitoisuuden 30,4 %:n lasku laskematta lämpötehokkuutta. Ennakoi- van kunnossapidon ja käyttövarmuuden kannalta voitiin koneoppimisen olevan hyödyllinen prosessien tilojen luokittelussa. Tässä tapauksessa esimerkiksi syöttövesipumpun vikatila on voitu ennustaa ennen varsinaista häiriötä koulutetun mallin avulla.

Koneoppimisen hyödyntäminen ja soveltaminen mahdollistavat siis lämpövoimalaitosten tehokkaamman ja turvallisemman laitoskokonaisuuden ylläpitämisen. Tehdyn kirjallisuuskatsauksen perusteella voidaan todeta, että koneoppiminen on hyvin lupaava apukeino lämpövoimalaitosten käytönaikaisessa optimoinnissa, mutta se vaatii usein hyvin suuren määrän dataa toimiakseen halutulla tavalla.

Avainsanat: Koneoppiminen, lämpövoimalaitos, käytönaikainen optimointi

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin Originality Check –ohjelmalla.

Opinnäytteessäni käytetyt tekoälytyökalut ja niiden käyttötarkoitukset on kuvailtu alla:

ChatGPT 4o:

Käytetty työn rakenteen hahmottelussa ja kielen tarkastamisessa

Scopus AI

Käytetty lähteiden etsimisen tukena

Olen tietoinen siitä, että olen täysin vastuussa koko opinnäytteeni sisällöstä, mukaan lukien tekoälyllä tuotetut osat, ja hyväksyn vastuun mahdollisista eettisten ohjeiden rikkomuksista.

SISÄLLYSLUETTELO

1. JOHDANTO	1
2. LÄMPÖVOIMALAITOKSET JA NIIDEN OPTIMOINTIKOHTTEET	3
2.1 Lämpövoimalaitosten toimintaperiaate	3
2.2 Optimointikohteet käytönaikaisessa optimoinnissa	7
3. KONEOPPIMINEN	10
3.1 Ohjattu oppiminen	10
3.2 Ohjaamaton oppiminen	13
3.3 Vahvistusoppiminen	15
3.4 Neuroverkot ja syväoppiminen	16
3.5 Koneoppimisen osa-alueiden yleiset käyttökohteet	18
4. KONEOPPIMINEN LÄMPÖVOIMALAITOSTEN KÄYTÖNAIKAISISSA OPTIMOINTIKOHTTEISSA	21
4.1 Prosessien tehokkuus	21
4.2 Ennakoiva kunnossapito ja käyttövarmuus	24
5. JOHTOPÄÄTÖKSET	27
LÄHDELUETTELO	28

1. JOHDANTO

Energian riittävyys ja päästöjen hallinta ovat olleet pitkään paljon puhuttavia aiheita ja niihin on pyritty löytämään mahdollisia ratkaisuja. Maailman väestön jatkaessa kasvuaan ja yhä enemmän energiaa tarvitsevien laitteiden ja koneiden määrän kasvaminen vaativat energiantuotannolta jatkuvaa kehittämistä. Energiantuotanto voidaan jakaa sähköntuotantoon, lämmöntuotantoon sekä niiden yhteistuotantoon. Maailman sähköntuotannosta noin 70 % tuotettiin lämpövoimalla vuonna 2023, josta fossiilisten polttoaineiden, kuten kivihillen ja maakaasun, osuus oli noin 61 % [1]. Lämpövoimalaitoksilla eli voimalaitoksilla, joiden tarkoituksena on tuottaa lämpöenergiaa ja muuntaa se sähköenergiaksi, on siis yhä suuri rooli energiantuotannossa, joten niiden optimointitarve on merkittävä.

Optimoinnilla tarkoitetaan yleisesti jonkin tavoitefunktion minimointia tai maksimointia tietyillä kriteereillä [2–4]. Optimointi pohjautuu algoritmeihin, joilla pyritään ratkaisemaan tietty optimointiongelma. Optimointiongelmat voivat olla hyvin monimutkaisia ja vaikeasti ratkaistavissa yksinkertaisilla algoritmeilla. Tähän yhtenä ratkaisuna esitetäänkin koneoppimista hyödyntävä optimointi, jossa algoritmien toiminta pohjautuu koneoppimiseen. Koneoppiminen on tekoälyn osa-alue, jolla tarkoitetaan koneen itsenäistä oppimista ilman, että sitä on siihen erikseen ohjelmoitu. [5] Koneoppiminen voidaan jaotella ohjattuun, ohjaamattomaan, puoliohjattuun ja vahvistettuun oppimiseen [5,6].

Tämän kandidaatintyön tavoitteena on tutkia koneoppimista osana käytönaikaista optimointia ja lämpövoimalaitoksia. Käytönaikaisella voimalaitoksen optimoinnilla pyritään optimoida voimalaitoksen reaaliaikaista käyttöä esimerkiksi prosessien tehokkuuden [2,7], päästöjen hallinnan [8] tai ennakoivan kunnossapidon [9,10] näkökulmasta. Mahdollisuudet koneoppimista hyödyntävällä optimoinnilla ovat siis hyvin merkittävät ja monipuoliset.

Työssä pyritään vastaamaan seuraaviin kysymyksiin: mitä optimointikohteita lämpövoimalaitoksissa on, millaisia koneoppimisalgoritmeja on olemassa ja miten koneoppimista voidaan hyödyntää lämpövoimalaitosten käytönaikaisessa optimoinnissa. Tutkimusmenetelmänä työssä käytetään kirjallisuuskatsausta aiheeseen soveltuvan kirjallisuuden pohjalta.

Lämpövoimalaitosten toimintaperiaatetta ja optimointikohteita esitellään luvussa 2 ja koneoppimisen teoriaa ja algoritmeja luvussa 3. Luvussa 4 tutkitaan tarkemmin koneoppimisen sovelluksia lämpövoimalaitosten käytönaikaisessa optimoinnissa tehtyihin tutkimuksiin perustuen. Lopuksi laaditaan johtopäätökset ja pyritään vastaamaan tutkimuskysymyksiin mahdollisimman kattavasti.

2. LÄMPÖVOIMALAITOKSET JA NIIDEN OPTIMOINTIKOHTTEET

Lämpövoimalaitokset ovat maailmanlaajuisesti keskeisiä energian tuotannon lähteitä ja niiden optimointi on merkittävä osa kestävästä ja tehokasta energiantuotantoa. Lämpövoimalaitosten tehtävänä on tuottaa sähköä ja lämpöä muuttamalla energian muotoa esimerkiksi polttoaineen kemiallisesta energiasta saatavaa lämpöenergiaa sähköenergiaksi termodynaamisten prosessien kautta. Tässä luvussa tarkastellaan ensin lämpövoimalaitosten toimintaperiaatetta ja termodynamiikkaa, johon toiminta perustuu. Tämän jälkeen tutkitaan tarkemmin lämpövoimalaitosten käytönaikaisia optimointikohteita ja ongelmia.

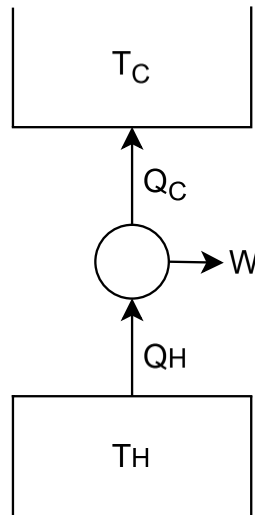
2.1 Lämpövoimalaitosten toimintaperiaate

Lämpövoimalaitosten toimintaperiaate pohjautuu sykleihin, joissa energia muuttaa muotoaan lämpöenergiasta sähköenergiaksi. Syklien perusperiaate pohjautuu termodynamiikkaan, joka pitää sisällään termodynamiikan pääsäännöt. Termodynamiikan ensimmäinen pääsääntö määrittää energian säilymisen eli systeemiin tuotu lämpö on verrannollinen systeemissä tehtyyn työmäärään. Tämän mukaan energiaa ei katoa, vaan se muuttaa muotoaan. [11] Tämä voidaan esittää suljetulle systeemille yhtälön 1 mukaisesti

$$\Delta U = Q + W, \quad (1)$$

jossa ΔU on systeemin sisäenergian muutos, Q siirtynyt lämpö ja W tehty työ.

Toinen termodynamiikan pääsääntö määrittää lämmönsiirtymisen suunnan. Tämä asettaa ehdon, että lämpö voi siirtyä itsestään korkeammasta lämpötilasta matalampaan, mutta ei toisinpäin. Toisen pääsäännön perusteella voidaan todeta, että systeemissä tuotettu työ ei voi olla yhtä suuri kuin systeemiin tuotu lämpö, joten systeemissä on aina hyötösuhde näiden termien välillä. [11] Termodynamiikan pääsääntöjen perusperiaatetta havainnollistetaan kuvassa 1.

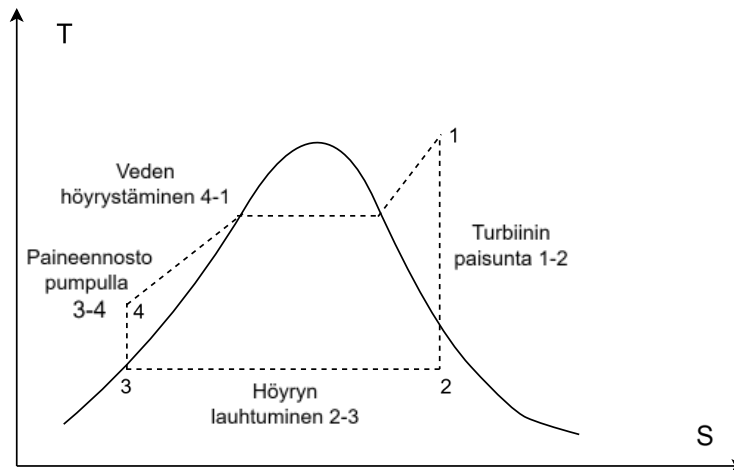


Kuva 1. Havainnollistus yksinkertaisesta lämpökoneesta. Perustuu lähteeseen [12].

