

Tampereen ammattikorkeakoulu
Auto- ja kuljetustekniikan koulutusohjelma
Auto- ja työkonetekniikka

Opinnäytetyö

Matti Mäkinen

**Työkonemoottorin imusarjan mallintaminen
virtauslaskennan avulla**

Työn ohjaaja
Työn tilaaja

Tampere 2009

Tekniikan lisensiaatti Tauno Kulojärvi
AGCO Sisu Power, valvojana suunnittelupäällikkö
Ismo Hämäläinen

Tampereen ammattikorkeakoulu
Auto- ja kuljetustekniikan koulutusohjelma
Auto- ja työkonetekniikka

Tekijä	Matti Mäkinen
Työn nimi	Työkonemoottorin imusarjan mallintaminen virtauslaskennan avulla
Sivumäärä	54
Valmistumisaika	Elokuu 2009
Työn ohjaaja	Tekniikan lisensiaatti Tauno Kulojärvi
Työn tilaaja	AGCO Sisu Power, valvojana suunnittelupäällikkö Ismo Hämäläinen

Tiivistelmä

Opinnäytetyön tarkoituksena oli tutustua virtauslaskentaan ja mallintaa työkonemoottorin imusarja virtauslaskennan avulla. Pää tavoite oli optimoida imusarjan virtaus tasaisesti imuputkien kesken. Lisäksi työn aikana tutkittiin virtauslaskentaohjelmiston soveltuvuutta suunnittelutyöhön. Työn tarkoituksena ei ollut suunnitella imusarjaa, joka jonain päivänä siirtyisi tuotantoon. Pääpaino oli nimenomaan virtauslaskentaohjelmistoon tutustumisessa.

Työhön sisältyi virtauslaskentaan tutustumista teorian osalta ja mallinnus- ja laskentatehtäviä pääasiassa CATIA-mallinnusohjelmistolla ja ANSYS CFX -virtauslaskentaohjelmistolla. Virtauslaskentaa varten luotiin 3D-geometria kuvitteellisesta imusarjasta. Imusarjamalleja kehitettiin työn teettäjän toivomusten ja työn tekijän parannusten myötä.

Valmis 3D-geometria siirrettiin virtauslaskentaohjelmistoon, jossa siihen määritettiin elementtiverkko, virtauskenttä ja työn tilaajan määräämät reunaehdot. Elementtiverkkoa tihennettiin kriittisistä kohteista, jotta laskennan tulokset olisivat mahdollisimman tarkkoja. Tämän jälkeen suoritettiin virtauslaskenta, jonka tuloksia analysoitiin tuotetun datan pohjalta. Tuloksien pohjalta 3D-geometriaa muokattiin edelleen, jotta virtausjakauma saatiin mahdollisimman tasaiseksi.

Työn tuloksena syntyi kymmeniä variaatioita imusarjan 3D-geometriasta, joiden virtausominaisuuksia vertailtiin keskenään parhaan mahdollisen kombinaation löytämiseksi. Virtauslaskentaohjelmiston katsottiin soveltuvan suunnittelutyön tueksi hyvin. Ohjelmiston sujuva käyttö edellyttää virtauslaskennan perusteiden osaamista.

Writer(s)	Matti Mäkinen
Thesis	Modeling an Intake Manifold of an Off-road Vehicle Engine using CFD
Pages	54
Month and Year of Completion	August 2009
Thesis Supervisor	Lic. Tech. Tauno Kulojärvi
Co-operating Company	AGCO Sisu Power, Supervisor: Design Manager Ismo Hämäläinen

Abstract

The purpose of this thesis was to model an intake manifold for an off-road vehicle engine using Computational Fluid Dynamics. The main objective was to distribute the airflow in the intake manifold evenly between the intake pipes. The purpose of this thesis was not to produce an intake manifold for future manufacturing. The main aim was to study the CFD software and to determine whether the software is an effective aid in a design process.

The project included tasks like modeling geometry and CFD files with CATIA and ANSYS software, and studying the theory behind CFD. A 3D-geometry of an intake manifold was created from scratch for this project. The 3D-geometry was modified constantly throughout the project according to the commissioning company's wishes and the writer's innovations. The modified 3D-geometry was imported to the CFD software where it was then meshed and prepared for the computing process. Afterwards, the results were processed and analysed.

The project produced dozens of intake manifold geometries that were carefully studied to find the optimal combination. It was determined that the CFD software is an effective tool for a professional engineer. The basics are easy to grasp, but to master the trade takes years.

Esipuhe

Idea virtauslaskentaan liittyvästä opinnäytetyöstä tuli auto- ja kuljetustekniikan koulutuspäälliköltä, tekniikan lisensiaatti Tauno Kulojärveltä, joka oli jo pitkään suunnitellut virtauslaskentaohjelmiston hankkimista koululle. Virtauslaskentaohjelmisto oli saatavissa ilmaiseksi akateemista tutkimusta varten Tieteen tietotekniikan keskukselta eli CSC:ltä. CSC lainasi myös supertietokonekapasiteettiaan vaativiin laskentaoperaatioihin.

