

Ilmastointijärjestelmän vikadiagnostiikka Menetelmät ja sovellukset

Satu Kärki

Sami Karjalainen

VTT Rakennustekniikka



ISBN 951-38-5452-3 (nid.)

ISSN 1235-0605 (nid.)

ISBN 951-38-5453-1 (URL: <http://www.inf.vtt.fi/pdf/>)

ISSN 1455-0865 (URL: <http://www.inf.vtt.fi/pdf/>)

Copyright © Valtion teknillinen tutkimuskeskus (VTT) 1999

JULKAISIJA – UTGIVARE – PUBLISHER

Valtion teknillinen tutkimuskeskus (VTT), Vuorimiehentie 5, PL 2000, 02044 VTT
puh. vaihde (09) 4561, faksi (09) 456 4374

Statens tekniska forskningscentral (VTT), Bergsmansvägen 5, PB 2000, 02044 VTT
tel. växel (09) 4561, fax (09) 456 4374

Technical Research Centre of Finland (VTT), Vuorimiehentie 5, P.O.Box 2000, FIN-02044 VTT, Finland
phone internat. + 358 9 4561, fax + 358 9 456 4374

VTT Rakennustekniikka, Rakennusfysiikka, talo- ja palotekniikka, Lämpömiehenkuja 3, PL 1804, 02044 VTT
puh. vaihde (09) 4561, faksi (09) 455 2408

VTT Byggnadsteknik, Byggnadsfysik, hus- och brandteknik, Värmemansgränden 3, PB 1804, 02044 VTT
tel. växel (09) 4561, fax (09) 455 2408

VTT Building Technology, Building Physics, Building Services and Fire Technology,
Lämpömiehenkuja 3, P.O.Box 1804, FIN-02044 VTT, Finland
phone internat. + 358 9 4561, fax + 358 9 455 2408

Toimitus Leena Ukoski

Libella Painopalvelu Oy, Espoo 1999

Kärki, Satu & Karjalainen, Sami. Ilmastointijärjestelmän vikadiagnostiikka. Menetelmät ja sovellukset [Fault diagnosis of air-conditioning systems. Methods and applications]. Espoo 1999, Valtion teknillinen tutkimuskeskus, VTT Tiedotteita – Meddelanden – Research Notes 1967. 92 s. + liitt. 2 s.

Avainsanat air conditioning, HVAC, air handling units, performance requirements, supply air, flow rate, evaluating, heating, capacity, fault diagnostic tools, FDD, software

Tiivistelmä

Ilmastointikoneen tärkeimmät suorituskykyvaatimukset liittyvät sillä tuotettavaan lämmitys- ja jäähdytystehoon, tuloilmavirtaan ja sen puhtauteen, energiataloudellisuuteen ja säätötekniiseen toimintaan. Monet järjestelmän elinkaaren aikaiset ratkaisut vaikuttavat järjestelmällä saavutettavaan lopulliseen suorituskykyyn. Tässä julkaisussa eritellään näiden ratkaisujen vaikutusta.

Diagnostisia menetelmiä ja työkaluja voidaan kehittää suorituskykykriteerien toteutumisen seurantaan. Jos näiden avulla todetaan, että järjestelmä ei toimi asetettujen vaatimusten rajoissa, havaitaan vika. Vian paikantamiseen on käytettävä muita menetelmiä. Tarkastellun ilmastointikoneen keskeisten suorituskykyvaatimusten perusteella kehitettiin käytännönläheisiä vikadiagnostiikkamenetelmiä. Menetelmät ovat yksinkertaisia ja helposti sovellettavissa käytäntöön eivätkä ne vaadi ylimääräistä instrumentointia. Julkaisussa esitetään menetelmät tarkasteltavan järjestelmän LTO-piirin tehojen seurantaan sekä lauhdelämmön ja kaukolämmön tehon seurantaan. Lisäksi käsitellään ominaiskäyrämenetelmän ja vikapuun käyttöä vikadiagnostiikassa.

Prosessin vikatietämys voidaan jäsentää vikapuun avulla. Vikapuita käytetään laajasti monilla aloilla, mutta yleensä vikapuissa esitetään vain vikojen välinen hierarkia, ei vikojen havaitsemiseen käytettäviä oireita, kuten tässä julkaisussa tehdään. Julkaisussa selvitetään sopivia työkaluja vikapuun kuvaaman asiantuntijatiedon käsittelyyn määriteltujen vaatimusten ja tarpeiden perusteella. Työkaluohjelmat jaoteltiin ohjelmoinnin perustyökaluihin, ohjelmointiympäristöihin, tietämyspohjaisiin työkaluohjelmiin, sumean logiikan kehitysohjelmiin, neuraaliverkko-ohjelmiin sekä muihin valmisohjelmiin. Jokaisesta työkaluohjelmatyypistä esiteltiin useita työkaluja kuvaamalla lyhyesti niiden keskeisimmät ominaisuudet.

Julkaisussa esitellään kolme ilmastointikoneen kunnan ja toiminnan seurantaan tarkoitettua prototyypisovellusta. Kehitetyt prototyypit ovat ominaiskäyräsovellus, ilmastointikoneen tehonäyttö ja vikapuusovellus. Tarkoituksena on demonstroida, millaisia toimintoja vikadiagnostiikassa hyödynnettävillä tietokonesovelluksilla voisi olla.

Kärki, Satu & Karjalainen, Sami. Ilmastointijärjestelmän vikadiagnostiikka. Menetelmät ja sovellukset [Fault diagnosis of air-conditioning systems. Methods and applications]. Espoo 1999, Valtion teknillinen tutkimuskeskus, VTT Tiedotteita – Meddelanden – Research Notes 1967. 92 p. + app. 2 p.

Keywords air conditioning, HVAC, air handling units, performance requirements, supply air, flow rate, evaluating, heating, capacity, fault diagnostic tools, FDD, software

Abstract

The most important performance requirements of air handling units are related to the heating and cooling capacity, the supply air flow rate and purity, the energy efficiency, and the control quality. Many decisions made during different life-cycle phases affect the final system performance. These decisions are discussed in this report.

Diagnostic tools and methods can be developed for monitoring the defined performance criteria. A fault is detected if, in some circumstances, it is noticed that the criteria are not fulfilled. In fault localization and diagnosis, other methods are used. Based on the defined performance requirements, practical methods for monitoring the system performance, and for fault detection and diagnosis (FDD) were developed. The methods are simple and use measurements which are always available from the considered system. These attributes are significant if the methods are applied in practice. This report describes the methods for monitoring the heat recovery unit, and the thermal heating powers of the waste heat and the district heating. In addition, using characteristic curves and fault-symptom trees for fault detection and diagnosis are discussed.

The fault information of a process can be defined using a fault tree. The fault trees are widely used in many fields. However, usually only the hierarchy between the faults are shown, not the symptoms which indicate that a fault has occurred as is done in this report. This report describes the process of selecting suitable software tools for implementation of expert knowledge of FDD-methods. The main focus is on the tools which could be used for implementing fault trees that include symptoms in all levels of the tree. First, the criteria and requirements of the software were defined. The software tools were classified into computer languages, language environments, knowledge-based tools, fuzzy logic, neural networks and other computer programs. An introduction of a wide range of software tools is given.

This report describes the prototype applications for monitoring the condition and operation of the air handling unit (AHU). The developed prototypes include applications for handling characteristic curves and fault-symptom trees, and monitoring AHU capacities. The objective of the prototype development was to demonstrate the type of functions the FDD applications could include.

Alkusanat

"Kehittyneet menetelmät ilmastointijärjestelmän säädön parantamisessa ja vika-diagnostiikassa" -projektin (eli ADMET-projektin) tavoitteena oli luoda ThermoNet-tekniikkakeskuksen säätö- ja diagnostiikkatoimintojen tuotekehitysedellytykset kehittämällä perusteita säätö- ja automaatiotekniikkaa hyödyntäville työkaluille ilmastointijärjestelmän toiminnan suunnitteluun ja parantamiseen. Projekti kuului Teknologian kehittämiskeskuksen (Tekesin) ThermoNet-tutkimusohjelmaan. Rahoittajina toimivat Tekes, Valtion teknillinen tutkimuskeskus (VTT) ja ABB Fläkt Oy.

Projekti jakautui viiteen toisistaan melko erilliseen tehtävään:

1. Talotekniikkajärjestelmien suorituskyky mittarit sekä niiden määrittämis- ja diagnostiikkamenetelmät elinkaaren eri vaiheissa
2. Menettelytavat talotekniikkajärjestelmien optimiajotavan määrittämiseen
3. Menetelmät ja ohjelmistot asiantuntijatiedon käsittelyyn
4. LON-tekniikan hyödyntäminen talotekniikkakokonaisjärjestelmien diagnostiikassa
5. Prototyypisovellukset.

Projektin johtoryhmä koottiin rahoittajien edustajista. Siihen kuuluivat Esko Virtanen (Tekes), Markku Virtanen (VTT Rakennustekniikka) vuoden 1997 ajan, Mikko Nyman (VTT Rakennustekniikka) vuoden 1998 ajan ja Juhani Hyvärinen (ABB Fläkt Oy).

Tutkimuksen projektipäällikkönä ja päätutkijana toimi diplomi-insinööri Satu Kärki. Lisäksi tutkimuksen toteuttamiseen osallistuivat diplomi-insinöörit Sami Karjalainen (päätekijä tehtävässä 1) ja Petri Pietarinen (päätekijä tehtävässä 4). Tämä tutkimusraportti käsittelee ADMET-projektin vikadiagnostiikkaosuuksia (tehtävät 1, 3 ja 5) ja sen on projektin työraporttien pohjalta koonnut Satu Kärki.

Kiitämme kaikkia tutkimukseen osallistuneita.

Espoossa 1999

Tekijät

Sisällysluettelo

| | |
|---------------------------------------------------------------------|----|
| TIIVISTELMÄ..... | 3 |
| ABSTRACT..... | 4 |
| ALKUSANAT | 5 |
| | |
| 1 JOHDANTO | 8 |
| 1.1 Tutkimuksen lähtökohdat | 8 |
| 1.2 Tarkasteltava järjestelmä | 9 |
| 1.2.1 Market-sovellus..... | 9 |
| | |
| 2 SUORITUSKYKYVAATIMUKSET JA DIAGNOSTIIKKAMENETELMÄT | 12 |
| 2.1 Suorituskyky- ja laatutekijät | 13 |
| 2.1.1 Sisäilmastoon liittyvät tekijät..... | 13 |
| 2.1.2 Energiankulutus | 16 |
| 2.1.3 Muut mahdolliset suorituskykytekijät..... | 19 |
| 2.2 Suorituskykyvaatimukset..... | 21 |
| 2.2.1 Sisäilmaston toteutumiseen liittyvät vaatimukset | 22 |
| 2.2.2 Energiatehokkuus..... | 26 |
| 2.2.3 Sääntötekninen toiminta..... | 27 |
| 2.3 Elinkaarivaiheiden vaikutus suorituskykyyn | 27 |
| 2.3.1 Suunnittelu | 28 |
| 2.3.2 Valmistus | 30 |
| 2.3.3 Asennus ja käyttöönotto..... | 32 |
| 2.3.4 Käyttö..... | 34 |
| 2.4 Mahdollisia diagnostiikkamenetelmiä..... | 34 |
| 2.4.1 LTO-piirin tehojen seuranta..... | 36 |
| 2.4.2 Lauhdelämmön ja kaukolämmön tehon seuranta..... | 38 |
| 2.4.3 Ominaiskäyrämenetelmä..... | 42 |
| 2.4.4 Säädön toiminnan seuranta | 42 |
| 2.4.5 Vikapuu..... | 43 |
| 2.4.6 Muita vikadiagnostiikkamenetelmiä..... | 46 |
| 2.5 Yhteenvedo luvusta 2 | 46 |
| | |
| 3 MENETELMÄT JA OHJELMISTOT ASiantuntijätiedon KÄSITTELYYN | 48 |
| 3.1 Vikadiagnoosijärjestelmän yleisrakenne | 48 |
| 3.2 Tarpeet ja vaatimukset..... | 51 |

| | |
|----------------------------------------------|----|
| 3.3 Työkaluohjelmia..... | 54 |
| 3.3.1 Ohjelmoinnin perustyökalut..... | 54 |
| 3.3.2 Ohjelmointiympäristöt | 56 |
| 3.3.3 Tietämispohjaiset työkaluohjelmat..... | 58 |
| 3.3.4 Sumean logiikan kehitysohjelmat | 66 |
| 3.3.5 Neuraaliverkko-ohjelmat..... | 68 |
| 3.3.6 Muut valmisohjelmat | 72 |
| 3.4 Yhteenveto luvusta 3 | 73 |
| 4 PROTOTYYPPISOVELLUKSET | 76 |
| 4.1 Ominaiskäyräsovellus..... | 76 |
| 4.1.1 Datan puhdistaminen..... | 77 |
| 4.1.2 Toteutettu Excel-sovellus..... | 78 |
| 4.2 Ilmastointikoneen tehonäyttö..... | 79 |
| 4.3 Vikapuusovellus | 83 |
| 4.4 Yhteenveto luvusta 4 | 85 |
| 5 JOHTOPÄÄTÖKSET | 86 |
| | |
| LÄHDELUETTELO | 88 |

Liite 1. Ohjelmistotuotteiden WWW-sivut

1 Johdanto

1.1 Tutkimuksen lähtökohdat

LVIS-2000-tutkimusohjelman LVI-prosessien prosessidiagnostiikan tutkimuksessa kehitettiin menettely LVI-järjestelmien toiminnan reaaliaikaiseen diagnostisointiin sekä kuvattiin rakennusautomaatiojärjestelmän osana toimivan vikadiagnoosijärjestelmän periaatteellinen rakenne. Lisäksi tutkittiin LVI-prosessien ja -järjestelmien säätöratkaisuja, jolloin todettiin, että väärät asetusarvot ja ohjaustavat voivat kasvattaa energian kulutusta ja huonontaa samanaikaisesti ilmastointijärjestelmän lämpöviivytyttä suhteessa 1:2.

Rakennusten vikadiagnoosia tutkittiin myös kansainvälisessä IEA Annex 25 -projektissa, jossa kehitettiin menetelmiä vikojen havaitsemiseen ja paikantamiseen. Kehitetyt menetelmät perustuvat tarkasteltavan prosessin oikeaa toimintaa kuvaavien yksinkertaisten mallien reaaliaikaiseen käyttöön. Menetelmäkehitystä tehtiin simulointia ja laboratoriomittauksia hyödyntäen.

Myös Raket-tutkimusohjelmassa ja Tekesin soveltavan tutkimuksen hankkeissa on kehitetty LVI-järjestelmien vikadiagnoosi- ja toiminnan seurantamenetelmiä sekä näiden toimintaa ja hyödyntämistä demonstroivia ohjelmistosovelluksia.

Tutkimustulosten hyödyntäminen käytännön tuotekehitystyössä vaatii varsinaisen tutkimuksellisen ajatuksen kehittämisen ja testaamisen lisäksi jatkokehittämistä, jossa tutkimustuloksesta tehdään työkalu jonkin ongelman ansiokkaaseen ratkaisuun. Tutkimustulokset kannattaa jalostaa tälle asteelle vasta siinä vaiheessa, kun niiden hyödyntäminen on ajankohtaista. ThermoNet-järjestelmien käyttövarmuusvaiheessa tunnistettiin joitakin menetelmiä ja lähestymistapoja, joita voidaan hyödyntää tuotekehityksessä. Tämän tutkimuksen tavoitteena oli näiden menetelmien jatkokehittäminen tuotekehityksen pohjaksi.

Käytännön tuotekehitystyötä ajatellen oli tärkeää, että käytännölliset ratkaisut seuraaviin vikadiagnostiikkaan liittyviin asioihin kehitettiin, ennen kuin tutkimustuloksia alettiin hyödyntää:

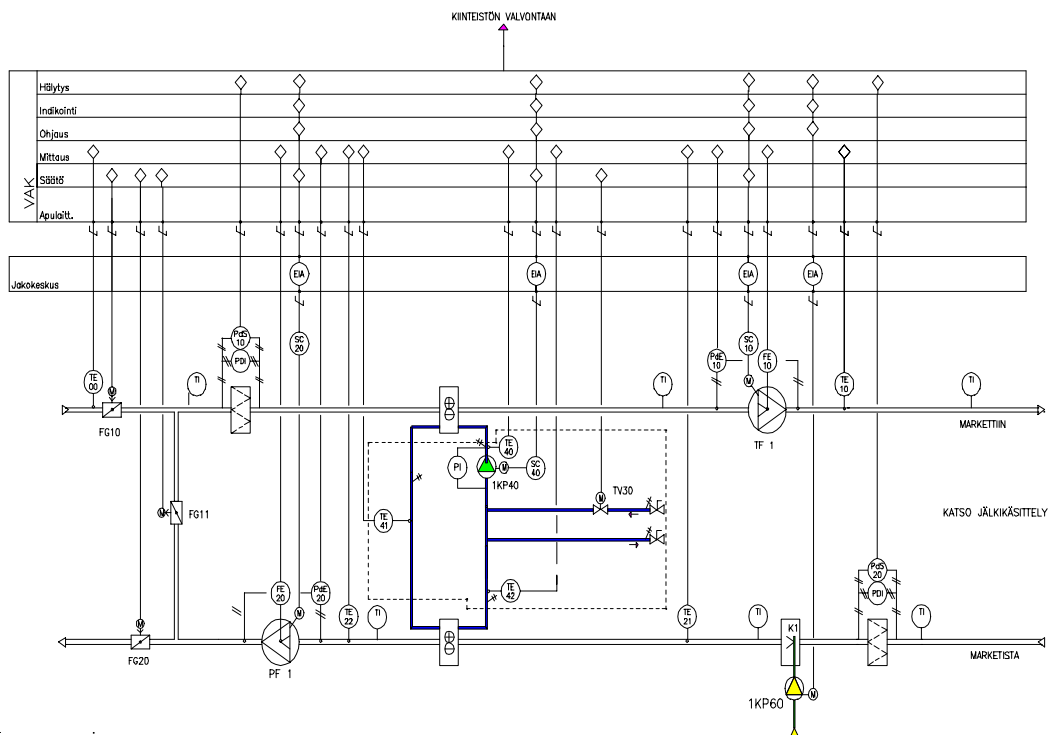
- sopiva metodiikka ominaiskäyrien tai muun vastaavan tiedon käsittelyyn
- saatavilla olevan tehon laskentatapa ja
- ohjelmistotyökalut asiantuntijatiedon (esimerkiksi vikapuun) käsittelyyn.

1.2 Tarkasteltava järjestelmä

ThermoNet® on ABB:n kehittämä talotekniikkajärjestelmä, joka soveltuu erityyppisille rakennuksille ja kiinteistöille. ThermoNet®-teknologia perustuu ilmankäsittelyyn ja -säättöön liittyvien toimintojen sekä laitteistojen yhdistämiseen. ThermoNet-talotekniikkajärjestelmän keskeiset innovaatiot liittyvät ilmastointikoneeseen, jota nimitetään myös tekniikkakeskukseksi. Siinä lämmityksen, jäähdytyksen ja poistoilman sekä muiden jätelämpöjen talteenotto on yhdistetty.

1.2.1 Market-sovellus

ADMET-projektissa tarkasteltiin esimerkkinä ThermoNet-tekniikkakeskuksen market-sovellusta. Muissa käyttökohteissa ThermoNet-järjestelmän kokoonpano on hieman erilainen, mutta tässä esitettävät asiat ovat pääosin sovellettavissa myös niihin.



Kuva 1. Periaatekuva ThermoNet-tekniikkakeskuksesta.

ThermoNet-tekniikkakeskuksen ilmankäsittelytoimintoja ovat

- suodatus
- lämmitys
- lämmöntalteenotto
- epäsuora kostutusjäähdytys (tarvittaessa)
- jäähdytys (tarvittaessa)

- ilmavirtojen tarpeenmukainen ohjaus
- palautusilman käyttö.

ThermoNet-tekniikkakeskuksen keskeisin osa on epäsuora lämmöntalteenotto piiri, johon on yhdistetty lämmitys ja tarvittaessa myös jäähdytys. Lämmönsyöttö LTO-piiriin voi tapahtua suoralla kytkennällä (kuva 1), jolloin sama lämmönsiirtoneste virtaa sekä LTO-piirissä että lämmönsiirtoverkostossa. Kytkentä voi olla myös epäsuora, jolloin lämmönsiirtoverkoston ja LTO-piiriin nesteet virtaavat erillisissä nestepiireissä. Tällöin välissä on lämmönsiirrin.

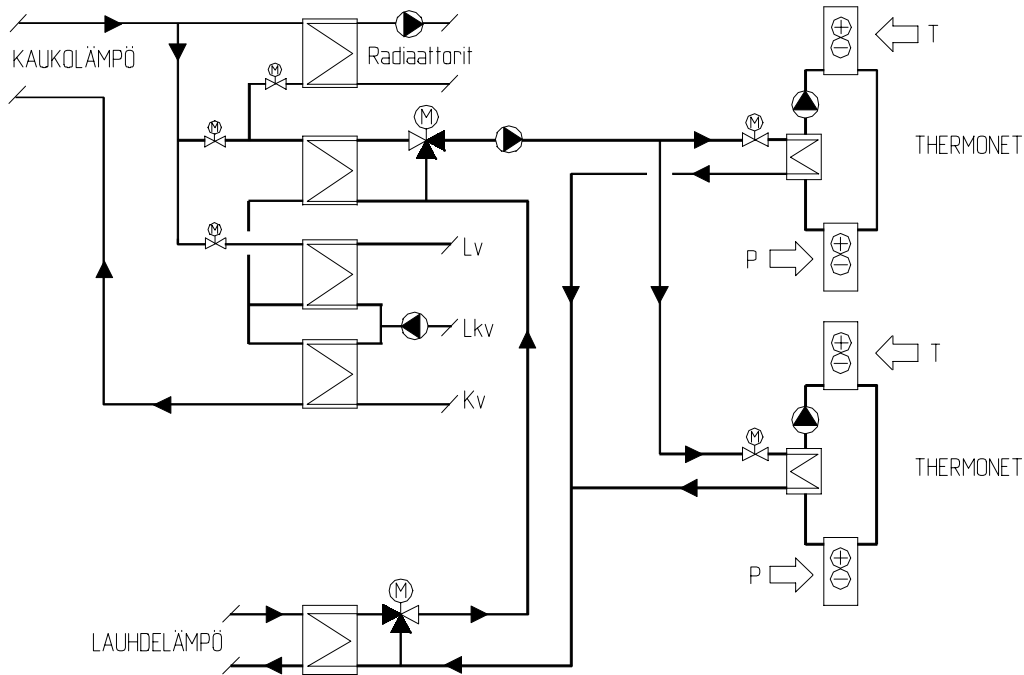
Puhaltimet on varustettu kierroslukusäädöllä. Tuloilmavirtaa ohjataan huoneen tehotarpeen mukaan ja ulkoilmavirtaa ilmanlaadun mukaan. Market-sovelluksessa ulkoilmavirtaa ohjataan sisäilman hiilidioksidipitoisuuden mukaan.

Marketin pääasiallinen lämmönjakotapa on ilmalämmitys. Aputiloissa on radiaattorilämmitys. Tuulikaapeissa käytetään kierrätysilmakojeita. Marketissa ilmalämmitysjärjestelmän tuloilman lämpötilaa ja ilmavirtaa ohjataan huonelämpötilan funktiona. Huonelämpötilan laskiessa tuloilman lämpötilan asetusarvo kasvaa. Jos tuloilman lämpötilan asetusarvo saavuttaa maksiminsa, kasvatetaan ilmavirtaa. Toiminta on vastaava jäähdytystilanteessa, mutta tuloilman lämpötilan asetusarvo asetetaan minimiin, minkä jälkeen kasvatetaan tarvittaessa ilmavirtaa. Periaatteena on siis tuottaa sekä ilmalämmitys- että ilmajäähdytysteho mahdollisimman pienellä ilmavirralla ja suurella lämpötilaerolla.

Pieniä kohteita lukuun ottamatta käytössä on yleensä useita tekniikkakeskuksia. Tällöin osa koneista on ilmalämmityskoneita ja osa ilmanvaihtokoneita. Ilmalämmityskone käy pääasiassa palautusilmalla ja ilmanvaihtokone ulkoilmalla.

Lämmönlähteet ja lämmönjako

Ilmanvaihdon lämmitykseen käytetään poistoilman lämpöä, kylmälaitteiden lauhdelämpöä ja täydentävänä energialähteenä esimerkiksi kaukolämpöä.



Kuva 2. Periaatekuva ThermoNet-tekniikkakeskuksen liittymisestä lämmönsiirtoverkostoon.

LTO-piirin pumppua ohjataan niin, että talteenotettu lämpöteho on maksimissaan eri toimintatilanteissa. Mikäli poistoilmasta talteenotettu lämpö ei riitä, käytetään ensin kylmälaitteiden lauhdelämpöä ja vasta sitten kaukolämpöä. Kaukolämpöä käytettäessä hyödynnetään ensin radiaattoripiiristä palaavan veden lämpö, ja mikäli tämä ei riitä, käytetään suoraan kaukolämpöä (kuva 2). Lämmönsiirtoverkoston mitoituslämpötila valitaan käyttökohteen mukaan. Menonesteen lämpötila voidaan kompensoida ulkolämpötilan mukaan.

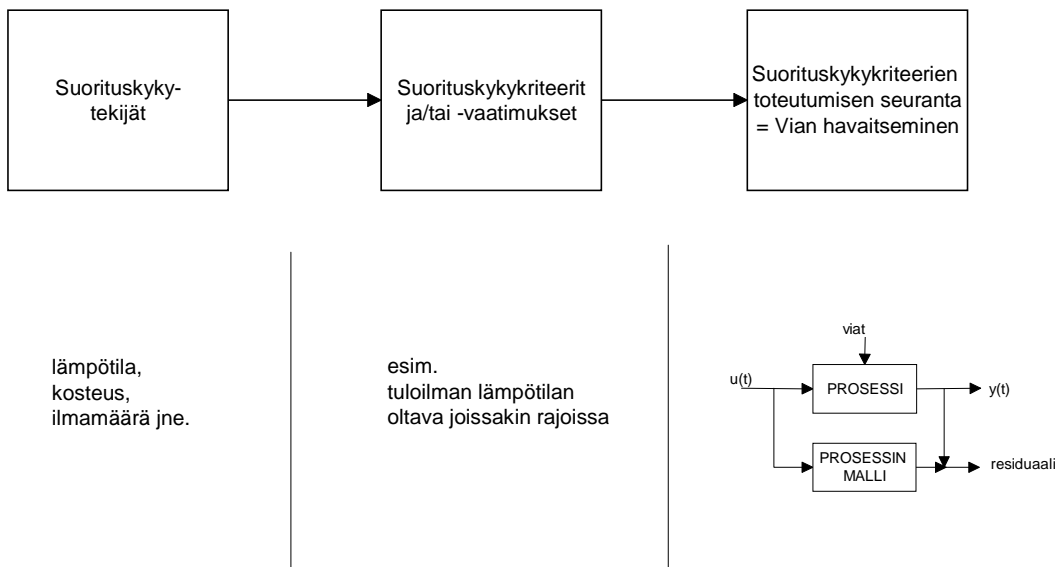
Jäähdytysteho tuotetaan ensisijaisesti epäsuoralla kostutusjäähdytyksellä. Mikäli tämä ei riitä, lisjäähdytystehoa voidaan syöttää ThermoNet-piiriin samalla tavalla kuin lisälämpöäkin.

2 Suorituskykyvaatimukset ja diagnostiikkamenetelmät

Tämän luvun tarkoituksena on esittää tarkasteltavan ilmastointijärjestelmän keskeiset suorituskykyvaatimukset sekä suorituskykytekijöiden vaihtoehtoisia diagnostiikkamenetelmiä.

Teknisellä suorituskyvyllä tarkoitetaan sitä, kuinka hyvin järjestelmä suoriutuu sille asetetuista teknisistä vaatimuksista eri käyttöolosuhteissa.

Kuva 3 esittää, miten suorituskykytekijät liittyvät vikadiagnostiikkaan (Kärki 1997). Kuvan yläosa kuvaa menetelmää, jolla suorituskykytekijöistä päädytään vian havaitsemiseen. Laatikoiden alla on annettu esimerkkejä ko. laatikon mahdollisesta sisällöstä. Siis kun suorituskykytekijät on määritetty, niille voidaan asettaa vaatimuksia. Näiden vaatimusten toteutumista seurataan. Jos jossakin tilanteessa todetaan, että kriteerit eivät toteudu, on kyseessä vian havaitseminen. Vian havaitsemisen jälkeen vika voidaan paikallistaa esimerkiksi vikapuiden avulla.



Kuva 3. Suorituskykytekijöiden liittyminen vikojen havaitsemiseen.

Järjestelmän suorituskykyvaatimuksia analysoimalla voidaan selvittää, millaisia kriteerejä järjestelmän toiminnan tulisi täyttää. Näiden perusteella määritetään, millaisia vikoja ja toimintahäiriöitä tulisi seurata ja pyrkiä havaitsemaan. Suorituskykykriteereitä voidaan joko sellaisenaan tai niihin liittyvien yhteyksien avulla käyttää apuna havaittaessa järjestelmän tai sen osien vikoja. Lisäksi suorituskyky-

vaatimusten perusteella on helppo kehittää käyttäjää avustavia järjestelmän oikean toiminnan seurantaan soveltuvia työkaluja.

2.1 Suorituskyky- ja laatutekijät

Ilmanvaihto- ja ilmalämmitysjärjestelmien keskeisinä tavoitteina ovat hyvä sisäilman laatu, alhainen energiankulutus, alhaiset käyttökustannukset ja hyvä käyttövarmuus. Vertailuarvoja antavat lait, määräykset, suositukset, ohjeet ja suunnittelu.

2.1.1 Sisäilmastoon liittyvät tekijät

Hyvä sisäilmasto on yksi rakentamisen keskeisistä tavoitteista. Sisäilmaston laatuun vaikuttavat lämmitys-, ilmanvaihto- ja ilmastointilaitteet, rakennustekniikka, rakennustöiden tekeminen ja käytetyt materiaalit sekä rakennuksen käyttö ja kunnossapito (Suomen Sisäilmayhdistys 1995). Hyvä sisäilmasto edellyttää siihen vaikuttavien asioiden huomioon ottamista suunnittelun, rakentamisen ja käytön kaikissa vaiheissa.

Sisäilmaston laatua koskevia määräyksiä, ohjeita ja luokituksia esitetään useissa julkaisuissa. Suomen rakentamismääräyskokoelman osa D2, Rakennusten sisäilmasto ja ilmanvaihto (Ympäristöministeriö 1987) antaa määräyksiä ja ohjeita sisäilman laadusta.

Sisäilmastoluokitus (Suomen Sisäilmayhdistys 1995) käsittelee lämpöoloja, melutasoa, ilmanvaihtoa ja ilman epäpuhtauksia. Siinä esitetään tärkeimmät lämmitys-, ilmanvaihto- ja ilmastointilaitteiden suunnittelussa tarvittavat sisäilmastosuureiden arvot, joita voidaan käyttää vertailuarvoina arvioitaessa sisäilman laatua. Sisäilmastoluokitukset ovat kolmiportaisia. Sisäilmastoluokka S1 vastaa parasta laatua ja luokka S3 lähinnä nykyistä viranomaistasoa.

Sisäilmastovaatimusten toteutumista voidaan arvioida myös vertaamalla toteutunutta sisäilmastoa suunnitteluarvoihin.

Sisäilmastovaatimukset voivat olla erilaisia rakennuksen eri osissa. Tarkasteltava market-sovellus on jaettu useisiin eri säätövyöhykkeisiin, joilla on erilaisia sisäilmastovaatimuksia. Säätövyöhykkeet ovat: käyttötavaraosasto, kassa-alue, ulkoseinävyöhyke, hedelmä- ja vihannesosasto ja muu elintarvikeosasto.

Sisäilman lämpötila

Rakentamismääräyskokoelman osa D2 (Ympäristöministeriö 1987) määrää lämpöoloista seuraavasti: "Huonetilan oleskeluvyöhykkeen lämpötilan tulee olla hallittu niin, ettei viihtyisyys tai suorituskyky huomattavasti alene eikä energiaa kulu tarpeettomasti." Myymälän ilman lämpötilan ohjearvo on 18 °C operatiivisen lämpötilan ohjearvon ollessa 15 °C. Kiinteissä työpisteissä (esimerkiksi kassat) ilman lämpötilan ohjearvo on 21 °C ja operatiivisen lämpötilan ohjearvo 20 °C.

Rakentamismääräyskokoelman ohjeiden mukaan oleskeluvyöhykkeen lämpötila ei saa ylittää arvoa 27 °C. Kun ulkolämpötila on yli 22 °C, voi sisäilman lämpötila ylittää ulkolämpötilan viiden tunnin keskiarvon korkeintaan 5 °C:lla. Operatiivisella lämpötilalla tarkoitetaan käytännössä keskimääräisen säteilylämpötilan ja ilman lämpötilan keskiarvoa.

Sisäilmastoluokitus (Suomen Sisäilmayhdistys 1995) ei anna tavoitearvoja erikseen erityyppisille rakennuksille, vaan luokitus koskee kaikkia tavanomaisia työ- ja asuintiloja. Sisäilman lämpötilan tavoitearvo on luokassa S3 talvella 20 - 24 °C ja kesällä 22 - 27 °C (35 °C). Kesällä lämpötila ei saa olla missään tilanteessa yli 35 °C:n. Kun ulkolämpötila on alle 15 °C, huonelämpötila ei saa olla yli 27 °C:n. Parhaan laadun luokassa, S1, sisäilman lämpötilan tavoitearvo on talvella 21 - 22 °C ja kesällä 22 - 25 °C. Tämän luokan suunnitteluarvoissa edellytetään lämpötilan huonekohtaista säädettävyyttä (± 2 °C).