Yksinkertaisen lämpökoneen tapauksessa lämpö siirtyy korkeammasta lämpötilasta T_H matalampaan lämpötilaan T_C siten, että osa lämmöstä Q_H muuntuu työksi W ja osa siirtyy lämpönä Q_C matalamman lämpötilan ympäristöön. Energiaa ei siis häviä, vaan se muuttuu osin muotoaan termodynamiikan ensimmäisen pääsäännön mukaisesti. Kuvan 1 mukaisessa tilanteessa teoreettinen maksimihyötysuhde, eli Carnot-hyötysuhde, voidaan laskea jakamalla tehty työ W tuotetulla lämmöllä Q_H . Tehty työ saadaan tuotetun lämmön Q_H ja matalamman lämpötilan ympäristöön siirtyneen lämmön Q_C erotuksesta, sillä energia säilyy ensimmäisen pääsäännön mukaisesti. [11] Carnot-hyötysuhteelle saadaan yhtälön 2 mukainen muoto

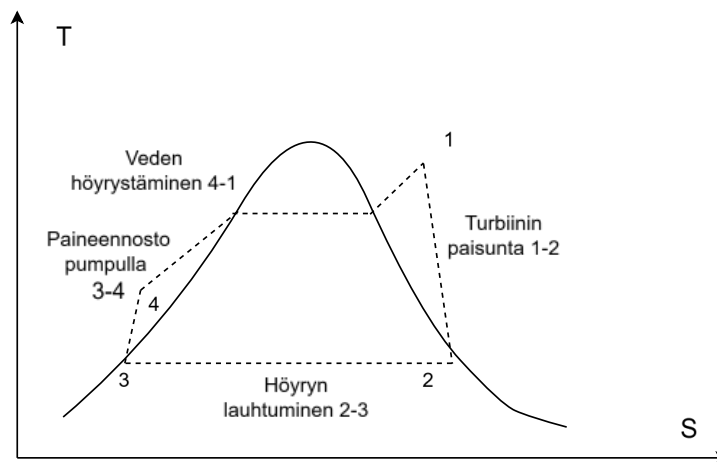
$$\eta = \frac{Q_H - Q_C}{Q_H} = 1 - \frac{Q_C}{Q_H}. \quad (2)$$

Lämpövoimalaitoksissa yksi yleisimmin käytetyistä sykleistä on höyryvoimalaitoksissa käytettävä Rankine-sykli, joka perustuu höyryn ja veden kiertoon. Tässä syklissä on yleisesti käytössä höyrygeneraattori, höyryturbiini, lauhtutin sekä syöttövesipumppu. Ideaalisessa Rankine-syklissä turbiinissa tapahtuvaa paisuntaa ja pumppujen paineen nostoa pidetään isentrooppisina, joissa entropia pysyy vakiona. Lisäksi ideaalisessa syklissä oletetaan höyryn tuottamisen ja lauhtumisen tapahtuvan vakio paineessa. [11] Ideaalinen prosessi havainnollistetaan TS-kuvaajana kuvassa 2. TS-kuvaajalla tarkoitetaan kuvaajaa, jossa lämpötila T muuttuu entropian S funktiona.



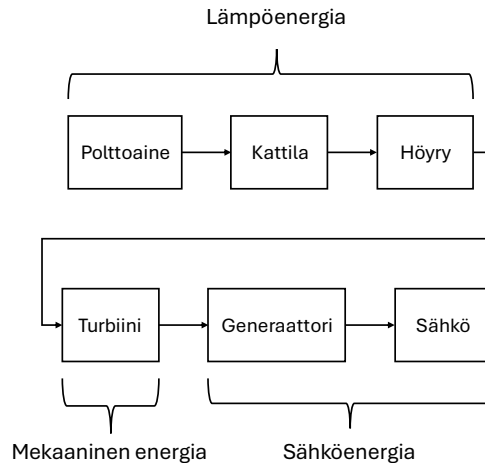
Kuva 2. Ideaalinen Rankine-sykli TS-kuvaajana. Perustuu lähteeseen [11].

Todellisessa Rankine-syklissä turbiinissa tapahtuva paisunta ja pumppujen paineen nosto eivät tapahdu isentrooppisesti, mikä nähdään kuvassa 3. Entropian muuttuminen paisunnassa ja paineen nostossa aiheutuu prosessien palautumattomuuksista, joita ideaalisessa syklissä ei oteta huomioon. Prosessien palautumattomuudet johtuvat muun muassa paine- ja kitkahäviöistä lämpövoimalaitoksen komponenteissa, mitkä vaikuttavat entropian muuttumiseen. [11]



Kuva 3. Todellinen Rankine-sykli TS-kuvaajana. Perustuu lähteeseen [11].

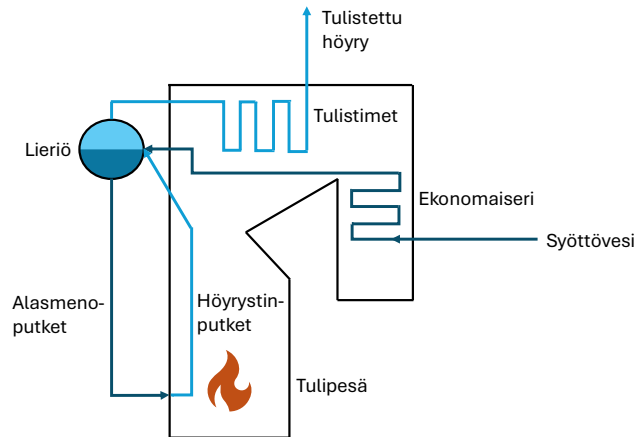
Yksinkertaisuudessaan Rankine-syklissä vesi höyrystetään kattilassa ja johdetaan turbiinille, jossa se paisuu tuottaen mekaanista energiaa turbiinin pyöriessä. Turbiinista poistuva kostea höyry johdetaan lauhttimelle, jossa se lauhtutetaan nestemäiseksi vedeksi. [11] Tämä taas pumpataan takaisin kattilaan halutulle painetasolle syöttöpumpulla ja sykli alkaa alusta [11,13]. Tämän kierron avulla toteutetaan veden tehokas kierto laitoksella sekä mahdollistetaan sähkön ja lämmön tuotanto. Tarkastellaan seuraavaksi yksityiskohtaisemmin energiantuotantoketjua, joka havainnollistetaan kuvassa 4.



Kuva 4. Lämpövoimalaitoksen energiantuotantoketju. Perustuu lähteeseen [9].

Höyrykattilassa syöttövesi höyrytetään hyödyntämällä energianlähteestä saatavaa energiaa esimerkiksi palamisen seurauksena tai fissioreaktiosta ydinvoimaloissa [11]. Maailmanlaajuisesti yleisimpänä energianlähteenä käytetään fossiilisia polttoaineita, kuten kivihiiltä ja maakaasua [1,11]. Fossiilisiin polttoaineisiin on varastoitunut suuri määrä kemiallista energiaa, joka vapautuu palamisreaktiossa hapettimen, kuten ilman sisältämän hapen vaikutuksesta [11]. Esimerkiksi maakaasun bruttolämpöarvo voi olla jopa noin 50 MJ/kg ja kivihiilen noin 25–30 MJ/kg riippuen kosteuspitoisuudesta ja laadusta [14]. Polttoaineen palaminen tapahtuu kattilan tulipesässä, josta vapautuva lämpö siirtyy kattilan lämmönsiirtopinnoille, joihin kuuluvat esimerkiksi ekonomaiserit, tulistimet sekä höyrystin [11,13]. Lämmönsiirtopinnoilta lämpö siirtyy syötetyn veden kuumentamiseen ja höyryttämiseen [11].

Veden kierto kattilassa riippuu kattilatyypistä, mutta yleisesti syöttövesi esilämmitetään ensin lähelle kyllästymislämpötilaa kattilan ekonomaiserissa, josta se johdetaan lieriölle. Lieriön tarkoituksena on erottaa höyry- ja nestefaasi veden kierrossa kattilassa. Nestemäinen vesi johdetaan lieriöstä alasmienoputkien kautta kattilan höyrystinputkille, jotka vastaanottavat lämpöä polttoaineen palamisesta ja toimivat näin veden höyrystiminä. [11] Lieriöltä erotettu kylläinen höyry johdetaan höyryn tulistimille, joissa höyryn lämpösältöä nostetaan halutulle tasolle turbiinia varten [11,13]. Tämä prosessi havainnollistetaan kuvassa 5. Kattila voi olla tyypiltään myös läpivirtauskattila, jossa lieriötä ei ole, vaan vesi höyrystyy yhdellä jatkuvalla kierrolla [13].



Kuva 5. Veden kierto höyrykattilassa. Perustuu lähteeseen [2].

Turbiinille saapuva korkeaenerginen höyry virtaa turbiinissa useiden siipisarjojen läpi. Osa siipisarjoista pysyy paikallaan, jolloin höyryn virtausnopeus kasvaa ja paine alenee suunnan muuttuessa. Sisään tulevan höyryn korkean paineen potentiaalienergia saadaan siis turbiinin paikallaan pysyvillä staattorisiivillä muutettua kineettiseksi energiaksi. [11] Staattorisiiviltä höyry johdetaan turbiinin pyöriville roottorisiiville, jolloin kineettinen energia muutetaan mekaaniseksi energiaksi turbiinin akselin pyörittämiseen [11,13]. Akseli on kytketty generaattoriin, joka muuntaa mekaanisen energian sähköenergiaksi [13]. Generaattorilta tuotettu sähköenergia siirretään sähköverkkoon käytettäväksi.

Turbiinilta matalapaineinen höyry johdetaan lauhduttimelle, jonka tarkoituksena on lauhduttaa höyry nestemäiseksi lämmönsiirron kautta. Turbiinista johdettu höyry virtaa lauhduttimen läpi, jossa lämpöä siirtyy korkeammassa lämpötilassa olevasta höyrystä matalammassa lämpötilassa olevaan jäähdytysveteen sekoittumatta keskenään. Tällöin höyry tiivistyy lauhduttimessa nesteeksi samalla luoden tyhjiön tilavuuden pienentyessä. [11] Lauhduttimesta poistuva tiivistynyt höyry, eli lauhde, pumpataan takaisin kattilaan, jossa kierto alkaa jälleen alusta.

2.2 Optimointikohteet käytönaikaisessa optimoinnissa

Lämpövoimalaitosten optimoinnilla pyritään parantamaan laitosten prosessien tehokkuutta ja luotettavuutta, mutta samanaikaisesti voidaan pyrkiä huomioimaan myös taloudellisuuteen ja päästöihin vaikuttavia tekijöitä monitavoiteoptimoinnin avulla [15]. Optimointiongelmat voivat olla hyvin monimutkaisia, sillä säädettäviä parametreja voi olla useita samassa ongelmassa. Lisäksi parametrit voivat vaikuttaa toisiinsa monin tavoin, mikä tekee optimoinnista haastavaa. [16]

Lämpövoimalaitoksissa prosessien tehokkuuden optimoinnilla voidaan vaikuttaa suoraan syntyviin päästöihin sekä laitosten hyötysuhteeseen, mikä tekee niistä merkittävän

optimointikohteen. Prosessien optimointi merkitsee energian tai raaka-aineiden tehokkaampaa käyttöä, kuten polttoaineen puhtaampaa palamista tai palamisesta saatavan lämmön käyttöä tehokkaammin prosesseissa, kuten esimerkiksi lämmönvaihtimissa. Käytönaikaisessa optimoinnissa prosesseja säädetään reaaliaikaisesti ja ympäristön tila voi vaihdella prosessien aikana, joten optimoinnin parametreille on tarve löytää optimaalisimmat arvot tehokkaasti. [17]

Lämpövoimalaitosten yhtenä ongelmana voidaan pitää myös polttoaineiden polttamisessa vapautuvia ympäristölle haitallisia päästöjä, kuten typen oksideja (NO_x), rikkidioksidia (SO_2) ja hiilidioksidia (CO_2). Hiilidioksidi kuuluu kasvihuonekaasuihin, joiden liiallinen kertyminen ilmakehään vaikuttaa ilman lämpötilan nousuun. Muodostunut rikkidioksidi taas voi reagoida ilman tai savukaasun kosteuden kanssa muodostaen rikkihappoa, joka tiivistyessään voi aiheuttaa kattilan komponenteissa korroosiota tai ilmakehään päästessään happosateita. [11] Päästöjen hallinta on siis merkittävä optimointikohde, jota voidaan tehostaa kehittämällä parempia optimointimenetelmiä, joissa keskitytään polttoprosessin tehokkuuteen sekä puhdistusjärjestelmien kehittämiseen, joilla minimoidaan haitallisten yhdisteiden muodostumista.