Työn tilaajaksi valikoitui AGCO Sisu Power, jonka palveluksessa opinnäytetyön tekijä oli jo kahtena edellisessä oltu harjoittelussa. Virtauslaskentaohjelmiston tarjoajan tiukoista lisenssirajoituksista johtuen päädyimme Sisu Powerin kanssa aiheeseen, jonka tarkoitus on yksinomaan akateeminen tutkimus.

Haluan kiittää kaikkia jotka osaltaan auttoivat minua opinnäytetyön valmiiksi saattamisessa. Suurin kiitos kuuluu rakkaalle vaimolleni Nooralle, joka aina tarpeen tullen antoi kannustavan potkun eteenpäin. Lisäksi erityisen kiitollisuuteni ansaitsevat Jani Ojala Process Flow:lta, Esko Järvinen CSC:ltä, Ismo Hämäläinen AGCO Sisu Powerilta sekä Harri Myllysoo ja Esa Huhtamäki omalta luokalta.

Tampereella 17. elokuuta 2009

Matti Mäkinen

Sisällysluettelo

Tiivistelmä	2
Abstract	3
Esipuhe	4
Sisällysluettelo	5
1 Johdanto	7
2 Virtauslaskennan teoria	8
2.1 Virtauslaskennan historiaa	8
2.2 Virtauslaskennan perusteet	9
2.3 Virtauslaskennan sovellukset	10
2.4 ANSYS CFX:ssä käytetty teorianmalli	11
2.4.1 Tilavuusmenetelmä	13
2.4.2 k -epsilon -turbulenssimalli	15
2.5 Virtauslaskenta prosessina	17
3 Työssä käytetyt ohjelmistot	18
3.1 CATIA	18
3.2 ANSYS Workbench ja CFX	18
3.3 SSH Secure Shell Client	18
4 3D-geometrian luominen virtauslaskentaa varten	19
4.1 Reunaehdot 3D-geometrialle	19
4.2 CATIAlla luodut 3D-geometriat	19
5 Virtauslaskentaprosessin kuvaus	23
5.1 ANSYS Workbenchin käyttö	23
5.1.1 DesignModelerin käyttö	25
5.1.2 Verkotuksen luominen	27
5.2 Virtauslaskennan suorittaminen ANSYS CFX:llä	29
5.2.1 Reunaehdot virtauslaskennalle	29
5.2.2 CFX-Pren käyttö	30
5.2.3 CFX-Solverin käyttö	32
5.2.4 CFX-Postin käyttö	35

5.3 CSC:n supertietokoneen käyttäminen	37
5.3.1 Interaktiivinen laskenta	38
5.3.2 Ei-interaktiivinen laskenta	40
6 Tulosten analysointi	42
6.1 Tulosten tarkkuuteen vaikuttavat tekijät	42
6.2 Virtausnopeudet eri kierrosnopeuksilla.....	42
6.3 Tekijät jotka vaikuttavat virtauksen tasaiseen jakautumiseen	45
7 Päätelemät	49
7.1 3D-mallinnuksen onnistuminen	49
7.2 Virtauslaskennan onnistuminen	50
7.3 ANSYS CFX -ohjelmiston soveltuvuus suunnittelutyöhön.....	51
7.4 Ajatuksia projektin jatkokehittämisestä	52
Lähteet.....	53
Liitteet	54
Liite 1: Esimerkki jobfile-tiedoston rakenteesta rinnakkaisajossa.....	54

1 Johdanto

Tämän opinnäytetyön tavoitteena on tutustua virtauslaskennan perusteisiin ja mallintaa ANSYS CFX 11.0 -virtauslaskentaohjelmiston avulla työkonemoottorin imusarja. Pääpaino työssä on virtauslaskentaohjelmiston käytön opettelussa, sillä tulevaisuudessa TAMKissa on tarkoitus teettää useampia opinnäytetöitä ja projekteja kyseisellä ohjelmistolla. Opinnäytetyön tilaajan puolelta työn tavoitteeksi asetettiin mahdollisimman tasaisen virtausjakauman saavuttaminen imuputkien kesken. Tähän pyritään muokkaamalla imusarjan 3D-geometriaa ja vertailemalla jokaisen mallin sisäisiä virtauksia.

Imusarjan 3D-geometria luodaan CATIAlla ja virtauslaskenta suoritetaan ANSYSin virtauslaskentaohjelmistolla. Imusarja luodaan täysin tätä opinnäytetyötä silmälläpitäen. Työn aikana on tarkoitus tutkia miten imusarjan 3D-geometrian muutokset vaikuttavat virtausten jakautumiseen imuputkissa. Lisäksi tarkastellaan virtauslaskentaohjelmiston soveltuvuutta insinöörin apuvälineeksi suunnittelutyössä. Virtauslaskenta suoritetaan jokaisella imusarjageometrialla kolmessa eri pisteessä: suurimman kierrosnopeuden alueella (2100 1/min), suurimman vääntömomentin alueella (1500 1/min) ja pienimmän kierrosnopeuden alueella (1000 1/min).

Opinnäytetyötä varten luodulla imusarjalla ei ole kaupallisia tavoitteita. Tämä johtuu osaltaan ANSYSin lisenssirajoituksista ja opinnäytetyön laajuudesta.