Työsuojeluhallituksen päätöksen mukaan työnantaja on velvollinen huolehtimaan teknisin toimenpitein siitä, että sisäilman lämpötila pysyy alle 28 °C:ssa, kun ulkoilman lämpötila on alle 25 °C.

Tarkasteltavassa market-sovelluksessa käytetyt mitoitusarvot esitetään taulukossa 1.

Taulukko 1. Sisäilman lämpötilan suunnitteluarvot marketissa.

| Tila | Lämmityskaudella | Kesällä (maksimi) |
|---------------------------------------------------|------------------|--------------------------------------------|
| Elintarvikeosasto | 18 - 20 °C | 21 °C |
| Käyttötavaraosasto | 20 - 22 °C | 25 °C |
| Pien- ja erikoismyymälät, leipomot, konttorit ym. | 21 °C | 25 °C (tapauskohtaisesti 21 - 26 °C) |
| Varastot, tavarantoimitus yms. | 19 - 21 °C | 25 - 27 °C |

Sisäilman kosteus

Rakentamismääräyskokoelman osa D2 (Ympäristöministeriö 1987) määrää sisäilman kosteudesta seuraavasti: "Sisäilman kosteuden tulee pysyä käyttötarkoituksen mukaisissa arvoissa, joiden mukaan rakenteet ja ilmanvaihtolaitos on suunniteltava. Kosteuden tiivistyminen rakenteisiin tai ilmanvaihtolaitokseen ei saa aiheuttaa kosteusvaurioita eikä mikro-organismien kasvua tai muuta terveydellistä haittaa."

Sisäilmastoluokituksen (Suomen Sisäilmayhdistys 1995) tavoitearvo on luokassa S1 talvella 25 - 45 % ja kesällä 30 - 60 %. Luokissa S2 ja S3 ei ole asetettu vaatimuksia.

Marketissa on vähän kosteuskuormia, eikä sisäilman kosteus ole marketin tärkeimpiä sisäilmastotekijöitä. Sen sijaan esimerkiksi uimahallissa ulkoilmavirtaa ohjataan halli-ilman kosteuden mukaan.

Tarkastellulle market-järjestelmälle ei ollut suunnitteluarvoja sisäilman kosteudelle.

Sisäilman puhtaus

Rakentamismääräyskokoelman osa D2 (Ympäristöministeriö 1987) määrää sisäilman puhtaudesta seuraavasti: "Sisäilmassa ei saa esiintyä haitallisessa määrin kaasumaisia tai hiukkasmuodossa olevia epäpuhtauksia eikä mikro-organismeja." D2 antaa ohjearvoja eri epäpuhtauksien pitoisuuksille. Hiilidioksidipitoisuuden tulee alittaa 2 500 ppm, josta ihmisperäisen hiilidioksidin osuus on enintään 1 500 ppm. Mikäli ulkoilmavirtoja säädetään hiilidioksidipitoisuuden perusteella, säätöarvona käytetään korkeintaan pitoisuutta 800 ppm.

Muiden epäpuhtauksien pitoisuus saa D2:n ohjeiden mukaan olla korkeintaan 1/10 työsuojeluhallituksen tiedotteen mukaisista haitalliseksi tunnetuista pitoisuuksista. Mikäli esiintyy useita haitallisiksi tunnettuja aineita, D2 esittää kaavan, jolla arvioidaan yhteisvaikutusta.

Sisäilmastoluokitus (Suomen Sisäilmayhdistys 1995) antaa tavoitearvoja ammoniakkipitoisuudelle, formaldehydipitoisuudelle, haihtuvien orgaanisten aineiden (TVOC) kokonaispitoisuudelle, hajuvoimakkuudelle, hiilidioksidipitoisuudelle, hiilimonoksidipitoisuudelle, otsonipitoisuudelle, pölypitoisuudelle ja radonpitoisuudelle. Sisäilmastoluokituksen tavoitearvo hiilidioksidipitoisuudelle luokassa S1 on alle 1 000 ppm, luokassa S2 1 250 ppm ja luokassa S3 1 500 ppm.

Tarkasteltavan market-järjestelmän elintarvikeosaston ja käyttötavaraosaston hiilidioksidipitoisuuden yläraja on 1 000 ppm.

2.1.2 Energiankulutus

Rakennuksen energiankulutus

Rakennuksen energiankulutuksen tavoitearvon laskemiseen on useita eri vaihtoehtoja. Tavoitearvo voidaan laskea Suomen rakentamismääräyskokoelman osan D5 (Ympäristöministeriö 1985) esittämällä tavalla. D5-laskenta voidaan tehdä vuosi- tai kuukausitasolla. Vastaavanlaisia laskentatapoja on useita muitakin, esimerkiksi Eurooppalaisen standardisointijärjestön CEN:n menetelmä.

Rakennuksen energiankulutus saadaan arvioitua nopeasti esimerkiksi VTT Rakennustekniikassa kehitetyllä WinEtana-ohjelmalla. Tämä ohjelma perustuu RakMk:n D5:een tietyin muutoksin. WinEtana käyttää hyväkseen oletusarvoja erityyppisille ja eri-ikäisille rakennuksille, joten karkean arvon saa helposti. Käyttäjä pystyy tarkentamaan laskentaa muuttamalla ohjelman ehdottamia arvoja.

Simulointiohjelmistojen (esimerkiksi TRNSYS, DOE, HVACSIM+) käyttäminen on työläämpää, mutta rakennus voidaan mallintaa niillä hyvinkin yksityiskohtaisesti. Rakennus voidaan esimerkiksi jakaa useisiin vyöhykkeisiin. Simulointiohjelmat mahdollistavat rakennuksen lämpöteknisen käyttäytymisen laskemisen eri toimintatilanteissa (kuten jäähdytyksessä), koska lämmön varastoituminen seiniin otetaan huomioon. Laskennan tarpeet määräävät käytettävän mallin tarkkuuden. Tarkempi malli vaatii enemmän lähtötietoja. Mallin tarkkuuden lisäksi tulosten tarkkuuteen vaikuttaa olennaisesti lähtöarvojen tarkkuus. Lämpötekniset simulointiohjelmat eivät yleensä ota huomioon lämpötilan kerrostumista, vaan ne käsittelevät huoneilman lämpötilaa yhtenä lämpötilana eli olettavat huoneilman täysin sekoittuneeksi.

Rakennuksen energiankulutuksen vertailuarvoina on käytetty paitsi em. laskennallisin menetelmin saatuja arvoja myös vertailuarvoja toteutuneista rakennuksista. Ominaisenergiankulutus (kWh/r-m^3) kertoo rakennuksen energiankulutuksen rakennuskuutiota kohti. Vertailu tilastollisiin arvoihin ei ole kuitenkaan aina suositeltavaa, sillä tällöin ei oteta huomioon kyseisen rakennuksen ominaispiirteitä.

Tarkasteltavan ilmastointikoneen energiankulutus

Tarkasteltavan ilmastointikoneen tehontarve riippuu mm. ulkoilman lämpötilasta, poistoilman lämpötilasta (entalpiasta), tuloilman lämpötilan asetusarvosta lämmöntalteenoton jälkeen, tulo- ja poistoilmavirroista, palautusilmaosuudesta sekä epäsuoran LTO:n hyötysuhteesta.

Ilmastointikoneen energiankulutuksen vertailuarvo saadaan järjestelmäsuunnittelusta. Energiankulutuksen vertailuarvoja voidaan tuottaa myös simuloimalla. Lisäksi simuloimalla voidaan selvittää esimerkiksi asetusarvojen valinnan tai säätö- ja ohjaustapojen vaikutusta energiankulutukseen.

Nyky menetelmin toteutunutta energiankulutusta seurataan kuukausittain. Tässä työssä esitetään menetelmä, jolla tarkasteltavan ilmastointikoneen energiankulutusta voitaisiin seurata reaaliajassa.

Puhaltimien energiankulutus

Puhaltimen tehontarve (kW) lasketaan seuraavasti:

$$P = \frac{q_v \Delta p_{tot}}{\eta_{tot}}, \quad (1)$$

jossa

q_v on ilmavirta (m^3/s)
 Δp_{tot} paine-ero (kPa)
 η_{tot} kokonaishyötysuhde (-).

Puhaltimelta tarvittava paineenkorotus riippuu painehäviöistä kanavistossa, patterissa, suodattimessa, pääte-elimissä jne. Puhallin valitaan siten, että sen toimintapiste eli puhaltimen ja laitoksen ominaiskäyrien leikkauspiste on mahdollisimman lähellä puhaltimen parasta hyötysuhdealuetta.

Kokonaishyötysuhde on puhaltimen, sähkömoottorin ja välityksen hyötysuhteiden tulo. Puhaltimen kokonaishyötysuhteen vähimmäisarvoja esitetään Suomen rakentamismääräyskokoelman osassa D2 (Ympäristöministeriö 1987). Esimerkiksi 1 kW:n puhaltimen kokonaishyötysuhteen tulee olla vähintään 0,35 ja 10 kW:n puhaltimen 0,52.

Koska puhaltimen painehäviöt ovat verrannollisia pyörimisnopeuden toiseen potenssiin, tehontarve on verrannollinen pyörimisnopeuden kolmanteen potenssiin.

$$P_x / P_0 = (n_x / n_0)^3 \quad (2)$$

Tunnusluku tulo- ja poistoilmakoneen puhaltimien energiatehokkuudelle lasketaan seuraavasti:

$$\frac{P_{tulo} + P_{poisto}}{q_{v,max}}, \quad (3)$$

jossa

P_{tulo} on tuloilmapuhaltimen sähköteho (kW)

P_{poisto} poistoilmapuhaltimen sähköteho (kW)

$q_{v,max}$ suurempi ilman tilavuusvirta (m^3/s).

Taulukossa 2 on esitetty puhaltimien energiatehokkuusluokittelu (Shemeikka et al. 1996) tulo-poistojärjestelmälle. Luokittelu koskee koko IV-järjestelmää, jossa tulo- ja poistopuhaltimien tehot on jaettu rakennuksen nimellisilmavirralla. Mikäli käyttöaika on vähintään 2 500 h/a (toimistokäyttöaika), tulisi käyttää vähintään luokan D järjestelmää. Luokan A saavuttaminen ei luultavasti ole taloudellisesti kannattavaa nykyisillä sähköenergian hinnoilla.

Pohjoismaisen LVI-alan yhdistyksen, SCANVAC:n, luokittelu on esitetty taulukossa 3. Tehontarve lasketaan jakamalla puhaltimien yhteenlaskettu sähkönkulutus rakennuksen nimellisilmavirralla.

Puhaltimen toiminnan vertailukohtana voidaan käyttää valmistajan taulukoita. Valmistaja ilmoittaa puhaltimelle ominaiskäyrän, joka kertoo puhaltimen paineenkorotuksen ja tilavuusvirran riippuvuuden toisistaan. Valmistajan antamista tiedoista selviää puhaltimen sähkötehotarve kussakin toimintapisteessä.

Taulukko 2. Tulo-poistojärjestelmän puhaltimien energiatehokkuusluokittelu (Shemeikka et al. 1996).

| Luokka | Tehontarve kW/(m^3/s) |
|--------|---------------------------|
| A | < 1,0 |
| B | 1,0 - 1,5 |
| C | 1,5 - 2,0 |
| D | 2,0 - 2,5 |
| E | 2,5 - 3,0 |
| F | 3,0 - 4,0 |
| G | > 4,0 |

Taulukko 3. Puhaltimien sähkötehokkuusluokittelu (Svenska Inneklimatinstitutet 1991).

| Luokka | Tehontarve kW/(m ³ /s) |
|-------------------|--------------------------------------|
| VAS 1500 | < 1,5 |
| VAS 2500 | < 2,5 |
| VAS 4000 | < 4,0 |
| ei VAS luokitusta | > 4,0 |

Pumppujen energiankulutus

Pumpun tehontarve lasketaan kuten puhaltimenkin (kaava 1). Kiinteistöissä käytettävien pumppujen nostokorkeudet ovat yleensä 1 - 5 m (vastaa noin 10 - 50 kPa). Myös pumppujen energiankulutus on verrannollinen pyörimisnopeuden kolmanteen potenssiin.

Pumpun ominaiskulutuksen tunnusluku on pumpun ottaman sähkötehon suhde vesivirtaan (W/(l/s)) (Shemeikka et al. 1996). ASHRAE-standardi 90.1-1989 esittää pumppauksen ominaisteholle maksimiarvon 300 W/(l/s). Arvo koskee kaikkia rakennuksen vesikiertopiirejä (lämmitys- ja jäähdytysvesipiirit). On huomattava, että ilmastointijärjestelmissä lämmönsiirtonesteenä on toisinaan vesi-etyleeniglykoli-seos, jolloin vedelle annetut arvot eivät ole suoraan sovellettavissa.

Shemeikka et al. (1996) ovat esittäneet pumppujen energiatehokkuusluokittelun asuinrakennuksille, mutta muille rakennuksille ei ole olemassa vastaavaa luokittelua. Pumpun toiminnan vertailukohtana voidaan käyttää myös valmistajan taulukoita. Valmistaja ilmoittaa pumpulle ominaiskäyrän, joka kertoo pumpun paineenkorotuksen ja tilavuusvirran riippuvuuden toisistaan. Valmistajan antamista tiedoista selviää pumpun sähkönkulutus kussakin toimintapisteessä.

2.1.3 Muut mahdolliset suorituskykytekijät

Käyttökustannukset

Ilmastointikoneen energiankustannukset riippuvat käytetystä lämmönlähteestä. Tarkasteltava ilmastointikone käyttää hyväkseen ensisijaisesti poistoilman lämpöä. Mikäli tämä ei riitä, käytetään kylmälaitteiden lauhdelämpöä, ja mikäli tämä ei riitä, käytetään kaukolämpöä.

Suunnittelutyökalu tuottaa vertailuarvot lämmöntalteenoton tuottamalle energialle, lauhdelämmölle ja kaukolämmölle. Tässä työssä esitetään menetelmä, jolla em. tehoja voitaisiin seurata reaaliajassa.

Käyttövarmuus

Käyttövarmuudella tarkoitetaan sitä, kuinka suuren osan ajasta järjestelmä on käytettävissä suhteessa siihen aikaan, jonka järjestelmä on esimerkiksi vikaantumisen tai huollon takia poissa käytöstä (Lyytikäinen 1987).

Käyttövarmuus jakaantuu toimintavarmuuteen, huollettavuuteen ja huoltovarmuuteen (LVI 70-40013). Toimintavarmuus ilmaisee, kuinka kauan järjestelmä toimii ilman vikoja. Toimintavarmuutta kuvataan esimerkiksi keskimääräisellä vikaantumisvälillä. Huollettavuus ja huoltovarmuus yhdessä määrittelevät sen ajan, jonka kuluessa järjestelmä saadaan vikaantumisen jälkeen käyttökuntoon. Huollettavuuden mittoja ovat mm. keskimääräinen aktiivinen korjausaika, tarkastusten välit ja tarvittava huoltotyömäärä käyttötuntia kohti. Huollettavuuteen vaikuttavat järjestelmän ja sen osien rakenne, luoksepäästävyys ja huoltotilat. Huoltovarmuus kuvaa huolto-organisaation kykyä järjestää edellytykset huoltoon varten. Siihen vaikuttavat mm. varaosavarasto, henkilöresurssit, huolto-ohjeet ja henkilökunnan osaaminen.

Käyttövarmuuden analysointimenetelmiä ovat poikkeamatarkastelu, vikojen vaikutusten tarkastelu, huollon vaikutusten tarkastelu sekä laskennalliset menetelmät (LVI 70-40013). Poikkeamatarkastelun (PT) tekee asiantuntijaryhmä käymällä tarkasteltavan järjestelmän systemaattisesti läpi. Poikkeamatarkasteluistunto on ideapalaveri, jossa pyritään keksimään poikkeamien syitä, seurauksia, havaitsemistapoja sekä parannusehdotuksia. Vika- ja vaikutusanalyysissa (VVA) tarkastellaan yksittäistä laitetta, jolle määritellään vioittumistavat, vian havaitsemistapa, vian vaikutus järjestelmän toimintaan sekä parannusehdotuksia vian tai sen vaikutuksen eliminoimiseksi. Huolto- ja vaikutusanalyysi (HVA) vastaa VVA-analyysia, mutta se kohdistuu laitteen tai järjestelmän huollettavuuden ja huoltovarmuuden parantamiseen. Laskennallisilla menetelmillä voidaan ennustaa järjestelmien käyttövarmuutta.

Yleisesti ottaen järjestelmän käyttövarmuus kasvaa sen mukaan, mitä enemmän siinä on luotettavuusteknisiä rinnakkaisrakenteita, ja pienenee sen mukaan, mitä enemmän siinä on luotettavuusteknisiä sarjarakenteita (LVI 70-40013).

Säätötekniinen toiminta

Virheet säädön toiminnassa voivat aiheuttaa epäviiltyvyyttä ja energiankulutuksen kasvua sekä lyhentää säätölaitteitten kestoikää.

Säädön tavoitteita ovat

- lyhyt asettumisaika,
- pieni poikkeama asetusarvosta,
- värähtelemättömyys.

Tyypillisiä säätöjärjestelmän vikoja on esitetty lähteessä Leivo & Sirén 1991. Tyypillisiä suunnitteluvirheitä ovat rakennuskohteeseen sopimattoman säätötavan valinta sekä ohjelmistoviat. Tyypillisiä asennuksen ja käyttöönoton virheitä ovat säätöpiirien puutteellinen viritys tai virittämättä jättäminen, mittausanturien väärä sijoitus sekä anturien ristiin kytkeminen. Tyypillisiä käytön aikaisia virheitä ovat ilmastointikoneiden käyntiajoissa olevat virheet sekä viritysparametrien tahattomasti tehty muuttaminen.

2.2 Suorituskykyvaatimukset

Ilmastointikoneen tehtävänä on tuottaa haluttu määrä puhdasta ilmaa sekä lämpö- tai jäähdytysteho. Muita vaatimuksia ovat energiataloudellisuus ja säätötekniisen toiminnan hyväksyttävyyys. Äänitasolle asetettavat vaatimukset on jätetty tarkastelun ulkopuolelle.

Suorituskykyvaatimusten perusteella voidaan kehittää ilmastointikoneen käyttäjää avustavia koneen oikean toiminnan seurantaan soveltuvia työkaluja. Jos vaadittua suorituskykyarvoa ei saavuteta, kyseessä voi olla suunnittelu- tai mitoitusvirhe, valmistuksessa, asennuksessa tai käyttöönotossa ilmennyt virhe tai käytönaikainen vika.

Mahdollisia ilmastointikoneen käytönaikaisia vikoja ovat mm.

- pellin, venttiilin tai toimimoottorin jumiutuminen
- tukos suodattimessa, patterissa tai ulko-, tulo- tai poistoilmasäleikössä
- puhaltimen tai pumpun pysähtyminen
- tukos tai vuoto putkistossa
- väärä lämmönsiirtonesteen pitoisuus
- anturivika tai säätölaitevika.

2.2.1 Sisäilmaston toteutumiseen liittyvät vaatimukset

Ilmastointijärjestelmän toiminta vaikuttaa keskeisesti kohdassa 2.1.1 kuvattujen sisäilmastotekijöiden toteutumiseen. Tässä kohdassa käsitellään sisäilmastotekijöiden laitejärjestelmälle asettamia suorituskykyvaatimuksia.

Lämpöteho

Ilmastointikoneen keskeisenä suorituskykyvaatimuksena on saada IV-koneesta kullakin hetkellä haluttu lämmitys- tai jäähdytysteho. Vakioilmavirtaisessa järjestelmässä säädetään ilmastointikoneen jälkeisen ilman lämpötilaa. Tarkasteltavan järjestelmän kaltaisessa muuttuvailmavirtaisessa järjestelmässä tehoa säädetään myös ilmavirran avulla.

Ilmastointikoneen tarvitsema teho (kW) on

$$\phi_{ik} = q_{v,i} \rho_i c_{p,i} (t_{ik} - t_u), \quad (4)$$

jossa

$q_{v,i}$ on ilman tilavuusvirta (m^3/s)

ρ_i ilman tiheys (kg/m^3)

$c_{p,i}$ ilman ominaislämpökapasiteetti (kJ/kgK)

t_{ik} ilmastointikoneelta lähtevän ilman lämpötila ($^{\circ}\text{C}$)

t_u ulkoilman lämpötila ($^{\circ}\text{C}$).

Tarkasteltavassa ilmastointikoneessa ulkoilman lämmittämiseen tarvittava teho koostuu palautusilman sekoituksesta saatavasta lämpötehosta, poistoilmasta talteenotetusta tehosta, puhaltimesta tuloilmaan siirtyvästä tehosta sekä lisälämmityksestä.

Tarkasteltavassa market-sovelluksessa lämpötilojen kerrostuminen estetään huoneilman sekoittamisella, joten poistoilman lämpötilan voidaan olettaa olevan huoneilman lämpötilassa. Jäähdytystilanteessa ilmanjako ylläpitää 2°C :n lämpötilakerrostuman, joten poistoilma lämpötila on noin 2°C oleskelutilan lämpötilaa korkeampi.

Koska marketissa pääasiallinen lämmitystapa on ilmalämmitys, on syytä tarkastella ilmalämmityksen huoneeseen tuomaa tehoa. Tuloilmalla voidaan tuoda lämmitystehoa tai jäähdytystehoa.

Tuloilman huoneeseen tuoma teho (kW) on

$$\phi_{ik} = q_{v,i} \rho_i c_{p,i} (t_{sp} - t_h), \quad (5)$$

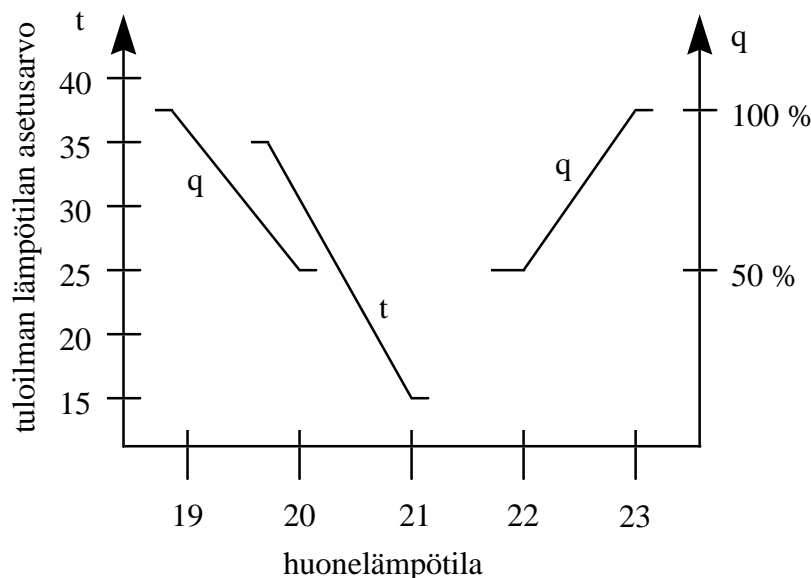
jossa

| | | |
|-----------|----|----------------------------------------------------------|
| $q_{v,i}$ | on | ilman tilavuusvirta (m^3/s) |
| ρ_i | | ilman tiheys (kg/m^3) |
| $c_{p,i}$ | | ilman ominaislämpökapasiteetti (kJ/kgK) |
| t_{sp} | | huoneeseen puhallettavan ilman lämpötila ($^{\circ}C$) |
| t_h | | huoneilman lämpötila ($^{\circ}C$). |

Koska järjestelmään ei kuulu jälkilämmitystä, huoneeseen puhallettavan ilman lämpötila on likimain ilmastointikoneelta lähtevän ilman lämpötila, jos lämpöhäviöt kanavassa ovat pieniä.

On huomattava, että lämpötehoa ei voida tuoda huoneeseen millä tahansa ilmavirran ja lämpötilan yhdistelmällä. Viihtyisyyseikat rajoittavat tuloilman lämpötilan tietylle alueelle. Myös tekniset syyt rajoittavat tuloilman lämpötilaa; puhallinmoottori ei siedä liian korkeaa lämpötilaa ja liian alhaisesta tuloilman lämpötilasta seuraa kondensoitumista, jos kanavia ei ole eristetty.

Tarkasteltavassa market-sovelluksessa tuloilman lämpötilaa ja ilmavirtaa ohjataan huonelämpötilan mukaan. Kuvassa 4 on esitetty esimerkkinä käyttötavaraosaston asetusarvoalueet. Jäähdytystilanteessa tuloilman lämpötilan asetusarvo on $15^{\circ}C$. Lämmitystilanteessa asetusarvo on $35^{\circ}C$. Ilmavirtaa kasvatetaan, mikäli tarvittava lämpötehoa ei voida tuottaa minimi-ilmavirralla.



Kuva 4. Ilmalämmityskoneen tuloilman lämpötilan ja ilmavirran ohjaus (ABB Installaatiot Oy 1995).

Suorituskykyvaatimuksena oleva lämpöteho toteutuu, jos tuloilman lämpötila ja ilmavirta saavuttavat asetusarvonsa. Jotta asetusarvot määräytyisivät oikein huonelämpötilasta, huonelämpötilan mittauksen tulee olla tarkka ja ohjelmistossa ei saa olla virheitä.

Tuloilmavirta ja tuloilman puhtaus

Rakentamismääräyskokoelman osa D2 (Ympäristöministeriö 1987) määrää, että "Oleskelutiloihin on johdettava tyydyttävän sisäilman puhtauden takaava määrä ulkoilmaa". D2 antaa ohjearvoja ilmavirralle. Ilmavirta määräytyy ensisijaisesti henkilöperusteen mukaan, mutta jos henkilömäärä ei ole tiedossa, voidaan käyttää pinta-alaperustetta. Tarkasteltavan järjestelmän järjestelmäkäsikirja antaa mitoitussulkoilmavirraksi 10 l/s,hlö tai 2 l/s,m².

D2 määrää tuloilman puhtaudesta seuraavasti: "Tuloilman puhtausvaatimus määräytyy sisäilman puhtaudesta asetettujen vaatimusten perusteella. Tuloilma on tarvittaessa puhdistettava." Käytännössä tuloilman on oltava huoneilmaa puhtaampaa, koska huonetiloissa tapahtuva toiminta sekä rakennusmateriaalit ym. synnyttävät epäpuhtauksia huonetilaan.

Palautusilman käytöstä D2 määrää seuraavasti: "Palautus- ja siirtoilmana saadaan käyttää vain ilmanpuhtaudeltaan samanarvoisten tai puhtaampien tilojen ilmaa, joka ei saa sisältää haitallisia määriä hiukkasia eikä kaasumaisia epäpuhtauksia."

Ulkoilman epäpuhtauslähteitä ovat mm. ajoneuvot, voimalaitokset, jätteenpolttolaitokset, jäteveden käsittelylaitokset, pöly, bakteerit ja siitepöly. D2 määrää, että "Ulkoilma-aukot on sijoitettava siten, että rakennukseen tuleva ulkoilma on mahdollisimman puhdasta." D2:n ohjeet antavat vähimmäisetäisyyksiä ulkoilma-aukkojen ja ulkoilman laatua pilaavien kohteiden välille.

Ilmastointikoneen suodattimet vähentävät epäpuhtauksia. Suodattimen valinta riippuu suodatettavan ilman epäpuhtauksista sekä sen tilan laatuvaatimuksista, johon ilma johdetaan. Suodattimet luokitellaan Eurovent-luokituksen mukaisesti EU1 - EU9-luokkiin. Luokkien EU1 - EU4 suodattimet ovat karkeasuodattimia ja luokkien EU5 - EU9 suodattimet hienosuodattimia. Tarkasteltavassa ilmastointikoneessa on vakiona tuloilmasuodatin EU7 ja poistoilmasuodatin EU5 (ABB Fläkt Oy 1996).

Suodattimen erotuskykyä huonontavat vuodot suodatinkehysten välistä sekä suodattimen jälkeisten ilmastointikoneen osien vaipan vuodot. Vuotojen suuruus riippuu koneen tiiviyydestä sekä koneen alipaineesta ympäristöönsä nähden. Mikäli va-

litaan hyvän erotuskyvyn suodatin, täytyy ilmanvaihtolaitoksen olla erityisen tiivis, sillä vuotojen vaikutus on sitä suurempi, mitä parempi suodatin on.

D2 määrää, että "Ilmanvaihtolaitoksen osien tulee olla riittävän tiiviit." Ohjearvona esitetään, ettei vuotoilmavirta saa käyttötilanteessa ylittää 6 % laitoksen kokonaisilmavirrasta. Tiiviysluokka (K,A,B,C) kertoo vuotoilmavirran ($\text{dm}^3/\text{s}, \text{m}^2$) paine-erosta riippuen.

Ulkoilman tai palautusilman lisäksi epäpuhtauslähteenä saattaa olla itse ilmastointikone. Liian harvoin vaihdettu/puhdistettu suodatin voi levittää epäpuhtauksia tuloilmaan. Myös ilmastointikoneen kostutin saattaa levittää epäpuhtauksia. Tarkasteltavassa ilmastointikoneessa kostutetaan vain poistoilmaa, joten kostutin ei voi levittää epäpuhtauksia tuloilmaan, kun kostutusjäähdytyksen ja palautusilman samanaikainen käyttö on estetty. IV-koneeseen voi kulkeutua epäpuhtauksia myös epäpuhtaissa tiloissa kulkevien alipaineisten kanavien vuotojen vuoksi. Rakennusvaiheessa kanavistoon jäänyt maa-aines tms. voi levittää epäpuhtauksia tuloilmaan.

Hiilidioksidipitoisuus

Marketissa ulkoilmavirtaa ohjataan huoneilman hiilidioksidipitoisuuden mukaan eli ilmastointikoneet siirtyvät käyttämään ulkoilmaa palautusilman sijaan huoneilman hiilidioksidipitoisuuden noustua yli raja-arvon (esim. 800 ppm:n tai 1 000 ppm:n). Mikäli hiilidioksidipitoisuus nousee yli 1 000 ppm:n ja IV-koneet ovat ulkoilmalla, kasvatetaan IV-koneiden ilmavirtaa. Yöaikana huoneilman hiilidioksidipitoisuus lähenee ulkoilman hiilidioksidipitoisuutta, joten aamulla IV-koneet käyttävät pelkästään palautusilmaa.

Mitä pienempi tuloilman hiilidioksidipitoisuus on, sitä pienemmällä ilmavirralla huoneilman hiilidioksidipitoisuus voidaan pitää asetusarvossaan. Ulkoilman hiilidioksidipitoisuus on noin 350 ppm. Hiilidioksidia ei pystytä suodattamaan pois tuloilmasta tai ainakaan se ei ole teknisesti ja taloudellisesti kannattavaa eikä tuloilmakone luovuta hiilidioksidia tuloilmaan, joten tuloilman hiilidioksidipitoisuus riippuu ulkoilman ja palautusilman hiilidioksidipitoisuudesta. Tosin alipaineisten kanavien vuotaminen saattaa hieman nostaa tuloilman hiilidioksidipitoisuutta. Hiilidioksidipitoisuuden mittaaminen tuloilmasta ei varmaankaan ole tarpeellista.

On huomattava, että hiilidioksidipitoisuus ei kerro mitään muiden epäpuhtauksien määrästä. Jos marketissa on ihmisten lisäksi muita epäpuhtauslähteitä, sisäilman puhtaus voi olla huono, vaikka hiilidioksidipitoisuus olisi hallinnassa.

2.2.2 Energiatehokkuus

Tarkasteltavan järjestelmän lämmönlähteenä käytetään ensisijaisesti poistoilman lämpöä ja toissijaisesti kylmälaitteiden lauhdelämpöä. Jos ne eivät riitä, käytetään kaukolämpöä.

Oleennaista on, millä kustannuksella voidaan tuottaa tietty energiamäärä. Lämmöntalteenotolla tuotettu energia on lähes ilmaista, sillä pumppauksesta aiheutuu vähän sähköenergian kuluja. Myös kylmälaitteiden lauhdelämpö on käytännössä ilmaista; perinteisesti lauhdutuslämpö ajetaan pihalle katolla olevien lauhduttimien kautta. Kaukolämmöstä maksetaan energiamaksua sekä perusmaksua, joka määräytyy tilausvesivirran mukaan. Kaukolämmöstä hyödynnetään ensisijaisesti radiaattoriverkoston paluuveden lämpöä.

Tarkasteltavassa järjestelmässä voidaan käyttää muitakin lämmönlähteitä, esimerkiksi jäteilmasta lämpöä talteenottavaa lämpöpumppua. Tällöin kustannuksia aiheutuu kompressorin sähkökulutuksesta. Lämpökerroin kertoo lauhdelämmön ja kompressorien sähkökulutuksen suhteen. Tariffeista riippuen voi olla mahdollista, että on kannattavampaa käyttää esimerkiksi kaukolämpöä kuin lämpöpumppujen lauhdelämpöä.

On myös huomattava, että lämmönlähteen hyödynnettävyys riippuu sen lämpötilatasosta. Tarkasteltavassa market-sovelluksessa lämmönsiirtoverkoston menonesteen lämpötilan asetusarvo on 30 - 40 °C. Kylmimpään aikaan se on 40 °C. Tätä alempi lämpötilataso ei riitä, koska tuloilman lämpötilan asetusarvo on suurimmillaan 35 °C. Kylmälaitteiden lauhdutuslämmön lämpötilataso on noin 30 °C, joten vaikka energiaa olisikin riittävästi tarjolla, sillä ei voida lämmittää menonestettä asetusarvoonsa. Tarvitaan siis kaukolämpöä. Myöskään radiaattoriverkoston paluuvesi ei pysty lämmittämään menonestettä 40 °C:seen.