Prosessien tehokkuuden ja päästöjen hallinnan optimoinnissa kattilan lämpötehokkuus on yksi merkittävä optimointikohde. Lämpötehokkuuteen vaikuttaa vahvasti kattilan tulipesässä tapahtuva palamisprosessi. Palamisprosessi taas riippuu merkittävästi polttoaineen ja hapen suhteesta, jota voidaan säätää prosessissa polttoaineen ja ilmavirtauksen syötöllä. Liian vähäisellä hapen määrällä palaminen on epätäydellistä, jolloin muodostuu esimerkiksi hiilimonoksidia (CO) eli häkää. Toisaalta taas, kun happea on enemmän, niin esimerkiksi typen oksideja muodostuu enemmän. Optimoinnin tavoitteena voi olla siis esimerkiksi typpipäästöjen vähentäminen sekä kattilan lämpöhyötysuhteen tehostaminen. [7] Tällöin optimaalisimman ratkaisun löytäminen voi vaikeutua, sillä tavoitteet eivät tue toisiaan ja niiden suhde toisiinsa on keskenään ristiriidassa [3,7]. Lisäksi höyrynkulutuksen minimointi siten, että sähköntuotanto ei muutu, on merkittävä ajoitustekninen optimointikohde [18]. Optimointiongelmien ratkaisemisen apuna voidaan käyttää prosessien dynaamista mallintamista, mutta se voi olla haastavaa, sillä esimerkiksi lämpöjärjestelmät voivat olla hyvin monimutkaisia [7].

Käytettävyys ja toimintavarmuus ovat myös merkittävä osa käytönaikaista optimointia, sillä suunnittelemattomat tuotannon katkokset voivat heikentää laitosten kustannustehokkuutta sekä vaikuttaa turvallisuuteen laitosympäristössä [19]. Reaaliaikaisten prosessien tilasta saadulla mittausdatalla mahdollistetaan ohjaus ja prosessien toiminnan eteneminen. Mittausdata mahdollistetaan anturiteknologian avulla, joilla ympäristön tila-

muuttujien reaaliaikaisia arvoja saadaan selville. Antureiden avulla voidaan havaita virhetilanteita prosessissa, mikä tekee käytettävien antureiden valinnasta merkittävän lämpövoimaloiden käytettävyyden ja toimintavarmuuden kannalta käytönaikaisessa optimoinnissa [9,10]. Yhtenä yleisimpänä katkoksia aiheuttavana tekijänä pidetään kattilaputkien vuotoa [10]. Huollon havaitseminen ja ennakoiva kunnossapito ovatkin laitoksen tehokasta ja turvallista käyttöä edistäviä kehityksen ja optimoinnin kohteita.

Lämpövoimalaitoksissa näihin ongelmiin on pyritty vastaamaan erilaisilla säätöjärjestelmillä sekä algoritmeilla. Nämä kuitenkin ovat usein rajallisia, sillä niiden toiminta perustuu usein lineaarisiin ja gradienttipohjaisiin malleihin, jolloin niiden mukautuminen muuttuviin olosuhteisiin on haastavaa [2]. Gradienttipohjaisten optimointialgoritmien ongelmana on, että ratkaisu voi päätyä esimerkiksi paikalliseen minimiin globaalin minimin sijasta, jolloin algoritmin antaman ratkaisun arvo ei ole optimaalinen [2,3]. Monimutkaisiin ja haastaviin optimointiongelmiin mahdollisena ratkaisuna ovat koneoppimista hyödyntävät mallit, joiden avulla optimointia voidaan tehostaa.

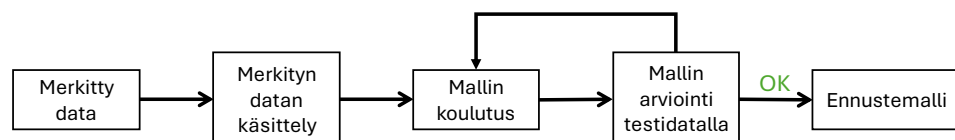
3. KONEOPPIMINEN

Koneoppimisella tarkoitetaan tekoälyn osa-aluetta, jossa koneet kykenevät kehittämään suoritustaan itsenäisesti kokemuksen tai datan perusteella ilman, että niitä on erikseen ohjelmoitu tiettyyn tehtävään [5]. Koneoppimisen prosessi voidaan yleisesti jaotella eri osiin, joihin kuuluvat prosessille syötetty data, abstraktio ja yleistys. Abstraktiossa koulutetaan syötetyn datan perusteella malli, joka voi olla esimerkiksi matemaattinen yhtälö tai looginen ryhmittely, riippuen syötetyn datan ja ongelman tyypistä. Yleistyksessä abstraktiovaiheen aikana luotu malli muokataan siten, että sitä voidaan käyttää uusien tapauksien ennustamiseen. [6]

Koneoppiminen voidaan jaotella oppimismenetelmän perusteella eri osa-alueisiin, joita ovat ohjattu oppiminen, ohjaamaton oppiminen, puoliohjattu oppiminen sekä vahvistusoppiminen [5,6]. Näitä menetelmiä ja algoritmeja käsitellään tässä luvussa tarkemmin, lukuun ottamatta puoliohjatun oppimisen menetelmiä, jotta työ ei paisuisi liikaa. Lisäksi syvennytään myös neuroverkkoihin, joilla pyritään mallintamaan biologisen hermojärjestelmän toimintaa [6].

3.1 Ohjattu oppiminen

Ohjatussa oppimisessa merkitystä koulutusdatasta pyritään luomaan malli, joka voi muodostaa ennusteita tai luokituksia uusille tiedoille [5,6]. Merkitty data koostuu syötteistä sekä niiden tunnisteista, jotka muodostavat koulutusdatan, jossa syöte yhdistetään kyseisen syötteen oikeaan vastaukseen tai mahdolliseen luokkaan [5]. Koulutusdatassa syötteet voivat olla esimerkiksi kuvia tai tekstiä ja tunnisteet esimerkiksi luokkia tai numeerisia arvoja [6]. Tässä koneoppimismenetelmässä syötteiden tunnisteet annetaan ohjaajan toimesta, joka voi olla ihminen tai kone [5]. Tästä ohjattu oppiminen saa myös nimensä. Ohjatun oppimisen prosessin toimintaperiaate havainnollistetaan kuvassa 6.



Kuva 6. Ohjatun oppimisen toimintaperiaate. Perustuu lähteeseen [6].

Ohjattu oppiminen voidaan jaotella kahteen osa-alueeseen, jotka ovat luokittelu ja regressio [5,6]. Luokittelussa mallin tarkoituksena on ennustaa koulutusdatan perusteella testidatan luokka eli kategorinen tunniste. Mallin luokittelupäätös tehdään koulutusdatan

perusteella, jossa syötteiden luokat ovat jo valmiiksi määritettyjä ohjaajan toimesta. Testidata sisältää uusia syötteitä, joiden luokat malli on opetellut tunnistamaan aiempien tietojen perusteella ja pystyy siten ennustamaan niiden luokat kuvan 6 prosessin mukaisesti. Luokittelussa yleisesti käytettyjä algoritmeja ovat muun muassa päätöspuu, satunnaismetsä, k-lähin naapuri, tukivektorikone ja Naiivi Bayes. [6]

Yksi yleisimmistä luokittelualgoritmeista on päätöspuu, jossa malli muodostetaan haarautuvilla solmupisteillä. Päätöspuussa koulutusdata jakautuu pienempiin osiin solmupisteissä, jolloin muodostuu haarautuva puuta muistuttava malli. Juurisolmu on päätöspuun ensimmäinen solmupiste ja se sisältää koko algoritmille syötetyn datan. Tämä jakautuu pienempiin kokonaisuuksiin ominaisuuksien jaottelulla, jota jatketaan, kunnes havaitaan jokin pysäytyskriteeri. Pysäytyskriteeri on olennainen osa päätöspuuta, sillä muuten jaottelu voisi jatkua loputtomasti ja ylisovittaa koulutusdatan, jolloin mallin yleistysvaihe ei toteudu oikein. Tämä on useiden koneoppimisalgoritmien heikkous. [6] Kriteerinä voi olla esimerkiksi solmussa olevien tietojen kuuluminen pääosin samaan luokkaan, jolloin jaottelu ei tuo prosessille enää lisäarvoa [6,20].

Useiden päätöspuiden yhdistämistä samaan malliin kutsutaan satunnaismetsäksi [6]. Tällöin satunnaismetsän sisältämien päätöspuiden tuloksista saadaan varianssia ja virheiden vaikutusta pienennettyä, sillä lopulliseksi tulokseksi otetaan enemmistön mukainen tulos tai päätöspuiden keskiarvo [6,20]. Satunnaismetsän datajoukko saadaan jaoteltua pienempiin kokonaisuuksiin päätöspuilla, mikä mahdollistaa myös suurempien datajoukkojen käsittelyn. [21]

Algoritmia, jossa luokittelu suoritetaan lähimpien datapisteiden perusteella, kutsutaan k-lähimmän naapurin menetelmäksi. Datapisteet koostuvat koulutusdatasta, jonka perusteella ennestään tuntemattomalle datapisteelle voidaan ennustaa luokka etäisyyksien avulla. [5,6] Yksi yleisimmistä käytetyistä etäisyyksimittauksista on euklidinen etäisyys, joka määritellään yhtälön 3 mukaisesti

$$d_E = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}. \quad (3)$$

Tässä datapisteiden välinen etäisyys d_E lasketaan pisteiden koordinaattien erotusten neliöiden summan neliöjuurena [9]. Naapureiden määrän huomioiminen tehdään arvon k valinnalla. Jos k -arvoksi valitaan esimerkiksi 3, menetelmässä otetaan huomioon kolme lähintä naapuria, joiden enemmistöluokka määrää uuden datapisteen luokan. Arvon k valitseminen on kriittinen vaihe luokittelun onnistumisen kannalta, sillä liian suuren k -arvon valitseminen voi johtaa lähimpien naapureiden vaikutuksen heikkenemiseen,

mikä mahdollistaa virheellisen luokittelun. Toisaalta myös liian pienen k -arvon valitseminen voi aiheuttaa virheellisen tuloksen, sillä datapisteissä voi esiintyä kohinaa sekä poikkeamia. Näihin ongelmiin yhtenä ratkaisuna on valita k -arvoksi koulutusdatan lukumäärän neliöjuuri, painottaa lähimpien naapureiden vaikutusta tai k -arvojen testaaminen useilla eri arvoilla ja tutkia niiden suorituskykyä. [6]

Tukivektorikoneessa datapisteiden välille luodaan hypertaso, jolla erotetaan kahteen eri luokkaan kuuluvat pisteet siten, että tason ja luokkien pisteiden välillä on mahdollisimman suuri marginaali. Tukivektorikoneessa pyritään siis maksimoimaan hypertason tai hypertasojen ja lähimpien datapisteiden eli tukivektoreiden välinen etäisyys. [5,6] Saata-villa oleva data ei aina ole lineaarisesti eroteltavissa, jolloin voidaan käyttää ydintempua (engl. kernel trick), jossa datajoukon ulottuvuus voidaan muuntaa korkeampaan ulottuvuuteen. Tämä voidaan toteuttaa ydinfunktiolla, joita ovat esimerkiksi lineaarinen ydinfunktio sekä polynominen ydinfunktio. Ydintempua käyttämällä tukivektorikoneen on mahdollista löytää optimaalinen hypertaso, sillä data voidaan erotella lineaarisesti korkeammassa ulottuvuudessa, vaikka se ei ole alkuperäisessä ulottuvuudessa mahdollista. [6] Tukivektorikone ei kuitenkaan sovi erittäin suurille datajoukoille, sillä oppimisprosessin laskenta muuttuu hyvin raskaaksi ja sitä kautta kalliiksi [21].