Virtauslaskenta suoritetaan avoimilla malleilla, eli sisääntuloaukosta tulee imusarjaan tapauskohtainen massavirta ja kaikki imuputket ovat päästään avoimia. Työmäärän räjähdysmäisen kasvamisen estämiseksi laskennassa ei oteta huomioon venttiilien avautumista ja sulkeutumista.

Tämän opinnäytetyön teknisen raportin on tarkoitus toimia johdatuksena virtauslaskennan maailmaan ja erityisesti ANSYS CFX -virtauslaskentaohjelmiston käyttöön ja opinnäytetyön aiheen mukaisesti keskittyä kohteen sisäisten virtausten tutkimiseen.

2 Virtauslaskennan teoria

Virtauslaskenta tutkii tietokonesimulaation avulla nesteiden ja kaasujen virtausten käyttäytymistä erilaisissa tilanteissa.

2.1 Virtauslaskennan historiaa

Virtauslaskenta sai alkunsa 1960-luvulla, jolloin tiedemiehet alkoivat tehdä tutkimuksia aineiden virtauksista lähinnä yritysten tutkimus- ja kehitysosastoilla. 1970-luvun puolivälin jälkeen ryhdyttiin kehittämään yleisiä virtauslaskentaratkaisuja, kun monimutkaisten algoritmien yleistämiseen tarvittavaa matemaattista taustaa alettiin ymmärtää. (ANSYS CFX Introduction 2006, 1)

1980-luvulla virtauslaskenta alkoi kehittyä nopeasti, käsi kädessä tietokoneiden kanssa. Silloisia tietokoneita alettiin käyttää monimutkaisten virtauslaskentojen ratkaisemiseen. Jo 1980-luvun alusta lähtien alkoi ilmestyä kaupallisia virtauslaskentakoodeja. Tämä tarkoitti sitä, että yritykset luopuivat vähän kerrallaan omien virtauslaskentakoodiensa kehittämisestä. (A Brief History of Computational Fluid Dynamics (CFD).)

Tietokoneiden edelleen kehittyessä virtauslaskennassa otettiin käyttöön virtausmallien graafinen ja interaktiivinen esittäminen. Malleja kyettiin myös muokkaamaan kolmiulotteisina tietokoneen näytöllä. Kaupallisten virtauslaskentaohjelmistojen toiminta perustui hyvin monimutkaiseen epälineaarisiin matemaattisiin malleihin, joista saatiin yhtälöt nesteiden virtauksen, lämmön- ja materiaalin siirron iteratiiviseen ratkaisemiseen. (A Brief History of Computational Fluid Dynamics (CFD).)

Nykyisin kaupalliset virtauslaskentaohjelmistot ovat niin helppokäyttöisiä, että jo lyhyellä aiheeseen ja ohjelmistoon tutustumisella voi ryhtyä tekemään yksinkertaisia laskentoja, kuten on asian laita tämänkin projektin kanssa. Virtauslaskentaa pidetään nykyään osana tietokoneavusteisen mallintamisen skaalaa. Se on käytössä lähes kaikilla teollisuuden sektoreilla.

Vapaaseen lähdekoodiin perustuva virtauslaskenta elää myös vahvana alan ammattilaisten keskuudessa. Alalla ammatikseen toimivat ovat edelleen hyvin korkeasti koulutettuja tutkijoita, jotka kehittävät tarvittaessa virtauslaskentaa varten oman laskentakoodin täysin laskentatapausta varten ja sen ehdoilla.

2.2 Virtauslaskennan perusteet

Virtauslaskennan perustana toimivat 1800-luvun alussa kehitetyt Navier-Stokesin yhtälöt, joilla kuvataan liikkeen, lämmön ja aineensiirron prosesseja. Yhtälöt muistuttavat paljon Eulerin kehittämiä yhtälöitä, mutta ottavat lisäksi huomioon virtauksen viskositeetin. Näihin yhtälöihin ei ole analyttistä ratkaisua, mutta ne voidaan diskretisoida ja ratkaista numeerisesti.

Tähän tekniikkaan perustuvat virtauslaskennan tärkeimmät menetelmät joista mainittakoon Finite Difference Method (FDM), Finite Element Method (FEM) ja Finite Volume Method (FVM). Kussakin menetelmässä virtauslaskentakoodi muodostetaan muokkaamalla Navier-Stokesin yhtälöitä muotoon, josta saadaan likiarvoisia tuloksia. (NASA, Navier-Stokes Equations)

Yleisesti, virtauslaskennassa tarkasteltava jatkuva alue jaetaan verkon avulla äärellisiin osiin. Verkko koostuu elementeistä, jotka puolestaan määrittelevät solmupisteet jotka sijaitsevat jossain elementtien alueella tai rajoilla, ratkaisumallista riippuen.