Lämmöntalteenotosta hyödynnettävään energiaan vaikuttaa poistoilman ja ulkoilman tilan lisäksi kiertopumpun virtaama. Ideana on optimoida pumpun nestevirta niin, että lämmöntalteenoton hyötysuhde on mahdollisimman korkea, jos on lämmöntarvetta. Epäsuoran LTO:n hyötysuhde on korkeimmillaan, kun nesteen lämpökapasiteettivirta on sama kuin ilmavirtojen. Tyypillinen epäsuoran LTO:n hyötysuhde on 50 - 60 %. LTO:n hyötysuhdetta rajoittaa kylmimpään aikaan huurtumisen esto, joka toteutetaan ensi vaiheessa pumpun kierroslukua kasvattamalla.

Puhaltimien ja pumppujen energiankulutusta on käsitelty aiemmin kohdassa 2.1.1 Energiankulutus.

2.2.3 Säädotekninen toiminta

Standardi SFS 5768 määrittelee säätötekni­sen minimitason ilmastointi­järjestelmille. Taulukossa 4 esitetään vaatimukset tuloilman lämpötilan säädölle.

Edellytykset säädön toiminnan tarkastamiselle ovat:

- Keskusilmastointikoneeseen tulevan, lämmitykseen tai jäähdytykseen käytettävän nesteen lämpötilan jatkuvaa värähtelyä ei saa esiintyä.
- Keskusilmastointikoneeseen tulevan ilman lämpötilan jatkuvaa värähtelyä tai äkillisiä muutoksia ei saa tapahtua.
- Mahdollisen ilmakehän staattisen paineen säädön tulee toimia sille asetettujen vaatimusten mukaisesti.

Taulukko 4. Standardin SFS 5768 mukaiset vaatimukset keskusilmastointikoneen jälkeisen ilman lämpötilan säädölle.

| Tilanne | Vaatus |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Käynnistys | <ul style="list-style-type: none">• asettumisaika $\pm 1\text{ }^{\circ}\text{C}$:n tarkkuudella lopullisesta arvosta enintään 15 min |
| Käyntiaika | <ul style="list-style-type: none">• lämpötilan keskiarvon suurin sallittu poikkeama asetusarvosta $\pm 1\text{ }^{\circ}\text{C}$• symmetrisesti keskiarvon molemmin puolin sijaitsevat rajat, joiden sisällä lämpötilan tulee olla vähintään 90 % ajasta $\pm 0,5\text{ }^{\circ}\text{C}$• jatkuvaa värähtelyä ei saa esiintyä |
| Asetusarvon muutos | <ul style="list-style-type: none">• asettumisaika $\pm 1\text{ }^{\circ}\text{C}$:n tarkkuudella lopullisesta arvosta enintään 10 min• lämpötilan asettumisajan jälkeisen keskiarvon suurin sallittu poikkeama uudesta asetusarvosta $\pm 1\text{ }^{\circ}\text{C}$• symmetrisesti asettumisajan jälkeisen keskiarvon molemmin puolin sijaitsevat rajat, joiden sisällä lämpötilan tulee olla vähintään 90 % ajasta $\pm 0,5\text{ }^{\circ}\text{C}$ |

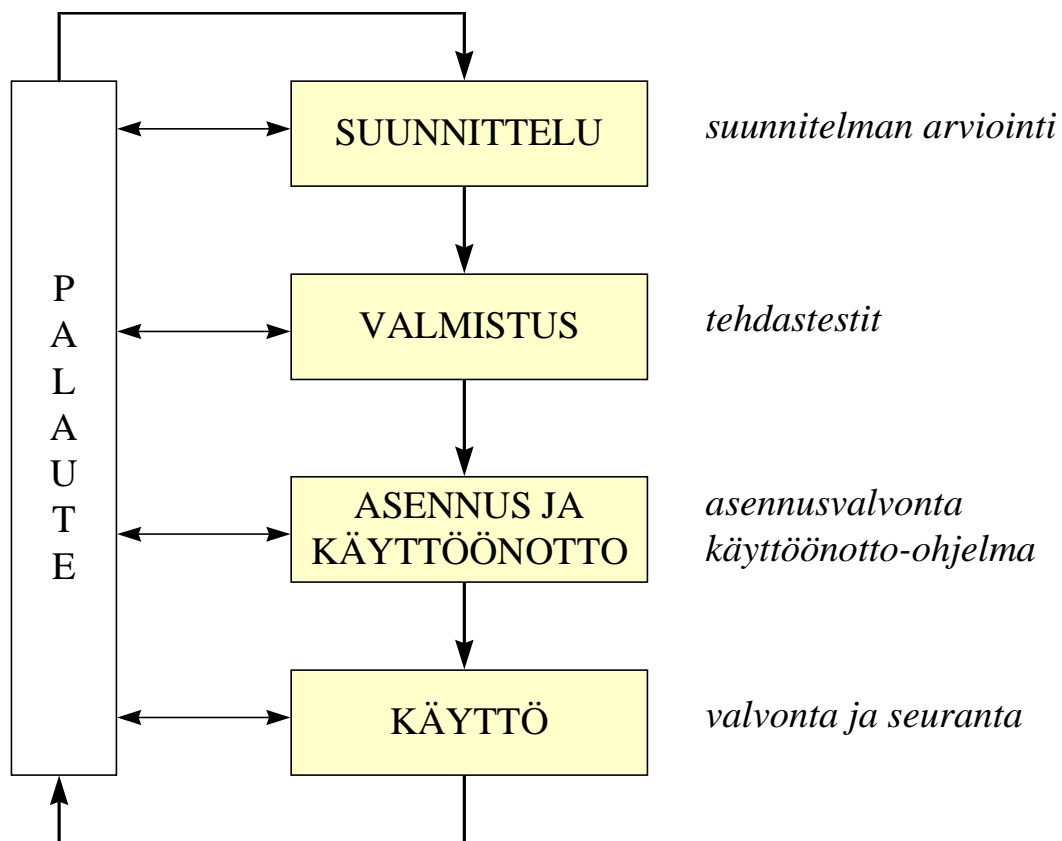
2.3 Elinkaarivaiheiden vaikutus suorituskykyyn

Taloteknisten järjestelmien suorituskykyyn vaikuttavat monet suunnittelun-, valmistuksen-, asennuksen- ja käytön aikaiset ratkaisut. Tässä kohdassa selvitetään sitä, miten suorituskykytekijät määritetään tuotteen eri elinkaarivaiheissa, jotta niiden toteutumista voitaisiin seurata vaiheittain.

Elinkaaren vaiheet jaetaan tässä seuraavasti:

- suunnittelu
- valmistus
- asennus ja käyttöönotto
- käyttö.

Kuva 5 esittää, miten elinkaarivaiheet ja suorituskyvyn seuranta kytkeytyvät toisiinsa tuotteen tai järjestelmän eri elinkaarivaiheissa. Tuotteen elinkaarivaiheista saadaan palautetta, jota voidaan hyödyntää vastaavan tuotteen seuraavan sukupolven toteutuksessa. Kaikkiin elinkaarivaiheisiin liittyy niille ominaisia menetelmiä, joilla suorituskykytekijöiden toteutumista voidaan seurata ja järjestelmän oikea toiminta varmentaa ennen seuraavaan vaiheeseen siirtymistä.



Kuva 5. Elinkaarivaiheet ja suorituskyvyn seuranta eri vaiheissa.

2.3.1 Suunnittelu

Suunnittelu luo pohjan suoritusteknisten vaatimusten toteutumiselle. Suunnitteluvaiheissa tehtävät ratkaisut vaikuttavat keskeisesti rakennuksen koko elinkaaren

aikaisiin käyttökustannuksiin. Tällöin tehdään myös rakennuksen energiataloutta koskevat tärkeimmät ratkaisut.

Rakentamismääräyskokoelman osa D2 (Ympäristöministeriö 1987) määrää ilmanvaihtosuunnitelmasta seuraavasti: "Ilmanvaihtosuunnitelmassa on esitettävä sisäilmaston tavoitearvot, laitoksen laatutaso sekä laitoksen mitoituksen, rakentamisen ja käytön kannalta olennaiset tiedot." IV-suunnitelma käsittää joukon asiapapereita, joista D2 määrää seuraavasti: "Ilmanvaihtolaitoksen suunnitelmiin tulee sisältyä laitoksen toimintasuunnitelma, toiminta- ja säätökaaviot, piirustukset, ilmanvaihtotöiden tekninen erittely, käyttöönotto-ohjelma ja järjestelmätekniiset käyttöohjeet."

Talotekniikan suunnittelun tehtäväluetteloa TATE 95 (LVI 03-10242) käytetään suunnittelijan tehtävälajisuuden määrittelyssä, suunnittelukokonaisuuden hallinnassa sekä osana suunnittelun laadunvarmistusta. Tehtäväluettelon ovat hyväksyneet rakennuttaja- ja konsulttitoimistoliitot.

Suunnitelmissa määritetään mm. seuraavat IV-laitoksen suorituskykyyn vaikuttavat tekijät:

1. järjestelmäkonfiguraatio ja säätötapa
2. lämmön- ja kylmänlähteet
3. ulko- ja jäteilmavirrat, tulo- ja poistoilmavirrat
4. laitteiden lämmitys-, jäähdytys- ja kostutustehot
5. puhaltimien ja pumppujen ominaiskäyrät
6. kanaviston ja kojeosien painehäviöt
7. kanaviston ja kojeosien tiiviysluokat
8. suodattimien suodatusluokat
9. mittauspisteiden sijainnit
10. ulkoilma-aukkojen sijainnit
11. säätöventtiilien mitoitus.

Taulukko 5. Tärkeimmät suunnitelmissa valittavat eri suorituskykytekijöihin vaikuttavat asiat. Taulukko on suuntaa-antava.

| Suorituskykytekijä | numero edellä olevassa listassa |
|------------------------------------|---------------------------------|
| lämpöteho | 1, 2, (3), 4 |
| tuloilmavirta ja tuloilman puhtaus | 3, 5, 7, 8, 10 |
| energiatehokkuus | 1, 2, (4), 5, 6 |
| säätötekniinen toiminta | 1, (4), 9, 11 |

Suunnittelussa valittavista tekijöistä lämpötehoon vaikuttavat lämmönlähteen riittävyys ja IV-koneen mitoitus. Tuloilman puhtaus riippuu ulkoilma-aukkojen sijainnista, suodatinvalinnasta sekä IV-koneen vuodoista. Energiatehokkuuteen

vaikuttavat mm. käytetty lämmönlähde, LTO:n hyötysuhde ja puhaltimien ja pumppujen hyötysuhde. Säätötekniiseen toimintaan vaikuttavista tekijöistä suunnitteluvaiheessa valitaan säätötapa ja antureiden sijainti. Säädetävyyteen vaikuttaa myös järjestelmätekniinen mitoitus.

Myös monet käyttövarmuuteen vaikuttavat tekijät kiinnitetään suunnitteluvaiheessa. Järjestelmän käyttövarmuutta parantaa esimerkiksi se, että järjestelmään suunnitellaan kaksi rinnakkaista pumppua, joista toista käytetään kerrallaan. Suunnitelmavaiheessa vaikutetaan myös huollettavuuteen mm. IV-konehuoneiden suunnittelulla.

Suunnittelussa tehdyillä valinnoilla tulee voida saavuttaa asetetut vaatimukset. Mikäli joudutaan tekemään muutoksia jälkikäteen, ne ovat useimmiten hankalia ja kalliita. Tyypillisiä IV-järjestelmän suunnittelun virheitä ovat yli- tai alimitoitus, jotka usein aiheutuvat puutteellisista lähtötiedoista. Yleisiä automaatio suunnittelun virheitä ovat virheelliset ja epäselvät säätökaaviot ja toimintaselostukset (Leivo & Sirén 1991 ja Kosonen et al. 1995). Virheitä vähennetään käyttämällä ammattitaitoista henkilöstöä sekä tarkistusmenettelyillä.

Useilla valmistajilla on ilmastointikoneen osien mitoitusta ja valintaa varten tietokoneohjelmia. Myös tarkasteltavaan ilmastointikoneeseen kuuluvien laitteiden mitoitus ja suunnittelu tehdään vastaavanlaisella ohjelmalla.

2.3.2 Valmistus

Suunnitelmien mukaiset osajärjestelmät kootaan joukosta tehdasvalmisteisia, keskenään yhteensopivia tuotteita. Tarkasteltava ilmastointikone on asennusvalmis kokonaisuus, joko sisältää LTO-piirin lisälämmön ja -jäähdytyksen syötöllä, puhaltimet, pumput ja venttiilit, sähkökeskuksen ja kaapelit sekä säätöjärjestelmän. Tarkasteltavaan ilmastointikoneeseen voi myös kuulua äänenvaimennusosa ja sekoitusosa. Tarkasteltava ilmastointikone voi olla yhdistetty tai jaettu kahteen osaan, jolloin toisessa on tuloilmayksikkö ja toisessa poistoilmayksikkö ja niiden välissä putkisto.

Ilmastointijärjestelmiin ja niiden osiin liittyy useita standardeja (taulukko 6). Ne asettavat vähimmäisvaatimuksia laitteille sekä esittävät hyväksyttävät mittausten menetelmät ja testitulosten esittämistavan. SFS 5358 asettaa vaatimuksia mm. keskusilmastointikoneen materiaalille, tiiviydelle, lujuudelle ja palonkestävyydelle. SFS 5404 esittää patterien testausmenettelyn sekä tulosten esittämistavan mm. kehyksen ilmatiiviydelle, lämpö- ja virtausmittauksille ja lämpöteholle. Venttiilistandardeissa esitetään mm. maksimivuotomäärä ja

standardissa SFS 5330 vaatimukset säätöpeltien tiiviydelle. SFS 5147 esittää puhaltimien hyväksyttävät mittausmenetelmät sekä täsmentää tulosten käsittelyä.

Taulukko 6. Ilmastointijärjestelmän suorituskykyvaatimuksiin liittyviä standardeja.

| Numero | Nimi |
|------------------------------|---------------------------------------------------------------------------------------------------------|
| SFS 3542 | Ilmastointikanavat. Lujuus- ja tiiviystestaus |
| SFS 3543 | Ilmastointikanavat. Lujuus- ja tiiviysvaatimukset |
| SFS 4120-4134, SFS 4341-4358 | Putkistovarusteet. Venttiilit. |
| SFS 4448 | Pumput. Keskipako-, puoliaksaali- ja aksiaalipumppujen testaus. Luokka C. |
| SFS 4699 | Ilmastointi. Ilmastointilaitosten tiiviysvaatimukset |
| SFS 5117 | Ilmastointi. Kanavaäänenvaimentimen vaimennusarvon määrittäminen |
| SFS 5147 | Ilmatekniikka. Puhaltimet. Suoritusarvojen virtaustekniset mittaukset |
| SFS 5148 | Ilmatekniikka. Puhaltimien käyttö ilmankäsittelylaitoksissa |
| SFS 5149 | Ilmatekniikka. Puhaltimet. Suoritusarvojen esittäminen |
| SFS 5150 | Ilmastointi. Ilmansuodattimien testausmenetelmät |
| SFS 5332 | Ilmastointi. Korkean erotusasteen (HEPA) ilmansuodattimien vuorotestaus. Käsitteet, yksiköt, menetelmät |
| SFS 5329 | Ilmastointi. Vakio- ja muuttuvan ilmavirran säätimet. Testaus |
| SFS 5330 | Ilmastointi. Säätö- ja sulkulaitteet. Laitevaatimukset |
| SFS 5331 | Ilmastointi. Säätö- ja sulkulaitteet. Testaus |
| SFS 5357 | Ilmastointi. Keskusilmastointikone. Määritelmät |
| SFS 5358 | Ilmastointi. Keskusilmastointikone. Vaatimukset ja testaus |
| SFS 5404 | Ilmastointi. Pakotetun ilmavirtauksen lämmittimet ja jäähdyttimet. Testaus |
| SFS 5436 | Ilmastointi. Metallilevystä valmistetut suorakaidekanavan osat. Mitat |

Eurooppalaisen standardisointijärjestön CEN:n standardien valmistelutyö on ollut käynnissä jo useita vuosia. Standardien soveltamisessa noudatetaan periaatetta, että ensin sovelletaan maailmanlaajuisia (esim. ISO), sitten eurooppalaisia (esim. CEN) ja viimeisenä kansallisia standardeja (esim. SFS).

Ympäristöministeriö voi ennalta hyväksyä rakennusalan tuotteen rakennustapaviranomaisia sitovasti antamalla tyyppihyväksynnän. Tyyppihyväksynnän tarkoituksena on rakennusvalvonnan yksinkertaistaminen ja yhtenäistäminen sekä rakennustuotannon teollistumisen edistäminen (Ympäristöministeriö 1996). Tyyppihyväksyntämenettelyyn kuuluu, että ulkopuolisella ja sisäisellä laadunvalvonnalla

varmistetaan, että tuote täyttää jatkuvasti sille tyyppihyväksyntäpäätöksessä asetetut vaatimukset.

ThermoNet-tekniikkakeskus on valmistettu ISO 9001 -standardin (Suunnittelun, tuotekehityksen, tuotannon, asennuksen ja huollon laadunvarmistusmalli) mukaisesti (ABB Fläkt Oy 1996). Se testataan tehtaalla mekaanisen rakenteen ja toimintojen osalta. Mekaanisiin tarkastuksiin kuuluu laitetoimintojen, tiiviiden, rakenteen ja asennustarvikkeiden tarkastus. Toiminnallisiin tarkastuksiin kuuluu kytkentöjen, toimintojen ja massavirtojen tarkastus.

2.3.3 Asennus ja käyttöönotto

LVI-RYL 92 määrittää hyvän asennustavan eli työsuoritusten vaatimustason, jonka sekä rakennuttajat että työn suorittajat hyväksyvät. LVI-RYL 92 ei ole säädös, mutta sitä käytetään työselitysten viiteasiakirjana, jolloin se on urakoitsijaa velvoittava.

Rakentamismääräyskokoelman osa D2 (Ympäristöministeriö 1987) määrää ilmanvaihtolaitoksen käyttöönotosta: "Ilmanvaihtolaitos on mitattava ja säädettävä ja sen toiminta todettava suunnitelmien mukaiseksi ennen rakennuksen käyttöönottoa. Laitoksen toiminta eri kuormitus- ja sääoloissa on tarkistettava ja säädettävä." D2:n rakennuksen käyttöönottoon liittyviä määräyksiä ovat myös: "Ilmanvaihtolaitoksista on tehtävä yksityiskohtaiset ja selkeät käyttö- ja huolto-ohjeet" ja "Rakennuksen käyttäjälle luovutettaviin piirustuksiin on merkittävä kaikki rakennusaikana suunnitelmiin tehdyt muutokset. Käyttöä ja huoltoa varten on laadittava riittävät piirustukset ja ohjeet. Ilmanvaihtolaitosta käyttöön otettaessa on kiinteistöhoitajaa perehdytettävä riittävästi laitoksen toimintaan, käyttöön ja huoltoon."

RT-kortin RT 16-10288 mukainen rakennuksen käyttöönotto sisältää seuraavat vaiheet:

- *Asennustapataarkastus.* Onko kaikki asennukset tehty ja täyttääkö asennusten laatu niille asetetut vaatimukset?
- *Toimintakokeet.* "Rakennuttaja tarkastaa urakoitsijan edustajan läsnäollessa, että laitteet on asennettu oikein paikoilleen ja että sähköllä toimivat laitteet saavat virtansa lopullisia virtayhteyksiä pitkin. Moottorien, peltien ja automaattikalaitteiden liikesuuntien on oltava oikeat. Samoin pakkokytkentöjen käynnistyshidastusten ja hälytysten on oltavat oikein kytketyt."
- *Koekäyttö.* Tutkitaan laitteiden kokonaistoimintaa eri olosuhteissa. Koekäyttö tehdään niille laitteille, jotka urakka-asiakirjoissa määritellään koekäytettäväksi.

Säätö- ja viritystoimenpiteet LVI-RYL 92:n mukaan:

- hälytysten toiminnan kokeilu yhteistyössä sähkö- ja automaatiourakoitsijoiden kanssa sekä toimipisteiden viritykset
- toimimoottorien ohjaamien toimilaitteiden käytön tarkistus
- lämpötila- ja kosteuspysyvyydet
- lämpötila-, paine-, kosteus- ym. mittarien ja automaatiolaitteiden kalibrointi ja raja-arvojen asettelu
- varolaitteiden asettelut ja kokeilut
- äänimittaukset
- automaatiojärjestelmien viritys ja säädön tarkistus säätöpiireittäin
- kellokojeistojen toimiaikojen asetukset ja tarkistukset
- mittaus- ja virityspöytäkirjojen laadinta.

LVI-järjestelmien käyttöönottotarkastuksia on parannettu viime vuosina (Pietiläinen & Salonen 1994), kun on havaittu, että ilman niitä laitoksiin jää paljon vikoja. Nykyisin rakennuttajat hyväksyvät RT-kortin mukaiset tehtävät. Huolellisten tarkastustenkin jälkeen LVI-laitoksiin on havaittu kuitenkin edelleenkin jäävän vikoja (Pietiläinen & Salonen 1994). Virheet eivät ole aina helposti havaittavissa, koska ne saattavat esiintyä vain tietyissä olosuhteissa. Virheiden havaitsemiseksi on kehitetty tarkistusmenetelmä (Kosonen et al. 1995), joka soveltuu ensivaiheessa vasta vakioilmavirtaiselle ilmastointikoneelle.

Kuvaa IV-järjestelmien toiminnasta antaa, että yhden konsultin testaustilaston mukaan 76 % järjestelmistä toimi epätydyttävästi (Kosonen et al. 1995). Testaustulokset ovat vuosilta 1986 - 94. Testattuja järjestelmiä oli yhteensä noin 1 500 kpl. Virheistä 80 % on pääasiassa automaatiourakoitsijan virheitä. Tavallisimmat automaatiourakoitsijan virheet liittyvät säätölaitteiden viritykseen, mittausten kalibrointiin ja sovellusohjelmistoihin. Useimmissa tapauksissa järjestelmä saadaan toimimaan täysin tyydyttävästi viritysarvoja muuttamalla. Tyypillisiä asennuksesta aiheutuvia virheitä ovat mittausantureiden väärä sijoitus sekä anturien kytkeminen ristiin (Leivo & Sirén 1991).

Kun lämpötila mitataan kanavasta, anturin asennuksen mahdollisia virheitä ovat (Leivo & Sirén 1991):

- Anturi sijaitsee kohdassa, jossa ilma ei ole riittävän sekoittunutta.
- Mittalaite on kiinnitetty kanavan eristekerroksen päälle siten, että itse anturiosa jää eristeen sisään.
- Anturi on sijoitettu liian lähelle kanavan pintaa, jolloin anturi mittaa ilman lämpötilan sijaan kanavan pintalämpötilaa.
- Tuloilmaa mittaava anturi on sijoitettu liian kauas ilmastointikoneesta, jolloin mittausviive on suuri.
- Anturi on sijoitettu kohtaan, jossa virtausnopeus on pieni.

Kun lämpötila mitataan huonetilasta, anturin asennuksen mahdollisia virheitä ovat (Leivo & Sirén 1991):

- Anturin lämpötila ei edusta koko ilmastoitavan tilan lämpötilaa (esimerkiksi tuloilmasuihku osuu siihen).
- Anturi tarkoitettu ulkolämpötilan mittaamiseen ja on epäherkkä normaalin huonelämpötilan alueella.

2.3.4 Käyttö

Rakentamismääräyskokoelman osa D2 (Ympäristöministeriö 1987) määrää, että "Ilmanvaihtolaitosta on käytettävä ja huollettava siten, että näiden määräysten terveydelliset ja energiataloudelliset vaatimukset jatkuvasti täytetään."

Ilmastointilaitosten hoidon taso vaihtelee kiinteistöhoitajien ammattitaidosta ja siihen käytetyistä resursseista riippuen. Laitteiden ja järjestelmien valvonnalla pyritään estämään häiriöt ja pitämään järjestelmät tarkoituksenmukaisessa toiminnassa ja kunnossa. LVI-kortti LVI 39-10073 esittää ilmanvaihtolaitoksen hoito-ohjeen (laitevalmistajilla on yksityiskohtaisempia huolto-ohjeita). Käyttö- ja huolto-ohjelmassa esitetään kohde-, tehtävä-, aikaväli- ja ajankohtatietoja. Käyttötarkastukset tehdään päivittäin tai lyhyin väliajoin. Huoltotarkastukset tehdään harvemmin.

Nykyiset rakennusautomaatiojärjestelmät havaitsevat ja hälyttävät joistakin IV-järjestelmien vioista. Tällaisia ovat ristiriitahälytykset (eli kun käyntitila ei vastaa ohjausta, esimerkiksi puhaltimen pysähtyminen), jäätymisvaara, tuloilman lämpötilan ylärajan ylittyminen tai alarajan alittuminen ja suodattimen painehäviön kasvaminen (suodatin likainen).

2.4 Mahdollisia diagnostiikkamenetelmiä

Diagnostiikkamenetelmiä voidaan käyttää apuna järjestelmän suorituskykytekijöiden seurannassa. Diagnostiikkamenetelmien avulla pyritään havaitsemaan ja tulkitsemaan suorituskykytekijöissä aiheutuneet poikkeamat.

Vikadiagnostiikkamenetelmät jakautuvat vian havaitsemismenetelmiin ja vian paikantamismenetelmiin (eli varsinaiseen diagnoosiin). Vian havaitsemismenetelmillä voidaan todeta, että järjestelmässä on jokin vika, mutta ne eivät kerro, mikä vika on kyseessä ja missä kohdassa järjestelmää se sijaitsee. Yksinkertaisimmillaan jo lämpötilan seuranta voidaan pitää vian havait-

semismenetelmänä. Vian syyn ja paikan määrittämiseen käytetään vian paikantamismenetelmiä (tai vikadiagnoosimenetelmiä).

Diagnostiikkajärjestelmät voidaan toteuttaa itsenäisinä sovelluksina siten, että käyttäjä kertoo kaikki havaitsemansa epäkohdat. Diagnostiikkajärjestelmä voi myös olla yhteydessä rakennusautomaatiojärjestelmään. Tällöin se joko voi olla sulautettu automaatiojärjestelmään tai toimii erillisenä sovelluksena mutta kiinteässä yhteydessä automaatiojärjestelmään saaden sen kautta esimerkiksi mittaustietoa prosessista.

Käytännölliselle vikadiagnoosijärjestelmälle voidaan esittää joitakin laadullisia ominaisuuksia, jotka sen olisi hyvä täyttää (Pakanen et al. 1996a):

- Se tarvitsee vain muutaman syöttötiedon käyttäjältä.
- Se tarvitsee sovellusalue-tietoa vain aloituksessa tai päivityksessä.
- Se mukautuu helposti uusia vikoja varten.
- Se ei häiritse prosessin normaalia toimintaa.
- Se ei tarvitse ihmisen apua vian havaitsemisen tai paikantamisen aikana.
- Se ei aiheuta lisää energian- tai polttoaineenkulutusta.
- Se tarvitsee vain lyhyen opetusjakson tai vähän opetusaineistoa.
- Se soveltuu monenlaisiin LVI-prosesseihin.
- Se soveltuu monenlaisiin vikoihin.
- Se tukee sekä vian havaitsemista että paikantamista.
- Se ei tarvitse yksityiskohtaisia prosessimalleja.
- Se on helposti muokattavissa uusiin sovelluksiin.
- Se on helppo sulauttaa ja integroida rakennusautomaatiojärjestelmään.
- Sen kehittäminen vaatii vain vähän tai kohtuullisesti työtä ja kustannuksia.
- Se ei tarvitse lisäinstrumentointia.

Lista on suuntaa-antava. Se ei esitä kaikkia mahdollisia vaatimuksia, eikä kaikkia listan ominaisuuksia voida vaatia yhdeltä vikadiagnostiikkajärjestelmältä. Koska vikadiagnostiikkajärjestelmä ei pysty seuraamaan kaikkia mahdollisia vikoja, sen tulisi keskittyä havaitsemaan viat, joilla on suurimmat vaikutukset tai joiden esiintyminen on todennäköisintä.

Oleennaista vikadiagnostiikkajärjestelmien toiminnalle on hälytyksen kynnyksrajan asettaminen. Jos kynnyksraja on liian alhaalla, tulee turhia hälytyksiä. Jos kynnyksraja on liian korkealla, vikoja jää havaitsematta. Oikean toiminnan rajat olisi pystyttävä määrittämään.

Kun vikadiagnostiikkajärjestelmä havaitsee poikkeaman järjestelmän toiminnassa, se luokittelee vian ja päättää, miten siihen reagoidaan. Vaihtoehdot ovat 1) ei hälytystä (mahdollisesti varoitus), 2) hälytys, 3) hälytys ja koneen pysäytys. Ilmoitus-

luokka riippuu poikkeaman suuruudesta, kestosta ja toistosta sekä siitä, millaisia seurauksia vialla voi olla. Esimerkiksi LTO-piirin nesteen lämpötilan vähäinen vaihtelu kuuluu asiaan, kohtuullinen poikkeama alaspäin voi olla merkki huurtumisesta, suuret poikkeamat voivat merkitä vakavaa vikaa ja aiheuttaa hälytyksen ja koneen pysäytyksen.

Tarkasteltavan ilmastointijärjestelmän suorituskyvyn kannalta tärkeimpiä seurattavia tekijöitä ovat:

- Onko huonelämpötila (ja huonekosteus) sopiva?
- Onko huoneilman hiilidioksidipitoisuus oikea?
- Ovanko tuloilman lämpötila ja ilmavirta sallituissa rajoissa?
- Onko tuloilma puhdasta?
- Saadaanko poistoilman lämpö talteen? Paljonko lämmitys- tai jäähdytystehoa käytetään?
- Mikä on käytetyn lämmitys- tai jäähdytystehon lähde?
- Mitkä ovat puhaltimien (ja pumppujen) hyötysuhteet?
- Toimivatko säädöt värähtelemättä ja muuten hyväksyttävästi?

Listan kolmen ensimmäisen kohdan toteutumista voidaan seurata nyky menetelmin. Oikean toiminnan rajat on määritettävä, ja alituksesta tai ylityksestä voidaan tehdä hälytys. Tuloilman puhtauden seuraaminen on vaikeaa. Tuloilman hiilidioksidipitoisuus voitaisiin halutessa mitata, mutta useimpien muiden epäpuhtauksien mittaaminen ei onnistu tai ei ole ainakaan kannattavaa.

Seuraavassa kohdassa esitetään menetelmä, jolla voidaan seurata tarkasteltavan ilmastointikoneen LTO-piirin tarvitsemää lisälämpö- ja lisäjäähdytystehoa. Myös lämmitystehon lähteen seurantaan esitetään menetelmä. Puhaltimien ja pumppujen toimintaa voidaan seurata ominaiskäyrämenetelmällä. Myös säädön toimintaa voitaisiin diagnosoida. Esitettävissä diagnostiikkamenetelmissä on pyritty yksinkertaisuuteen ja helppoon sovellettavuuteen.

2.4.1 LTO-piirin tehojen seuranta

LTO-piiri on keskeinen osa tarkasteltavaa järjestelmää, joten olisi olennaista seurata sen toimintaa. Diagnostiikkajärjestelmän tulisi havaita poikkeamat LTO-piirin toiminnassa, esimerkiksi liian korkean lisälämmitystehon tai -jäähdytystehon käyttö.

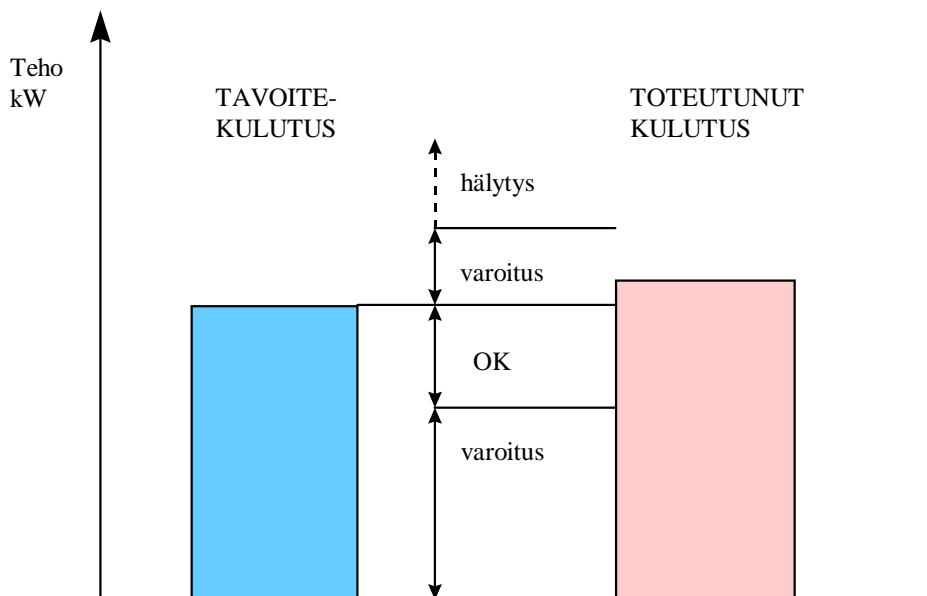
LTO-piirin lisätehon tarve riippuu mm. ilmavirroista ja niiden suhteesta, ulkoilman lämpötilasta, poistoilman lämpötilasta ja kosteudesta, ilmastointikoneen jälkeisen ilman lämpötilasta (sen asetusarvosta), palautusilmaosuudesta sekä

patterien mitoituksesta ja niiden dynamiikasta. LTO-piiriin syötetty lisälämmitys- tai lisäjähdytysteho voidaan mitata. Ongelmana on se, mistä saadaan vertailuarvot, jotta tiedetään, milloin käytetty teho on hyväksyttävää tasoa.