Naiivi Bayes -luokittelualgoritmi perustuu Bayesin teoreemaan, jolla pyritään ennustamaan todennäköisyys eri luokille annettujen tietojen perusteella [6,21]. Tässä luokittelussa koulutusdatan ominaisuudet oletetaan tasavertaisiksi ja riippumattomiksi, mikä tekee siitä hyvin yksinkertaisen ja tehokkaan luokittelijan [6]. Toisaalta nämä oletukset eivät kuitenkaan aina pidä paikkaansa, sillä esimerkiksi jotkin koulutusdatan ominaisuudet voivat olla riippuvaisia toisistaan, jolloin luokittelijan suorituskyky voi heikentyä. Bayesin teoreema voidaan esittää yhtälön 4 mukaisesti

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \quad (4)$$

jossa $P(A|B)$ on posteriori-todennäköisyys, joka kuvaa tapahtuman A todennäköisyyttä, kun tapahtuma B on jo toteutunut. $P(A)$ taas tarkoittaa priori-todennäköisyyttä eli aiempaan tietoon tai oletuksiin perustuvaa todennäköisyyttä tapahtumalle ennen tapahtuman B toteutumista. $P(B)$ kuvaa marginaalitodennäköisyyttä eli todennäköisyyttä sille, että tapahtuma B toteutuu riippumatta tapahtuman A toteutumisesta ja $P(B|A)$ on todennäköisyys tapahtumalle B , kun tapahtuma A on jo toteutunut. [6] Bayesin teoreemaa hyödyntämällä saadaan siis ennustettua todennäköisin luokka koulutusdatan perusteella.

Regressiossa tavoitteena on ennustaa numeerinen arvo tietylle ongelmalle. Regressiossa pyritään löytämään yhteys ennustettavan muuttujan sekä siitä funktionaalisesti riippuvien muuttujien välille, joita käytetään ennustettavan muuttujan ratkaisemiseen. Yksi yleisimmistä regressioalgoritmeista on lineaarinen regressio. [6]

Yksinkertaisessa lineaarisessa regressiossa oletetaan lineaarinen riippuvuus riippumattoman muuttujan X ja riippuvan muuttujan Y välille, mitä voidaan kuvata yhtälöllä 5

$$Y = a + bX + e, \quad (5)$$

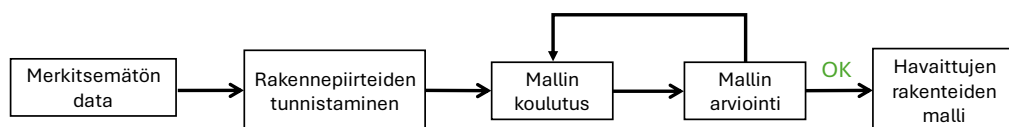
jossa a kuvaa vakiotermiä eli Y -akselin leikkauspistettä, b kulmakerrointa eli muuttujan Y muutoksen suhdetta muuttujan X muutokseen ja e mallin poikkeamaa todellisista arvoista. Hajontakaavio saadaan lisäämällä sekä riippumaton, että riippuva muuttuja kaavioon. Riippuvana muuttujana voi olla esimerkiksi kiinteistön hinta ja riippumattomana kiinteistön pinta-ala. Hajontakaavioon sovitetaan regressiosuora pienimmän neliösumman menetelmällä, jossa virheiden neliöiden summa pyritään minimoimaan. [6] Saadun suoran avulla voidaan ennustaa muuttujan Y arvoja annetuilla muuttujan X arvoilla. Yksinkertaisessa lineaarisessa regressiossa on mukana vain yksi ennustemuuttuja, mutta todellisuudessa voi olla useampia. Moninkertaisessa lineaarisessa regressiossa huomioidaan useampi ennustemuuttuja, jolloin saadaan yhtälön 6 mukainen lauseke

$$Y = a + b_1X_1 + b_2X_2 + \dots + b_nX_n + e, \quad (6)$$

jossa n kertoo ennustemuuttujien määrän. b_1 , b_2 ja b_n kuvaavat muuttujan Y muutosta tietyn ennustemuuttujan suhteen, kun muut ennustemuuttujat pidetään vakioina. [6]

3.2 Ohjaamaton oppiminen

Ohjatussa oppimisessä koulutusdata oli merkittyä, jossa tiedettiin syötteiden tunnisteet, kuten esimerkiksi luokat tai numeeriset arvot. Ohjaamattomassa oppimisessä syötedata ei ole merkittyä, vaan sen sijaan algoritmin tarkoituksena on löytää datasta yhtenäisiä rakenteita ja ryhmittymiä ilman ennalta annettuja tulosuuttuja [5,6]. Ohjaamaton oppiminen ei siis ennusta luokkia tai numeerisia arvoja ohjatun oppimisen tapaan merkitystä datasta, vaan sen sijaan pyrkii löytämään samankaltaisuuksia merkitsemättömästä datasta [6,21]. Ohjaamaton oppiminen havainnollistetaan kuvassa 7.



Kuva 6. Ohjaamattoman oppimisen toimintaperiaate. Perustuu lähteeseen [6].

Ohjaamattomaan oppimiseen kuuluvia menetelmiä ovat klusterointi sekä assosiaatioanalyysi. Klusteroinnilla tarkoitetaan prosessia, jossa datajoukosta ryhmitellään samankaltaiset kohteet joukkoihin niiden ominaisuuksien perusteella kuvan 7 prosessin mukaisesti. [6] Näitä joukkoja kutsutaan myös klustereiksi. Klusterointia voidaan käyttää esimerkiksi datan louhinnassa ryhmittelemällä suuren tietojoukon omaava data pienempiin osiin. [5,6] Klusterointimenetelmät voidaan yleisesti jaotella jakamismenetelmiin, hierarkkisiin menetelmiin sekä tiheysperusteisiin menetelmiin [6].

Jakamismenetelmissä datajoukon ryhmittelyssä voidaan käyttää alaryhmien datapisteiden keskiarvoihin perustuvaa k-keskiarvot-algoritmia [5]. Algoritmista pyritään maksimoimaan klustereiden väliset eroavaisuudet datapisteiden välisten etäisyyksien avulla [6]. Algoritmin peruserä on jakaa datapisteet tiettyyn lukumäärään klustereita, jonka määrää arvo k [5,6]. Jokainen piste pyritään ryhmittelemään klusteriin, jonka keskipiste on sille lähimpänä [21]. Keskipiste määräytyy klusterin pisteiden keskiarvosta, jota päivitetään iteraatiokierrosten aikana, kunnes keskipiste ei enää muutu [5,6,21]. Arvo k voidaan valita käyttämällä esimerkiksi kynnärpää metodia, jossa mitataan klusterien sisäisiä samankaltaisuuksia tai eroavaisuuksia muuttamalla arvoa k [6]. Tiettyssä pisteessä arvon k kasvattaminen ei muuta klusterien sisäisiä samankaltaisuuksia enää merkittävästi, jolloin arvon k kasvattaminen ei ole tarpeellista iteraatiokierrosten aiheuttamien laskentakustannuksien suhteen. Tätä pistettä kutsutaan kynnärpääpisteeksi. [6]

Hierarkkisissa menetelmissä datajoukosta pyritään löytämään hierarkkinen rakenne. Tämä voidaan toteuttaa siten, että erotetaan kaikki datapisteet ensin omiksi klustereikseen, jonka jälkeen niitä yhdistetään samankaltaisuuksien perusteella suuremmiksi klustereiksi. Tällöin muodostuu puumainen hierarkkinen rakenne datapisteiden välille, joka päättyy valittuun lopetusehtoon. Hierarkkinen rakenne voidaan muodostaa myös aloittamalla yhdestä suuresta klusterista, joka jaotellaan iteratiivisesti pienempiin klustereihin. Menetelmässä, jossa klustereita yhdistetään, lopetusehtona voi olla esimerkiksi kaikkien alaryhmien kuuluminen samaan ryhmään tai ennalta määritetyn klusterimäärän saavuttaminen. Toisessa menetelmässä lopetusehtona taas voi olla esimerkiksi kaikkien datapisteiden kuuluminen omiin alaryhmiin. [6]

Tiheysperusteisissa menetelmissä datajoukon jaottelu ryhmiin perustuu datajoukon pisteiden muodostamien alueiden tiheyksiin [6]. Tämä mahdollistaa samankaltaisuuksiin perustuvien ryhmien eli klustereiden muodon monipuolisemman vaihtelevuuden verrattuna jakamis- ja hierarkkisiin menetelmiin. Yksi yleisimmistä tiheysperusteisten menetelmien algoritmeista on DBSCAN, jolla luodaan klustereita perustuen datapisteiden tiheyksiin. [6,22] Algoritmi tarvitsee toimiakseen rajapituuden tiheysalueelle sekä minimimäärän pisteitä, jotka sijaitsevat tämän alueen sisällä. Pisteiden väliset etäisyydet voidaan

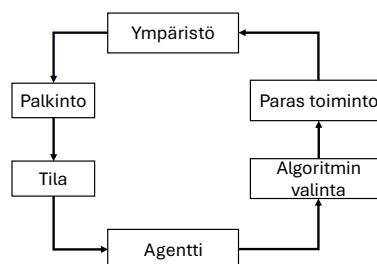
määrittää euklidisella etäisyydellä, kuten myös aiemmissa klusterointimenetelmissä [6]. Pisteiden välistä etäisyyttä verrataan valittuun rajapituuteen ja mikäli se on pienempi tai yhtä suuri, niitä voidaan pitää kuuluvan samaan klusteriin. Klustereiden muodostumiseen vaikuttavat myös määritetty klusterin pisteiden minimimäärä, jolla asetetaan ehto vähimmäismäärälle pisteitä rajapituuden muodostaman alueen sisälle. Klusteri voidaan muodostaa, jos molemmat edellä mainitut ehdot täyttyvät. [22]

Ohjaamattoman oppimisen toisessa osa-alueessa, assosiaatioanalyysissä, data-aineistosta tunnistetaan useasti esiintyviä suhteita yksittäisten objektien välillä. Suhteiden perusteella muodostetaan sääntöjä, jotka perustuvat tiettyjen tapahtumien esiintymisien todennäköisyyksiin. Säännöt voidaan esittää esimerkiksi muodossa: jos tapahtumat 1 ja 2 tapahtuvat, niin todennäköisesti myös tapahtuma 3 toteutuu. Assosiaatiosääntöjen löytymistä voidaan tehostaa käyttämällä apriori-algoritmia, jolla suodatetaan epäolennaimmat säännöt pois. Tällöin suuria datajoukkoja käsiteltäessä sääntöjen muodostaminen tehostuu, sillä vain olennaimmat säännöt otetaan huomioon. Esimerkiksi, jos tietyt tapahtumat ovat harvinaisia, niiden suhteita ei ole tarvetta käsitellä. Epäolennainten sääntöjen suodattamisessa käytetään minimitukirajaa ja -luottamusrajaa, jotka määrittävät hyväksyttävien sääntöjen määrittämisen. [6]

3.3 Vahvistusoppiminen

Vahvistusoppimisessa toiminta perustuu oppijan eli agentin tekemiin toimintoihin ja päätöksiin, joista ympäristö antaa palautetta [5]. Agentti oppii saamastaan palautteesta ja muuttaa toimintaansa sen mukaan, mikäli se on tarpeellista. Jos agentti tekee oikean päätöksen, se saa siitä positiivisen palkinnon, jota ei väärästä päätöksestä saada. [6] Päätöksenteko jatkuu niin kauan, että päädytään haluttuun lopputulokseen eli palkkion maksimoimiseen [21].