Diskretisointia käyttäen alueella tutkittavasta virtauksesta ei ratkaista jatkuvan alueen jokaisen pisteen tuntemattomia, vaan tyydytään ratkaisemaan solmupisteiden tuntemattomat. Solmupisteiden ratkaistavat yhtälöt määräytyvät virtauslaskennassa käytettävän ratkaisumallin ja reunaehtojen mukaisesti. (Bhaskaran & Collins, 3)

Solmupisteiden ulkopuoliset arvot saadaan selville solmupisteistä interpoloimalla. Interpolointi johtaa likiarvoisiin tuloksiin, joiden tarkkuutta voidaan parantaa käyttämällä tiheämpää verkkoa, joka sisältää enemmän solmuja. (Bhaskaran & Collins, 3)

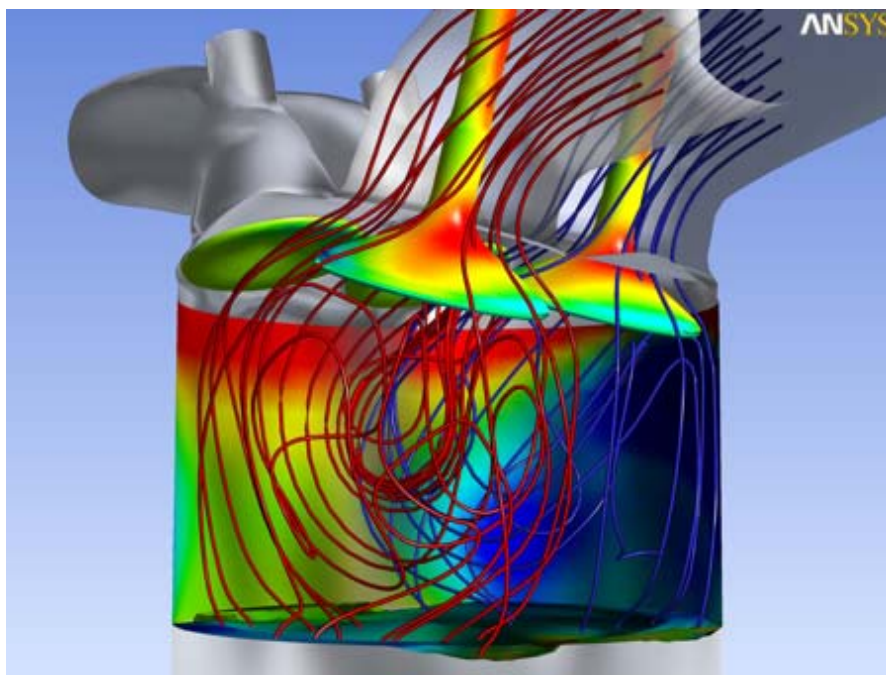
Tällainen jatkuvasta alueesta rajattu diskretisoitu systeemi koostuu suuresta määrästä diskreettejä muuttujia, jotka sisältävät algebrallisia yhtälöryhmiä. Näiden

solmuyhtälöiden ratkaiseminen johtaa suureen määrään toistuvia laskentoja, joiden ratkaisemiseen käytetään nykyään tietokoneita tai supertietokoneklustereita. (Bhaskaran & Collins, 3)

2.3 Virtauslaskennan sovellukset

Nykyisin virtauslaskentaa käytetään lähes jokaisella teollisuuden sektorilla. Virtauslaskentaa sovellettiin alkuvuosista lähtien lentokoneiden aerodynamiikan tutkimiseen. Lentokoneen nostetta, vastus- ja sivuttaisvoimia tutkittiin, jotta voitiin päätellä tarvittava moottoriteho, rahtikapasiteetti ja polttoaineen kulutus. Aluksi virtauksia tutkittiin tuulitunneleissa, mutta nykyään on siirrytty tietokonesimulointiin. (Shaw, C.T., 2)

Autoteollisuudessa käytetään myös paljon virtauslaskentaa. Sillä voidaan simuloida esimerkiksi virtauksia ajoneuvon ulkopuolella, ilmastoinnin virtauksia ajoneuvon sisäpuolella ja polttomoottorin palotapahtumaa (kuvio 1) ja pakokaasujen virtausta. (ANSYS CFX Introduction 2006, 2)



Kuvio 1: ANSYSin virtauslaskentaohjelmistolla mallinnettu polttomoottorin palotila ja venttiilit (Kuva: www.ansys.com)

Prosessiteollisuudessa voidaan esimerkiksi tutkia monimutkaisia kemiallisia reaktioita palamisprosesseissa. Lääketieteen sektorilla voidaan simuloida veren virtausta verisuonissa. Rakennusteollisuudessa voidaan simuloida huoneiston ilmastointiin liittyviä seikkoja. Elektroniikkateollisuudessa voidaan tutkia piirilevyjen lämmönjohtumista. Sovelluskohteita on nykyään tuhansia. (ANSYS CFX Introduction 2006, 2–3)

Nykyisin tietokoneilla toteutetulla simuloinnilla voidaan säästää miljoonia euroja kehityskustannuksissa, kun jokaista prototyypin osaa ja mallia ei tarvitse erikseen testata käytännössä. Esimerkiksi aerodynamiikkaan liittyvässä simuloinnissa säästetään tuntuvasti, koska aidon kokoisia, tai edes pienemmän mittakaavan, tuulitunnelimalleja tarkasteltavista kohteista ei tarvitse valmistaa.