LTO-piiri voitaisiin mallintaa simulointiohjelmalla, jolloin toteutunutta ja ohjelman laskemaa tehoa pystyttäisiin vertaamaan toisiinsa. Tapa ei ole yksinkertainen eikä helposti sovellettavissa, vaan se vaatii lisäinstrumentointia, kohtuullisesti laskentakapasiteettia ja lisäkehitystyötä. Menetelmää voitaisiin jatkokehittää simuloimalla LTO-piiriä ja vertaamalla simulointituloksia oikein toimivasta (ja viallisesta) laitoksesta saataviin arvoihin.

Tehon seurantamenetelmillä voidaan tarkastella ilmastointikoneesta haluttua, saatavaa ja maksimitehoa (Kärki & Hyvärinen 1997). Tehokomponentit voidaan jaotella osiin, jotta saadaan selville, mikä osuus tarvittavasta tehosta voidaan tuottaa ilmaisenergioilla ja mikä on tuotettava muilla tavoilla.

LTO-piirin seuranta ei välttämättä vaadi tarkkaa mallintamista, vaan tarvittava lisälämmitysteho voitaneen arvioida ulkolämpötilan, palautusilmaosuuden ja ilmavirran perusteella. Laskentatarkkuus tietenkin huononee, mistä voi seurata turhia hälytyksiä tai havaitsematta jääneitä vikoja. Yksinkertaistetun laskentamenetelmän kehittämisessä ja oikean toiminnan rajojen etsimisessä voidaan käyttää hyväksi simulointeja.

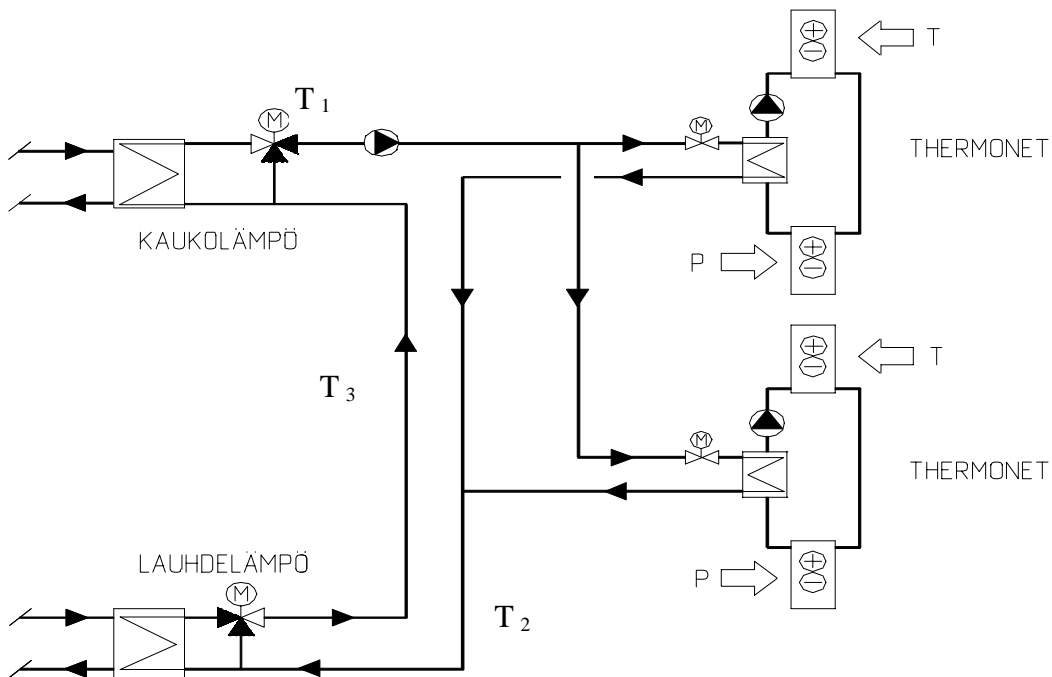


Kuva 6. Esimerkki LTO-piiriin syötetyn lisälämmitys- tai lisäjähdytystehon tavoitekulutuksen ja toteutuneen kulutuksen mahdollisesta vertailusta. Tässä tapauksessa diagnostiikkajärjestelmä varoittaisi liian korkeasta kulutuksesta.

Kuvassa 6 havainnollistetaan esimerkinomaisesti, kuinka tehon näyttömallilla verrattaisiin tavoitekulutusta toteutuneeseen kulutukseen. Jos lämmöntalteenottoa käytetään tarkoituksellisesti ei-optimaalisesti (esimerkiksi lauhdelämmön hyödyntämisen takia), ei oteta huomioon varoituksia ja hälytyksiä.

2.4.2 Lauhdelämmön ja kaukolämmön tehon seuranta

Tarkasteltavan järjestelmän lisälämmönlähteenä käytetään kylmlaitteiden lauhdelämpöä ja kaukolämpöä (tai yleisesti ostoenergiaa). Tavoitteena on käyttää kaukolämpöä mahdollisimman vähän. Vikadiagnostiikkajärjestelmän tulisi seurata lisälämpötehon jakautumista eri lämmönlähteiden kesken ja havaita kaukolämmön liian suuri käyttö. Kuva 7 esittää, miten ThermoNet-tekniikkakeskus voi liittyä lämmöntuottoon. Kuvasta näkyvät myös seuraavissa kaavoissa esiintyvien lämpötilojen sijainnit.



Kuva 7. ThermoNet-tekniikkakeskuksen liittyminen lämmöntuotantoon.

Kaukolämmön teho (kW) on

$$\phi_{kl} = q_{v,n} \rho_n c_{p,n} (T_1 - T_3), \quad (6)$$

jossa

$q_{v,n}$ on nesteen tilavuusvirta (m^3/s)

ρ_n nesteen tiheys (kg/m^3)

$c_{p,n}$ nesteen ominaislämpökapasiteetti (kJ/kgK)

T_3 nesteen lämpötila ennen kaukolämpösiirrintä ($^{\circ}\text{C}$)
 T_1 nesteen lämpötila kaukolämpösiirtimen jälkeen ($^{\circ}\text{C}$).

Lauhdelämmön teho (kW) on vastaavasti

$$\phi_{ll} = q_{v,n} \rho_n c_{p,n} (T_3 - T_2), \quad (7)$$

jossa

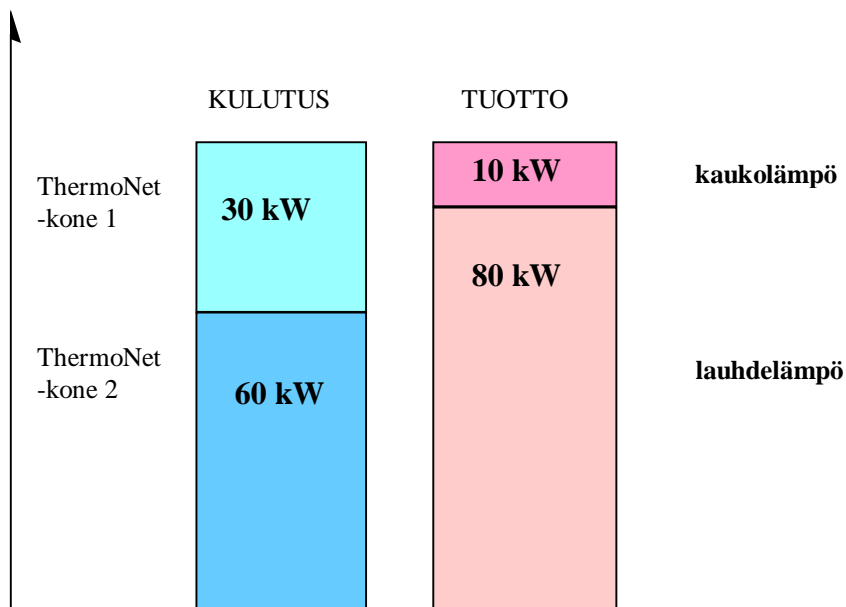
T_2 on nesteen lämpötila ennen lauhdelämpösiirrintä ($^{\circ}\text{C}$)
 T_3 nesteen lämpötila lauhdelämpösiirtimen jälkeen ($^{\circ}\text{C}$).

Kaavoissa mainitut lämpötilat täytyy mitata. Lämpötilojen perusteella voidaan laskea kaukolämmön ja lauhdelämmön suhde. Mikäli halutaan muuttaa lämpötilaerot tehoiksi, täytyy nesteen tilavuusvirta mitata (tai arvoida). Tiheydelle ja ominaislämpökapasiteetille voidaan käyttää vakioarvoja tai niiden arvot voidaan laskea lämpötiloista.

Jos lämpöhäviöitä ei oteta huomioon, ThermoNet:n LTO-piireihin syötetty teho on

$$\phi_{lto} = \phi_{kl} + \phi_{ll}. \quad (8)$$

Mikäli piiriin on kytketty useita ThermoNet-koneita, ϕ_{lto} on niiden yhteenlaskettu teho. Mikäli tarvittavat mittaukset tehdään, teho voidaan jakaa eri koneiden kesken.



Kuva 8. Kaukolämmön ja lauhdelämmön tehon seuranta.

Tehojen seurantaohjelma voisi esittää hetkelliset tehot esimerkiksi seuraavan kuvan 8 tapaan. Tämän lisäksi järjestelmän tulisi pystyä piirtämään käyriä tehoista esimerkiksi viimeisen päivän tai viikon ajalta. Tehon seurantaohjelman tulisi esittää lisäksi ainakin mitatut nesteen lämpötilat ja tilavuusvirrat sekä ulkolämpötila.

Oleennaista on löytää vertailuarvot kaukolämmön ja lauhdelämmön tehoille, jotta vikadiagnostiikkajärjestelmä osaisi päätellä, toimiiko järjestelmä tarkoituksenmukaisesti.

Lauhdutusteho on ilmeisesti useimmissa tapauksissa suurempi kuin ThermoNet-koneissa enimmillään hyödynnettävä teho. Jos näin on, kaukolämpöä käytetään vain silloin, kun lauhdelämmön lämpötilataso ei riitä. Lauhdelämmön lämpötilataso ei riitä kovilla pakkasilla, kun menolämpötilan asetusarvo on 40 °C.

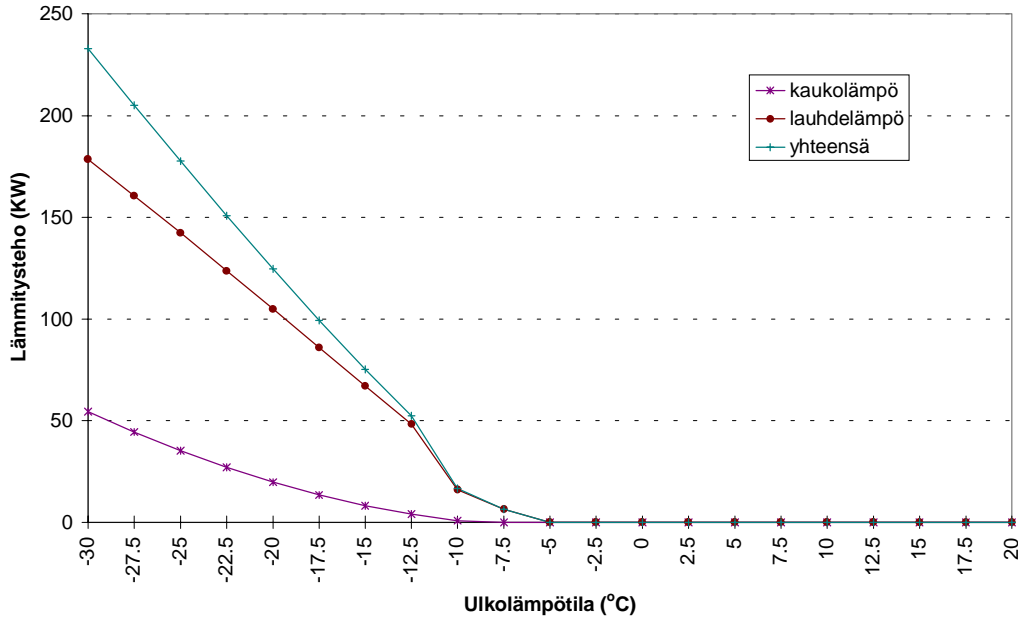
Lauhdelämmön ja kaukolämmön kulutuksen tavoitearvot tulisi esittää ulkolämpötilan funktiona. Kuvissa 9 ja 10 esitetään simuloituja arvoja eräästä marketista. Laskelmissa on oletettu lauhdelämmön nostavan lämmönsiirtoverkoston menonesteen lämpötilan aina noin 33 °C:seen. Jos lämmönsiirtoverkoston menonesteen lämpötila on tätä korkeampi, menoneste on lämmitetty asetusarvoonsa kaukolämmöllä. Kuvat eivät ole yleispäteviä, mutta ne antavat jonkinlaisen arvion lauhdelämmön ja kaukolämmön keskinäisistä suhteista.

Kuvista havaitaan, että mitä kylmempi ulkoilma on, sitä suurempi on kaukolämmön (ostoenergian) osuus koko kulutuksesta. Ostoenergiatehon ja lauhdelämpötehon suhteelle voidaan laskea tunnusluku, joka riippuu ulkolämpötilasta. Koska kaukolämmön osuus kulutuksesta on suurempi yöaikaan, kun ollaan palautusilmalla¹, tarvitaan eri tunnusluvut palautusilmakäytölle ja raitisilmakäytölle.

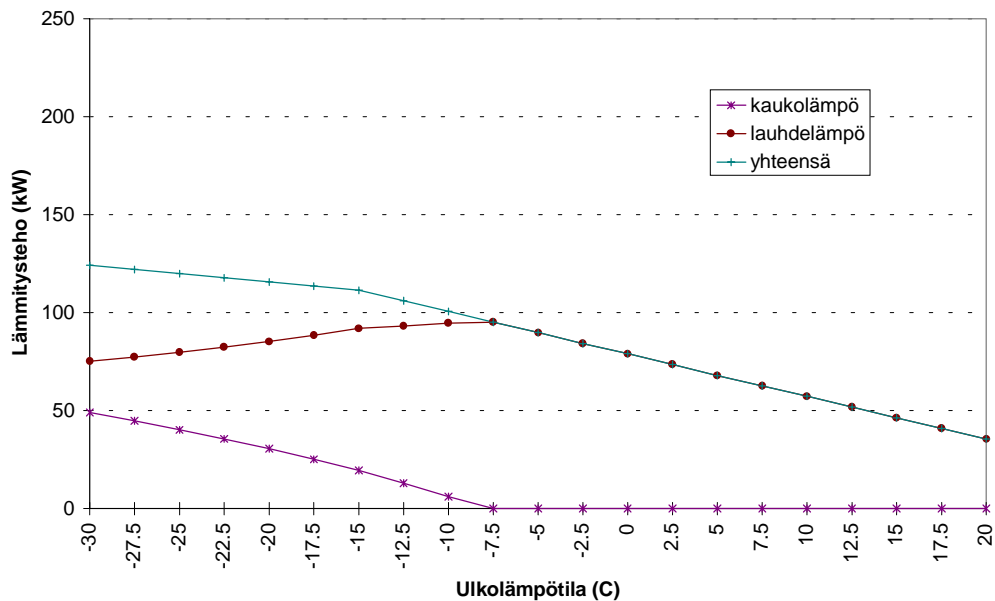
Mikäli arvioidaan aika, jonka koneet käyttävät palautusilmaa ja raitisilmaa, ja tarkastellaan tehojen sijaan energioita, voidaan laskea energiankäytön tavoitteelle yksi tunnusluku. Jos oletetaan, että koneet käyvät puolet ajasta raitisilmalla ja puolet ajasta palautusilmalla, kuvista voidaan päätellä, että ulkolämpötilalla

¹ Tähän on syynä se, että ThermoNet-piiristä palaavan nesteen lämpötila on korkeampi yöaikana. Esimerkiksi ulkolämpötilalla -20 °C ThermoNet-piiristä palaavan nesteen lämpötila on päivätilanteessa 11 °C ja yötilanteessa 21 °C. Lauhdelämpö lämmittelee molemmissa tapauksissa nesteen noin 33 °C:seen ja kaukolämmöllä neste lämmitetään 37 °C:seen. Nämä arvot on saatu simuloimalla ThermoNet-piiriä, joka on kytketty lämmönsiirtoverkoston suoralla kytkennällä (kuva 1), ei lämmönsiirtimen välityksellä (kuva 7).

- 25 °C kaukolämmön kulutuksen tulisi olla noin 25 % kokonaiskulutuksesta ja ulkolämpötilalla -15 °C noin 15 % kokonaiskulutuksesta.



Kuva 9. Lämpötehot ulkolämpötilasta riippuen päiväaikaan (IV-koneet käyttävät ulkoilmaa).



Kuva 10. Lämpötehot ulkolämpötilasta riippuen yöaikaan (IV-koneet käyttävät palautusilmaa).

Vertailuarvot kaukolämmön ja lauhdelämmön tehoille (ja niiden suhteelle) on laskettava tapauskohtaisesti. Parhaat vertailuarvot saataneen muodostamalla ne seu-

rattavasta järjestelmästä itsestään silloin, kun järjestelmä toimii tarkoituksenmukaisesti. Lauhdelämmön ja kaukolämmön tehoa seuraamalla saadaan reaaliaikaista tietoa järjestelmän toiminnasta sekä havaitaan järjestelmän epänormaali toiminta, johon voi olla syynä esimerkiksi venttiilin jumiutuminen, vuoto putkistossa tai venttiilissä, vioittuneet eristeet tms.

2.4.3 Ominaiskäyrämenetelmä

Ominaiskäyrät ovat yksinkertaisia prosessimalleja. Ominaiskäyrät esittävät kahden tai usean muuttujan riippuvuussuhteita. Tyypillisin esimerkki ominaiskäyrien käytöstä vikadiagnostiikassa on niiden käyttäminen puhaltimien ja pumppujen toiminnan seurantaan. Kuitenkin ne soveltuvat monien muidenkin prosessien seurantaan, esimerkiksi edellisessä luvussa esitetyt kuvat lämpötehojen ulkolämpötilariippuvuudesta ovat myös ominaiskäyriä. Ominaiskäyrien käyttöä vikojen havaitsemisessa käsitellään myös kohdassa 4.1.

Ominaiskäyrää kuvaavan mallin yleinen muoto on

$$y = f(x_1, x_2, \dots, x_n). \quad (9)$$

Ominaiskäyrä voidaan kuvata graafisesti, jolloin se on helposti ymmärrettävä ja havainnollinen. Jos prosessimalli kuvaa kahden muuttujan riippuvuussuhteita, toinen muuttuja esitetään x-akselilla ja toinen y-akselilla. Mikäli ominaiskäyrä esittää kolmen muuttujan riippuvuussuhteet, muodostuu käyräparvi. Laskennallisia tarpeita varten ominaiskäyrä esitetään kaavana.

Puhaltimen ja pumpun ominaiskäyrä kertoo paineenkorotuksen ja tilavuusvirran riippuvuuden toisistaan. Prosessin todellisen ominaiskäyrän muodostaminen vaatii puhaltimen paineenkorotuksen ja tilavuusvirran mittaamista. Kullakin ajanhetkellä voidaan laskea yksi ominaiskäyrän piste. Ominaiskäyrän seurannalla voidaan havaita esimerkiksi moottorin laakerivaurio tai pysähtyminen, hihnan katkeaminen, puhaltimen väärä pyörimissuunta tai epävakaa toimintapiste. Tarkasteltavan ilmastointijärjestelmän puhaltimien ja pumppujen pyörimisnopeus on portaattomasti säädettävissä. Puhaltimen ominaiskäyrä riippuu pyörimisnopeudesta, joten ominaiskäyrät muodostavat käyräparven.

2.4.4 Säädön toiminnan seuranta

Säädön tavoitteet ovat lyhyt asettumisaika, pieni poikkeama asetusarvosta ja värähtelemättömyys. Tavoitteiden toteutumista voitaisiin seurata diagnostiikkamenetelmien avulla.

Seurannan kohteena voisi olla tuloilman lämpötila (keskusilmastointikoneen jälkeisen ilman lämpötila). Menetelmä voisi olla seuraavankaltainen. Tuloilman lämpötila mitataan tietyin väliajoin ja riittävän tiheästi, jotta prosessin dynamiikka käy ilmi. Tietyn ajanjakson (viimeiset puoli tuntia tms.) aikana tehdyistä mittauksista lasketaan keskiarvo ja hajonta. Asetusarvon tulisi pysyä vakiona tänä aikana. Tuloksia voidaan verrata standardin SFS 5768 (taulukko 4) vaatimukseen, joiden mukaan lämpötilan keskiarvon suurin sallittu poikkeama asetusravosta saa olla ± 1 °C ja lämpötilan tulee olla vähintään 90 % ajasta $\pm 0,5$ °C keskiarvon molemmiin puolin. Jos nämä rajat ylitetään, menetelmä varoittaa ja ehdottaa korjaavia toimenpiteitä (esimerkiksi mihin suuntaan viritysarvoja tulisi muuttaa).

Tuloilman lämpötilan säätöä seurattaessa tulisi samalla pystyä seuraamaan ilmastointikoneeseen tulevan lisälämpö- tai lisäjähdytysnesteen lämpötilaa, sillä värähdellessään se vaikeuttaa tuloilman lämpötilan säätöä.

Säädön hyvyttä voidaan arvioida myös hyvyyskriteereillä, joita ovat esimerkiksi IAE-kriteeri (Integral Absolute Error), ITAE-kriteeri (Integral Time Absolute Error) ja ISE-kriteeri (Integral Squared Error). Virhekriteereissä summataan poikkeamia asetusravoista eri perusteilla. Mitä pienempi virhekriteerin arvo on, sitä paremmin säätö toimii.

Säädön toimintaa ei ehkä tarvitse diagnostisoida jatkuvasti, vaan tietyin väliajoin, esimerkiksi kerran viikossa. Säädön toimintaa voidaan seurata normaalissa käyttötilanteessa tai sitä voidaan testata erikseen tehtävällä testillä, esimerkiksi tekemällä portaittainen muutos asetusravoon. Mikäli järjestelmään kuuluu säädön toimintaa seuraava diagnostiikkamenetelmä, sitä voidaan hyödyntää laitosta käyttöönotettaessa.

2.4.5 Vikapuu

Prosessiin liittyvä vikatietämys voidaan jäsentää vikapuun avulla. Sen käyttö on havainnollinen tapa päätellä, missä järjestelmän kohdassa toiminta poikkeaa normaalista tai on viallinen ja mihin kaikkeen järjestelmässä tämä vaikuttaa (Kärki & Hyvärinen 1997). Vikapuu on yleiskäyttöinen menetelmä, jota voidaan soveltaa mihin tahansa järjestelmään. Vikapuuta käytetään vian paikantamiseen.

Vikapuu on organisoitu graafinen esitys ehdoista tai muista tekijöistä, jotka aiheuttavat tai myötävaikuttavat määrätyn, ei-toivottavan tapahtuman, ns. huipputahtuman, esiintymiseen (IEC 1025). Vikapuu on erityisen sopiva sellaisten monimutkaisten järjestelmien analyysiin, jotka muodostuvat useista erilaisista toimintata-

voitteet omaavista, toiminnallisesti toisiinsa liittyvistä tai toisistaan riippuvista osajärjestelmistä. Jotta vika puut olisivat mahdollisimman käyttökelpoisia, ne tulisi muodostaa käyttövarmuusanalyysin perusteella, jolloin niissä esiintyvät todennäköisimmät ja merkittävimmät viat.

Jotta vika puuanalyysi onnistuu, tarvitaan yksityiskohtaista tietoa järjestelmästä. Vika puun kehittäminen aloitetaan huipputapahtuman määrittelyllä (IEC 1025). Huipputapahtuma voi olla vaarallinen olosuhde tai järjestelmän kyvyttömyys täyttää toivottu toiminto. Huipputapahtuma tulisi määrittellä mahdollisuuksien mukaan mitattavissa olevissa yksiköissä.

Vika puussa huipputapahtuma on vika puun juurena ja huipputapahtuman synty miseen vaikuttavat syyt puun oksina (Kärki & Hyvärinen 1997). Perinteisessä vika puussa osoitetaan vain syyt tai syiden yhdistelmät, jotka voivat johtaa määrättyyn huipputapahtumaan. Kuitenkin vika puun yhteydessä voidaan käyttää oirejoukkoja, jotka kuvaavat vikojen oireet ja niiden liittymiset vikoihin. Vika puu muodostuu hierarkkisesta rakenteesta, jossa yhtä ylempään tason vika voi selittää yksi tai useampi alemman tason vika. Oirejoukkoja voidaan käyttää merkinä vian ilmaantumisesta. Sama oire voi olla merkittynä yhteen tai useampaan vikaan. Suorituskykyvaatimukset muodostavat osan vika puun vikojen oireista.

Vika puut voidaan määrittellä esimerkiksi seuraaville ilmastointikoneen huippu tapahtumille (Kärki & Hyvärinen 1997):

- Ilmastointikoneesta lähtevän tuloilman lämpötila on liian korkea.
- Ilmastointikoneesta lähtevän tuloilman lämpötila on liian matala.
- Ilmastointikoneesta ei saada riittävästi tehoa.
- Ilmastointikoneesta lähtevän tuloilman lämpötilan asettumisaika on liian pitkä.
- Ilmastointikoneesta lähtevän tuloilman lämpötila poikkeaa liikaa asetusarvosta.
- Ilmastointikoneesta lähtevän tuloilman lämpötila värähtelee liikaa.
- Lämmityspatterista lähtevän veden lämpötila on liian matala.
- Ilmastointikoneesta ei tule tarpeeksi ilmaa.
- Ilmastointikoneesta tulee liikaa ilmaa.
- Puhaltimien hyötysuhteet ovat liian korkeat.
- Puhaltimien hyötysuhteet ovat liian matalat.
- Puhaltimet kuluttavat liikaa sähköenergiaa.
- Ilmastointikone kuluttaa liikaa energiaa.
- Ilmastointikone kuluttaa liian vähän energiaa.
- Liian korkea CO₂-pitoisuus tilassa (jonka aiheuttaa ilmastointikone, ei esimerkiksi liika ihmismäärä).
- Ilmastointikoneen äänitaso on liian korkea.
- Ilmastointikoneesta lähtevän ilman kosteuden asettumisaika on liian pitkä.
- Ilmastointikoneesta lähtevän ilman kosteus värähtelee.

- Lämmöntalteenottolaitteen lämpötilahyötysuhde on liian matala.
- Lämmöntalteenottolaitteen lämpötilahyötysuhde on liian korkea.
- Ilmastointikone pysähtyy odottamattomasti.

Jotkut vikapuista voivat olla toistensa osavikapuita. Esimerkiksi huippu-
tapahtuman "ilmastointikone kuluttaa liikaa energiaa" yhtenä syynä voi olla toinen
huipputapahtuma "ilmastointikoneesta lähtevän tuloilman lämpötila värähtelee
liikaa". Kaikista mahdollisista huipputapahtumista ei tehdä vikapuuta vaan vain
niistä, jotka ovat kulloinkin kyseessä olevalle järjestelmälle oleellisia.

Seuraavassa esitetään esimerkki erään ilmastointikoneen vikapuusta (Kärki & Hy-
värinen 1997). Viat on esitetty isoilla kirjaimilla ja oireet pienillä kirjaimilla. Jos
mitattua oiretta ei yleensä mitata, se on merkitty sulkuihin.

LIIAN SUURI CO₂-PITOISUUS TILASSA

mitattu CO₂ ylittää tavoitellun tason

KONE PYSÄHTYNYT

koneesta ei lähde ääniä

SÄHKÖHÄIRIÖ

IV-HÄTÄPYSÄYTYS

TULOILMAPUHALLIN PYSÄHTYY

mitattu paine-ero on mitätön

(mitattu ilmavirta on mitätön)

(ei sähkövirtaa)

MOOTTORI RIKKI

HIHNA POIKKI

tarkastus

SULAKKEET PALANEET

tarkastus

LÄMPÖSUOJA RIKKI

PALO-OHJAUS

AUTOMAATIOVIKA

VÄÄRÄ AIKAOHJELMA

AUTOMAATIOLAITTEEN KELLO SEKAISIN

OHJELMOINTIVIRHE

PIENI TULOILMAVIRTA

(ilmavirta merkittävästi alle tavoitearvon)

MITOITUSVIRHE

ULKOILMAPELTI JUUTTUNUT

tarkastus

SUODATIN TUKOSSA

mitattu paine-ero on suuri

LTO TUKOSSA, KUN OHITUSPELTI JUUTTUNUT KIINNI
(mitattu paine-ero on suuri)

tarkastus

LÄMMITYSPATTERI TUKOSSA

tarkastus

PUHALLIN PYÖRII VÄÄRINPÄIN

(painesuhteet väärin)

AUTOMAATIOVIKA

VÄÄRÄ AIKAOHJELMA

AUTOMAATIOLAITTEEN KELLO SEKAISIN

OHJELMOINTIVIRHE

CO₂-ANTURIN VIKA

vertailu toisen mittarin näyttämään

valitukset

VÄÄRÄ KALIBROINTI

EI KYTKETTY

VÄÄRÄ PAIKKA

2.4.6 Muita vikadiagnostiikkamenetelmiä

Kansainvälisessä IEA Annex 25 -projektissa (Real Time Simulation of HEVAC Systems for Building Optimisation, Fault Detection and Diagnosis (BOFD)) kehitettiin LVI-järjestelmien vikadiagnostiikkamenetelmiä. Kärki & Hyvärinen (1997) ovat esittäneet yhteenvedon tässä projektissa kehitetyistä ilmastointikoneen vikadiagnostiikkamenetelmistä. Tarkasteltavan ilmastointijärjestelmän kannalta näistä menetelmistä mielenkiintoisimpia voisivat olla vikapuuovellus ilmastointikoneelle (Yuzawa et al. 1996) sekä puhallintehon käyttö vian havaitsemisessa ja ilmanvaihtojärjestelmien toiminnan seurannassa (Norford & Little 1993).

Myös muut kuin ominaiskäyrään perustuvat menetelmät ovat mahdollisia pumpujen ja puhaltimien seurannassa. Rosenhave (1997) on esittänyt menetelmän, joka perustuu siihen, että pumpun ääni muuttuu sen vioittuessa.

2.5 Yhteenveto luvusta 2

Ilmastointikoneen suorituskykyvaatimusten perusteella voidaan kehittää ilmastointikoneen toiminnan seurantaan soveltuvia työkaluja. Luvussa 2 on esitetty useita mahdollisia tarkasteltavan järjestelmän diagnostiikkamenetelmiä. Esitetyt menetelmät keskittyvät keskeisten suorituskykytekijöiden seuraamiseen.

Suorituskykykriteerien perusteella voidaan toteuttaa tiettyyn järjestelmään räätälöityjä sovelluksia järjestelmän toiminnan seurantaan. Suorituskykyvaatimukset liittyvät vian havaitsemiseen, jolloin vian paikantaminen on tehtävä muilla keinoilla. Paikantamisessa voidaan käyttää apuna esimerkiksi vikapuita. Suorituskykytekijöiden määrittäminen ja vikapuu ovat menetelminä yleiskäyttöisiä eli niitä voidaan soveltaa myös muihin järjestelmiin kuin tarkasteltavaan järjestelmään.

Ilmastointikoneen suorituskyvyllä ei ole olemassa yksityiskohtaisia viranomaismääräyksiä, lähinnä voidaan puhua väljistä suosituksista. Ilmanvaihto- ja ilmalämmitysjärjestelmien keskeiset tavoitteet ovat hyvä sisäilman laatu, alhainen energiankulutus, alhaiset käyttökustannukset ja käyttövarmuus. Työssä on esitetty vertailuarvoja, jotta voitaisiin todentaa em. tavoitteiden toteutuminen. Vertailuarvoja antavat lait, määräykset, suositukset, ohjeet ja suunnittelu. Tarkasteltavan ilmastointikoneen tärkein tehtävä on tuottaa haluttu määrä puhdasta ilmaa sekä lämmitys- tai jäähdytystehoa. Tavoitteisiin tulisi päästä energiataloudellisesti. Taloteknisten järjestelmien suorituskykyyn vaikuttavat monet suunnittelun-, valmistuksen-, asennuksen- ja käytönaikaiset ratkaisut. Eri elinkaarivaiheissa tehtävien ratkaisujen vaikutusta järjestelmän lopulliseen suorituskykyyn on eritelty.

Vikadiagnostiikkamenetelmiä käytetään apuna järjestelmän suorituskykytekijöiden seurannassa. Niiden avulla pyritään havaitsemaan ja tulkitsemaan suorituskykytekijöissä aiheutuneet poikkeamat. Vikadiagnostiikkamenetelmien tulisi keskittyä havaitsemaan viat, joilla on suurimmat vaikutukset tai joiden esiintyminen on todennäköisintä. Olennaista vikadiagnostiikkamenetelmien toiminnalle on hälytyksen kynnyksrajan asettaminen. Liian alhainen kynnyksraja aiheuttaa turhia hälytyksiä. Jos kynnyksraja on liian korkea, vikoja jää havaitsematta. Työssä on esitetty erilaisia diagnostiikkamenetelmiä, joissa on pyritty yksinkertaisuuteen ja helppoon sovellettavuuteen.