Agentin ja ympäristön välistä vuorovaikutusta havainnollistetaan kuvassa 8, jossa agentti havaitsee ympäristön tilan ja valitsee parhaan toiminnon sopivaa algoritmia käyttäen. Ympäristö reagoi toimintoon ja antaa mahdollisen palkinnon, jota agentti käyttää hyödykseen seuraaviin päätöksiin.



Kuva 8. Agentin ja ympäristön välinen vuorovaikutus. Perustuu lähteeseen [6].

Yksi vahvistusoppimisen algoritmeista on Q-oppiminen, joka ei tarvitse ennakkoon tietoa tilasiirtymämallista tai palkkiofunktioista toimiakseen. Q-oppimisessa optimaalinen toiminta opitaan kokeilemalla eri toimintoja, joista saatujen palautteiden avulla päivitetään algoritmin Q-arvoja. Tämä koostuu kahdesta vaiheesta, joista ensimmäisessä arvioidaan tilan ja toiminnan arvot, jotka perustuvat palkkioon sekä arvioituun seuraavan tilan arvoon. Toisessa vaiheessa arvioita muokataan hyödyntämällä aikaisempaa tietoa sekä uusia havaintoja, joiden perusteella valitaan optimaalisin toiminto. Ensimmäinen vaihe voidaan esittää yhtälön 7 mukaisesti

$$\hat{q}^n(s^n, a^n) = r(s^n, a^n) + \gamma \max_{a'} \bar{Q}^{n-1}(s', a'), \quad (7)$$

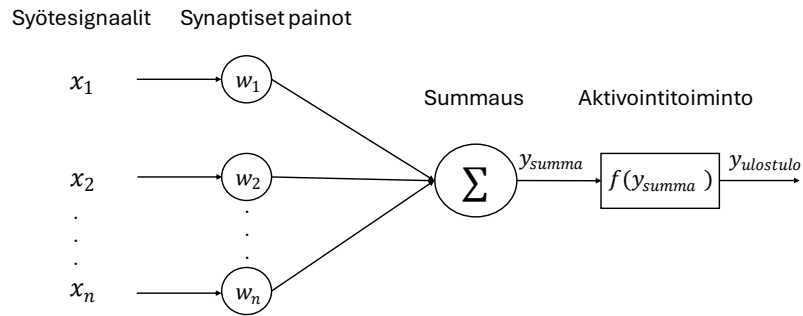
jossa $\hat{q}^n(s^n, a^n)$ on arvio tapahtumalle suorittaessa toiminto a^n tilassa s^n . Arvio lasketaan hyödyntämällä saatua palkkiota $r(s^n, a^n)$ ja arvioitua arvoa $\bar{Q}^{n-1}(s', a')$ valitsemalla optimaalisin seuraava toiminto a' tilassa s' . Tekijä γ määrittää tulevien palkkioiden suhteen nykyisiin palkkioihin. Toinen vaihe voidaan esittää yhtälön 8 mukaisesti

$$\bar{Q}^n(s^n, a^n) = (1 - \alpha_{n-1})\bar{Q}^{n-1}(s^n, a^n) + \alpha_{n-1}\hat{q}^n(s^n, a^n), \quad (8)$$

jossa $\bar{Q}^n(s^n, a^n)$ on päivitetty Q-arvo, joka saadaan yhdistämällä vanha ja uusi havainto. Tekijä α_{n-1} määrittää uusien havaintojen vaikutuksen aikaisempiin arvoihin. [23] Tämä ei kuitenkaan ole täysin ongelmaton, sillä oppimisen tehokkuus riippuu paljon palkintojen toteuttamisesta ja ajoituksesta, jotka määrittävät algoritmin oikeaoppisen toiminnan. [21]

3.4 Neuroverkot ja syväoppiminen

Ihmisen biologinen hermojärjestelmä koostuu keskushermostosta sekä ääreishermostosta, jotka koostuvat yksittäisistä neuroneista. Neuronit ovat hermoston tiedonsiirtäjiä ja -käsittelijöitä, jotka ovat yhteydessä muihin hermoston neuroneihin kemiallisin ja sähköisin signaalein. [6] Keinotekoiset neuroverkot (ANN) ovat koneoppimisen osa-alue, jolla pyritään simuloimaan biologisen hermojärjestelmän rakennetta ja toimintaa monimutkaisten regressio- ja luokitteluongelmien ratkaisemiseksi [24]. ANN koostuu useista keino-tekaisista neuroneista, joiden rakenne havainnollistetaan kuvassa 9.



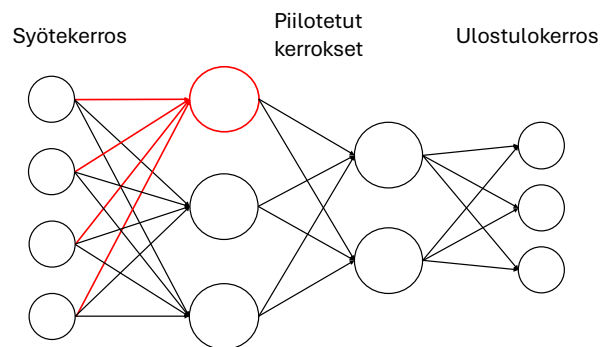
Kuva 9. Keinotekoisien neuronin rakenne. Perustuu lähteeseen [6].

Keinotekoisien neuronin toiminta perustuu syötesignaalien ja synapsien, eli neuronien välisien yhteyksien, painotettuun summaan [5]. Painotettu summa saadaan laskemalla yhteen syötesignaalien ja synaptisten painojen tulot. Yleisesti summaan lisätään myös lisäsyöttö, jolla mahdollistetaan kynnsarvon siirtäminen ja siten aktivointitoiminnon säätö. Lopullinen painotettu summa y_{summa} voidaan esittää yhtälön 9 mukaisesti

$$y_{summa} = b + \sum_{i=1}^n w_i x_i, \quad (9)$$

jossa n kuvaa syötesignaalien lukumäärää, b lisäsyöttöä ja $\sum_{i=1}^n w_i x_i$ synaptisten painojen w_i ja syötesignaalien x_i tulojen summaa. [5,6] Aktivointitoiminto ottaa syötteenä painotetun summan ja tuottaa ulostulosignaalin, jos syöte ylittää määritetyn kynnsarvon [6]. Aktivointitoiminto käsittelee syötteen tilanteeseen määritetyllä funktiolla, joita voivat olla esimerkiksi askel-, lineaari-, ja sigmoidifunktiot [5]. Yleisesti aktivointitoiminnon ulostulo voidaan määrittellä kuvan 9 tapaan $y_{ulosulo} = f(y_{summa})$ [6].

ANN voi koostua useista kerroksista neuroneita, jotka toimivat edellä esitetyllä tavalla siirtäen tietoa eteenpäin. Monikerroksisissa neuroverkoissa kerroksille käytetään nimityksiä syötekerros, piilotettu kerros sekä ulostulokerros [5,6]. Tämä rakenne esitetään kuvassa 10.



Kuva 10. Yksinkertaisen monikerroksisen neuroverkon rakenne, jossa yksittäinen neuroni korostettu punaisella. Perustuu lähteeseen [5].

Monikerroksisessa neuroverkossa jokainen yksittäinen neuroni saa syötteen muilta neuroneilta ja tekee niiden perusteella toiminnon eli aktivoituu, mikäli kynnyсарvo ylittyy [5,6]. Verkon kouluttamisen tavoitteena on oikeiden neuronien aktivoituminen tietyille syötteelle, jotta saavutetaan haluttu lopputulos [5].

Neuroverkkojen kouluttamiseen voidaan käyttää ohjattuun oppimiseen perustuvaa vastavirta-algoritmia, jossa ulostulokerrokselta saatuja arvoja verrataan todellisiin arvoihin. Muodostunut virhe välitetään aiemmille kerroksille, joissa kerroksien välisiä painoja säädetään muodostuneen virheen perusteella ja muodostetaan uusi ulostulo. [6] Oppimisenopeus on vastavirta-algoritmia käyttävissä malleissa tärkeä parametri, sillä se ohjaa painojen säätämistä. Liian suurella oppimisenopeudella neuroverkosta tulee epävakaa ja liian pienellä taas oppiminen voi olla hyvin hidasta. [21] Vastavirta-algoritmilla pyritään siis minimoimaan ulostulokerrokselta saatujen arvojen ja todellisten arvojen erotuksesta muodostettu virhefunktio iteraatiokierrosten aikana ja siten löytää optimaalisimmat painot neuroverkolle [5].

Syväoppimisella tarkoitetaan koneoppimisen osa-aluetta, jossa ongelmien ratkaisu perustuu keinotekoisii neuroverkkoihin [25]. Syviksi oppijoiksi voidaan erityisesti kutsua neuroverkkoja, jotka koostuvat yli kolmesta kerroksesta [6,23]. Kerrosten lisääntyessä neuroverkoissa, laskentatehon tarve kasvaa merkittävästi [6]. Lisäksi syötetyn tietojoukon on oltava suuri, jotta oppimisprosessista saadaan tarkka, mikä on yksi neuroverkkojen haasteista [25]. Syväoppimisella voidaan kuitenkin mahdollistaa hyvin monimutkaisien optimointiongelmiien ratkaiseminen, sillä usean kerroksen neuroverkot oppivat mallintamaan syötteestä epälineaarisia suhteita ja ominaisuuksia, jotka voivat olla yksinkertaisille malleille haastavia. Tätä voidaan hyödyntää esimerkiksi kuvantunnistuksessa, joka vaatii monimutkaisien ominaisuuksien tunnistamista. [21]

3.5 Koneoppimisen osa-alueiden yleiset käyttökohteet

Koneoppimisen osa-alueiden toimintaperiaatteet eroavat toisistaan, joten tiettyihin käyttökohteisiin kaikki koneoppimisen osa-alueiden algoritmit eivät ole yhtä tehokkaita. Tästä syystä algoritmin valinnalla on suuri merkitys optimaalisen lopputuloksen saavuttamiseksi. Koneoppimisen osa-alueiden yleiset käyttökohteet ja algoritmit havainnollistetaan taulukossa 1.

Ohjatussa oppimisessa käyttökohteet jakautuvat luokittelun ja regression käyttökohteisiin. Luokittelussa käyttökohteet ovat yleisesti asioiden tunnistamista merkityn datan perusteella. Näitä ovat muun muassa kuvantunnistus, käsialantunnistus, puheentunnistus

sekä poikkeavuuksien tunnistus [6,25]. Regressiossa käyttökohteena on yleisesti numeerisen arvon ennustaminen aiempaan dataan perustuen, joten käyttökohde voi olla esimerkiksi asunnon hinnan ennustaminen [6].