2.4 ANSYS CFX:ssä käytetty teorianmalli

ANSYS CFX ratkaisee niin sanottuja kolmiulotteisia epävakaita Navier-Stokesin yhtälöitä, jotka koostuvat massan säilyvyyden jatkuvuusyhtälöstä (1), kolmesta liikkeen säilyvyysyhtälöstä (2, 3, 4) ja energian säilyvyysyhtälöstä (5).

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \quad (1)$$

$$\begin{aligned} \text{X:} \quad & \frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \frac{\partial(\rho uw)}{\partial z} = -\frac{\partial p}{\partial x} + \frac{1}{\text{Re}} \left(\frac{\partial \tau_{xx}}{\partial x} + \right. \\ & \left. \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{xz}}{\partial z} \right) \quad (2) \end{aligned}$$

$$\begin{aligned} \text{Y:} \quad & \frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho v^2)}{\partial y} + \frac{\partial(\rho vw)}{\partial z} = -\frac{\partial p}{\partial y} + \frac{1}{\text{Re}} \left(\frac{\partial \tau_{xy}}{\partial x} + \right. \\ & \left. \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{yz}}{\partial z} \right) \quad (3) \end{aligned}$$

$$Z: \quad \frac{\partial(\rho w)}{\partial t} + \frac{\partial(\rho u w)}{\partial x} + \frac{\partial(\rho v w)}{\partial y} + \frac{\partial(\rho w^2)}{\partial z} = -\frac{\partial p}{\partial z} + \frac{1}{Re} \left(\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} \right) \quad (4)$$

$$\begin{aligned} \frac{\partial(E_t)}{\partial t} + \frac{\partial(uE_t)}{\partial x} + \frac{\partial(vE_t)}{\partial y} + \frac{\partial(wE_t)}{\partial z} &= -\frac{\partial(up)}{\partial x} - \frac{\partial(vp)}{\partial y} - \frac{\partial(wp)}{\partial z} - \\ &\frac{1}{RePr} \left(\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} \right) + \frac{1}{Re} \left[\frac{\partial}{\partial x} (u\tau_{xx} + v\tau_{xy} + w\tau_{xz}) + \right. \\ &\left. \frac{\partial}{\partial y} (u\tau_{xy} + v\tau_{yy} + w\tau_{yz}) + \frac{\partial}{\partial z} (u\tau_{xz} + v\tau_{yz} + w\tau_{zz}) \right] \end{aligned} \quad (5)$$

Kaavoissa ρ on tiheys, t on aika, p on paine, τ on jännitystensorin komponentti, E_t on kokonaisenergia, q on lämpövuoto, Re on Reynoldsin luku (6) ja Pr on Prandtin luku (7). x , y ja z ovat koordinaatit ja u , v ja w ovat vastaavat nopeuskomponentit. (NASA, Navier-Stokes Equations)

Reynoldsin luku ilmaisee virtausosaseen vaikuttavien inertiaivoimien ja viskositeettivoimien suhdetta, joka voidaan laskea kaavalla

$$Re = \frac{w \cdot L}{\nu} \quad (6)$$

jossa w on nopeus, ν on kinemaattinen viskositeetti ja L on virtausta luonnehtiva pituus.

Reynoldsin luvun avulla voidaan päätellä, onko virtaus luonteeltaan laminaarista vai turbulentista. Esimerkiksi pyöreissä putkissa virtaus on laminaarista, eli virtaviivaista, jos Reynoldsin luku on pienempi kuin 2320. Reynoldsin luvun ollessa yli 4000 pyöreissä putkissa esiintyy turbulentista eli pyörteistä virtausta. (Bohl 1971, 85–86)

Prandtlin luku ilmaisee viskositeettivoimien ja termisen diffuusion suhdetta, joka voidaan laskea kaavalla

$$Pr = \frac{\mu c_p}{\lambda} \quad (7)$$

jossa μ on dynaaminen viskositeetti, c_p on ominaislämpökapasiteetti ja λ on lämmönjohtavuus.

Prandtlin luku vaihtelee käytettävän fluidin mukaan noin 0,004 ja 100 000 välillä. Esimerkiksi nestemäisillä metalleilla Prandtlin luku on alle 0,01, kaasuilla noin 0,7–1,0 ja öljyillä 50–100 000. (TKK, Rakennusfysiikan luentomateriaali 2008, 5–6)