Tarkasteltavan ilmastointikoneen lisälämpötehon jakautumista kylmlaitteiden lauhdelämmön ja kaukolämmön (tai yleisesti ostoenergian) kesken voidaan seurata työssä kehitetyllä menetelmällä. Olennaista on löytää vertailuarvot kaukolämmön ja lauhdelämmön tehoille, jotta voidaan päätellä, toimiiko järjestelmä tarkoituksenmukaisesti. Nämä vertailuarvot on todennäköisesti laskettava tapauskohtaisesti.

Puhaltimen ja pumpun toimintaa voidaan seurata ominaiskäyrän avulla. Kullakin ajanhetkellä voidaan laskea yksi todellisen ominaiskäyrän piste. Sitä voidaan verrata annetun ominaiskäyrän pisteeseen, tai todellista ominaiskäyrää voidaan verrata annettuun ominaiskäyrään.

3 Menetelmät ja ohjelmistot asiantuntijatiedon käsittelyyn

Tämän tehtävän tavoitteena oli määritellä, millaisia ominaisuuksia vikadiagnoosissa tarvittavan asiantuntijatiedon käsittelyssä käytettävillä ohjelmistotyökaluilla tulisi olla. Lisäksi tavoitteena oli selvittää, onko sopivia tuotteita markkinoilla.

3.1 Vikadiagnoosijärjestelmän yleisrakenne

Mallipohjainen vikadiagnoosi hyödyntää vian selvittämisessä tarkasteltavan prosessin komponenttien ja osaprosessien malleja. Mallit voivat kuvata prosessin normaalia tai vikaantunutta toimintaa. Prosessin mallina voidaan käyttää esimerkiksi dynaamista mallia, staattista mallia tai ominaiskäyrää, prosessista laskettuja tunnuslukuja, kvalitatiivista mallia, neuraaliverkkoon perustuvaa mallia tai heuristista mallia.

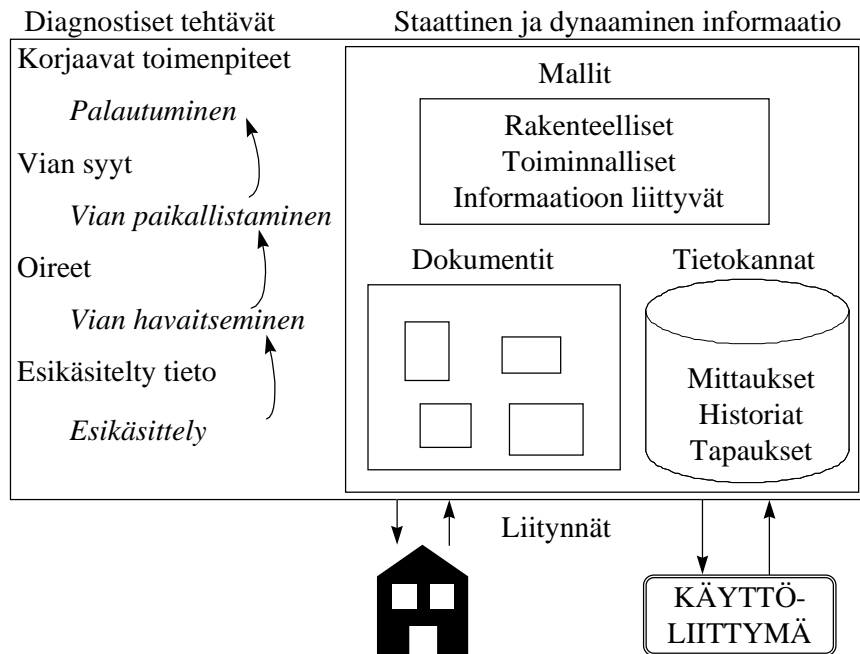
Malleille voidaan esittää esimerkiksi seuraavia vaatimuksia (Kurki 1995):

- Mallien kehitystyön tulee olla kannattavaa ja joustavaa.
- Mallien tulee olla helposti sovellettavissa ja räätälöitävissä.
- Mallien päivittämisen pitää olla joustavaa.
- Sovellusalueen asiantuntijoiden pitää pystyä muokkaamaan ja päivittämään malleja.

Kurki (1995) on esittänyt väitöskirjassaan mekatronisten järjestelmien mallipohjaisen vikadiagnoosijärjestelmän yleisen rakenteen, jota voidaan soveltaa myös rakennuksen vikadiagnoosijärjestelmien perustana (kuva 11). Sovelluskohtainen tietämys ja tieto esitetään malleina. Diagnostiset toiminnot implementoidaan diagnostisina tehtävinä, jotka käyttävät mallien sisältämää tietoa ja tietämystä. Vikadiagnoosijärjestelmä voidaan kuitenkin implementoida paljon suppeammassakin laajuudessa kuin mitä yleinen rakennekuvaus antaa ymmärtää.

Diagnostiset tehtävät ohjaavat vikadiagnoosijärjestelmän toimintaa sekä generoivat perussyöttötiedoista oireita, vikojen syitä ja tarvittavia korjaavia toimenpiteitä sekä automaattisesti että käyttäjän avustamana. Esikäsittely (preprocessing) muuttaa automaattisesti syöttötiedot, esimerkiksi prosessimittaukset, vian havaitsemisessa tarvittavaan muotoon sekä suodattaa vioittuneiden antureiden arvot. Vian havaitseminen (fault detection) generoi oireet mallien ja esikäsittelyn datan perusteella. Vian paikantaminen (fault localization)

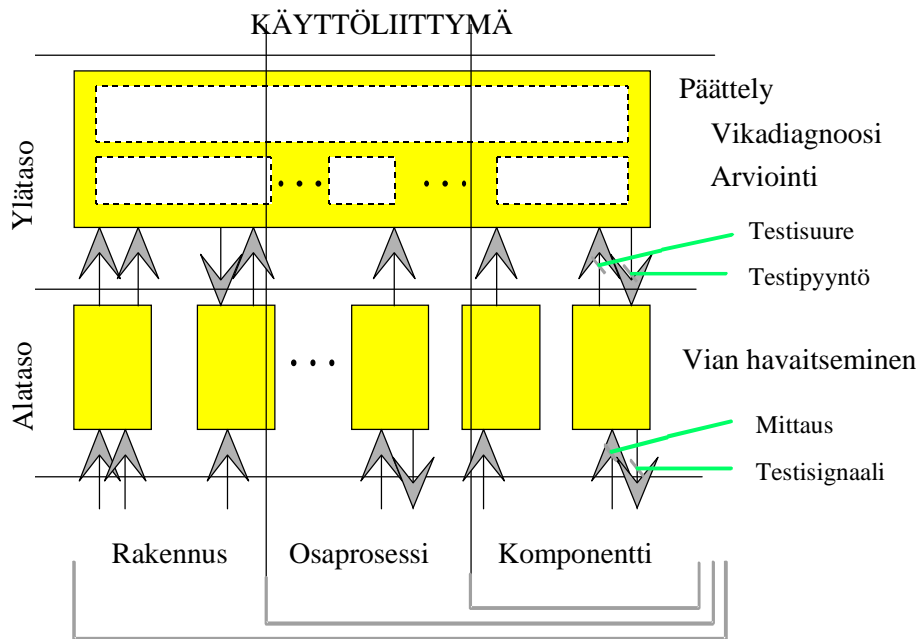
määrittelee vikojen syyt generoitujen oireiden perusteella. Vian syyn tietäminen ei välttämättä tarkoita sitä, että oikea korjaava toimenpide seuraa ilmeisenä vasteena, vaan se määritetään palautumistehtävässä (recovery).



Kuva 11. Mallipohjaisen vikadiagnoosijärjestelmän perusrakenne ja siihen liittyvät diagnostiset tehtävät.

Vikadiagnoosijärjestelmä voi sisältää diagnostisiin tehtäviin liittyvää staattista ja dynaamista informaatiota. Staattista informaatiota ovat esimerkiksi kaikki dokumentit. Tietokannat sisältävät tietoa aikaisemmasta tapaushistoriasta ja dynaamista mittaustietoa, jota kerätään järjestelmän toiminnan aikana.

Mallit ovat aina sovellusaluekohtaisia ja voivat kuvata esimerkiksi sovelluksen rakennetta tai toimintaa. Rakenteelliset mallit voivat kuvata sovelluksen graafista tai loogista rakennetta. Graafinen rakenne on selvintä esittää kuvana tai kaaviona. Looginen rakenne voi olla esimerkiksi järjestelmän oliopohjainen luokkarakenne. Vikapuukin (Kärki & Hyvärinen 1997) esittää järjestelmän rakenteen siinä mielessä, että järjestelmätasolta edetään komponenttitasolle tai toisinpäin. Vikadiagnostiikan kannalta toiminnalliset mallit ovat tärkeimpiä, koska niiden avulla seurataan järjestelmän toimintaa ja havaitaan vikoja. Informaatioon liittyvät mallit voivat kuvata esimerkiksi järjestelmän vikatietämystä eli vikapuu on myös tällainen malli.



Kuva 12. Vikadiagnosijärjestelmän toiminnallinen rakenne.

Edellä esitetty Kurjen lähestymistapa ottaa osittain kantaa jo vikadiagnosijärjestelmän toteutukseenkin staattisen ja dynaamisen informaation esittämisen osalta, vaikka perusrakenteessa ei olekaan määritelty yksittäisiä työkaluja. Ruokonen & Lautala (1986) ovat esittäneet reaaliaikaiselle vikadiagnosijärjestelmälle hierarkkista ja modulaarista järjestelmärakennetta, joka perustuu toimintoihin ja on toteutettavissa palasina. Hyvärinen (1996) on soveltanut tätä ideaa rakennuksen vikadiagnosijärjestelmän rakennemäärittelyyn (kuva 12).

Vikadiagnosijärjestelmän alatason menetelmät ovat modulaarisia ja riippumattomia muista moduuleista. Jokainen moduuli seuraa yksittäistä, hyvin määriteltyä prosessiyksikköä (rakennusta, osaprosessia tai komponenttia). Yhden prosessiyksikön toimintaa saattaa valvoa useita seurantamoduuleita, joista jokainen seuraa tiettyä virhetoimintaa tai käyttää tiettyä menetelmää.

Jokainen alatason moduuli generoi mittauksista yhden tai useamman testisuureen tai oireen, joka/jotka välitetään ylätasolle päättelyyn. Päättelyn tavoitteena on diagnosoida eli paikantaa vika. Päättelyssä voidaan käyttää apuna myös käyttäjältä saatavaa tietoa. Tässä työssä oli päätavoitteena selvittää juuri tällä tasolla tarvittavan asiantuntijatiedon käsittelyyn soveltuvia ohjelmistotyökaluja.

3.2 Tarpeet ja vaatimukset

Hjerpe (1983) ja Williams (1984) ovat luetelleet älykkäältä, tietokantapohjaiselta hakujärjestelmältä vaadittavia ominaisuuksia seuraavasti:

1. Käyttäjä voi halutessaan esittää tiedontarpeensa luonnollisella kielellä. Tavoitteena on, että käyttäjä voi tiedontarpeen sijasta kuvailla ongelman, jonka ratkaisemiseen hän tarvitsee tietoa.
2. Järjestelmä valitsee sopivat hakusanat ja muodostaa hakustrategian käyttäjän kanssa käymänsä keskustelun perusteella.
3. Järjestelmä valitsee sopivat tietokannat hakua varten.
4. Järjestelmä muuttaa hakustrategian kunkin tietokantajärjestelmän vaatimaan muotoon.
5. Järjestelmä pystyy iteratiivisesti parantamaan hakustrategiaa keskustelemalla käyttäjän kanssa.
6. Järjestelmä muistaa käyttäjän aikaisemmat kyselyt, ts. muodostaa kuvan käyttäjän tiedontarpeista.
7. Järjestelmä pystyy arvioimaan haun tuloksen hyvyttä pyytämällä käyttäjältä palautetta ja oppimaan tehokkaammaksi.
8. Järjestelmä yhdistelee useammista tietolähteistä hankkimaansa tietoa, mukaan lukien käyttäjän omat tiedot.

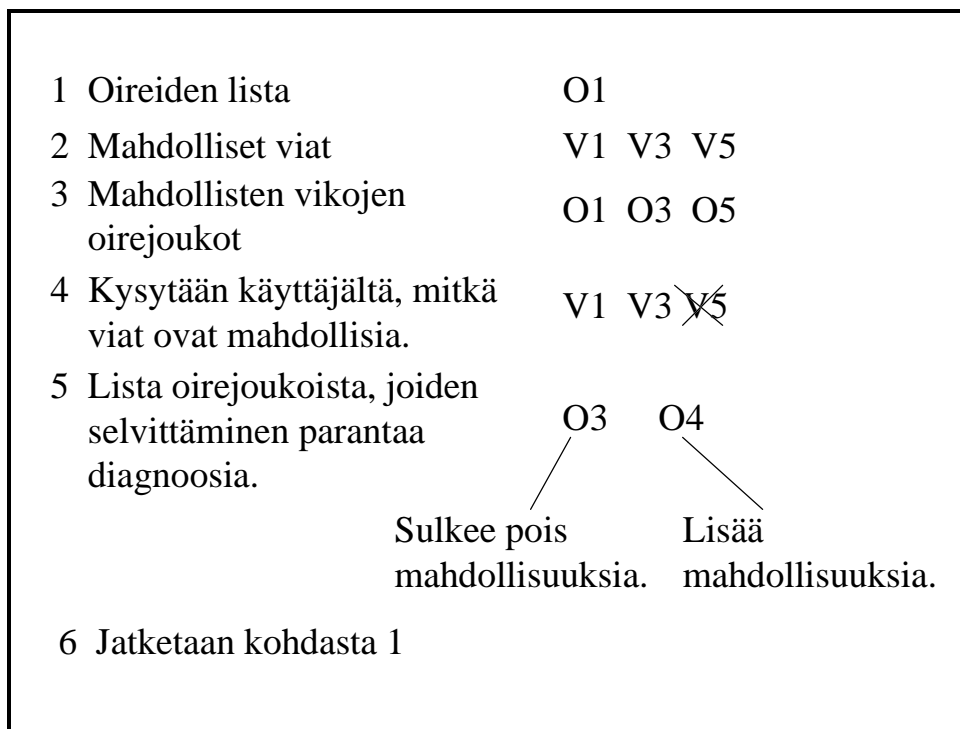
Hyvärinen (1987) on esittänyt perustoimintoja, jotka vikadiagnoosijärjestelmän (kuva 12) käyttöliittymällä voisi olla.

1. Järjestelmä tulostaa jatkuvasti näytölle luetteloä vioista, joita kunkin hetken testisuureiden arvojen ja päättelyn mukaan valvottavassa prosessissa on. Nämä viat ovat vikadiagnoosijärjestelmän kannalta todennäköisimpiä vikoja.
2. Käyttäjä voi tarkastella kohdan 1 luettelon kuhunkin vikaan liittyviä päättelyketjuja tai testisuureiden arvoja eli pyytää selitystä ko. vikaan liittyvästä päättelystä.
3. Käyttäjä voi tarkastella kohdan 1 luettelon kuhunkin vikaan liittyvää luetteloä sellaisista muista mahdollisista vioista, jotka voisivat aiheuttaa samanlaisen oirejoukon. Jokaiselle tämän luettelon vialle täytyisi myös saada selitys siihen johtaneesta päättelystä. Jako todennäköisimpään ja muihin mahdollisiin vikoihin tehdään päättelysäännöissä määriteltyjen kriteerien mukaan.
4. Käyttäjä voi hylätä molempien em. luetteloiden vikoja eli ilmoittaa, että ne eivät missään tapauksessa ole häiriötilanteen aiheuttajia. Hylkäämisten tuoman lisätiedon perusteella molemmat luettelot täytyisi päivittää uudelleen. Hylkäämistä sijasta voisi diagnostisointia parantaa käyttäjän ja järjestelmän välisen "keskustelun" avulla.
5. Käyttäjä voi hyväksyä jonkin vian eli todeta, että häiriön syy on joko järjestelmän ehdottama todennäköisin vika tai jokin todennäköisimpään vikaan liittyvä muu mahdollinen vika. Hyväksymisen jälkeen diagnostisoidusta viasta

seuraavat vikadiagnoosijärjestelmän testisuureisiin ja mahdollisesti myös automatiojärjestelmän prosessisuureisiin liittyvät hälytykset voisi poistaa näytöstä tai ne voisi merkitä selitetyiksi.

6. Järjestelmä tulostaa pyydettyä selvitetyä vian korjausohjeet.
7. Järjestelmä ylläpitää luetteloa, johon on merkitty hälytystilassa olevat testisuureet.
8. Käyttäjä voi tarkastella kaikkien vikadiagnoosijärjestelmään liittyvien testisuureiden arvoja trendi- ja/tai numeeristen näyttöjen avulla.

Kärki & Hyvärinen (1996) ovat esittäneet yleisen konseptin siitä, miten vikadiagnostiikassa hyödynnettävän päättelyjärjestelmän tulisi toimia (kuva 13). Tästä lähestymistavasta on annettu esimerkkejä vikapuusovelluksista, mutta se soveltuu käytettäväksi yleisemminkin diagnostisoitaessa vikoja. Tässä lähestymistavassa käyttöliittymä näyttää ensin kaikki aktiiviset oireet. Tämän jälkeen päätellään, mitkä viat ovat mahdollisia, ja listataan ne käyttäjälle. Samassa yhteydessä kerrotaan mahdollisten vikojen kaikki oireet eli nekin, joita ei ole havaittu. Jos samanlainen oirejoukko osoittaa useita mahdollisia vikoja, kysytään käyttäjältä, mitkä viat todella ovat mahdollisia. Lopuksi näytetään lista oirejoukoista, joiden selvittäminen parantaa diagnoosia joko sulkemalla pois joitakin mahdollisuuksia tai lisäämällä mahdollisuuksia. Kaikki nämä vaiheet käydään läpi niin monta kertaa, että vika on paikannettu tai havaittu ennalta tuntemattomaksi.



Kuva 13. Päättelyn etenemisperiaate vikadiagnoosisovelluksessa.

Pobil & Arroyo (1996) ovat esittäneet vaatimuksia asiantuntijajärjestelmien kehitysympäristöjen valinnalle, mutta osaa esitetyistä vaatimuksista voidaan soveltaa myös muiden työkalujen valintaan. Nämä vaatimukset voidaan ryhmitellä seuraavasti:

- Käyttäjään liittyvät ominaisuudet
 - Lopullisen käyttöliittymän ominaisuudet
 - Selitysvalmius
 - Graafiset tulokset
 - Käytön opastus
 - Ikkunat ja menut
 - Täydellinen ja helppotajuinen dokumentointi
 - Kehitysvaiheen ominaisuudet
 - Dokumentaatio
 - Grafiikat
 - Käytön opastus
 - Ikkunat ja menut
 - Selitysvalmius
 - Nopea prototyypin kehitys
 - Muokkaus- ja vianetsintäominaisuudet
- Tekniset vaatimukset
 - Järjestelmäominaisuudet
 - Siirrettävyys
 - Sulautettavuus
 - PC-tuki
 - Tuki monikäytölle
 - Pääteläkoneen ominaisuudet
 - Tieto-ohjattu eli etenevä eli eteenpäin ketjutettu haku (forward chaining)
 - Tavoiteohjattu eli perääntyvä eli taaksepäin ketjutettu haku (backward chaining)
 - Sääntöjen prioriteetin määrittäminen
 - Luotettavuuskertoimet
 - Tietämiskannan ominaisuudet
 - Tuotantosäännöt
 - Kehykset
 - Sääntöjen jakaminen joukoiksi
 - Pääteläkoneen mekanismit
 - Dataliittynät
 - Liittynät tietokantoihin
 - Liittynät ohjelmointikieliin
 - Liittynät muihin ohjelmiin
- Hinta

- Myyjä
 - Käyttäjätuki
 - Yrityksen asiantuntemus

3.3 Työkaluohjelmia

Vikadiagnostiikkasovelluksissa ja -kehitysympäristöissä käytettävät ohjelmistot voidaan luokitella varsinaisiin ohjelmointikieliin, tietämyspohjaisiin ohjelmiin, sumean logiikan ohjelmiin, neuraaliverkkoihin sekä Matlabin-tyyppisiin valmisohjelmiin. Kutakin eri ohjelmistotyyppiä voidaan käyttää erilaisten päättelytapahotumien kuvaamiseen. Seuraavassa kuvataan lyhyesti joitakin jaottelunmukaisia ohjelmia. Ohjelmistotuotteiden WWW-osoitteet löytyvät liitteestä 1.

3.3.1 Ohjelmoinnin perustyökalut

Päättelyrakenteiden suora ohjelmoiminen ohjelmointikielillä on mahdollista ja pienissä sovelluksissa viisasta. Yksinkertaista päättelyrakennetta ei kannata työstää ohjelmointikieliin perustuvilla sovelluskehittimillä tai muilla työkaluilla, jos sen tekeminen perinteisillä ohjelmointikielillä on joustavaa. Usein ohjelmointikieliet myös suoraan tukevat muun kehitysympäristön rakentamista.

C-ohjelmointikieli

C on erityisesti systeemiohjelmointiin mutta myös numeeriseen laskentaan, tekstinkäsittelyyn yms. soveltuva lausekieli (Korpela & Larmela 1989). C käsittelee ensisijaisesti sellaisia tietoalkioita, joilla konekäskyt tyypillisesti operoivat: lukuja, merkkejä ja osoitteita. Ohjausrakenteet ovat suhteellisen kehittyneet ja tukevat rakenteista ohjelmointia. C tarjoaa käytettäväksi laajan joukon yksinkertaisia operaatioita sekä välineet monimutkaisten kokonaisuuksien rakentamiseksi. C-kääntäjiä ilmestyy markkinoille tasaista tahtia. Uusimmat kääntäjät tukevat hyvin Windowsin eri ominaisuuksia ja mahdollistavat erilaisten linkkien, kuten OLE, DDE ja DLL, tekemisen C-ohjelmista. C++ on C:n laajennus, jonka tärkein lisäys ovat luokat. C++:n tärkeimpiä etuja ovat luotettavuuden lisääntyminen ja määrättyyn ongelmaan sopivien ohjelmaelementtien luominen.

Java

Java (http://www.jyu.fi/~jankinnu/harjoitus/tko_d10/java.html) on Internet-ympäristössä laajalti käytetty ohjelmointikieli. Sen suosio on paljolti ilmaisjakelun ansiota. Java-kieli soveltuu Internet-käyttöön erityisesti laiteriippumattomuutensa ansiosta. Laiteriippumattomuus vähentää myös ohjelmistokehityksen kustannuksia poistamalla tarpeen ohjelmoida useammille laiteympäristöille omat koodit. Java on kehitetty ehkä maailman käytetyimmistä korkean tason ohjelmointikielestä C++:sta tehden siitä erittäin kehittyneen oliosuuntautuneen kielen. Java-kielen leviämistä suuren yleisön käyttöön nopeuttaa ohjelmistotalojen sekä vapaaehtoisvoimin luotujen kehitysympäristöjen ilmestyminen.

Helpon Java-kieleen törmää selaillessaan WWW-sivuja jollain Java-kieltä tukevalla selaimella. HTML-dokumenttiin on mahdollista liittää Java-ohjelmia, jolla saadaan aikaan erilaisia operaatioita. Appletit eli pienet verkossa toimivat Java-kieliset ohjelmat, ovat levinneet laajalti juuri Internetissä. Applettien käyttömahdollisuuksia on rajaton määrä ja oikeastaan vain mielikuvitus on rajana siihen, mitä niillä voi tehdä.

LISP

LISP-kieli, joka kehitettiin 1950-luvulla, on vahvasti liittynyt tekoälytutkimukseen. LISP on funktionaalinen ohjelmointikieli, jossa on prosessuaalisia laajennuksia. LISP (LISt Processor) suunniteltiin erityisesti epäyhtenäisten listojen, tyypillisesti symbolilistojen, käsittelyyn. LISPin ominaisuuksia, jotka tekivät siitä houkuttelevan tekoälytutkijoille, ovat: ajoaikainen tyyppin tarkastus, korkean tason funktiot, automaattinen muistinhallinta ja vuorovaikutteinen ympäristö. Valtaosa markkinoilla olevista symbolien käsittelyn, tekoälyn ja luonnollisen kielen ohjelmistoista on laadittu LISPillä (Hyvönen & Seppänen 1987). Monet tekoälyssä käytettävät tekniikat perustuvat LISP-kielen erikoisiin ominaisuuksiin.

LISPin ominaisuuksia:

- Tiedon ja ohjelmien esitysmuotona olevat listat koostuvat muistisoluista, joiden sijainnilla ja järjestyksellä muistissa ei ole merkitystä.
- Järjestelmä varaa ja vapauttaa muistia automaattisesti tarpeen mukaan.
- Funktionaalinen ohjelmointi perustuu siihen ajatukseen, että jokaisesta toimituksesta syntyy tulokseksi arvo.
- LISPissä voidaan käyttää myös tavanomaista peräkkäisiin käskyihin perustuvaa ohjelmointia.
- LISP on ensisijaisesti tulkitseva kieli.

- LISPissä symbolien, muuttujien, listojen, funktioiden ym. olioiden nimiä ei ole ennalta kiinnitetty mihinkään tietotyyppiin.
- LISP on samalla sekä sovellutus- että systeemiohjelmointikieli.

Prolog

Toinen vahvasti tekoälyyn liittyvä kieli on PROLOG (PROgramming in LOGic), joka kehitettiin 1970-luvulla. Prolog perustuu ensimmäisen kertaluvun logiikkaan. Prolog sisältää lukuisia ominaisuuksia, joita ei löydy normaaleista ohjelmointikielistä (Doores et al. 1987). Näitä ominaisuuksia ovat tehokas haku- ja päättelymekanismi, tehokas hahmontunnistuskonkreettinen ja yksinkertainen mutta hieno tietorakenne. Prologissa tieto ja ohjelma ovat eri tapoja katsoa Prolog-olioita. Koska ohjelmaa ja tietoa ei erotella, Prologilla on syötetyn tiedon perusteella kyky muuttaa Prolog-ohjelmaa suoritettaessa. Prologissa ei ole osoittimia, määräyksiä eikä hyppylauseita. Rekursio on hyvin luonnollinen ohjelmointitekniikka Prologissa. Pelkistettyjä DOS-versioita Prologista on saatavilla Internetistä. Windows-ympäristöön Prologista on saatavilla myös kaupallisia kehitysversioita, jotka tukevat graafisten käyttöliittymien tekemistä.

3.3.2 Ohjelmointiympäristöt

Amzi! Prolog + Logic Server

Amzi! Prolog + Logic Server (Amzi! Inc.) on Prolog-kehitysympäristö, jolla voi tehdä Prolog-pohjaisia osia sulatettavaksi esimerkiksi C/C++:lla, Delphillä, Visual Basicilla tai Javalla tehtyyn ohjelmaan. Amzi!:lla voi tehdä myös logiikkasääntöjä, jotka voidaan integroida ODBC-tietokantoihin. Amzi! edellyttää Prologin sekä sen työkalun osaamista, johon se integroidaan.

Delphi

Delphi on suorituskykyinen, vakaa ja ominaisuuksiltaan monipuolinen ohjelmointityökalu sekä tavanomaisten että asiakas-palvelinsovellusten tekoon Windows-ympäristössä (Becks 1995). Delphin tehokkuutta sovellusten rakentamisessa kuvaa hyvin se, että Delphi itse on ohjelmoitu Delphillä. Delphin ohjelmointikieli on itse asiassa Pascal, tarkemmin sanottuna Borlandin Object Pascal. Delphin vahvuutena pidetään myös sitä, että samalla välineellä voidaan tehdä niin prototyyppi kuin lopullinen sovelluskin. Delphiä verrataan usein Visual Basiciin, joskin sen sovellusalue on laajempi ja ohjelmointiympäristö monipuolisempi. Pakanen et al. (1996b) esittävät esimerkin Delphin käytöstä LVI-järjestelmän vikadiagnostiikkasovelluksessa.

GCLISP Developer

GCLISP Developer on Windows-ympäristössä toimiva graafinen LISP-ohjelmointi- ja -kehitysympäristö, joka tukeutuu Common LISPiin. Ympäristö tukee DDE- ja DLL-linkkejä. Osa sovelluksesta voidaan ohjelmoida myös C-kielillä. Kehitysympäristöllä voidaan tuottaa Runtime-sovelluksia kaupallisiin tarkoituksiin.

Visual Basic

Visual Basic on helppokäyttöinen ohjelmointiympäristö ja sopii monenlaisten Windows-ohjelmien tekoon (Suvisaari & Suvisaari 1995). Visual Basicia käytetään paljon tietokantaohjelmien edustaohjelmien tekoon. Sillä ei kuitenkaan kannata tehdä isoa piirto-ohjelmaa eikä tekstinkäsittelyä. Visual Basicia ei myöskään kannata käyttää, jos ohjelmaprojektiin kuuluu monta tekijää, koska tällöin ohjelman hallinta on hankalaa. Visual Basicin puutteita voi kuitenkin kiertää tekemällä osan ohjelmasta C- tai C++-kielillä. Visual Basic -ohjelmat kuluttavat paljon muistia ja järjestelmäresursseja. Vikadiagnostiikkaympäristöä ajatellen Visual Basic soveltuu hyvin ympäristön rungoksi ja sillä on helppo tehdä käyttöliittymä. Kuitenkin vain yksinkertaiset päättelyrakenteet kannattaa toteuttaa suoraan Visual Basicilla. Pakanen et al. (1996b) ovat havainnollistaneet vikadiagnosioisovellusten tekemistä Visual Basic -ympäristössä.

Visual Prolog

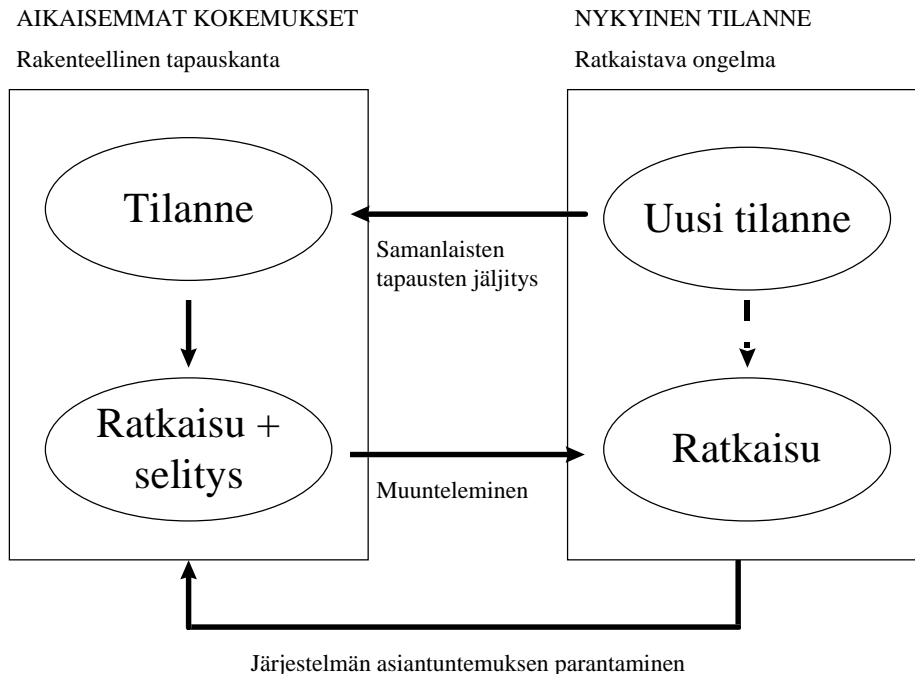
Visual Prolog (Prolog Development Center) on Windows-pohjainen Prolog-kehitysympäristö. Visual Prolog -ohjelmat voi linkata yhteen C/C++:lla tai Assemblerilla tehtyjen kirjastojen tai ohjelmien kanssa. Toimintoja voi kutsua myös Excelistä tai Wordistä.

WIN-PROLOG

WIN-PROLOG (LPA - Logic Programming Associates) on johtava (<http://www.lpa.co.uk/win.html>) Prolog-kääntäjä ja -kehitysympäristö. Siinä on kaksisuuntainen DLL-liityntä ja muokattavissa oleva DDE-liityntä.

3.3.3 Tietämuspohjaiset työkaluohjelmat

Tapauspohjaiset päättelytyökalut



Kuva 14. Tapauspohjaisen päättelyn periaate.

Tapauspohjainen päättelyprosessi käyttää aikaisempia kokemuksia ja perustuu siihen ehtoon, että ihmiset käyttävät analogista päättelyä tai kokemusperäistä päättelyä oppiakseen ja ratkaistakseen monimutkaisia ongelmia. Kuva 14 esittää tapauspohjaisen päättelyn dynamiikan, jossa nykyisessä tilanteessa ratkaistavana olevaa ongelmaa lähestytään vertaamalla tilannetta aikaisempiin kokemuksiin. Tapauskannasta saattaa löytyä suoraan ratkaisu ja siihen kuuluva selitys tai sitten uuden ongelman ratkaisu etsitään muuntelemalla aikaisemmista tapauksista. Tällöin tapauskanta on päivitettävä sisältämään myös juuri ratkaistu ongelma.

Tapauspohjainen päättely on käyttökelpoista, kun käytettävissä on vähän tietoa tai se on epätäydellistä. Tapaukset sisältävät tietoa tilanteesta, ratkaisun, vastaukset käyttäen ratkaisua ja avainattribuutteja, joita voidaan käyttää samanlaisten tilanteiden nopeaan hakuun. Jos riittävän sopivaa tapausta ei löydy, voidaan käsiteltävä tapaus ja ihmisen siihen luoma ratkaisu lisätä tapauskantaan ja siten antaa järjestelmän oppia.