Ohjaamattomassa oppimisessa käyttökohteet jakautuvat puolestaan klusteroinnin ja assosiaatioanalyysin käyttökohteisiin. Klusteroinnissa käyttökohteet painottuvat samankaltaisten ryhmien muodostamiseen, joten esimerkiksi datan louhinta tai poikkeavuuksien tarkistus merkitsemättömästä datasta on tälle mahdollinen käyttökohde. Assosiaatioanalyysin käyttökohteet perustuvat useasti esiintyvien suhteiden havaitsemiseen, kuten esimerkiksi ostoskorianalyyssissä tehdään. Ostoskorianalyyssissä tunnistetaan ihmisten ostokäyttäytymisestä useasti esiintyviä tapahtumia, kuten tiettyjen tuotteiden ostotapahtumia samalla kerralla. [6]

Vahvistusoppimisen käyttökohteissa mallit oppivat ympäristöltä saamastaan palautteesta, jonka perusteella uusi päätös tehdään [5,6]. Tähän soveltuvia käyttökohteita ovat esimerkiksi älykkäät robotit sekä itseajavat autot, sillä päätöksiä tehdään jatkuvasti muuttuvissa tilanteissa itsenäisesti [6,21].

Taulukko 1. Koneoppimisen osa-alueiden yleiset käyttökohteet sekä algoritmit

Koneoppimisen osa-alue	Oppiminen	Yleiset käyttökohteet	Yleiset algoritmit
Ohjattu oppiminen	Oppii merkitystä datasta, jossa syönteillä on tietyt tunnisteen ^{a,b,c}	Luokittelu: kuvantunnistus, käsialan tunnistus, puheentunnistus, poikkeavuuksien tunnistus ^{b,f} Regressio: numeerisen arvon ennustaminen ^b	Päätöspuu, satunnaismetsä, k-lähin naapuri, tukivektori-kone, Naivi Bayes, lineaarinen regressio ^{b,f}
Ohjaamaton oppiminen	Löytää datasta samankaltaisuuksiin perustuvia rakenteita ^{a,b,c}	Klusterointi: datan louhinta, poikkeavuuksien tarkistus ^b Assosiaatioanalyysi: ostoskorianalyysi ^b	k-keskiarvot, DBSCAN, apriori ^{b,d}
Vahvistusoppiminen	Oppii tekemällä päätöksiä ympäristöltä saadun palautteen avulla ^{a,b,c}	Älykkäät robotit, itse ajavat autot ^{b,c}	Q-oppiminen ^{b,c,e}

a [5], b [6], c [21], d [22], e [23], f [25]

Koneoppimisen osa-alueiden käyttäminen liittyy monesti siihen, mikä asia ongelman ratkaisussa on tavoitteena. Ohjattua oppimista käytetään usein tilanteissa, joissa tiedetään,

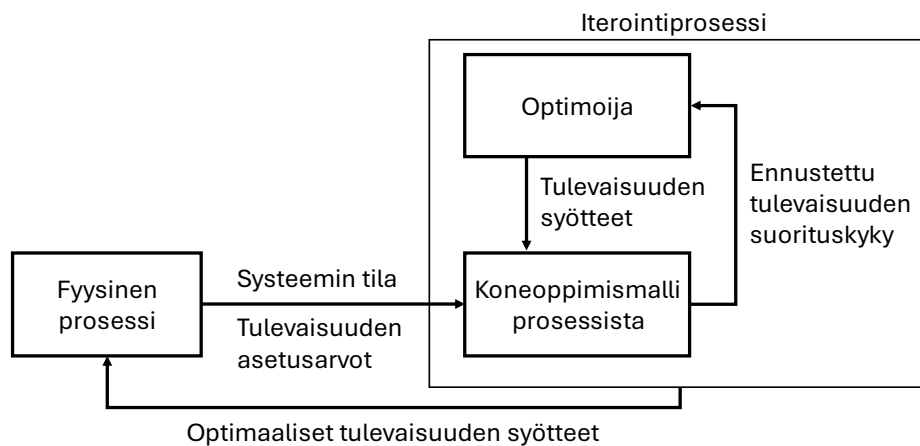
mitä halutaan etsiä, kuten esimerkiksi asunnon hinnan ennustamisessa. Jos taas ei tiedetä tarkalleen etsittävää asiaa, ohjaamaton oppiminen on siihen parempi ratkaisu, sillä se kykenee ryhmittelemään datajoukon ilman tietoa tunnisteista. Vahvistusoppiminen taas soveltuu parhaiten dynaamisiin tilanteisiin, joissa päätöksiä tehdään jatkuva-aikaisesti ympäristön palautteen perusteella. Vahvistusoppiminen on siis tämän perusteella dynaamisiin optimointitilanteisiin hyvin soveltuva. Toisaalta myös ohjattu ja ohjaamaton oppiminen kykenevät tukemaan dynaamista optimointia ja päätöksentekoa esimerkiksi ennustamalla arvoja tai tunnistamalla datajoukosta piileviä rakenteita.

4. KONEOPPIMINEN LÄMPÖVOIMALAITOSTEN KÄYTÖNAIKAISISSA OPTIMOINTIKOHTEISSA

Lämpövoimalaitosten käytönaikainen optimointi on merkittävä osa laitosten entistä tehokkaampaa ja turvallisempaa käyttöä. Optimoinnissa voidaan käyttää apuna koneoppimiseen perustuvia malleja ja algoritmeja, joita esiteltiin edellisessä luvussa. Tässä luvussa käydään läpi mahdollisia koneoppimisen sovelluskohteita ja mahdollisuuksia lämpövoimalaitosten optimoinnissa. Ensin keskitytään koneoppimisen hyödyntämiseen prosessien tehokkuuden näkökulmasta, jolla on merkittävä vaikutus sekä laitoksen taloudellisuuteen, että päästöjen hallintaan. Tämän jälkeen siirrytään tutkimaan koneoppimisen hyödyntämistä ennakoivassa kunnossapidossa ja käyttövarmuudessa.

4.1 Prosessien tehokkuus

Käytönaikaisessa prosessien tehokkuuden optimoinnissa keskeisenä asiana pidetään mallien ja optimointialgoritmin parametrien säätämistä, jotta optimointiongelmat pystytään ratkaisemaan mahdollisimman tehokkaasti ja tarkasti [17]. Koneoppimismalleja voidaan käyttää apuna prosessien dynaamisessa mallintamisessa ja optimaalisten syötteiden ennustamisessa yhdessä erilaisten optimointialgoritmin kanssa [2,7]. Esimerkki koneoppimisen roolista prosessin optimoinnissa havainnollistetaan kuvassa 11.



Kuva 11. Koneoppiminen osana prosessin optimointia. Perustuu lähteeseen [2].

Prosessin dynaaminen mallinnus voidaan toteuttaa esimerkiksi keinotekoisien neuroverkkojen (ANN) mukautetulla mallilla, jossa tietoa käsitellään ajan suhteen, ottamalla huomioon myös aikaisemmat tilat [2]. Malli koulutetaan harjoitusdatalla, jonka perusteella koulutetaan optimaalisimmat parametrit iteratiivisesti neuroverkon painoja säätämällä [2,6]. Virheiden pienentämistä voidaan tehostaa käyttämällä vastavirta-algoritmia

neuroverkon kerrosten välillä [7]. Harjoitusdata voi koostua sekä nykyisistä, että aikaisemmista prosessin tilan tiedoista, jotka parantavat mallin oppimista [2]. Esimerkiksi AL-Dura-voimalaitoksella prosessin mallintamisessa neuroverkot koulutettiin ennustamaan ja säätämään prosessin toimintaa syötteiden, kuten prosessin paineen ja lämpötilan, perusteella. Neuroverkoilla ennustettujen tulosten osoitettiin mallintavan erittäin hyvin kokeellisia tuloksia. [24] Prosessin parametrien ennustamisessa voidaan käyttää myös muita koneoppimismenetelmiä, kuten satunnaismetsää [20].

Mallinnettava prosessi voi olla esimerkiksi palamisprosessi lämpövoimalaitosten höyrykattiloissa [2,7]. Yhtenä optimointitavoitteena voi olla polttoaineenkulutuksen minimointi siten, että tehon tuotanto pysyy vakaana ja turvallisuusvaatimukset, kuten riittävän ylimäärähapen pitoisuus, toteutuvat [2]. Tarkan ennustetun mallin avulla voidaan ennakoita prosessissa tapahtuvia muutoksia ja siten parantaa palamiseen liittyvää säätöä [7]. Säätäminen voidaan toteuttaa esimerkiksi ilman syötöllä, jolla palaminen pyritään saamaan mahdollisimman tehokkaaksi kattilan lämpötehokkuuden parantamiseksi. Koneoppimista hyödyntävällä mallilla, yhdessä kuvan 11 mukaisella optimointikokonaisuudella, on mahdollistettu 485 MW:n hiilikäyttöisessä laitoksessa simuloidun kattilan noin 1 %:n lämpötehokkuuden nousu. Pitkällä aikavälillä tämä voi mahdollistaa hyvin merkittäviä taloudellisia säästöjä, sillä energia käytetään tehokkaammin. [2]

Polttoaineenkulutuksen tavoitefunktion minimoinnin lisäksi voidaan optimoida höyryturbiinin ja kattilan ajoitustehokkuus. Tämän optimointiongelman ratkaisuun voidaan hyödyntää syvävahvistusoppimisen menetelmää, jonka toiminta perustuu Q-oppimisen laajennettuun algoritmiin. Tässä optimointiprosessin tavoitteena on minimoida tietyn painetasen höyryn kokonaiskulutus siten, että sähkö- ja lämpökuormat toteutuvat. Optimaaliset oppimisen parametrit päivittyivät vahvistusoppimiseen perustuvalla algoritmillä jatkuvasti tilan arvofunktoita ja verkkoja säätämällä, joiden perusteella parhaat toiminnot voitiin valita. Tällä mahdollistettiin optimointiprosessin mallin iteratiivinen ratkaiseminen ajoituksen optimoimiseksi. [18]

Palamisprosessin mallintamisessa koneoppimista voidaan käyttää myös liekkikuvien tulkintaan perustuvaan palamisen poikkeavuuksien havaitsemiseen. Malli voidaan kouluttaa esimerkiksi syvillä neuroverkoilla tunnistamaan kuvista alhaisia ilmamääriä palamisessa muodostuneen noen perusteella, joka voidaan havaita kuvissa tummina alueina. Mallin havaittiin määrittävän kuvista palamistilan ja ylimääräisen ilman välisen suhteen keskimäärin 87 %:n tarkkuudella. [26] Tämä ei siis tuota täysin varmoja tuloksia, mutta tietoa voidaan käyttää optimaalisten parametrien löytämisen tukena optimointiprosessissa.