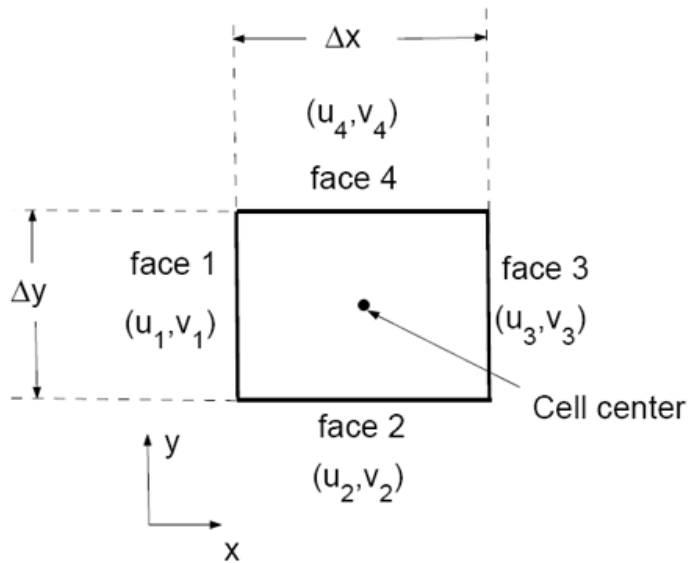
2.4.1 Tilavuusmenetelmä

ANSYS CFX:n käyttämä virtauslaskennan menetelmä on nimeltään Finite Volume Method eli tilavuusmenetelmä. Tässä ratkaisumallissa tutkittava alue jaetaan alivyöhykkeisiin, joita kutsutaan kontrollitilavuuksiksi. Säilyvyysyhtälöt sijoitetaan integraalimuodossa solun määrittelemään kontrollitilavuuteen. 3D-muodossa nämä solut ovat muodoltaan pääasiassa heksaedrejä, tetraedrejä tai prismoja. Yhtälöt diskretisoidaan soluille ja ratkaistaan iteratiivisesti jokaiselle kontrollitilavuudelle.

Tasaisen kokoonpuristumattoman virtauksen massan säilyvyyden jatkuvuusyhtälön integraalimuoto on

$$\int_S \vec{V} \cdot \hat{n} dS = 0 \quad (8)$$

Integrointi tapahtuu kontrollitilavuuden pinnan S yli ja \hat{n} on pinnan ulospäin suuntautuva normaali. Tämä tarkoittaa käytännössä sitä, että tilavuusvirta kontrollitilavuuden sisään on nolla. Kuviossa 2 on kuvattuna suorakulmainen solu.



Kuvio 2: Suorakulmainen kaksiulotteinen solu (Bhaskaran & Collins, 5)

Kuvion 2 mielivaltaisella pinnalla (face) i oleva nopeus on

$$\vec{V}_i = u_i \hat{i} + v_i \hat{j} \quad (9)$$

Sijoittamalla massan säilyvyyslaki (8) solun määrittelemään kontrollitilavuuteen, saadaan

$$-u_1 \Delta y - v_2 \Delta x + u_3 \Delta y + v_4 \Delta x = 0 \quad (10)$$

Tämä on solun diskretisoitu jatkuvuusyhtälö. Se vastaa kontrollitilavuuteen tulevan massavirran summaa, joka on yhtä kuin nolla, joten solun massa säilyy. Pintojen yksittäiset arvot (esim. u_2, v_3, \dots) saadaan selville interpoloimalla vierekkäisten solujen solmupisteiden (cell center) arvoista.

Samaan tapaan voidaan johtaa diskretisoidut yhtälöt liikkeen ja energian säilyvyyslaeille. Laskennassa ohjelmiston koodi pyrkii siis sellaiseen ratkaisuun, jossa jokaisen solun vaikuttavat suureet säilyvät. Tuloksena saadaan likiarvo jokaisesta alueen solmupisteen muuttujasta, joiden avulla voidaan tarkastella kokonaisvaltaisesti virtauksen luonnetta. (Bhaskaran & Collins, 5–6)

2.4.2 k -epsilon -turbulenssimalli

Projektia varten mallinnetut imusarjat olivat geometrialtaan melko vaihtelevia, joten voitiin hyvällä syyllä olettaa, että virtausnopeudet ja virtausominaisuuksien muutokset huomioon ottaen imusarjassa syntyy turbulenttisia virtauksia. Tämän vuoksi virtauslaskennat suoritettiin käyttäen turbulenssin mallintamista.

Projektissa käytettiin turbulenssin mallintamiseen k -epsilon -mallia. Sitä pidetään virtauslaskentateollisuuden piirissä perusmallina, koska se on vakaa ja sen tulosten ennustava tarkkuus on hyvä. Yleissimulaatioihin se tarjoaa hyvän kompromissin tarkkuuden ja laskennan monimutkaisuuden osalta. (ANSYS CFX-Solver Modeling Guide 2006, 99)

k on turbulenssin kineettinen energia, joka määritellään nopeudenheilahtelujen vaihteluina. Sen yksikkö voi olla esimerkiksi muotoa m^2 / s^2 . ϵ on turbulenssin pyörrehäviö eli nopeus jolla nopeudenheilahtelut häviävät. ϵ :n yksikkö voi olla esimerkiksi m^2 / s^3 .

k - ϵ -mallin jatkuvuusyhtälö on

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho U) = 0 \quad (11)$$

ja liikeyhtälöksi saadaan

$$\frac{\partial \rho U}{\partial t} + \nabla \cdot (\rho U \otimes U) - \nabla \cdot (\mu_{eff} \nabla U) = -\nabla p' + \nabla \cdot (\mu_{eff} \nabla U)^T + B \quad (12)$$

jossa B on tilavuusvoimien summa, μ_{eff} on turbulenssin tehollinen viskositeetti ja p' on modifioitu paine jonka määrittelee

$$p' = p + \frac{2}{3} \rho k + \frac{2}{3} \mu_t \bar{\nabla} \cdot U \quad (13)$$

k - ε -malli perustuu pyörteen viskositeettimalliin, jossa pätee

$$\mu_{eff} = \mu + \mu_t \quad (14)$$

jossa μ_t on turbulenssin viskositeetti.