Tapauspohjaisissa päättelyjärjestelmissä on useita tapoja kuvata, painottaa, indeksoida, jäljittää ja sovittaa tapauksia (Harmon 1992). *Nearest-neighbor indexing* sallii järjestelmän paikallistaa tapauskannasta sellaiset tapaukset, jotka ovat

jollakin tavalla samanlaisia kuin uusi tarkasteltava tilanne. *Case-based induction* on helpompi implementoida ja toimii nopeammin mutta soveltuu vain tilanteisiin, joissa tapauksia on riittävästi ja kun jäljityskriteerit ovat hyvin määriteltyjä. Kolmas vaihtoehto on järjestää tapaukset jonkun *hierarkian* mukaisesti.

CaseAdvisor

CaseAdvisor™ 3.1 (Case-Based Reasoning Group - CBRG) on tapauspohjaisten järjestelmien kehitys-, ylläpito- ja käyttöympäristö.

CBR Express® Acquisition Products

CBR Express Professional Author 3.0 (Inference Corporation) on CBR Express -tuoteperheen päätuote tapauspohjaisten tietämyssovellusten luomiseen ja ylläpitoon. *CBR Express Generator 3.0*:lla voi luoda tapauskantoja erilaisista dokumenteista, esimerkiksi teknisistä asiakirjoista, manuaaleista tai Web-sivuista. *Development Kits*:llä voi kerätä uutta tietoa tapauskantaan ja hallita kannan olemassa olevaa tietoa.

CBR-Works

CBR-Works (TECINNO GmbH) mahdollistaa olemassa olevien tietokantojen käytön tietämyksen lähteenä. Se tarjoaa oliopohjaisen tietämyksen esittämismahdollisuuden ja tukee kansainvälisiä standardeja, kuten CORBAa, varmistaakseen ohjelmistokomponenttien integrointimahdollisuudet myös tulevaisuudessa. CBR-Works:n arkkitehtuuri perustuu Componentwareen, jonka avulla järjestelmä voidaan muokata asiakkaan tarpeisiin ja integroida client-server-ympäristöihin. environments. CBR-Works on saatavissa seuraaviin ympäristöihin: MS-Windows-pohjaiset PC:t, OS/2, Apple Macintosh, eri Unix-järjestelmät ja Digital Alpha Unixilla tai Windows NT:llä. CBR-Works tukee seuraavia tietokantoja: DB/2, Oracle, Sybase ja GemStone sekä ODBC-yhteensopivat järjestelmät.

KATE-CBR

KATE-CBR on yksi AcknoSoftin tuotteista. Se sisältää kaksi moduulia, joista toinen vertailee nykyistä ongelmaa jo ratkaistuihin ongelmiin löytääkseen mahdollisimman lähellä olevan tapauksen. Toista moduulia käytetään hakemaan soveltuvat tapaukset tehokkaasti ja päivittämään tapauskanta.

ESTEEM

ESTEEM 1.1 on kirjoitettu IntelliCorpin Kappa-PC sääntöpohjaisella asiantuntijakehittimellä, mutta ESTEEM 2 (Esteem Software) kirjoitettiin uudelleen C++:lla. ESTEEM on saatavilla Windows 3.0 -, Unix- ja X/Motif-ympäristöihin. ESTEEM sallii loppukäyttäjän tekemän päätöksiä käytön aikana, joten se soveltuu käyttäjille, jotka haluavat kysyä tapauskannasta what-if-kysymyksiä ja joille haun nopeus ei ole kriittinen asia. ESTEEM tukee tapaushierarkioita.

ReCall

ReCall (ALICE) on C++:lla ohjelmoitu CBR-ohjelma, jolla voidaan kehittää sovelluksia vaihe vaiheelta. Se tukee monitasoista perintää, dynaamisia linkkejä, relaatioita, funktioita ja taksonomiaa. Ohjelma pystyy lukemaan tietoa relaatiotietokannoista ODBC:n kautta. Ohjelmaan on saatavissa kirjastoja, joilla C/C++-kutsuja voidaan sulauttaa sovellukseen. ReCall on saatavissa Windows- ja Unix/Motif-ympäristöihin.

ReMind

ReMind (Cognitive Systems, Inc.) on kirjoitettu C++:lla. ReMindilla voi kehittää hyvin monimutkaisia CBR-sovelluksia, muttei monimutkaisia asiantuntijajärjestelmäsovelluksia, jotka yhdistävät tapauspohjaisen päättelyn, säännöt ja oliotekniikan samassa sovelluksessa. ReMindiin voidaan yhdistää tietokantoja, joista voidaan tuoda tapauksiin liittyvää tietoa.

Sääntöpohjaiset päättelytyökalut

Asiantuntijajärjestelmistä useimmat ovat tällä hetkellä sääntöpohjaisia. Sääntöpohjaisten asiantuntijajärjestelmien rakenteeseen kuuluvat käyttöliittymä, tietokannat, päättelyosa ja tietämyskanta. Tietokannat sisältävät osan käsiteltävän järjestelmän tiedoista ja saattavat toimia on-line-yhteydessä yrityksen tai julkiseen tietokantaan. Myös ihmiskäyttäjää voidaan pitää tietokantana. Päättelyosa sisältää yleiset ongelmanratkaisutiedot ja -menetelmät sekä tulkitsee ja käsittelee säännöt. Sääntöpohjaisen järjestelmän hakumenetelmä hyödyntää myös heuristista tietoa. Tietämyskanta tai sääntökanta sisältää suurimman osan ongelmanratkaisutiedoista. Säännöt ovat muodoltaan IF-THEN-tyyppisiä. Säännöt voivat olla erityisesti tiettyyn kohteeseen liittyviä tai heuristisia peukalosääntöjä. Säännöt voivat olla myös ketjutettuja (esimerkiksi "Jos A, niin B." ja "Jos B, niin C.", joten A -> B -> C, jolloin "Jos A, niin C."). Varmuuskertoimet esittävät luotettavuutta, että asia on totta tai sääntö ajankohtainen.

Sääntöpohjaisten järjestelmien etuja (Hayes-Roth 1984):

1. Tietämyskanta voi muodostua sekä dokumentoidusta, todennetusta ja objektiivisesta informaatiosta (so. simulointimallit ja tekninen informaatio) että asiantuntijan subjektiivisista näkemyksistä.
2. Asiantuntijajärjestelmän toimintaa ohjataan päättelyjärjestelmän avulla, joka hyödyntää järjestelmän ohjauksessa *pattern matching* -periaatetta. Se on ei-proseduraalinen ja joustava keskusteluväline vastakohtana tavanomaisissa systeemeissä käytetyille ennalta sidotuille haarautumismenetelmille.
3. Päättelyjärjestelmä ja tietämyskanta ovat erilliset, mikä suuresti yksinkertaistaa asiantuntijajärjestelmän ylläpitoa ja päivittämistä. Tietämyskanta on usein modulaarinen, jolloin sen muuttaminen on helppoa.
4. Asiantuntijajärjestelmä sisältää tietorakenteet, joita tarvitaan epätarkan ja epätäydellisen informaation käsittelemiseksi. Tällä tarkoitetaan luotettavuusarvojen (certainty factors) yhdistämistä syötteisiin ja johtopäätöksiin.
5. Asiantuntijajärjestelmä pystyy antamaan vastauksen **miksi**-kysymykseen. Koska se muistaa käyttämänsä päättelyketjun, se voi sisällyttää antamansa ongelman ratkaisun selvityksen niistä askeleista, jotka se on päättelyssä tehnyt. Käyttäjä voi kysyä järjestelmältä, miksi se haluaa tiettyä informaatiota tai **kuinka** se päätyi juuri tiettyyn vastaukseen. Edistyneeltä järjestelmältä voi kysyä myös sitä, **miksi ei** joku asia ole mahdollinen.
6. Asiantuntijajärjestelmän käyttö ei edellytä erityisasiantuntemusta. Kaikki informaation analysointi tehdään järjestelmässä, joten käyttäjälle jää ainoastaan joukko suositusvaihtoehtoja, jotka on järjestetty luotettavuusfaktoreiden mukaan. Tavanomaisten järjestelmien yhteydessä käyttäjän täytyy tulkita esitettyjen tulosten merkitys.

Sääntöpohjaiset asiantuntijajärjestelmät sisältävät myös puutteita: ns. "maalaisjärjen" puute, luovuuden puute, oppimattomuus, aistihavaintojen puute ja käyttöalueen rajallisuus. Ihmiasiantuntijalla on ns. tervettä järkeä tai maalaisjärkeä suuren teknisen tiedon määrän lisäksi. Toistaiseksi ei tunneta tapaa, jolla asiantuntijajärjestelmille voitaisiin antaa maalaisjärkeä. Ihmiasiantuntija voi reagoida luovasti epätavallisiin tilanteisiin, asiantuntijajärjestelmä ei voi. Ihmiasiantuntija mukautuu automaattisesti muuttuvaan ympäristöön, asiantuntijajärjestelmiä pitää yksikäsitteisesti päivittää. Tapauspohjainen päättely ja neuraaliverkot ovat menetelmiä, joihin voi liittää oppimista. Ihmiasiantuntijoilla on käytettävissä laaja joukko aistihavaintoja, asiantuntijajärjestelmät ovat toistaiseksi riippuvaisia symbolisesta syöttötiedosta. Sääntöjen ilmaisuvoima on rajallinen kuvattaessa staattisia objekteja ja niiden välisiä riippuvuuksia. Asiantuntijajärjestelmät eivät ole hyviä havaitsemaan, milloin vastausta ei ole olemassa tai milloin ongelma on niiden asiantuntemuksen ul-

kopuolella. Kun sääntöjen määrä kasvaa yli sadan, alkaa olla todennäköistä, että niiden muuttamisella on haitallisia sivuvaikutuksia (Hayes-Roth 1984).

ACQUIRE

ACQUIRE (Acquired Intelligence Inc.) on tietämyksen hankintajärjestelmä (knowledge acquisition system) ja asiantuntijajärjestelmien kehitystyökalu. Se on täydellinen kehitysympäristö tietämyspohjaisten sovellusten rakentamiseen ja ylläpitoon. Siinä sovellusalueen asiantuntija voi askel askeleelta strukturoida ja koodata tietämystä. Sen ominaisuuksia ovat rakeenteellinen lähestymistapa tietämyksen hankintaan, hahmontunnistukseen perustuva tietämyksen hankinnan malli, olioina esitettävä tietämys, tuotantosäännöt ja päätöksentekotaulukot, epävarmuuden hallinta kvalitatiivisesti, ei-numeeriset proseduurit, perusteelliset tietämyskannat, raporttien kirjoitusmahdollisuudet sekä itsedokumentoituvat tietämyskannat hypertext-ympäristössä. ACQUIRE-SDK on ohjelmaan liittyvä ohjelmiston kehityspaketti, johon sisältyy mm. DLL:ien käyttömahdollisuus ja WWW:ssä pyörivien sovellusten tekemahdollisuus.

CLIPS

CLIPS (C Language Integrated Production System) on edullinen, NASAn kehittämä, ANSI C:llä kirjoitettu sääntö- ja oliopohjaisten asiantuntijajärjestelmien kehitystyökalu tai -moduuli. CLIPSillä tehdyn osuuden voi sulauttaa muihin sovelluksiin. CLIPS sisältää oliokielen nimeltään COOL (CLIPS Object-Oriented Language), joka on suoraan integroitu päättelykoneeseen. CLIPS on saatavissa moniin ympäristöihin.

Design++

Design++ (Design Power, Inc.) on lähinnä teknisten järjestelmien suunnitteluun tarkoitettu tietämyspohjainen työkalu. Älykkäät osat ja kokonaisuudet on kirjoitettu, joten niitä voidaan käyttää kaikissa malleissa. Design++ käyttää kehyspohjaista, dynaamista olioympäristöä. Ohjelmasta on liitännät CAD-ohjelmiin ja joihinkin relaatiotietokantoihin.

EXSYS

EXSYS Professional (EXSYS, Inc.) on asiantuntijajärjestelmien kehitysympäristö (Wendt 1993), joka soveltuu DOS-, Windows-, Macintosh-, Sun Open Look ja VAX/VMS Motif -ympäristöihin. EXSYS on kirjoitettu C:llä. EXSYSin avainominaisuus on sen kyky kehittää ammattilaissovelluksia eri ympäristöihin yhdestä lähdekoodista. EXSYSin säännöt voidaan luoda ja niitä voidaan muokata

perinteisillä tekstinkäsittelijöillä tai EXSYS:n omalla sääntöeditorilla. Jokainen sääntö rakennetaan ehdoista (condition) ja valinnoista (choice). Ehdot voivat olla joko tekstimuotoisia tai matemaattisia. Kun ehto on luotu, EXSYS antaa käyttäjä sille missä tahansa säännössä viittaamalla ehdon nimeen tai numeroon. Valinnat ovat mahdollisia ongelman ratkaisuja, joista asiantuntijajärjestelmä päättää. Varmuuskertoimet voidaan antaa seuraavilla viidellä eri tavalla: 0 tai 1 (kyllä tai ei), asteikko välillä 0 - 10, asteikko välillä -100 ... 100, kasvava/vähenevä tai määritelty kaava. EXSYSistä on liitännät SQL-tietokantoihin, yleisimpiin tietokantoihin, LINDO-ohjelmaan (lineaarinen optimointi), NeuralWorks-neuraaliverkko-ohjelmaan ja PI-prosessisäätöön. EXSYS RuleBook soveltuu järjestelmien luomiseen puukaavioista.

FLEX

FLEX (LPA - Logic Programming Associates) on asiantuntijajärjestelmien kehitysohjelma, joka tukee kehyspohjaista päättelyä. Siinä logiikkaohjelmointiympäristöön on integroitu perintä, sääntöpohjainen ohjelmointi ja tieto-ohjatut (data-driven) proseduurit. FLEX on implementoitu Prologilla.

G2

G2 (Gensym) on sekä toiminnallisuudessa että markkinaosuudessa (Årzén 1993) hallitseva graafinen, oliopohjainen ympäristö älykkäiden reaaliaikaisten järjestelmien luomiseen. G2 yhdistää kolme ominaisuutta: sääntöpohjaisen ohjelmoinnin, oliotekniikan ja prosessuaalisen ohjelmoinnin. G2 on kehitetty LISP-kielillä, mutta käyttäjä ei pääse käsiksi LISP-tasoon. G2:n päättelymekanismi perustuu staattisiin yhteyksiin. Perustietotyypit ovat: numerot, loogiset muuttujat, symbolit, merkit, listat ja oliot. Metodeja ei suoraan tueta. Jokainen G2-olio esitetään graafisesti ikonina. Oliot voidaan yhdistää G2-liitännöillä. Täten graafiset rakenteet, kuten prosessikaaviot tai vikapuut, voidaan esittää perinteisesti. Ei-graafiset liitännät olioiden välillä esitetään relaatioilla. G2:ssa on simulointiosa, joka sisältää numeeriset integrointirutiinit differentiaali- ja differenssiyhtälöiden ratkaisuun. G2 on saatavissa Unix-, Open VMS ja Windows-ympäristöihin. G2 sisältää SQL-liitännän. G2:n ongelmia ovat sen korkea hinta ja kovat laitteistovaatimukset. Lisäksi ohjelmaa ei ole toteutettu modulaarisesti, mikä mahdollistaisi vain sovelluksessa tarvittavien osien hankinnan.

GoldWorks

GoldWorks (Gold Hill Inc.) on kohdassa 3.3.2 kuvatun GCLISP Developerin päälle rakennettu suhteellisen monipuolinen asiantuntijajärjestelmien kehitystyökalu. Sillä voidaan luoda asiantuntijajärjestelmiä joko osaamatta LISP-kieltä tai käyttämällä suoraan apuna Common LISPiä. Ohjelmistossa voidaan toteuttaa monenlaisia graafisia näyttöjä. GoldWorks mahdollistaa liittynät C:llä ja C++:lla ohjelmoituihin osiin sekä tukee DDE:tä. Ohjelmassa ongelma voidaan osittain kuvata käyttäen kehyksiä (frames), jotka tavallaan perustuvat luokkarakenteeseen. Kehykset mahdollistavat hierarkian, jossa lapsikehyksellä voi olla yksi tai useampi kehysvanhempi, ja se voi silti periä attribuutteja kaikilta esi-isiltään. GoldWorksissa epävarmaa tietoa esitetään varmuuskertoimilla. Ohjelma osaa selittää, miksi päädyttiin johonkin johtopäätökseen. Ohjelmalla voidaan tuottaa myös Runtime-sovelluksia.

LEVEL5 OBJECT

LEVEL5 OBJECT (Inso Corporation) on monipuolinen kehitystyökalu Windows-ympäristöön (Creswick 1995a). Tiedon esitystapa sisältää luokat, instanssit ja attribuutit. Ajonaikaisten instanssien määrä voi olla yli 32 000. Käyttöliittymään voidaan liittää WAV- ja MIDI-audiotiedostoja ja AVI-videotiedostoja. Ohjelma on OLE2- ja ODBC-yhteensopiva. Versio 3.6 on optimoitu 386/486-prosessoreille. LEVEL5 OBJECT:ssa on viisi pääkehitysikkunaa. OBJECT Development Kitiä (ODK:ta) voi käyttää C- ja C++-moduulien tekemiseen.

M.4

M.4 (Teknowledge Corporation) on DOS- ja Windows-ympäristöissä toimiva sääntöpohjaisten asiantuntijajärjestelmien kehitysympäristö (Schmuller 1993). M.4:llä voidaan tehdä sovelluksiin sulautettua älykkyyttä, koska M.4 sisältää tuen MS Visual Basicille, MS Visual C++:lle ja Asymetrix Toolbookille. Pakanen et al. (1996b) kokeilivat huonolla menestyksellä M.4:n käyttöä vikapuusovelluksen implementoinnissa.

Nexpert Object ja Smart Elements

Nexpert Object (Neuron Data) on laajalti käytetty olio- ja sääntöpohjaisten sovelusten kehitystyökalu. Nexpert Objectilla tehdyt sovellukset voidaan sulauttaa olemassa oleviin järjestelmiin. Valinnaisella tietokantamoduulilla voidaan lukea ja kirjoittaa relaatiotietokantoja. Nexpert Object on saatavissa 35:lle ympäristölle sisältäen mm. Windowsin. Smart Elements sisältää Nexpert Objectin, datan käsitteilytoiminnot ja graafisen käyttöliittymän.

OPS/R2

OPS/R2:ta (Production Systems Technologies) voi käyttää joko itsenäisten asiantuntijajärjestelmien tekemiseen tai sellaisten sääntömoduulien luomiseen, jotka linkataan olemassa oleviin C- tai C++-ohjelmiin. OPS/R2 käyttää standardeja C:n aliohjelmaliitoksia ja tietotyyppejä. Kun OPS/R2-säännöt linkataan C-tai C++-ohjelmiin, tuloksena saadaan tavallinen ajettavissa oleva ohjelma, jota ajetaan juuri samalla tavalla kuin tavallista C- tai C++-ohjelmaa. Tällöin ei ole aikaavieviä viiveitä ladattaessa sääntöjä käynnistyksen yhteydessä toisin kuin muissa sääntöpohjaisissa järjestelmissä.

PowerModel (entinen Kappa)

PowerModel (IntelliCorp) on oliopohjainen mallinnusympäristö ja tietämyspohjaisten järjestelmien kehitysympäristö. PowerModelin järjestelmäarkkitehtuuri käyttää modulaarista kerrostettua suunnittelua. Tämä kerrostettu arkkitehtuuri varmistaa, että sovellukset sisältävät vain ne osat järjestelmästä, jotka tarvitaan ajon aikana. PowerModelin ydin sisältää täydellisen ANSI C API -rajapinnan. Ytimen yläpuolella ovat graafiset kehitystyökalut. Ylin kerros sisältää luokkakirjastot. PowerModel-sovellukset voidaan kääntää esimerkiksi DLL:ksi.

Rete++

Rete++ (The Haley Enterprise) generoi Rete-algoritmiin perustuvan C++-luokkarakenteen käytettäväksi C++-sovelluksissa. Rete++-sovelluksiin on saatavissa moduuleita, joilla voidaan tehdä tapauspohjaiseen päättelyyn perustuvia sovelluksia sekä integroida sovellus tietokantoihin.

XpertRule® KBS

XpertRule (Attar Software Limited) on tietämyspohjaisen järjestelmän kehitys- ja ohjelmistotyökalu (Creswick 1995b), jossa tietämys esitetään päätöksentekopuina. XpertRule tarjoaa menetelmät älykkääseen ongelmanratkaisuun. Analysointiongelmiin (diagnoosiikka, arviointi, valinta, luokittelu ja sääntöjen tulkinta) suositellaan käytettäväksi päätöksentekopuita ja yhdistelmäongelmiin (ohjeet, resurssien hallinta, luonnostelu ja suunnittelu) geneettisiä algoritmeja. Käytännöllisenä lähestymistapana resurssienallokointiongelmiin voidaan käyttää yhdistettyä heuristis-geneettistä algoritmia. XpertRule soveltuu Windows-ympäristöihin. XpertRule ei käytä oliotekniikkaa eikä tue OLE2:ta. Sen sijaan se tukee DDE-, DLL- ja ODBC-linkkejä.

3.3.4 Sumean logiikan kehitysohjelmat

DataEngine

DataEngine (MIT - Management Intelligenter Technologien GmbH) on ohjelmistotyökalu datan analysointiin, jossa sumeat sääntöpohjaiset järjestelmät, sumeat ryhmittelymenetelmät, neuraaliverkot ja neuro-sumeat järjestelmät on implementoitu yhdessä perinteisten matemaattisten, tilastotieteellisten ja signaalinkäsittelymenetelmien kanssa.

FlexTool(FS)

FlexTool(FS) (Flexible Intelligence Group, LLC.) on modulaarinen ohjelmistoympäristö sumeiden järjestelmien soveltamiseen. Versio M2.2 toimii MATLAB-ympäristössä.

FUZZLE

FUZZLE (MODiCO, Inc.) on PC-pohjainen sumeiden järjestelmien kehitysympäristö. Ohjelmisto tukee graafisia näytöjä ja hiiritoimintoja. Ohjelma tuottaa C- tai FORTRAN-kielistä lähdekoodia, jotka voidaan liittää sovellusympäristöön. Lisäksi FUZZLEssa on oma ajokelpoinen moduuli, jota ei tarvitse ohjelmoida ja jota voi vapaasti levittää.

Fuzzy Knowledge Builder

The Fuzzy Knowledge Builder (FUZZY SYSTEMS ENGINEERING) on sumeiden järjestelmien suunnittelijoille tarkoitettu ohjelmistotuote. Se on grafiikkapohjainen sääntö- ja sumeiden joukkojen editor. Tuloksena saadaan sovelluksen lähdekoodiin liitettävä tiedosto.

FuzzyCLIPS

FuzzyCLIPS on versio kohdassa 3.3.3 esitellystä CLIPSistä, johon sisältyvät sumeiden totuusarvojen ja sääntöjen esittämiseen ja käsittelyyn tarvittavat laajennukset. CLIPSin toimintojen lisäksi FuzzyCLIPS pystyy käsittelemään täsmällistä, sumeaa (tai epätäsmällistä) ja yhdistettyä päättelyä mahdollistaen asiantuntijajärjestelmän sumeiden ja normaalien ehtojen yhdistelyn säännöissä ja faktoissa.

fuzzyTECH

fuzzyTECH (INFORM GmbH ja Inform Software Corporation) soveltuu useimpiin tietokoneympäristöihin tarkoitettuihin tuotteisiin ja tukee kaikkia sumean logiikan soveltamisen vaiheita. *fuzzyTECH* voidaan liittää ja integroida useimpien muiden ohjelmistotuotteiden kanssa. Se tukee monia standardiliittymiä (DLL, OLE, ANSI-C jne.) sekä spesifisiä "plug-and-play"-tyyppisiä liityntöjä (Simulink, InTouch, Excel, VisualBasic jne.). *fuzzyTECH* toimitetaan API-rajapinnalla (Application Programming Interface), jolla voidaan ohjata useimpia *fuzzyTECH*-ohjelmiston toimintoja. Tämä API on nimeltään FTRCU-DLL (*fuzzyTECH* Remote Control Unit), ja se sisältää kaikki tarvittavat ominaisuudet *fuzzyTECH*in integroimiseen omaan sovellukseen.

O'INCA

O'INCA (Intelligent Machines, Inc.) on Windows-pohjainen sumean logiikan, neuraaliverkkojen ja neuro-sumeiden järjestelmien kehitysympäristö. O'INCAlla voidaan kehittää itsenäisesti toimivia sovelluksia sekä sulautettuja ohjelmistoja päätöksentekoa, hahmontunnistusta, säätöä tai järjestelmämallinnusta vaativiin sovelluksiin. Tällaisia voivat olla päättelyä vaativat sovellukset huonosti määritellyin parametrein, muuttuvien toimintaolosuhtein ja epälineaarisuuksin. O'INCAssa käyttäjä määrittelee moduuleja sovelluskohtaisille algoritmeille. Koko projektista tai yksittäisistä moduuleista voidaan tuottaa C-koodia. Ohjelmalla voidaan myös simuloida ja etsiä virheitä.

Sumeiden järjestelmien toteutuksissa voidaan käyttää useita päättelymenetelmiä, täsmällistämistapoja, loogisia operaattoreita ja jäsenyysfunktioimuotoja. Sääntöjen painot voivat olla sekä staattisia että dynaamisia. Syöttötietojen, ulostulotietojen, sääntökantojen, sääntöjen, sääntöihin liittyvien ehtojen ja jäsenyysfunktioiden määrä on rajoittamaton.

Neuraaliverkkototeutuksissa voidaan käyttää monikerroksisia myötäkytkentöjä ja perus- tai muokattua back-propagationia. Voidaan luoda verkko ja muokata sitä graafisesti. Piilokerroksia voi olla useita ja kussakin kerroksessa voi olla 150 neuronin. Verkkoa voidaan opettaa ja testata.

TilShell

TilShell (Togai InfraLogic) on Windows-pohjainen ohjelmiston kehitystyökalu, joka mahdollistaa käyttäjän suunnitella ja testata sumean logiikan asiantuntijajärjestelmiä sisältäen myös sulautetut säätöjärjestelmät. Ohjelmisto sisältää

- reaaliaikaisen on-line-testauksen, sääntöjen, jäsenyysfunktioiden ja sääntöjen painojen virityksen sekä sääntöjen lisäyksen ja tuhoamisen
- graafisen, "osoita ja klikkaa" -case-työkalun
- käyttäjän määrittelemät päättelymenetelmät
- värillisen sääntökannan ja käyttäjän visualisointityökalut 3-D-muodossa
- graafisen sumeiden järjestelmien ja perinteisten menetelmien simuloinnin
- C-koodin generoinnin.

WINROSA

WINROSA (MIT GmbH) tuottaa sääntöjä käytettävissä olevasta datasta, joka voidaan tuoda esimerkiksi MATLABista, Dora-Fuzzysta tai DataEnginestä.

3.3.5 Neuraaliverkko-ohjelmat

Neuraaliverkko koostuu suuresta määrästä samanlaisia, rinnakkaisia yksiköitä, joita kutsutaan neuroneiksi (Koikkalainen 1992). Jokainen neuroni on kytketty useaan muuhun neuroniin ja jokaisella neuronilla on yksi ulostulo, joka haarautuu vastaaviin kytkentöihin muihin neuroneihin. Neuraaliverkossa ei sovelleta Boolean algebraa, vaan jatkuva-arvoisia funktioita, joiden arvo määräytyy ns. kytkentävahvuuksien ja sisääntulosignaalien arvojen pohjalta. Kytkentäarvo on jatkuva-arvoinen reaalityttö, jolla yleensä painotetaan (kerrotaan) neuronin sisääntulosignaalin arvot. Jokaisella signaalilla on omat painonsa ja painotetut arvot lasketaan yleensä yhteen mahdollisesti kynnystäen digitaalisiksi 0-1-arvoiksi.

Neuraaliverkot eroavat toisistaan sekä rakenteeltaan että opetustavaltaan. Yksinkertaisimmillaan verkon painokertoimet voidaan määrätä jonkin matemaattisen optimointitehtävän ratkaisemiseksi, jolloin varsinaista oppimista ei tarvita. Oppivat verkot ovat joko ns. itseorganisaatioon tai ohjattuun oppimiseen perustuvia. Ohjatussa oppimisessa verkkoa opetetaan esittämällä syöte ja haluttu ulostulo eli kysymys ja vastaus vektorimuodossa. Itseorganisaatio on ohjaamatonta oppimista, missä verkko organisoituu pelkästään annetun syöteen pohjalta halutun ulostulon ollessa määräämätön.

Neurolaskennan ohjelmistotuotteet noudattavat muutamaa perusrakennetta, jotka määräytyvät käyttötarkoituksen ja käyttömahdollisuuksien mukaan. Ohjelmistoluokat ja niiden käyttötarkoitukset voidaan jaotella seuraavasti (Koikkalainen 1994):

- Sovelluskohtaiset. Kyseessä on tiettyä sovellusta varten kehitetty neurolaskennan perustuva työkalu.

- Menetelmätoteutukset. Suosituimmille neurolaskennan menetelmille on runsaasti ohjelmistototeutuksia. Useimmiten nämä ovat ohjelmia, joissa liitänä ulkomaailmaan tehdään tiedostojen kautta.
- Kehitysympäristöt. Näiden tavoitteena on yleiskäyttöinen neurolaskennan suunnitteluympäristö, jonka puitteissa voi toteuttaa useita erilaisia neuroverkkoalgoritmeja ja kokeilla menetelmien soveltuvuutta tehtäviin.
- Aliohjelmakirjastot ja työkalupakit. Nämä ovat ohjelmoinnin apuvälineitä, joissa neurolaskennan menetelmille on valmiiksi määritetyt tietorakenteet ja niitä käyttävät aliohjelmat tai funktiot.
- Neurokielet. Markkinoilla on muutamia neurolaskentaan suunniteltuja ohjelmointikieliä.

Brainmaker

Brainmaker on kaupallinen ohjelma, joka on tarkoitettu lähes pelkästään suosittujen monikerros-preceptron-verkkojen sovelluksiin. Keskeisimpiä ominaisuuksia ovat:

- Opetuksen suunnittelua tukevat työkalut
- Hyvät opetusaineiston käsittelyominaisuudet
- Ymmärtää tunnetuimpien taulukkolaskinohjelmien ja PC-tietokantojen tiedostorakenteita.

ECANSE

ECANSE (Siemens Nixdorf Advanced Technologies GmbH) on graafinen kehitysympäristö neuraaliverkkoja tai muita vastaavia tekniikoita hyödyntävien sovellusten suunnitteluun, simulointiin ja testaukseen.

Explorenet 2000

Vuokaavioajatteluun perustuva neurolaskennan kaupallinen kehitysympäristö. Neuroverkkosovellus kootaan graafisesti kytkemällä ikoneita toisiinsa linkeillä.

FlexTool(LNN)

FlexTool(LNN) (Flexible Intelligence Group, LLC.) on modulaarinen ohjelmistotyökalu neuraaliverkkojen soveltamiseen. Se sisältää Back-Propagation-algoritmit.

LVQ_PAK, SOM_PAK

TKK:n Informaatiotekniikan laboratorion ilmaisjakeluun antamat versiot oppivasta vektorikvantisaatiosta ja itseorganisaatiokartasta.

Matlab/Neural Networks Toolbox

Matlab on laaja-alaisesti käytetty kaupallinen numeerisen matematiikan työkalu-ohjelmisto, joka perustuu vektori- ja matriisilaskentaan. Perusympäristöä voi kasvattaa hankkimalla nk. työkalupakkeja (toolbox) esimerkiksi neurolaskentaan, signaalinkäsittelyyn tai tilastotieteeseen. Neurolaskennan työkalupakki on tehty kokonaan Matlab-ytimen avulla ja on siten käyttäjän kannalta avoin muutoksille ja liitännöille muihin järjestelmiin.

NeuFrame

NeuFrame (NCS) on lähinnä neuraalilaskennan 32-bittinen, oliopohjainen kehitysympäristö. Siinä on toteutettu backpropagation- ja Kohosen algoritmit. Ympäristössä voidaan tehdä myös neuro-sumeita verkkoja. NeuFrame 3 -versiossa voidaan tuottaa myös C/C++- ja Java-koodia sovelluksesta.

NeuralWorks

NeuralWorks Professional II Plus on tunnetuimpia neurolaskennan kaupallisia ohjelmia. Se on täydellinen kehitysympäristö. NeuralWorks perustuu melko yksinkertaiseen yleiskäyttöiseen neuromalliin. Ohjelma ymmärtää tunnetuimpien tietokantojen ja taulukkolaskinten käyttämiä tiedostorakenteita. Lisäksi ohjelmaan on mahdollista liittää käyttäjän itse ohjelmoimia liittymiä. Ohjelmassa on suuri joukko tunnetuimpia menetelmiä valmiiksi toteutettuna sekä suuri joukko opetussääntöjä ja muita neuraalilaskennan perusoperaatioita kirjastoituna.

NeuroGenetic Optimizer&trade or NGO&trade

NGO&trade (BioComp Systems, Inc.) on neuroverkkojen kehitystyökalu, joka käyttää geneettisiä algoritmeja optimoimaan neuroverkon syöttötiedot ja rakenteen.