Ennustetarkkuutta on myös mahdollista tehostaa yhdistämällä ennustemalliin koneoppimiseen perustuva tietojoukon ominaisuuksien ryhmittelijä. Esimerkiksi ohjaamattomaan oppimiseen perustuvalla k-keskiarvot-algoritmilla ja hierarkkisella klusteroinnilla tietojoukko jaoteltiin ominaisuuksien perusteella omiin klustereihin. Klusterit muodostivat uuden tietojoukon, jota käytettiin ohjattuun oppimiseen perustuvilla regressiomalleilla koulutusdatana. Tämän havaittiin parantavan mallien ennustetarkkuutta merkittävästi. Esimerkiksi lineaarisen regression keskimääräinen absoluuttinen virhe energiantuotannon ennustamisessa puolittui. [27] Hybridimalleilla saadaan siis huomattavia parannuksia mallien ennustetarkkuuksiin.

Prosessien tehokkuuden kannalta olennainen osa on myös päästöjen hallinta, joka on ympäristön näkökulmasta hyvin merkittävä. Kattilaprosessissa tapahtuvan palamisen ollessa puhtaampaa, happea on prosessissa enemmän mukana, jolloin typen oksideja (NO_x) muodostuu myös enemmän. Tällöin optimaalisten prosessiparametrien löytäminen vaatii tehokasta optimointiprosessia, sillä optimointitavoitteet ovat keskenään ristiriidassa. Prosessien tehokas ja tarkka dynaaminen mallintaminen korostuu entisestään, joten koneoppimisella on merkittävä rooli. Tässä tapauksessa koulutetulle koneoppimis-mallille syötetään palamisprosessin parametrit, joiden käsittelyn jälkeen ulostulona muodostetaan sekä kattilan lämpöhyötysuhde, että NO_x -pitoisuus. [7]

245 MW:n kuormituksen voimalaitoksessa koneoppimista hyödyntävällä monitavoiteoptimointiprosessilla kattilan lämpöhyötysuhteen huomattiin nousevan 0,02 % samalla, kun NO_x -pitoisuuden todettiin laskevan 7,77 mg Nm^{-3} perinteisen palamisen optimointimenetelmän arvoihin verrattuna. Tämä NO_x -pitoisuuden lasku vastaa noin 30,4 %:n vähennystä alkuperäiseen pitoisuuteen verrattuna. Lämpöhyötysuhteen kasvu ja NO_x -pitoisuuden lasku todettiin myös 194 MW:n kuormitukselle. [7]

Hiilikäyttöisissä lämpövoimalaitoksissa koneoppimismallin yksi käyttökohteista on myös hiilen ominaisuuksien analysointi, jonka perusteella NO_x -pitoisuutta voidaan ennustaa yhdessä muiden parametrien kanssa [8]. Ennustetulla NO_x -pitoisuudella mahdollistetaan myös tarvittavien säätöjen tekeminen etukäteen, jolloin päästörajoihin on mahdollista päästä nopeammin. Koneoppimisen hyödyntämisen menetelmät ja sovelluskohteet prosessien tehokkuuden näkökulmasta esitellään taulukossa 2 tiivistettynä.

Taulukko 2. Koneoppimisen hyödyntäminen prosessien tehokkuuden optimoinnissa tiivistettynä

Kohde	Menetelmä	Esimerkki
Dynaaminen mallinnus ^a	Keinotekoiset neuroverkot, satunnaismetsä ^{a,d}	Parametrien ennustaminen ^{a,e}
Ajoituksen optimointi ^c	Syvävahvistusoppiminen ^c	Höyryn kulutuksen minimointi, sähkö- ja lämpökuorien toteutuminen ^c
Mallien tehokkuuden optimointi ^f	Hybridimalli (klusterointi + regressio) ^g	Ennustetarkkuuden parantaminen ^g
Palamisprosessin optimointi ^{a,b}	Monitavoiteoptimointi, kuvantulkinta ^{b,f}	Lämpötehokkuuden parantaminen, NO _x -pitoisuuden vähentäminen ^{a,b}

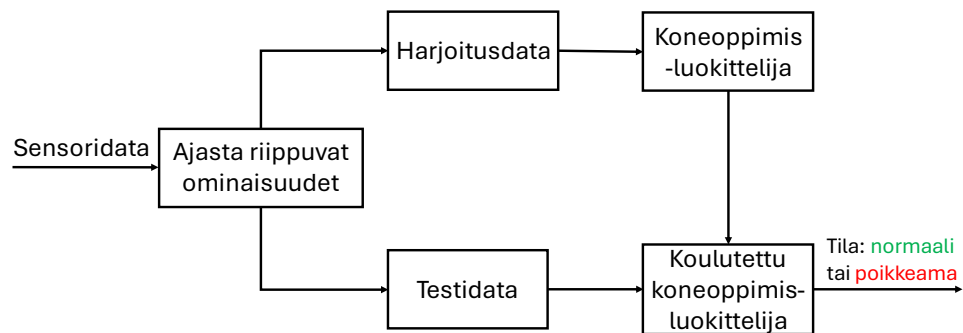
a [2], b [7], c [18], d [20], e [24], f [26], g [27]

Prosessien tehokkuuden koneoppimista hyödyntävällä optimoinnilla on siis hyvin monipuoliset mahdollisuudet lämpövoimalaitosten kehittämisessä. Nämä mahdollistavat sekä prosessien tehokkaan energiankäytön, että päästöjen hallinnan edistämisen.

4.2 Ennakoiva kunnossapito ja käyttövarmuus

Koneoppimismallien hyödyntäminen ei ainoastaan perustu prosessien tehokkuuden mallintamiseen ja optimointiin, vaan niitä voidaan hyödyntää myös ennakoivan kunnossapidon apuvälineenä. Tällöin keskitytään laitteistojen vikaantumisen ja mahdollisten häätätilojen ennustamiseen, mikä mahdollistaa huoltokustannuksien ja vaaratilanteiden vähentämisen.

Koneoppimista voidaan käyttää esimerkiksi datan luokittelussa lämpövoimalaitoksen komponenttien, kuten kattilan tai turbiinin, kunnon analysoinnissa [9,10]. Tässä koneoppimisprosessissa malli koulutetaan ensin harjoitusdatalla luokittelemaan komponentin kunnon tilat ja siten havaitsemaan mahdolliset viat [9]. Voimalaitoskomponentilta saatava anturidata voi olla datajoukoltaan hyvin suuri, joten optimaalinen anturin valinta on keskeinen osa tehokasta optimointiongelman ratkaisua [10]. Optimaalisen anturin valinta voidaan tehdä käyttämällä korrelaatioanalyysia, jossa poistetaan epäolennaisten anturien data vertailemalla anturisignaalien välisiä suhteita [9,10]. Olennaiset anturit voidaan myös luokitella esimerkiksi käyttämällä päätöspuita [10]. Tämä koneoppimisprosessi esitetään kokonaisuudessaan kuvassa 12.



Kuva 12. Koneoppimisprosessin rakenne vikatilojen ennustamiseen. Perustuu lähteeseen [9].

Koneoppimisluokittelijoina vikatilojen ennustamisessa voivat toimia esimerkiksi ohjattuun oppimiseen perustuvat tukivektorikone, k-lähin naapuri ja Naiivi Bayes [9,10]. Näiden kouluttamista varten datajoukkona käytetään lämpövoimalaitoksen normaalin tilan dataa sekä vikatilain dataa, jotta tilojen poikkeamat voidaan kouluttaa luokittelijalle [10].

Luokittelijoiden suorituskykyjen on huomattu parantuneen optimaalisella anturin valinnalla esimerkiksi k-lähimmän naapurin menetelmässä jopa 2,6 % ja tukivektorikoneella 2,3 % [9]. Tämän lisäksi optimointiprosessin laskentatehoa on saatu vähennettyä, sillä optimaalisella anturin valinnalla anturidatan määrä pienenee merkittävästi. Esimerkiksi höyryvoimalan kattilaputkien vuodon ennustamisessa on saatu syötteeseen otettavien antureiden lukumäärää laskettua noin 44 %. [9] Turbiinin vikatilanteen ennustamiseen olennaisten antureiden määrää on saatu laskettua jopa noin 55 % [10].

Yksi kunnon arvioinnin menetelmistä lämpövoimalaitoksen komponenteille ovat erilaisilla antureilla toteutetut värinämittaukset [28]. Värinämittauksia voidaan mitata esimerkiksi turbiinin laakereista [29]. Ohjatun oppimisen koneoppimisalgoritmilla, kuten satunnaismetsällä ja tukivektorikoneella, mahdollistetaan voimalaitoksen kunnon valvonnan jäljittelevän synteettisen datan analysointi ja mahdollisten värinämittauksien poikkeamien avulla tilan luokittelu [30]. Luokiteltava data voidaan tuottaa koneoppimista hyödyntävällä laitoksen mallintamisella, jossa lasketaan ennustettujen ja todellisten arvojen väliset poikkeamat tilojen luokittelijoille [29].

Koneoppimista hyödyntävällä laitteistojen kunnon valvonnalla on havaittu esimerkiksi lämpövoimalaitoksen syöttövesipumpun toimintahäiriö ennen varsinaista hätätilannetta. Toimintahäiriö havaittiin pumpun datasta useita tunteja ennakkoon poikkeavuuden ilmaisimella, joka perustui koneoppimismallin tuottamiin poikkeamiin. [29] Samaan tapaan myös lämmönvaihtimien häiriöiden ennakointi on mahdollistettu koneoppimisella. Mikäli koneoppimisalgoritmi havaitsee datasta tilanteen, joka viittaa lämmönvaihtimen mahdolliseen häiriöön, se voi suositella ennakoivia huoltotoimenpiteitä operaattoreille ennen

varsinaista häiriötä. [17] Ennakoivan kunnossapidon hyödyntämisen menetelmät ja sovelluskohteet esitellään tiivistettynä taulukossa 3.

Taulukko 3. Koneoppimisen hyödyntäminen ennakoivassa kunnossapidossa tiivistettynä

Kohde	Menetelmä	Esimerkki
Prosessin tilojen arviointi ^{a,b}	Luokittelu ^{a,b}	Vikatilojen ennustaminen ^{a,b}
Anturidatan esikäsittely ^{a,b}	Luokittelu ^{a,b}	Käsiteltävän datan vähentäminen, luokittelijoiden suorituskyvyn parantaminen ^{a,b}
Tärinämittaukset ^{d,e}	Luokittelu ^{d,e}	Poikkeamien tunnistaminen, häiriöiden ennakointi ^{c,d,e}

a [9], b [10], c [17], d [29], e [30]

Nämä ovat varsin merkittäviä havaintoja, sillä varhaisessa vaiheessa mahdollisten hätätilojen havaitseminen kasvattaa laitoksen turvallisuutta sekä käyttövarmuutta. Lisäksi varhainen häiriön havaitseminen vähentää mahdollisia korjauskustannuksia, sillä hätätilanteen tapauksessa kustannukset voivat olla huomattavasti suuremmat.

5. JOHTOPÄÄTÖKSET

Tämän työn tavoitteena oli selvittää, mitä optimointikohteita lämpövoimalaitoksissa on, millaisia koneoppimisalgoritmeja on olemassa ja miten koneoppimista voidaan hyödyntää lämpövoimalaitosten käytönaikaisessa optimoinnissa. Tämä toteutettiin tässä työssä kirjallisuuskatsauksen muodossa.