k - ε -mallissa oletetaan että turbulenssin viskositeetilla on yhteys turbulenssin kineettiseen energiaan ja häviämiseen

$$\mu_t = C_\mu \rho \frac{k^2}{\varepsilon} \quad (15)$$

jossa C_μ on vakio.

k :n ja ε :n arvot tulevat suoraan turbulenssin kineettisen energian ja turbulenssin häviämisen differentiaalin siirtymäyhtälöistä

$$\frac{\partial(\rho k)}{\partial t} + \nabla \cdot (\rho U k) = \nabla \cdot \left[\left(\mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right] + P_k - \rho \varepsilon \quad (16)$$

$$\frac{\partial(\rho \varepsilon)}{\partial t} + \nabla \cdot (\rho U \varepsilon) = \nabla \cdot \left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \nabla \varepsilon \right] + \frac{\varepsilon}{k} (C_{\varepsilon 1} P_k - C_{\varepsilon 2} \rho \varepsilon) \quad (17)$$

joissa $C_{\varepsilon 1}$, $C_{\varepsilon 2}$, σ_k ja σ_ε ovat vakioita. P_k on viskoosi- ja nostovoimista johtuva turbulenssin kehitys, joka mallinnetaan kaavalla

$$P_k = \mu_t \nabla U \cdot (\nabla U + \nabla U^T) - \frac{2}{3} \nabla \cdot U (3\mu_t \nabla \cdot U + \rho k) + P_{kb} \quad (18)$$

Puristumattoman virtauksen tapauksissa $\nabla \cdot U$ on pieni, ja toinen termi oikealla ei juurikaan vaikuta kehitykseen. Puristuvan virtauksen tapauksissa $\nabla \cdot U$ on suuri vain

alueilla, joilla on suuria nopeuseroja, kuten iskuissa. (ANSYS CFX-Solver Theory Guide 2006, 75–76)

2.5 Virtauslaskenta prosessina

Tyypilliseen virtauslaskentaprosessiin sisältyy neljä vaihetta:

1. geometrian ja elementtiverkon luominen
2. laskenta- ja reunaehtojen asettaminen
3. virtauslaskentaongelman ratkaiseminen
4. lopputulosten visualisointi ja analysointi.

Lopputulosten analysoinnin pohjalta voidaan myöhemmin tehdä perusteltuja muutoksia alkuperäiseen geometriaan ja elementtiverkkoon ja viedä prosessi uudelleen läpi parempien tulosten toivossa. Vaativimpia vaiheista sekä ammatillisesti että ajallisesti ovat, lähinnä käytettävissä olevasta laskentatehosta riippuen, geometrian ja elementtiverkon luominen ja lopputulosten käsittely. (ANSYS CFX Introduction 2006, 4)

3 Työssä käytetyt ohjelmistot

3.1 CATIA

Imusarjan 3D-geometrian luomiseen ja muokkaamiseen käytettiin CATIAN versiota V5R18. Ohjelmistoa käytettiin opinnäytetyön tekijän opiskeluaikana CAE-kurssilla, joten se oli ensisijainen vaihtoehto 3D-mallinnusohjelmaksi. ANSYSin virtauslaskentaohjelmiston käyttö ei rajoittanut 3D-mallinnusohjelman valintaa, sillä se kykenee käsittelemään kaikkien yleisimpien 3D-mallinnusohjelmien tuottamia tiedostomuotoja.

3.2 ANSYS Workbench ja CFX

Virtauslaskentaan käytettiin ANSYS-ohjelmiston versiota 11.0. Ohjelmisto saatiin ilmaiseksi akateemiseen käyttöön hakemalla tarvittavat käyttöluvat Tieteen tietotekniikan keskukselta CSC:ltä. CSC:ltä saatiin myös laskenta-aikaa supertietokoneille vaativimpia laskentoja varten.

ANSYS-ohjelmisto asennettiin TAMK:n autolaboratorion tietokoneelle, johon oli asetettu kiinteä IP-osoite ohjelmistolisenssin toiminnan takaamiseksi. Ohjelmiston asentamisessa oli apuna Esko Järvinen CSC:ltä. Hän antoi ensimmäiset ohjeet ohjelmiston käyttöön ja tarjosi myös esimerkkitehtäviä tutustumisen avuksi.

Myöhemmin apua ohjelmiston käyttöön haettiin Process Flowlta, joka toimii ANSYS-ohjelmiston maahantuojana.

3.3 SSH Secure Shell Client

SSH clienttia käytettiin suojattuun tiedonsiirtoon CSC:n Murska-supertietokoneklusterin ja koululla sijaitsevan tietokoneen välillä. Clientilla suoritettiin supertietokoneen etäkomentaminen käyttämällä linux-pohjaisia komentoja. Ohjelmalla onnistui myös tiedostojen lähettäminen molempiin suuntiin. Supertietokoneelle lähetettiin laskentatiedot sisältävät tiedostot ja koulun tietokoneelle imuroitiin supertietokoneelta valmiit tulostiedostot.