NeuroShell

NeuroShell® 2 Professional (Ward Systems Group, Inc.) on Windows-ympäristössä toimiva neuraaliverkkojen kehitysympäristö, jossa on käyttöliittymä sekä aloittelijalle että perehtyneelle käyttäjälle. Ohjelmaan kuuluu myös Runtime-

versio, jolla voi luoda DLL-versioita tehdystä neuraaliverkosta. Ohjelma lukee ja kirjoittaa myös ASCII- ja binäärimuotoisia tiedostoja sekä tunnettujen tietokantaohjelmien tiedostomuotoja. Ohjelmalla on mahdollista toteuttaa 15 erilaista neuraaliverkkoarkkitehtuuria. Tärkeimpiä näistä ovat erilaiset backpropagation-versiot (oppimisalgoritmi monikerroksisten epälineaaristen perceptron-verkon ohjattuun opettamiseen) ja Kohosen itseorganisoituva kartta (tunnetuin ohjaamattomaan oppimiseen perustuvista neuroverkkomalleista). Ohjelmaan sisältyy myös grafiikkaominaisuuksia.

NeuroSolutions

NeuroSolutions (NeuroDimension, Inc.) on oliopohjainen ympäristö neuraaliverkototeutusten suunnitteluun, prototyypin tekemiseen, simulointiin ja kehittämiseen (Principe 1995). Käytettävissä olevia neuroverkkomalleja ovat monikerros-perceptron (MLP), yleistetty myötäkytkentä, modulaarinen myötäkytkentä, Jordan/Elman-verkot, itseorganisoituvat kartat (SOFMs), pääkomponenttianalyysi (PCA), radiaaliskantafunktioverkko (RBF) ja viivästetyt toistuvat verkot (Time-Lag Recurrent networks). Ohjelmasta on saatavissa erilaisia versioita, joista laajemmissa voidaan tuottaa ANSI-yhteensopivaa C++-lähdekoodia ja liittää niihin omia algoritmeja DLL:n kautta.

NeuroWindows

NeuroWindows™ (Ward Systems Group, Inc.) on DLL-neuraaliverkkokirjasto, joka on erityisesti suunniteltu liitettäväksi Visual Basic -ohjelmaan mutta soveltuu käytettäväksi myös muiden DLL:iä kutsuvien ohjelmointikielien kanssa. Kirjasto sisältää optimoituun koodin esimerkiksi backpropagationille ja Kohosen kartalle ja sitä suositellaan käytettäväksi vain NeuroShell-ohjelman kanssa.

O'INCA

Ohjelma on kuvattu kohdassa 3.3.4.

OWL

OWL on tunnetuin kaupallinen neuroverkkojen aliohjelmakirjasto, joka koostuu C-ohjelmointikielille määritellyistä tietorakenteista ja joukosta tyypillisimpiä neurlaskennan operaatioita. Tämä lähestymistapa takaa täysin avoimen ympäristön, jossa ulkopuolisten liityntöjen tekeminen on normaalia ohjelmointia.

Trajan

Trajan 2.1 Professional (Trajan Software Ltd.) on Windows-pohjainen neuroverkkojen simulointipaketti. Se tukee useita arkkitehtuuria ja opetusalgoritmeja. *Trajan 2.1 Shareware* on edullinen, tehokas ja helppokäyttöinen neuroverkkopaketti.

WinNN

WinN on Microsoft Windows -ympäristössä toimiva ilmaisohjelma, joka sopii parhaiten aloittelijoille edullisena vaihtoehtona monille kalliille ohjelmille.

3.3.6 Muut valmisohjelmat

CAFTA

CAFTA on Windows-ympäristössä toimiva, kallis, suhteellisen monipuolinen vikapuiden kehitysohjelma, jolla voidaan muodostaa, analysoida ja ylläpitää suuriakin vikapuita. Puihin liittyvä tieto tallennetaan tietokantoihin, joiden kentät käyttäjä voi mainosesitteen mukaan määrittellä, mutta demo-ohjelmassa kentät olivat valmiiksi määriteltäviä eivätkä olleet muokattavissa. Esitteen mukaan käytettävä tietokanta lukee ja kirjoittaa tunnettujen taulukko-ohjelmien (esim. Excel) tiedostomuotoja, mutta ainakaan demo-ohjelma ei tähän pystynyt. Ohjelman puutteena on myös se, ettei yksittäisiin vikoihin voida suoraan liittää oireita. Mikäli tietokantaliityntä toimisi kuten on esitetty, oireet voitaisiin liittää sitä kautta.

FAULT

FAULT on vanha VTT:ssä kehitetty vikapuiden laadintaohjelma (Räty 1986), joka on tarkoitettu luotettavuustekniikan vikapuuovellusten tekemiseen.

FaultTree+

FaultTree+ on vika- ja tapahtumapuiden piirtämiseen tarkoitettu Windows-ohjelma, jolla on mahdollista laskea myös vikapuuanalyysiin liittyvät todennäköisyydet eri tapahtumille. Ohjelma on melko helppokäyttöinen ja noudattelee kohtuullisen hyvin standardin IEC 1025 mukaista vikapuuesitystä. Ohjelman puutteena on kuitenkin se, että siinä voidaan esittää vikapuurakenteen viat mutta ei vikoihin liittyviä oireita.

GeneHunter

GeneHunter® (Ward Systems Group, Inc.) on optimointiongelmien ratkaisemiseen tarkoitettu Windows-ohjelma, joka hyödyntää geneettisiä algoritmeja. GeneHunterissa tarvittava tieto (data) syötetään Excel-taulukkoon ja GeneHunterin dialoginäytöllä määritellään ongelman ratkaisuparametrit. Ohjelma sisältää myös geneettisen algoritmin kirjaston, joka on saatavissa DLL-muodossa.

Matlab

MATLAB on monipuolinen interaktiivinen numeerisen laskennan ja visualisoinnin ympäristö, joka yhdistää satoja kehittyneitä matematiikka- ja grafiikkafunktioita korkean tason kielellä. Koska MATLAB-kieli on matriisiorientoitunutta, se on luonnollista kieltä tekniseen ongelmanratkaisuun. MATLABissa on osia numeeriseen analyysiin, matriisilaskentaan, signaalinkäsittelyyn, 2- ja 3-ulotteiseen grafiikkaan jne. MATLABissa on lisäosia, jotka laajentavat sen käyttöä digitaaliseen signaalinkäsittelyyn, säätöjärjestelmien suunnitteluun, neuraaliverkkoihin, järjestelmän identifiointiin, dynaamiseen järjestelmäsimulointiin ja optimointiin.

Sugal

Sugal 2.1 (Trajan Software Ltd.) on ilmainen Windows-pohjainen geneettisten algoritmien simulaattori. Sugalista on saatavissa täydellinen lähdekoodi.

3.4 Yhteenveto luvusta 3

Tehtävän tavoitteena oli selvittää sopivat ohjelmistotyökalut vikadiagnoosissa tarvittavan asiantuntijatiedon käsittelyyn. Työn pääpaino oli vikapuun (Kärki & Hyvärinen 1997) kuvaaman asiantuntijatiedon hyödyntämiseen soveltuvien työkalujen selvittämisessä, mutta aihetta tarkasteltiin, mahdollisuuksien mukaan, myös laajemmasta näkökulmasta. Näin ollen osa esitetyistä työkaluista soveltuu varsinaisesti vikadiagnoosijärjestelmän muiden kuin asiantuntijatiedon käsittelyn osien toteutukseen.

Työn aluksi kuvattiin vikadiagnoosijärjestelmän yleisrakenne, joka osittain rajaa mahdollisia, toteukseen sovellettavia ohjelmia. Mallipohjaisessa vikadiagnoosijärjestelmässä ohjelmia käytetään staattisen ja dynaamisen informaation käsittelyyn, joka avustaa diagnostisten tehtävien toteutusta. Vikadiagnoosijärjestelmä saa erilaisten liityntöjen kautta tietoa rakennuksesta ja käyttäjältä sekä antaa itse esimerkiksi palautetietoa tai tarkennuspyyntöjä. Vikadiagnoosijärjestelmä voi olla toi-

minnoiltaan modulaarinen havaiten ja paikantaen rakennus-, osaprosessi- tai komponenttitason vikoja.

Työssä kuvattiin lähteistä löytyneitä ohjelmistojen tarpeita ja vaatimuksia, joita esitettiin älykkäälle, tietokantapohjaiselle hakujärjestelmälle, vikadiagnoosijärjestelmän käyttöliittymän perustoiminnoille, vikadiagnoosisovelluksen päättelyn etenemisperiaatteelle sekä asiantuntijajärjestelmien kehitysympäristöjen valinnalle. Esitetyt vaatimukset ovat aika tiukkoja ja tuskin yksikään ohjelmisto täyttää niitä kaikkia. Ohjelmiston valinnassa on oleellisinta, että ohjelmistolla on mahdollista toteuttaa kuvattu päättelykonsepti. Muita tärkeitä vaatimuksia ovat esimerkiksi soveltuvuus vikapuiden implementointiin (mm. vikapuun rakenteen muuttamismahdollisuus), sekä ohjelmiston että mahdollisen Runtime-lisenssin hinta, käyttöön tarvittavat tietokonelaitteistot (hardware), helppokäyttöisyys, erilaiset liityntämahdollisuudet, sulautettavuus, varsinaiset ominaisuudet ja dokumentointi.

Työkaluohjelmat jaoteltiin ohjelmoinnin perustyökaluihin, ohjelmointiympäristöihin, tietämyspohjaisiin työkaluohjelmiin, sumean logiikan kehitysohjelmiin, neuraaliverkko-ohjelmiin sekä muihin valmisohjelmiin. Tietämyspohjaiset työkaluohjelmat jaettiin vielä tapauspohjaisiin sekä sääntöpohjaisiin päättelytyökaluihin. Käytettävissä olleiden resurssien puitteissa ei ollut mahdollista perehtyä mihinkään ohjelmistoon tai työkaluohjelmaan perusteellisesti, joten esitettävät arvioinnit perustuvat pääasiassa aikaisempiin kokemuksiin, lähteistä löytyneisiin tai käyttäjiltä kuultuihin arvioihin tai esitteiden antamiin tietoihin.

Asiantuntijatietoa hyödyntävien sovellusten tekeminen on aina mahdollista toteuttaa perusohjelmointikielillä, joista tähän tarkoitukseen parhaiten soveltuvat C, LISP ja Prolog. Usein keskeisimpänä ongelmana on kehitetyn sovelluksen vaikeakäyttöisyys, koska esimerkiksi Prologia käytettäessä on tunnettava komentojen oikea muoto. Perusohjelmointikieliä käytettäessä myös sovelluksen tekeminen ja päivittäminen on usein varsin työlästä.

Ohjelmointiympäristöissä hyödynnetään jotakin perusohjelmointikieltä, mutta sen lisäksi ne sisältävät esimerkiksi käyttöliittymän tai ulkoisten liityntöjen tekemiseen tarkoitettuja toimintoja, joilla nämä voidaan tehdä perusohjelmointikieliin verrattuna paljon helpommin ja nopeammin. Esimerkiksi Delphiä ja Visual Basicia on aikaisemmin käytetty vikadiagnostiikkasovellusten toteutuksiin. Jos halutaan käyttää logiikkaohjelmointia, kyseeseen voisivat tulla Amzi! tai WIN-PROLOG. LISP-kielen perustuvista ympäristöistä GCLISP Developer voisi olla mahdollinen.

Tapauspohjaiset päättelytyökalut todennäköisesti soveltuisivat hyvin vikapuun implementointiin ja muihinkin vikadiagnostiikkasovelluksiin. CBR-työkalujen keskeinen ongelma on kuitenkin niiden korkea hinta.

Asiantuntijajärjestelmistä useimmat ovat sääntöpohjaisia. Yksi asiantuntijajärjestelmän etu on, että se pystyy antamaan vastauksen miksi-kysymykseen. Merkittävänä haittana on, että sääntöjen määrän kasvaessa yli tietyn rajan niiden muuttamisella on haitallisia sivuvaikutuksia. Sääntöpohjaisia asiantuntijajärjestelmien kehitystyökaluja on olemassa runsaasti ilmaisohjelmista erittäin kalliisiin kehitysympäristöihin. Työkalujen hyvyttä on kokeilematta mahdoton arvioida, mutta todennäköisesti nimenomaan vikapuusovelluksiin saattaisivat parhaiten soveltua oliopohjaiset tai kehystekniikkaa hyödyntävät kehittimet. Tällaisia ovat esimerkiksi ACQUIRE, FLEX, GoldWorks, LEVEL5 OBJECT, Nexpert Object ja PowerModel.

Esitetyistä sumean logiikan kehitysohjelmista, neuraaliverkko-ohjelmista tai muista valmisohjelmista tuskin mikään soveltuu vikapuiden implementointiin. Varsinaiset -vikapuuohjelmatkin (CAFTA, FAULT ja FaultTree+) soveltuvat vain sellaisten vikapuiden käsittelyyn tai lähinnä niiden piirtämiseen, joissa yksittäisiin vikoihin ei ole liitetty oireita.

Järjestelmän vikapuu on monipuolinen työkalu jo paperille piirrettynäkin, mutta kun toteutus automatisoidaan, käytettävyys lisääntyy entisestään. Vaikka yksikään tarkastelluista ohjelmistoista ei sellaisenaan täysin sovellu vikapuun implementointiin, automatisointi voidaan tehdä käyttämällä usempaa ohjelmistotyökalua. Vikapuu on parhaiten implementoitavissa käyttämällä tietokantaa ja ohjelmointiympäristöä yhdessä (esimerkiksi MS ACCESS ja MS Visual Basic).

Vikapuun hierarkkinen rakenne voidaan kuvata relaatiokantaan. Tietokannan käsittelyyn (esimerkiksi haut ja päivitykset) sekä vikojen syiden ja seurausten päättelyyn ja esittämiseen voidaan tehdä käyttöliittymä, jossa vikapuu esitetään File Manager -tyyppisesti. Todetut oireet voidaan joko kysyä käyttäjältä tai lukea esimerkiksi tiedostosta. Toteutus voi olla myös yhteydessä automaatiojärjestelmään, jolloin osa oireista havaitaan sieltä tulevien mittaus- tai ohjaustietojen perusteella. Kun vikapuun toteutus automatisoidaan, uusien vikojen lisäys puuhun sekä vanhojen päivitys ja poisto on helppoa.

4 Prototyypisovellukset

Tämän tehtävän tavoitteena oli kehittää 1 - 2 ilmastointikoneen kunnon ja toiminnan seurantaan helpottavaa prototyypisovellusta. Näiden ensisijaisina sovelluskohteina projektin käynnistysvaiheessa olivat tehon tarkkailu ja ominaiskäyrien tai muun vastaavantyyppisen tiedon käsittely. Kehitettävien sovellusten haluttiin olevan myös sellaisia, jotka avustavat koneen käyttäjää mahdollisten vikatilanteiden havaitsemisessa. Lisäksi tavoitteena oli, että sovellukset ovat mahdollisimman yleiskäyttöisiä.

Prototyypisovelluksia tehtiin kolme: ominaiskäyräsovellus, ilmastointikoneen tehonäyttö ja vikapuusovellus. Prototyypit esitellään pääpiirteittäin seuraavassa.

4.1 Ominaiskäyräsovellus

Ominaiskäyrät auttavat visualisoimaan fysikaalisten prosessien malleja (Madjidi 1996). Monet LVI-järjestelmien laitevalmistajat julkaisevat niitä kuvaamaan tuotteidensa ominaisuuksia. Visualisointivaatimus rajoittaa muuttujien määrää ominaiskäyrien matemaattisessa esitysmuodossa. Ominaiskäyriä käytetäänkin useimmiten kuvaamaan kahden muuttujan välistä riippuvuutta. Jos halutaan esittää kolmen muuttujan välisiä yhteyksiä, päädytään kaksiulotteisessa esitysmuodossa käyräparviin. Kolmea useamman muuttujan esittäminen graafisesti ei ole enää kovin havainnollista.

Käytettäessä ominaiskäyriä vian havaitsemisessa on ensin muodostettava referenssikäyrä, johon prosessin toimintaa verrataan (Hyvärinen 1993). Referenssikäyrä kuvaa prosessin toimintaa normaalitilassa. Se voi olla valmistajan toimittama tai muodostettu vastaanotto- tai käynnistysvaiheessa. Referenssikäyrä voi olla myös ominaiskäyrä, joka on muodostettu järjestelmän normaalin toiminnan aikana ja sitten valittu referenssikäyräksi. Tällöin se kuvaa jotakin referenssitoimintaa, jonka oletetaan olevan "oikeaa" toimintaa.

Vikojen paljastamiseksi voidaan käyttää esimerkiksi seuraavia menetelmiä:

- Toimintapisteen arvoa verrataan referenssikäyrän vastaavaan arvoon.
- Verrataan keskenään muodostettua ominaiskäyriä ja referenssikäyriä.
- Verrataan muodostettavan ominaiskäyrän tilastollisia ominaisuuksia referenssikäyrän ominaisuuksiin.

Toimintapisteen arvon vertaaminen referenssikäyrään on käytännössä sama kuin mallin ulostulon vertaaminen mitattuun tai laskettuun arvoon. Ominaiskäyriä voi-

daan verrata referenssikäyrään esimerkiksi laskemalla käyrien väliin jäämä pinta-ala tai havainnoimalla erot visuaalisesti. Pinta-alaa laskettaessa on otettava huomioon, että ominaiskäyrässä saattaa olla pisteitä, joita ei ole päivitetty äskettäin.

Ainakin neljänlaisia testisuureita voidaan käyttää (Ruokonen 1986)

- tilastollisia ominaisuuksia (varianssi ja jakauma kaikissa toimintapisteissä)
- referenssikäyrän ja mitatun arvon eroa
- referenssikäyrän ja ominaiskäyrän vertailua
- ominaiskäyrästä määritettyjä parametriestimaatteja ja jotakin fysikaalista ominaisuutta vastaavia parametriestimaatteja.

4.1.1 Datan puhdistaminen

Havaintojen joukossa voi esiintyä yksittäisiä huomattavasti poikkeavia arvoja, jotka ovat ilmeisesti mittausvirheitä (Karttunen 1994). Jos poikkeamia on vain muutama, ne voi poistaa aineistosta. Jos poikkeavat arvot kuitenkin ovat todellisia, niitä ei saa poistaa tai muuttaa.

Jos aineistoon on sovitettu jokin funktio f , hajontaa kuvaa sovituksen rms-virhe (root-mean-square)

$$R = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2}, \quad (10)$$

missä alkuperäiset havaintopisteet ovat (x_i, y_i) , $i = 1, \dots, n$. Yleisesti käytetty kriteeri on eliminoida sellaiset pisteet, joiden poikkeama sovitetun funktion kuvaamasta arvosta on suurempi kuin $3R$ eli

$$|y_i - f(x_i)| > 3R. \quad (11)$$

Virheiden noudattaessa normaalijakaumaa näin suuri poikkeama esiintyy pelkästään sattumalta vain todennäköisyydellä 0,001. Kun vialliset arvot on poistettu, funktion sovitus on syytä tehdä uudestaan käyttämällä puhdistettua aineistoa.

Toinen keino poikkeavien arvojen poistamiseen on mediaanisuodin, joka korvaa kunkin datapisteen arvon kyseisen pisteen pienen ympäristön mediaanilla (Karttunen 1994). Esimerkiksi kolmen pisteen mediaanisuodin on

$$g(x_i) = \text{med}(f(x_{i-1}), f(x_i), f(x_{i+1})), \quad (12)$$

missä $\text{med}(x, y, z)$ tarkoittaa suuruudeltaan keskimmäistä luvuista x , y ja z .

4.1.2 Toteutettu Excel-sovellus

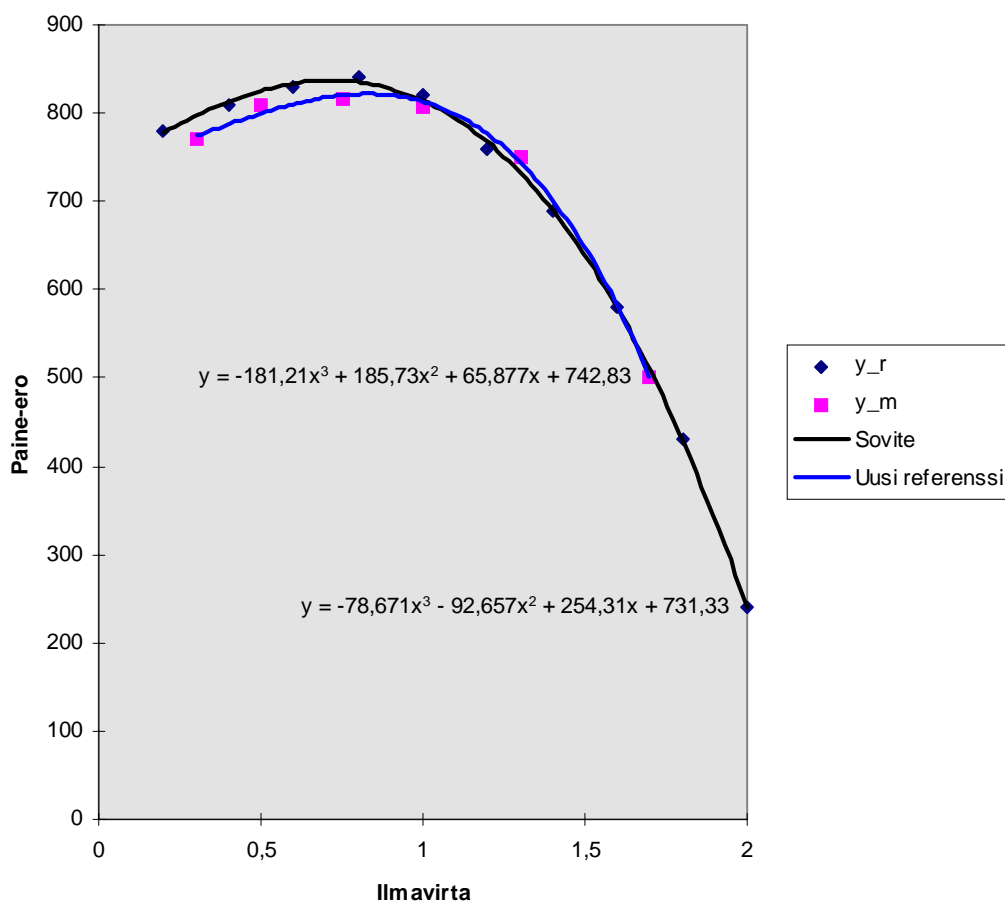
Ominaiskäyräsovellus tehtiin MS Excelin makroilla. Excel-pohjaan annetaan ensin muutama yleistieto käsiteltävästä datasta. Tällaisia ovat referenssikäyrän ja mittausdatan pisteparien lukumäärät, x- ja y-akselien otsikkotiedot sekä ominaiskäyrän otsikko. Tämän jälkeen referenssikäyrän data ja mittausdata voidaan joko syöttää käsin tai kopioida toisesta Excel- tai muusta tiedostosta. Sovellus piirtää automaattisesti referenssikäyrän ja lisää kuvaan mittauspisteet, kun klikataan näihin toimintoihin liittyviä toimintonappeja. Jos referenssikäyrä halutaan päivittää mittauspisteiden mukaiseksi, sillekin on oma automaattinen toimintonsa. Sovellukseen sisältyvät myös kuvan ja referenssi- ja mittausdatojen tuhoamiseen tarvittavat painikkeet.

Kuva 15 esittää ominaiskäyräsovelluksella tehtyä puhaltimen ominaiskäyrää, jossa on ensin syötetty referenssikäyrä (y_r) ja sitten mitatut pisteparit (y_m). Sen jälkeen on todettu, että mitatut pisteparit vastaavat paremmin todellisuutta, ja piirretty niiden perusteella uusi referenssikäyrä.

Tehty sovellus on yleiskäyttöinen kahden muuttujan välisten riippuvuuksien tarkastelun prototyyppi eli se soveltuu myös muiden laitteiden ja järjestelmien kuin puhaltimien ominaiskäyrien tarkasteluun. Sovellus ehdottaa käyrän sovitemuodoksi aina kolmannen asteen polynomimuotoa, mutta tämän voi Excelin omilla toiminnoilla muuttaa. Sovellus on lähinnä ominaiskäyrien piirto- ja tarkastelutyökalu. Se ei diagnostisoi, mistä mittauspisteiden poikkeamat referenssikäyrästä aiheutuvat, vaan tämä jätetään sovelluksen käyttäjän tehtäväksi.

Excel valittiin toteutustyökaluksi, koska se on niin laajasti käytössä oleva ohjelma. Siinä on kuitenkin omat rajoituksensa. Esimerkiksi sovitekäyrien matemaattiset muodot saadaan esiin vain tekstimuodossa, jolloin niitä ei voida automaattisesti suoraan hyödyntää diagnostisoinnissa tai datan puhdistamisessa.

Ominaiskäyräsovellus



Kuva 15. Ominaiskäyräprototyypillä tehty puhaltimen alkuperäinen ja todettu ominaiskäyrä.

4.2 Ilmastointikoneen tehonäyttö

Kärki & Hyvärinen (1997) ovat esittäneet menetelmiä ilmastointikoneen tehon seurantaan. Menetelmissä seurataan ilmastointikoneesta haluttua, siitä saatavaa ja maksimitehoa. Kyseessä on vian havaitsemismenetelmä, jonka avulla on mahdollista havaita, että kone toimii ei-toivotulla tavalla. Havaittujen vikojen paikantamiseen on käytettävä muita menetelmiä, esimerkiksi vikapuita.

Tehomenetelmää sovellettiin ThermoNet-järjestelmän ilmanvaihtokoneeseen, joka ei sisällä ilman sekoitusta. Menetelmässä pyrittiin käyttämään mahdollisimman vähän mittauksia. Siis jos käytössä on enemmän tai esitetystä poikkeavia mittauspisteitä, laskentaperiaatetta on muutettava niille soveltuvaksi. Ei-mitattaville

muuttujille käytetään joko säätöstrategian perusteella määritettävissä olevia arvoja tai suunnittelu- tai vakioarvoja.

Käytettävät mittaukset:

- tuloilman tilavuusvirta ($q_{v,t}$)
- ilmastointikoneelta lähtevän tuloilman lämpötila (T_{sp})
- ulkoilman lämpötila (T_u)
- LTO-nesteen lämpötila lisälämmityksen jälkeen (T_{40})
- poistoilman lämpötila ennen LTO-patteria (T_{21}).

Ilmastointikoneelta haluttu teho (kW) lasketaan seuraavasti:

$$\phi_{ik,h} = q_{v,t} \rho_i c_{p,i} (T_{sp,as} - T_u), \quad (13)$$

jossa

$q_{v,t}$ on tuloilman tilavuusvirta (m^3/s)
 ρ_i ilman tiheys (kg/m^3)
 $c_{p,i}$ ilman ominaislämpökapasiteetti (kJ/kgK)
 $T_{sp,as}$ ilmastointikoneelta lähtevän tuloilman lämpötilan asetusarvo ($^{\circ}C$)
 T_u ulkoilman lämpötila ($^{\circ}C$).

Saatava teho on

$$\phi_{ik,s} = q_{v,t} \rho_i c_{p,i} (T_{sp} - T_u), \quad (14)$$

jossa

T_{sp} on ilmastointikoneelta lähtevän tuloilman lämpötila ($^{\circ}C$).

Ilmastointikoneen maksimiteho on

$$\phi_{ik,max} = (\phi_{LTO})_{max} + (\phi_{puh,t})_{max}, \quad (15)$$

jossa

$(\phi_{LTO})_{max}$ on LTO-piiristä saatava maksimiteho (kW)
 $(\phi_{puh,t})_{max}$ puhaltimen maksimilämpöteho (kW).

LTO-piiristä saatava maksimiteho muodostuu poistoilmasta tuloilmaan siirtyvästä maksimi-ilmaistehosta ja lisälämmitysventtiilin kautta saatavasta maksimitehosta (kaava 16). Toisaalta se voidaan laskea maksimivirtaaman ja huurtumisen raja-lämpötilan perusteella.

$$(\phi_{LTO})_{max} = (\phi_{LTO,T})_{max} + (\phi_{lisä})_{max} = (q_{v,n})_{max} \rho_n c_{p,n} (T_{40} - T_{41,räjä}), \quad (16)$$

jossa

$(q_{v,n})_{max}$ on nestepiirin maksimitilavuusvirta (m^3/s)
 ρ_n LTO-nesteen tiheys (kg/m^3)
 $c_{p,n}$ LTO-nesteen ominaislämpökapasiteetti (kJ/kgK)

T_{40} LTO-nesteen lämpötila lisälämmityksen jälkeen (°C)
 $T_{41,raja}$ huurtumisen eston rajalämpötila (°C).

LTO:sta ilman lisälämmitystä saatava maksimiteho on

$$(\phi_{LTO,T})_{\max} = (q_{v,t})_{\max} \rho_i c_{p,i} (T_{12} - T_u), \quad (17)$$

jossa

$(q_{v,t})_{\max}$ on ilmastointikoneen maksimituloilmavirta (m^3/s)
 T_{12} tuloilman lämpötila LTO:n jälkeen (°C)
 T_u ulkolämpötila (°C).

Tuloilman lämpötila LTO:n jälkeen saadaan seuraavasti:

$$T_{12} = T_u + \eta(T_{21} - T_u), \quad (18)$$

jossa

η on LTO:n hyötysuhde (0-1)
 T_{21} poistoilman lämpötila ennen LTO-patteria (°C).

Kun tiedetään LTO:sta ilman lisälämmitystä saatava maksimiteho, voidaan kaavan 16 perusteella termien erotuksena laskea venttiilin kautta saatavissa oleva maksimilisälämmitysteho.

Kun ilmavirtaa säädetään puhaltimen pyörimisnopeutta muuttamalla, puhaltimen sähköteho on

$$P_{\text{puh,t}} = P_{\max} \left(\frac{q_{v,t}}{q_{v,\max}} \right)^3, \quad (19)$$

jossa

P_{\max} on puhaltimen maksimisähköteho (kW)
 $q_{v,t}$ tuloilmavirta (m^3/s)
 $q_{v,\max}$ maksimituloilmavirta (m^3/s)

Puhaltimesta siirtyvä lämpöteho on

$$\phi_{\text{puh,t}} = \text{osa } P_{\text{puh}}, \quad (20)$$

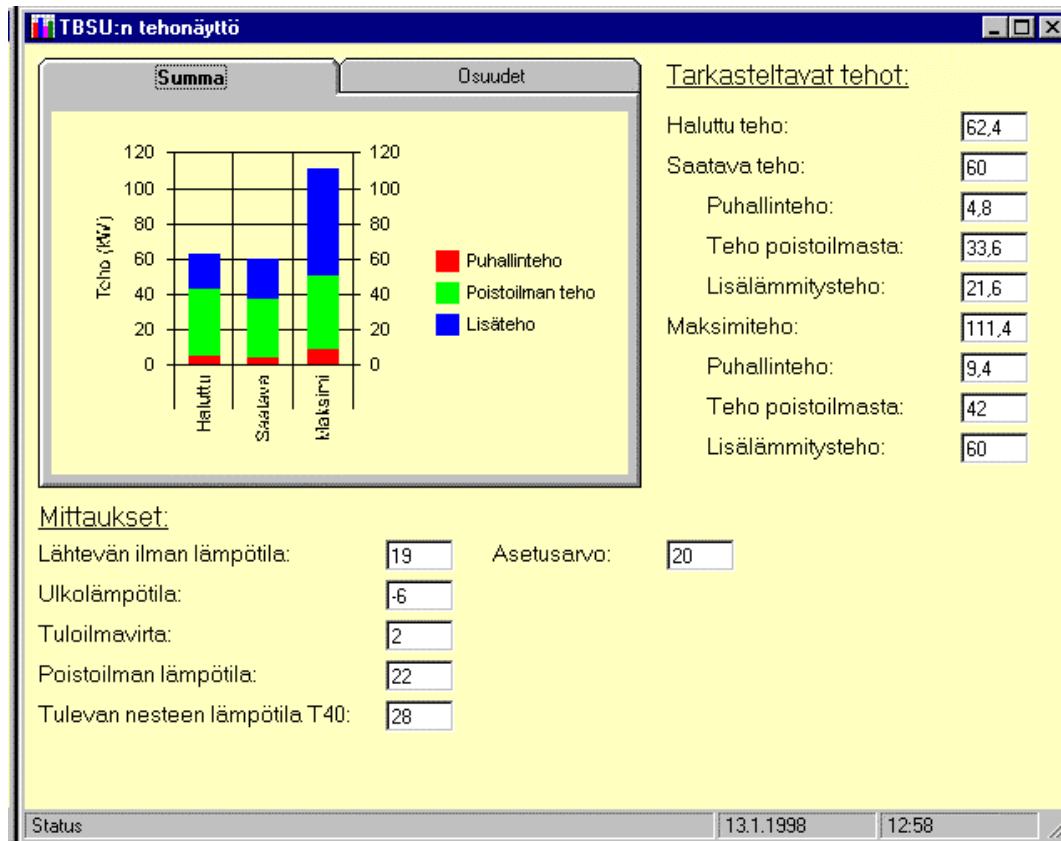
jossa

osa on puhaltimen sähkötehosta ilmavirtaan siirtyvän lämpötehon osuus (0...1).

Kun halutaan jaotella, miten ilmastointikoneesta saatava teho jakautuu eri prosesseista saatuihin tehoihin, käytetään puhallintehon laskennassa kaavoja 19 ja 20. Maksimitehon laskennassa ko. kaavoissa tuloilmavirran tilalle sijoitetaan maksimituloilmavirta. Kun jaotellaan eri prosessien tuottamat tehot ilmastointikoneelta saatavassa tehossa (kaava 14), käytetään kaavoja 15 - 18 siten, että maksimivirtausten sijasta laskennassa käytetään toteutuvia.