Koneoppimisalgoritmit voitiin kirjallisuuskatsauksen perusteella jaotella niiden oppimistyyppin mukaan ohjattuun oppimiseen, ohjaamattomaan oppimiseen ja vahvistusoppimiseen. Näiden käyttö riippuu pitkälti siitä, mikä ongelman ratkaisussa on tavoitteena. Lämpövoimalaitosten käytönaikaisessa optimoinnissa optimointikohteiksi korostuivat prosessien tehokkuus sekä ennakoiva kunnossapito ja käyttövarmuus. Näihin kohteisiin voitiinkin esittää koneoppimisen hyödyntämistä, jolla voitiin mahdollistaa reaaliaikaisten prosessien analysointi ja mallintaminen dataan perustuen. Tämän lisäksi ennakoivaan kunnossapitoon koneoppimisen todettiin olevan kykenevä prosessien tilojen luokittelulla, mikä taas mahdollistaa laitoksen turvallisuuden, huollon ja kustannuksien edistämisen.

Koneoppimista hyödyntävän optimoinnin on todettu mahdollistavan voimalaitoskattilan lämpötehokkuuden parantamisen ja muodostuvien päästöjen vähentämisen samanaikaisesti monitavoiteoptimoinnilla. Lisäksi koneoppimista käyttämällä on mahdollistettu huoltojen ennakoiva havaitseminen, jolloin mahdollisten vakavampien häiriötilanteiden synty voidaan ehkäistä. Näiden havaintojen perusteella koneoppimisen voidaan todeta olevan hyvinkin merkittävä tutkimuskohde myös jatkossa, sillä nämä edistävät sekä laitoksen käyttövarmuutta, että laitosten kustannustehokkuutta.

Tämän työn esitetyt johtopäätökset perustuvat tieteellisiin lähteisiin sekä sovellusesimerkkeihin, mitkä tukevat tulosten luotettavuutta. Tässä työssä ei kuitenkaan toteutettu kokeellista osuutta eri koneoppimismenetelmien testaamiseksi, mikä rajoittaa työssä esitetyjen menetelmien toimivuuden arviointia voimalaitosympäristössä. Koneoppimisalgoritmit vaativat usein myös suuren määrän dataa toimiakseen oikein ja mallien mahdollinen ylisovittaminen voi tuottaa uusilla syötteillä epäluotettavaa tietoa, joten tutkimusta olisi hyvä tehdä näiden osalta lisää.

Esitettyjä havaintoja voidaan kuitenkin yleistää samankaltaisiin voimalaitosympäristöihin, kunhan prosessidataa on riittävästi käytettävissä ja mallien koulutus on tehokasta. Koneoppimisen hyödyntäminen ja lisääntyvä kiinnostus lämpövoimalaitosten käytönaikaisessa optimoinnissa mahdollistaa entistä tehokkaamman sekä turvallisemman laitosten toiminnan.

LÄHDELUETTELO

- [1] "Report-Global-Electricity-Review-2024.pdf". Viitattu: 23. tammikuuta 2025. [Verkossa]. Saatavissa: <https://ember-energy.org/app/uploads/2024/05/Report-Global-Electricity-Review-2024.pdf>
- [2] L. D. Blackburn, J. F. Tuttle, K. Andersson, J. D. Hedengren, ja K. M. Powell, "Dynamic machine learning-based optimization algorithm to improve boiler efficiency", *Journal of Process Control*, vsk. 120, ss. 129–149, joulu 2022, doi: 10.1016/j.jprocont.2022.11.002.
- [3] Z. Tang, X. Hu, ja J. Périaux, "Multi-level Hybridized Optimization Methods Coupling Local Search Deterministic and Global Search Evolutionary Algorithms", *Archives of Computational Methods in Engineering*, vsk. 27, nro 3, ss. 939–975, heinä 2020, doi: 10.1007/s11831-019-09336-w.
- [4] S. M. Sinha, *Mathematical Programming: Theory and Methods*. Chantilly, INDIA: Elsevier Science & Technology, 2006. Viitattu: 22. huhtikuuta 2025. [Verkossa]. Saatavissa: <http://ebookcentral.proquest.com/lib/tampere/detail.action?docID=404733>
- [5] M. Mohammed, M. B. Khan, ja E. M. Bashier, *Machine Learning*. Viitattu: 30. tammikuuta 2025. [Verkossa]. Saatavissa: <https://learning.oreilly.com/library/view/machine-learning/9781315354415/xhtml/intro.xhtml>
- [6] S. Dutt, S. Chandramouli, ja A. K. Das, *Machine Learning, 1st Edition by Pearson*. Viitattu: 30. tammikuuta 2025. [Verkossa]. Saatavissa: <https://learning.oreilly.com/library/view/machine-learning-1st/9789353067373/>
- [7] L. Han *ym.*, "Optimization of Circulating Fluidized Bed Boiler Combustion Key Control Parameters Based on Machine Learning", *Energies*, vsk. 16, nro 15, Art. nro 15, tammi 2023, doi: 10.3390/en16155674.
- [8] N. Li ja Y. Hu, "The Deep Convolutional Neural Network for NOx Emission Prediction of a Coal-Fired Boiler", *IEEE Access*, vsk. 8, ss. 85912–85922, 2020, doi: 10.1109/ACCESS.2020.2992451.
- [9] S. Khalid *ym.*, "Intelligent Steam Power Plant Boiler Waterwall Tube Leakage Detection via Machine Learning-Based Optimal Sensor Selection", *Sensors*, vsk. 20, nro 21, Art. nro 21, tammi 2020, doi: 10.3390/s20216356.
- [10] S. Khalid, H. Hwang, ja H. S. Kim, "Real-World Data-Driven Machine-Learning-Based Optimal Sensor Selection Approach for Equipment Fault Detection in a Thermal Power Plant", *Mathematics*, vsk. 9, nro 21, Art. nro 21, tammi 2021, doi: 10.3390/math9212814.
- [11] D. Sarkar, *Thermal Power Plant: Design and Operation*. Chantilly, UNITED STATES: Elsevier, 2015. Viitattu: 24. tammikuuta 2025. [Verkossa]. Saatavissa: <http://ebookcentral.proquest.com/lib/tampere/detail.action?docID=2193598>
- [12] M. Feidt, Toim., *Carnot Cycle and Heat Engine Fundamentals and Applications*. Basel, Switzerland: MDPI - Multidisciplinary Digital Publishing Institute, 2020.
- [13] W. Muhammad Ashraf *ym.*, "Optimization of a 660 MWe Supercritical Power Plant Performance—A Case of Industry 4.0 in the Data-Driven Operational Management. Part 2. Power Generation", *Energies*, vsk. 13, nro 21, Art. nro 21, tammi 2020, doi: 10.3390/en13215619.
- [14] "Fossil vs. Alternative Fuels - Energy Content". Viitattu: 7. helmikuuta 2025. [Verkossa]. Saatavissa: https://www.engineeringtoolbox.com/fossil-fuels-energy-content-d_1298.html
- [15] S. Zhong, X. Wang, H. Wu, Y. He, B. Xu, ja M. Ding, "Energy hub management for integrated energy systems: A multi-objective optimization control strategy based on distributed output and energy conversion characteristics", *Energy*, vsk. 306, s. 132526, loka 2024, doi: 10.1016/j.energy.2024.132526.

- [16] A. M. Kler ja P. V. Zharkov, "An effective method for optimization of continuous and discrete parameters of heat and power plants", *Thermophysics and Aeromechanics*, vsk. 25, nro 2, ss. 305–316, maaliskuu 2018, doi: 10.1134/S0869864318020166.
- [17] S. Wankhede, R. Lobo, ja P. Pesode, "Evaluating machine learning algorithm for real-time heat exchanger optimization and automatic issue detection device: experimental analysis", *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vsk. 18, nro 7, ss. 4409–4420, syyskuu 2024, doi: 10.1007/s12008-023-01709-7.
- [18] P. Wang ja Y. Liu, "Research on optimal scheduling of thermal power system in thermal power plant based on deep reinforcement learning", *Journal of Physics: Conference Series*, vsk. 2442, nro 1, s. 012009, helmikuu 2023, doi: 10.1088/1742-6596/2442/1/012009.
- [19] S. Kong, X. Yu, Z. Ling, P. Tang, ja N. Jin, "Analysis of Special Consequences and Losses of Equipment Accident in Thermal Power Plant", teoksessa *E3S Web of Conferences*, Les Ulis, France: EDP Sciences, 2021. doi: 10.1051/e3sconf/202125202018.
- [20] K. He ja H. Ding, "Prediction of NOx Emissions in Thermal Power Plants Using a Dynamic Soft Sensor Based on Random Forest and Just-in-Time Learning Methods", *Sensors*, vsk. 24, nro 14, s. 4442, 2024, doi: 10.3390/s24144442.
- [21] S. Marsland, *Machine Learning, 2nd Edition*. Viitattu: 2. huhtikuuta 2025. [Verkossa]. Saatavissa: <https://learning.oreilly.com/library/view/machine-learning-2nd/9781466583283/>
- [22] M. Ester, H.-P. Kriegel, ja X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise", *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*.
- [23] I. Halperin, "Reinforcement Learning and Stochastic Optimization: A Unified Framework for Sequential Decisions: by Warren B. Powell (ed.), Wiley (2022). Hardback. ISBN 9781119815051.", *Quantitative Finance*, vsk. 22, nro 12, ss. 2151–2154, joulukuu 2022, doi: 10.1080/14697688.2022.2135456.
- [24] H. Anead, K. Sultan, ja R. Abd-Kadhum, "Evaluation and Improvement Performance of a Boiler in a Thermal Power Plant Using Artificial Neural Network", *Engineering and Technology Journal*, vsk. 36, nro 6A, ss. 656–663, kesäkuu 2018, doi: 10.30684/etj.36.6A.10.
- [25] J. P. Mueller ja L. Massaron, *Deep Learning for Dummies*. Newark, UNITED STATES: John Wiley & Sons, Incorporated, 2019. Viitattu: 13. maaliskuuta 2025. [Verkossa]. Saatavissa: <http://ebookcentral.proquest.com/lib/tampere/detail.action?docID=5751857>
- [26] E. Butakov ja S. Abdurakipov, "The application of machine learning techniques to detect combustion modes in a pulverised coal boiler", *E3S Web of Conferences*, vsk. 459, s. 07012, 2023, doi: 10.1051/e3sconf/202345907012.
- [27] P. S. K. Lanka ja P. Raja Rao, "A Hybrid Approach for Energy Output Prediction through Integration of Supervised and Unsupervised Learning Strategies", teoksessa *2024 3rd International Conference for Advancement in Technology (ICONAT)*, syyskuu 2024, ss. 1–6. doi: 10.1109/ICONAT61936.2024.10775009.
- [28] L. Zhu, J. Dai, ja G. Bai, "Sensor Placement Optimization of Vibration Test on Medium-Speed Mill", *Shock and Vibration*, vsk. 2015, nro 1, s. 690196, 2015, doi: 10.1155/2015/690196.
- [29] A. A. Korshikova ja A. G. Trofimov, "Model for Early Detection of Emergency Conditions in Power Plant Equipment Based on Machine Learning Methods", *Thermal Engineering*, vsk. 66, nro 3, ss. 189–195, maaliskuu 2019, doi: 10.1134/S0040601519030042.
- [30] A. W. K. Fahmi, K. Reza Kashyzadeh, ja S. Ghorbani, "Smart maintenance strategies in the combined cycle power plant", *Journal of Computational Applied Research in Mechanical Engineering (JCARME)*, nro Online First, kesäkuu 2024, doi: 10.22061/jcarme.2024.10797.2415.