Kuva 16 esittää kehitetyn tehonäyttöohjelman pääikkunaa. Siitä voidaan nähdä käytettävien mittausten arvot kulloisellakin tarkasteluhetkellä. Lähtevän ilman lämpötilasta esitetään myös sen asetusarvo. Tarkasteltavia tehoja ovat haluttu, saatava ja maksimiteho, jotka esitetään sekä lukuarvoina että pylväsdiagrammeina. Tehotermit on jaoteltu puhaltimen luovuttamaan, poistoilmasta saatavaan ja lisälämmityksestä saatavaan osatehoon. Näytöstä voidaan tarkastella tehopylväitä siten, että eri teho-osuudet näkyvät samassa pylväessä tai omina pylväinä.

Halutussa tehossa osatehot on määritelty siten, että puhallinteho on 10 %, poistoilmasta saatava 60 % ja lisäteho 30 % kokonaistehosta. Halutussa tehossa jako osatehoihin on kuitenkin viitteellinen, koska se riippuu järjestelmän kulloisestakin käyttötavasta ja käyttöolosuhteista.



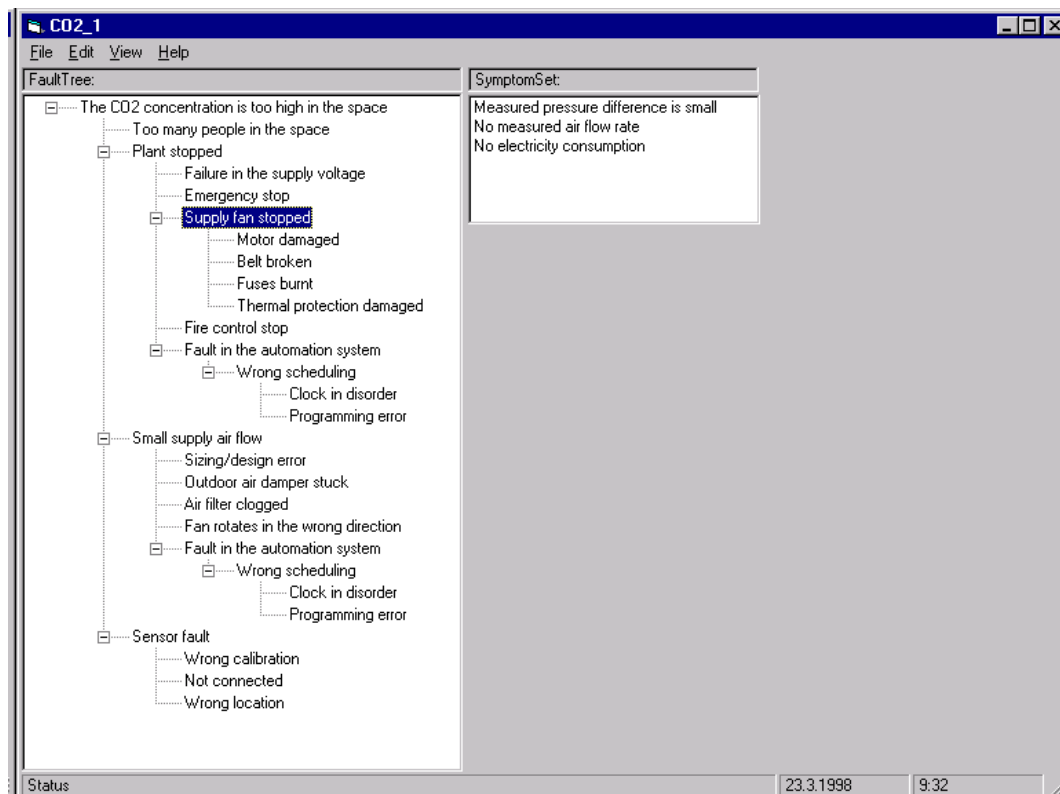
Kuva 16. Kehitetyn tehonäyttöohjelman päänäyttö.

4.3 Vikapuusovellus

Vikapuu on yksinkertainen vikadiagnoosimenetelmä, jonka tulkitsemiseen ja käyttöön ei tarvita monimutkaisten teorioiden ymmärtämistä. Kun ilmastointikoneen vikapuuhun liitetään vikojen havaitsemiseen tarvittavat oireet, vikapuu on ongelmatilanteissa tehokas koneen käyttäjän apukeino.

Vikapuusovellus on toteutettu siten, että vikapuun hierarkkinen rakenne on kuvattu MS ACCESS -tietokantaan. Viat on tallennettu tietokannan toiseen tauluun ja oireet toiseen tauluun. Sovelluksen käyttöliittymä ohjelmoitiin Visual Basic -ohjelmointikielellä.

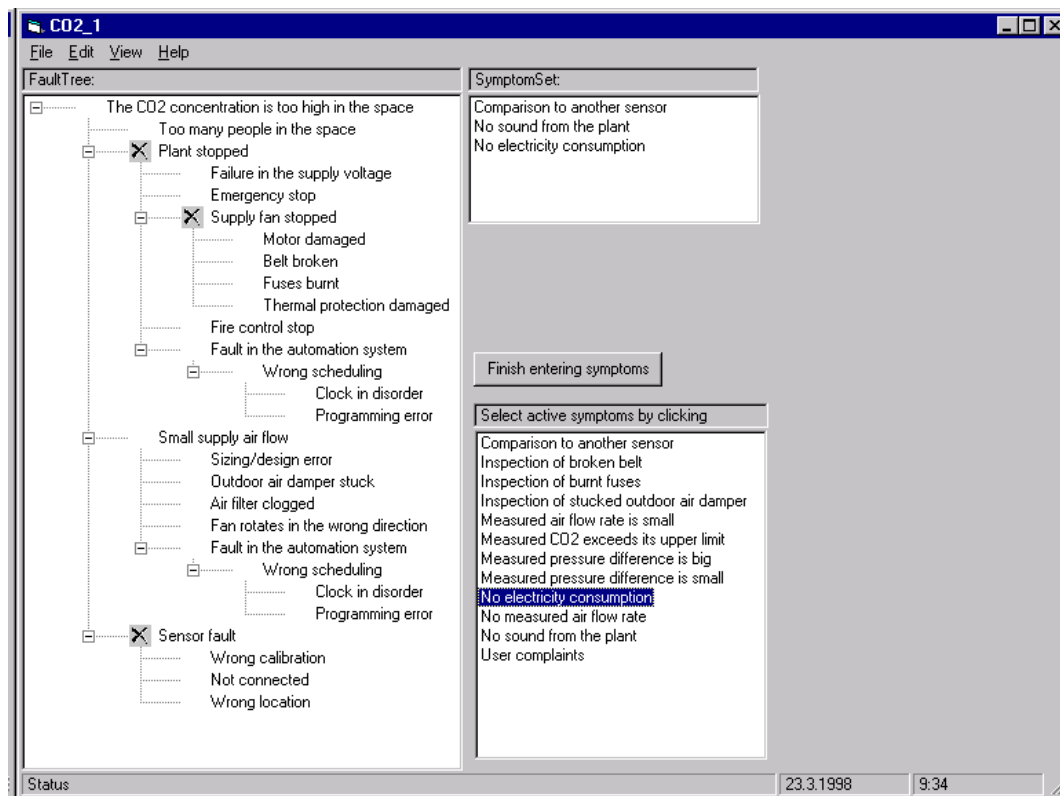
Käynnistettäessä ohjelma kysyy ensin salasanaa. Kun se on annettu oikein, vikapuusovellus käynnistyy selaustilassa (kuva 17). Prototyypisovellus on toteutettu siten, että se käsittelee aina samaa tietokantaa, jossa huippuvikana on "liian suuri CO₂-pitoisuus tilassa". Ohjelma on kuitenkin toteutettu siten, että sitä voitaisiin käyttää myös muiden vikapuutietokantojen käsittelyyn, jos ohjelmaan lisättäisiin rutiini, jossa kysyttäisiin tarkasteltavan tietokannan nimi. Ohjelma kuitenkin edellyttää, että tietokanta on rakenteeltaan samanlainen kuin prototyypisovelluksessa.



Kuva 17. Vikapuusovelluksen näyttö, kun sovellusta käytetään vikapuun selaukseen. Valitun vian oirejoukko näkyy listana.

Ohjelman toimintoja ovat:

1. Tiettyä vikaa klikkaamalla sitä vastaavat oireet näkyvät viereisessä listassa.
2. Edit-valikon alta voidaan lisätä vika, poistaa vika ja lisätä johonkin vikaan oireita. Uutta vikaa lisättäessä hyväksytään korkeintaan viisi oiretta, mutta myöhemmin niitä voi lisätä erillisellä toiminnolla.
3. Edit-valikosta käynnistyy oireiden valintatoiminto (Select Symptoms), jolloin näytetään lista kaikista mahdollisista oireista. Tästä oirelistasta voidaan aktiiviset oireet valita hiiren klikkauksella, jolloin ne siirtyvät yläpuoliseen oirejoukkoon, ja vastaavat viat merkitään puuhun (kuva 18). Kun oireet on valittu, valitaan "Finish entering symptoms". Tämän jälkeen vikapuusta voi tiettyä vikaa klikkaamalla hylätä sen vian. Toiminnosta palataan takaisin vikapuun selaukseen ja editointiin valitsemalla "Finish inferring fault causes".
4. Tietyn vian raahaus toiseen paikkaan toimii, jos ensin on tehty jotain muuta puulle. Muuten siitä seuraa virhe. Raahaus tapahtuu, kun Shift-nappia painetaan hiiren vasemman napin kanssa samanaikaisesti.



Kuva 18. Vikapuusovelluksen näyttö, kun aktiiviset oireet valitaan listasta. Valitut oireet siirtyvät oirejoukkoon ja vastaavat viat merkitään puusta.

Tietokannan käyttö ei tue oirejoukon mahdollisten sisäisten riippuvuuksien (esim. Boolean operaattoreiden) toteuttamista, joten sovelluksessa kaikki oireet ovat samanarvoisia. Tällöin käyttäjä ratkaisee, toteutuuko vika vai ei. Nämä riippuvuudet

voitaisiin toteuttaa myös ohjelmallisesti, mutta se ei ole kovin järkevää, koska silloin luovuttaisiin ohjelman generisestä luonteesta. Myös oirejoukon päivitysten tekeminen olisi silloin mahdotonta.

4.4 Yhteenveto luvusta 4

Tässä luvussa esiteltiin kolme ilmastointikoneen kunnon ja toiminnan seurantaan tarkoitettua prototyyppisovellusta. Tarkoituksena oli demonstroida, millaisia toimintoja vikadiagnostiikassa hyödynnettävillä tietokonesovelluksilla voisi olla. Toteutettavaksi valittiin ominaiskäyräsovellus, ilmastointikoneen tehonäyttö ja vikapuusovellus. Valintaan vaikuttivat ensisijaisesti projektin tavoitteet. Keskeisenä valintakriteerinä oli myös se, että sovellukset ovat helposti ymmärrettäviä.

Ominaiskäyräsovellus toteutettiin Excelin makroilla. Sovellus piirtää annetuista pistepareista polynomimuotoisen ominaiskäyrän ja antaa polynomisovitteen myös kaavamuodossa. Jos saatu referenssikäyrä poikkeaa toteutuvasta, voidaan sovelluksella piirtää uusi vertailukäyrä. Sovellus on tarkoitettu kahden muuttujan välisten riippuvuuksien visuaaliseen tarkasteluun, eikä se päättelee, mikä aiheuttaa poikkeaman ominaiskäyrissä.

Ilmastointikoneen tehonäyttö kehitettiin ThermoNet-ilmanvaihtokoneelle. Menetelmä esitettiin koneelle, jossa on käytössä mahdollisimman vähän mittauksia. Myös tehomenetelmä on vian havaitsemismenetelmä, jonka avulla on mahdollista havaita koneen poikkeava toiminta mutta ei paikantaa tämän syytä.

Vikapuusovelluksessa vikapuun hierarkkinen rakenne kuvattiin tietokantaan. Tietokannan käsittelyyn tehtiin Explorer-tyyppinen käyttöliittymä, jonka avulla vikapuuta voidaan päivittää ja selata. Käyttäjä voi myös valita aktiiviset oireet kaikkien oireiden joukosta, jolloin mahdolliset viat merkataan vikapuusta. Näistä vioista käyttäjä voi hylätä ei-mahdolliset. Vikapuusovellusta käytetään vian paikantamiseen.

5 Johtopäätökset

Tarkasteltavan ilmastointikoneen tärkeimmät suorituskykyvaatimukset liittyvät ilmastointikoneella tuotettavaan lämmitys- ja jäähdytysteeseen, tuloilmavirtaan ja sen puhtauteen, energiataloudellisuuteen ja säätötekniiseen toimintaan. Suunnittelun aikana suorituskykytekijöille asetetaan vaatimukset, vaikka yksityiskohtaisia viranomaismääräyksiä ko. vaatimuksille ei olekaan olemassa. Monet järjestelmän elinkaaren aikaiset ratkaisut vaikuttavat järjestelmällä saavutettavaan lopulliseen suorituskykyyn. Tässä julkaisussa eritellään suunnittelun-, valmistuksen-, asennuksen- ja käytönaikaisten ratkaisujen vaikutusta.

Diagnostisia menetelmiä ja työkaluja voidaan kehittää suorituskykykriteerien toteutumisen seurantaan. Jos näiden avulla todetaan, että järjestelmä ei toimi asetettujen vaatimusten rajoissa, havaitaan vika. Vian paikantamiseen on käytettävä muita menetelmiä. Menetelmä- ja työkalukehityksessä tulisi keskittyä vikoihin, jotka ovat todennäköisimpiä tai joilla on suurin vaikutus energiankulutukseen tai viihtyisyyteen ja turvallisuuteen.

Tarkastellun ilmastointijärjestelmän keskeisten suorituskykyvaatimusten perusteella kehitettiin käytännönläheisiä vikadiagnostiikkamenetelmiä. Menetelmät ovat yksinkertaisia ja helposti sovellettavissa käytäntöön eivätkä ne vaadi ylimääräistä instrumentointia. Julkaisussa esitetään menetelmät tarkastellun järjestelmän LTO-piirin tehojen seurantaan sekä lauhdelämmön ja kaukolämmön tehon seurantaan. Lisäksi käsitellään ominaiskäyrämenetelmän ja vikapuun käyttöä vikadiagnostiikassa.

Ominaiskäyrät ja vikapuut ovat yleisiä menetelmiä, joita voidaan käyttää myös muiden järjestelmien vikojen havaitsemiseen, paikantamiseen ja tulkintaan. Ominaiskäyriä voidaan käyttää esimerkiksi puhaltimissa ja LTO:ssa esiintyvien vikojen havaitsemiseen. Tässä julkaisussa kuvataan kehitetty, yleinen prototyypiohjelma ominaiskäyrien käsittelyyn.

Prosessin vikatietämys voidaan jäsentää vikapuun avulla. Vikapuita käytetään laajasti monilla alueilla, mutta yleensä vikapuissa esitetään vain vikojen välinen hierarkia, ei vikojen havaitsemiseen käytettäviä oireita. Tässä julkaisussa selvitettiin sopivia työkaluja vikapuun kuvaaman asiantuntijatiedon käsittelyyn määriteltyjen vaatimusten ja tarpeiden perusteella. Työkaluohjelmat jaoteltiin ohjelmoinnin perustyökaluihin, ohjelmointiympäristöihin, tietämyspohjaisiin työkaluohjelmiin, sumean logiikan kehitysohjelmiin, neuraaliverkko-ohjelmiin sekä muihin

valmisohjelmiin. Jokaisesta työkaluohjelmatyypistä esiteltiin useita työkaluja kuvaamalla lyhyesti niiden keskeisimmät ominaisuudet.

Koska mikään tarkastelluista työkaluohjelmista ei täytä kaikkia esitettyjä tarpeita ja vaatimuksia, kehitettiin prototyypiosuudessa vikapuusovellus, jossa vikapuun hierarkkinen rakenne on kuvattu relaatiotietokantaan, ja tietokannan käsittelyyn tehtiin Explorer-tyyppinen käyttöliittymä. Kehitettyä vikapuusovellusta voidaan käyttää vika-oirepuun selaukseen ja päivitykseen sekä aktiiviset oireet valitsemalla voidaan päätellä mahdolliset viat.

Tämän työn yksi tavoite oli kehittää ja esitellä tarkastellun ilmastointijärjestelmän toiminnan seurantaan ja vikadiagnostiikkaan sopivia menetelmiä. Muutamia lupaavimmat menetelmät toteutettiin prototyypitasolla. Näiden hyödyllisyyden ja käytettävyyden arviointi tehdään käytännössä, vaikka vikadiagnoosimenetelmien toimivuutta on vaikea arvioida täysin luotettavasti. Kuitenkin ensivaiheen testauksen perusteella varsinkin vikapuusovellus vaikuttaa lupaavalta.

Lähdeluettelo

- ABB Fläkt Oy. 1996. ThermoNet-talotekniikkakeskus. Suunnittelijan opas 04.96. 8 s.
- ABB Installaatiot Oy. 1995. ThermoNet järjestelmäkäsikirja. Market. 69 s.
- Becks, A. 1995. DELPHI Sovelluksen tekijän opas. Espoo: Suomen ATK-kustannus Oy. 624 s. ISBN 951-762-289-9
- Creswick, M. 1995a. LEVEL5 OBJECT Professional Release 3.6. PC AI, July/August 1995, s. 39 - 42.
- Creswick, M. 1995b. XpertRule - More Than the Name Suggests. PC AI, November/December 1995, s. 39 - 42.
- Doores, J., Reiblein, A.R. & Vadera, S. 1987. PROLOG - Programming for Tomorrow. Southampton, Great Britain: Sigma Press. 149 s. ISBN 0-905104-52-8
- Harmon, P. (ed.) 1992. Case-based reasoning III. Intelligent Software Strategies, vol. 8, no. 1, s. 1 - 12.
- Hayes-Roth, F. 1984. Knowledge-based expert systems. Computer 17, s. 263 - 273.
- Hjerpe, R. 1983. What artificial intelligence can, could and can't do for libraries and information services. Seventh International Online Information Meeting Conference Proceedings. London, December 6 - 8, 1983. S. 7 - 25.
- Hyvärinen, J. 1987. Vikadiagnoosijärjestelmä turvevoimalaitoksen turpeen syötön ja savukaasujärjestelmän valvomiseksi. Espoo: Teknillinen korkeakoulu, sähkötekniikka osasto. 96 s. Diplomityö.
- Hyvärinen, J. 1993. Static models and characteristic curves. Teoksessa: Hyvärinen, J. & Kohonen, R. (toim.). Building Optimisation and Fault Diagnosis system concept. Helsinki: Painatuskeskus Oy. S. 96 - 103. ISBN 952-9601-16-6
- Hyvärinen, J. 1996. Building optimization and fault diagnosis system concept. Teoksessa: Hyvärinen, J. & Kärki, S. (toim.). Building Optimization and Fault Diagnosis Source Book. Helsinki: Oy Edita Ab. S. 22 - 44. ISBN 952-5004-10-4

- Hyvönen, E. & Seppänen, J. 1987. LISP-maailma. Johdatus kieleen ja ohjelmointiin. Espoo: Kirjayhtymä Oy. 360 s. ISBN 951-26-2787-6
- IEC 1025. 1990. International standard on fault tree analysis (FTA). International Electrotechnical Commission. 39 s.
- Karttunen, H. 1994. Datan käsittely. CSC - Tieteellinen laskenta Oy. 224 s. ISBN 952-9821-12-3
- Koikkalainen, P. 1992. Neurocomputing Systems: Formal Modeling and Software Implementation. Lappeenranta: Lappeenrannan teknillinen korkeakoulu. 142 s. (Tieteellisiä julkaisuja 23). Väitöskirja.
- Koikkalainen, P. (toim.) 1994. Neurolaskennan mahdollisuudet. Helsinki: Paino-Center Oy. 151 s. (TEKES Julkaisu 43/94). ISBN 951-47-1950-6
- Korpela, J. & Larmela, T. 1989. C-ohjelmointikieli. 3. korjattu painos. Espoo: OtaDATA ry. 231 s. ISBN 951-767-049-4
- Kosonen, R., Laitila, P., Bitter, R., Laine, T. & Lahdenperä, R. 1995. LVI-järjestelmien säädön toiminnan tarkistaminen. Espoo: Valtion teknillinen tutkimuskeskus. 48 s. + liitt. 13 s. (VTT Tiedotteita 1629). ISBN 951-38-4744-6
- Kurki, M. 1995. Model-based fault diagnosis for mechatronic systems. Espoo. Technical Research Centre of Finland. 116 s. (VTT Publications 223). ISBN 951-38-4761-6
- Kärki, S. 1997. Suorituskykytekijät ilmastointikoneen vikadiagnostiikan lähtökohdina. Teoksessa: RAKET. Tutkimusseminaari 1997. Espoo: VTT Rakennustekniikka, 1997. S. 55 - 61.
- Kärki, S. & Hyvärinen, J. 1996. Implementation of a fault tree method. Teoksessa: Adey, R.A., Rzevski, G. & Sunol, A.K. (toim.). Applications of Artificial Intelligence in Engineering XI. 11th International Conference on Applications of Artificial Intelligence in Engineering (AIENG XI), Clearwater, Florida, USA, 11 - 13 September 1996. Southampton: Computational Mechanics Publications. S. 131 (abst.), artikkeli julkaisuun liittyvällä CD-ROMilla. ISBN 1-85312-410-9
- Kärki, S. & Hyvärinen, J. 1997. Ilmastointikoneen suorituskyvyn seuranta. Espoo: Valtion teknillinen tutkimuskeskus. 44 s. (VTT Tiedotteita 1832). ISBN 951-38-5110-9

- Leivo, A. & Sirén, K. 1991. Ilmastointijärjestelmien säätöön liittyvät viat ja virheetoiminat. Espoo: Teknillinen korkeakoulu, LVI-tekniikan laboratorio. 97 s. + liitteet. (Raportti B29).
- LVI 03-10242. 1995. Talotekniikan suunnittelun tehtäväluettelo. TATE95. Helsinki: Rakennustieto Oy. 12 s. (Ohjetiedosto LVI 03-10242).
- LVI 39-10073. 1985. Ilmanvaihtolaitoksen hoito-ohje. Helsinki: Rakennuskirja Oy. 6 s. (Ohjetiedosto LVI 39-10073).
- LVI 70-40013. 1991. LVI-järjestelmien käyttövarmuuden tarkastelumenetelmät. Helsinki: Rakennuskirja Oy. 3 s. (Tiedonjyväkortti LVI 70-40013).
- LVI-RYL 92. 1992. LVI-töiden yleiset laatuvaatimukset. 2., täysin uusittu laitos. Helsinki: Rakennustieto Oy. 440 s. ISBN 951-682-226-6
- Lyytikäinen, A. 1987. Käyttövarmuuskäsikirja. Espoo: Valtion teknillinen tutkimuskeskus. 147 s. + liitt. 6 s. (VTT Tiedotteita 678). ISBN 951-38-2633-3
- Madjidi, M. 1996. Characteristic curves. Teoksessa: Hyvärinen, J. & Kärki, S. (toim.). Building Optimization and Fault Diagnosis Source Book. Helsinki: Oy Edita Ab. S. 163 - 174. ISBN 952-5004-10-4
- Norford, L. & Little, R. D. 1993. Fault detection and load monitoring in ventilation systems. ASHRAE Transactions. Vol. 99, Part 1.
- Pakanen, J., Hyvärinen, J., Kuismin, J. & Ahonen, M. 1996a. Fault diagnosis methods for district heating substations. Espoo: Technical Research Centre of Finland. 70 s. (VTT Research Notes 1780). ISBN 951-38-4975-9
- Pakanen, J., Kärki, S. & Möttönen, V. 1996b. Kehitysympäristö LVI-prosessien vikadiagnoosimenetelmille. Espoo: Valtion teknillinen tutkimuskeskus. 44 s. (VTT Tiedotteita 1773). ISBN 951-38-4968-6
- Pietiläinen, P. & Salonen, J. 1994. LVI-järjestelmien käyttöönotto - onko seula liian harva? Talotekniikka, nro 7, s. 52 - 57.

Pobil, A.P. & Arroyo, V. 1996. Technical criteria for selecting expert system shells for engineering applications. Teoksessa: Adey, R.A., Rzevski, G. & Sunol, A.K. (toim.). Applications of Artificial Intelligence in Engineering XI. 11th International Conference on Applications of Artificial Intelligence in Engineering (AIENG XI), Clearwater, Florida, USA, 11 - 13 September 1996. Southampton: Computational Mechanics Publications. S. 99 - 100 (abst.), artikkeli julkaisuun liittyvällä CD-ROM:lla. ISBN 1-85312-410-9

Principe, J. 1995. NeuroSolutions. An Object Oriented Evolution. PC AI Magazine, November/December 1995, Volume 9, Number 6, s. 49 - 51.

Rosenhave, P.-E. 1997. Acoustic multivariate condition monitoring - AMCM. Teoksessa: Jantunen, E., Holmberg, K. & Rao, R. (toim.). COMADEM '97. 10th International Congress and Exhibition on Condition Monitoring and Diagnostic Engineering Management. Vol. 1. Espoo, Finland 9 - 11 June 1997. Espoo: VTT Manufacturing Technology, Operational Reliability. S. 193 - 202. (VTT Symposium 171). ISBN 951-38-4562-1

RT 16-10288. 1985. Urakkarajaliitteen malli. Helsinki: Rakennuskirja Oy. 6 s. (Ohjetiedosto RT 16-10288).

Ruokonen, T. 1986. Ominaiskäyrien käyttö kunnonvalvonnassa. Helsinki: Teknillinen korkeakoulu, Sääätötekniikan laboratorio. 36 s. (Laboratorioraportti 66).

Ruokonen, T. & Lautala, P. 1986. Mallipohjainen päättely teollisuusprosessin vikadiagnoosissa. Teoksessa: Suomen Tekoälypäivät STeP-86, 20. - 22.8.1986, III-julkaisu. 9 s.

Räty, J. 1986. FAULT - vikapuiden laadintaohjelmisto. Espoo: Valtion teknillinen tutkimuskeskus. 35 s. + liitt. 5 s. (VTT Tiedotteita 568). ISBN 951-38-2563-9

Schmuller, J. 1993. M.4 Something for Everyone. PC AI, November/December 1993, s. 30 - 34.

SFS 5768. 1993. Ilmastointijärjestelmien säädön toiminnalle asetettavat vaatimukset. Helsinki: Suomen Standardisoimisliitto. 6 s.

Shemeikka, J., Kosonen, R., Hoving, P., Laitila, P., Pihala, H. & Laine, T. 1996. Rakennuksen sähköenergiankulutuksen tavoitearvot. Espoo: Valtion teknillinen tutkimuskeskus. 123 s. + liitt. 9 s. (VTT Tiedotteita 1756). ISBN 951-38-4911-2

Suomen Sisäilmayhdistys. 1995. Sisäilmayhdistyksen julkaisu 5. Sisäilmaston, rakennustöiden ja pintamateriaalien luokitus. Helsinki: Suomen Sisäilmayhdistys. 31 s.

Suvisaari, S. & Suvisaari, O. 1995. Visual Basic 3 - Ohjelmoijan opas. 2. laajennettu painos. Espoo: Suomen ATK-kustannus Oy. 203 s. ISBN 951-762-314-3

Svenska Inneklimatinstitutet. 1991. SCANVAC. Klassindelade luftdistributionssystem - riktlinjer och specifikationer. Riktlinjeserien R2, version 1.0. Stockholm: Svenska Inneklimatinstitutet. 27 s.

Wendt, L. 1993. We had joy, we had fun - EXSYS Professional. AI Expert, October 1993, s. 17 - 19.

Williams, P.W. 1984. A model for an expert system for automated information retrieval. Eight International Online Meeting Conference Proceedings. Learned Information, Oxford, 1984. S. 139 - 149.

Ympäristöministeriö. 1985. Suomen rakentamismääräyskokoelma, osa D5. Rakennusten lämmityksen teho- ja energiantarpeen laskenta. Ohjeet 1985. 9 s. + liitteet.

Ympäristöministeriö. 1987. Suomen rakentamismääräyskokoelma, osa D2. Rakennusten sisäilmasto ja ilmanvaihto. Määräykset ja ohjeet 1987. 21 s.

Ympäristöministeriö. 1996. Rakennusalan tyyppihyväksyntä. Ympäristöopas 12. Helsinki: Ympäristöministeriö, Asunto- ja rakennusosasto. 28 s.

Yuzawa, H., Suzuki, M., Takei, H. & Yoshida, H. 1996. The FTA system for application to HVAC systems. Teoksessa: Hyvärinen, J. (toim.). Real Time Simulation of HVAC Systems for Building Optimization, Fault Detection and Diagnosis. Technical Papers of IEA Annex 25. Helsinki: Oy Edita Ab. S. 129 - 144. ISBN 952-5004-11-2

Årzén, K.-E. 1993. A survey of commercial real-time expert system environments. Teoksessa: Verbruggen, H.B. & Rodd, M.G. (toim.). Artificial Intelligence in Real-Time Control, Delft, Hollanti 16. - 18.6.1992. Oxford: Pergamon. S. 483 - 490.

Ohjelmistotuotteiden WWW-sivut

Taulukko 1. Ohjelmointiympäristöjen WWW-sivut.

| Tuote | Kotisivu |
|-----------------------------|---------------------------------------------------------------------------------------------------------|
| Amzi! Prolog + Logic Server | http://www.amzi.com/amzi_faq.txt |
| Delphi | http://www.borland.com/delphi/ |
| GCLISP Developer | http://www.goldhill-inc.com/developer.html |
| Visual Basic | http://www.microsoft.com/vbasic/ |
| Visual Prolog | http://www.pdc.dk/vip/html/vp_info_index.htm |
| WIN-PROLOG | http://www.lpa.co.uk/win.html |

Taulukko 2. Tapauspohjaisten päättelytyökalujen WWW-sivut.

| Tuote | Kotisivu |
|-------------|-------------------------------------------------------------------------------------------------------------------------|
| CaseAdvisor | http://www.cs.sfu.ca/cbr/shell.html |
| CBR Express | http://www.inference.com/products/author.html |
| CBR-Works | http://www.tecmath.de/tecinno/tecinno_e/ecbrwork.htm |
| KATE-CBR | http://www.acknosoft.com/Tools.html |
| ESTEEM | http://www.shai.com/esteem.html |
| ReCall | http://www.alice.fr/products/recall.html |
| ReMind | ei tiedossa |

Taulukko 3. Sääntöpohjaisten päättelytyökalujen WWW-sivut.

| Tuote | Kotisivu |
|----------------------------------|---------------------------------------------------------------------------------------------------|
| ACQUIRE | http://vuv.com/ai/ |
| CLIPS | http://www.jsc.nasa.gov/~clips/CLIPS.html |
| Design++ | http://www.dpe.fi/ |
| EXSYS | http://www.exsysinfo.com/ |
| Flex | http://www.lpa.co.uk/ |
| G2 | http://www.gensym.com/ |
| GoldWorks | http://www.goldhill-inc.com/index.html |
| LEVEL5 OBJECT | http://www.l5r.com/ |
| M.4 | http://www.teknowledge.com/M4/ |
| Nexpert Object ja Smart Elements | http://www.neurondata.com/ |
| OPS/R2 | ei tiedossa |
| PowerModel | http://www.intellicorp.com/Default.htm |
| Rete++ | http://www.haley.com/ |
| XpertRule | http://www.attar.com/ |

Taulukko 4. Sumean logiikan kehitysohjelmien WWW-sivut.

| Tuote | Kotisivu |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| DataEngine | http://www.mitgmbh.de/mit/products/de/de.html |
| FlexTool(FS) | http://www.flextool.com/ftfs.html |
| FUZZLE | http://users.aol.com/fuzzle20/hmpage2.html |
| Fuzzy Knowledge Builder | http://www.fuzzysys.com/fkbprod.html |
| FuzzyCLIPS | http://ai.iit.nrc.ca/fuzzy/fuzzy.html |
| <i>fuzzyTECH</i> | http://www.fuzzytech.com/e_fted.htm |
| O'INCA | ei tiedossa |
| TILShell | http://www.ortech-engr.com/fuzzy/TilShell.html |
| WINROSA | http://www.mitgmbh.de/mit/products/winrosa/winrosa.html |

Taulukko 5. Neuraaliverkko-ohjelmien WWW-kotisivut.

| Tuote | Kotisivu |
|---------------------------------|-----------------------------------------------------------------------------------------------------------|
| BrainMaker | http://www.calsci.com/ |
| ECANSE | http://www.snat.de/nc9/welcome.htm |
| Explorenet 2000 | ei tiedossa |
| FlexTool(LNN) | http://www.flextool.com |
| LVQ_PAK, SOM_PAK | http://nucleus.hut.fi/nnrc/nnrc-programs.html |
| Mtialab/Neural Networks Toolbox | http://www.mathworks.com/ |
| NeuFrame | http://www.demon.co.uk/skylake/ |
| NeuralWorks | http://www.neuralware.com/index.html |
| NGO&trade | http://www.bio-comp.com/prod01.htm |
| NeuroShell | http://www.wardsystems.com/Products.htm |
| NeuroSolutions | http://www.nd.com/ |
| NeuroWindows | http://www.wardsystems.com/Products.htm |
| O'INCA | ei tiedossa |
| OWL | ei tiedossa |
| Trajan | http://www.trajan-software.demon.co.uk/ |
| WinNN | ei tiedossa |

Taulukko 6. Muiden valmisohjelmien WWW-sivut.

| Tuote | Kotisivu |
|------------|-----------------------------------------------------------------------------------------------|
| CAFTA | ei tiedossa |
| FAULT | ei tiedossa |
| FaultTree+ | http://www.itemsoft.com/ |
| GeneHunter | http://www.wardsystems.com/Products.htm |
| Matlab | http://www.mathworks.com/ |
| Sugal | http://www.trajan-software.demon.co.uk/ |