4 3D-geometrian luominen virtauslaskentaa varten

4.1 Reunaehdot 3D-geometrialle

Imusarjan 3D-geometrialle ei ollut tiukkoja erityisehtoja työn tilaajan puolesta. Työhön riitti se, että imusarja on nelisylinteriseen työkonemoottoriin ja ulkomuodoltaan samantapainen kuin henkilöautoissa käytetty, eli koostuu paineentasauskammioista ja tasapituisista imuputkista. Imuputkien keskilinjojen väliseksi etäisyydeksi asetettiin 132 mm.

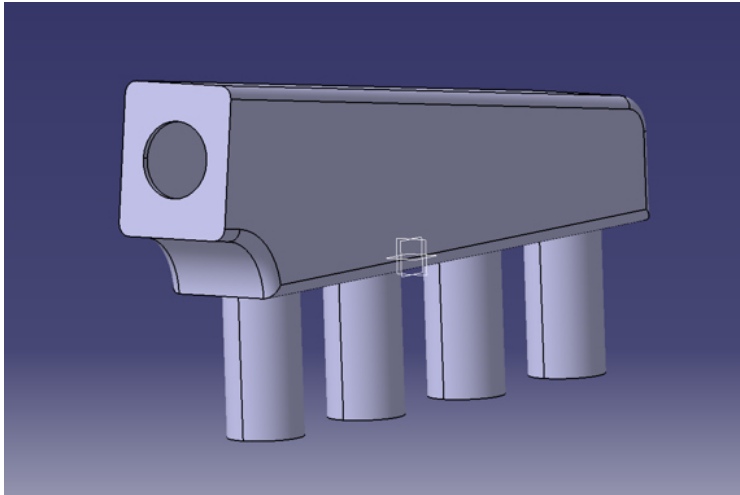
4.2 CATIAlla luodut 3D-geometriat

Lopullista mallisarjaa lukuun ottamatta kaikki imusarjamallit tehtiin Part Design –moduulissa, ja niistä tehtiin piirustukset työn tilaajan tarkasteltaviksi. Imusarjasta luotiin ontto malli, jonka imuputkien seinämäpaksuus oli 2 mm ja paineentasauskammion 5 mm. Jokainen mallisarja sisälsi useampia 3D-geometrioita, joita muokattiin aina virtauslaskennan tulosten analysoinnin seurauksena.

ANSYSiin siirtämistä varten CATIAlla luodut geometriat tallennettiin IGES-muodossa. Tämä tiedostomuoto osoittautui parhaaksi, sillä sen siirtämisessä ei havaittu mitään ongelmia. CATIAN oman CATPart-muodon kanssa siirtäminen ei onnistunut ongelmitta, vaikka se ANSYSin mukaan käykin siirrettäväksi tiedostomuodoksi.

Ensimmäinen mallisarja

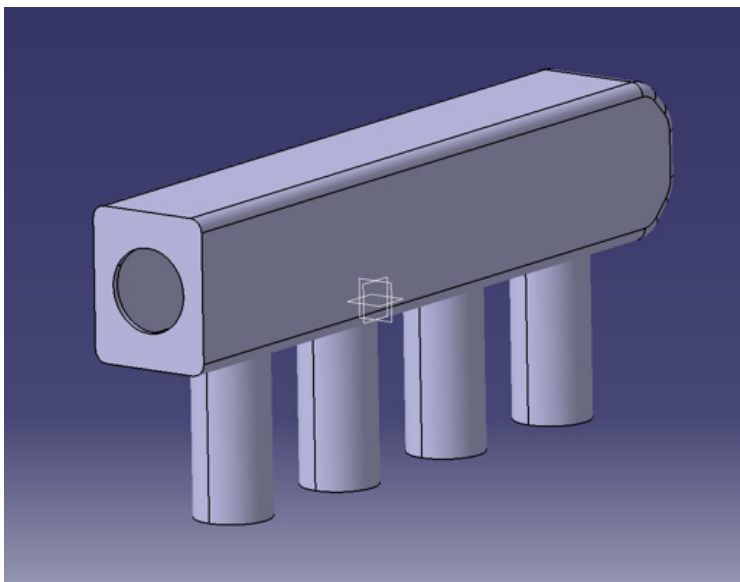
CATIAlla luotiin monia erimallisia imusarjoja projektia varten. Ensimmäisessä mallistossa (kuvio 3) oli halkaisijaltaan 60 mm sisääntuloaukko imusarjan päädyssä alaviistoon suunnattuna, paineentasauskammio oli hieman kalteva ja valtaosa kulmista oli pyöristetty. Imuputket ulottuivat 25 mm paineentasauskammion sisäpuolelle ja olivat pituudeltaan 140 mm. Imuputkien ulkohalkaisija oli 60 mm ja sisähalkaisija 55 mm. Ensimmäinen malli näytti päällisin puolin onnistuneelta, mutta toimivuudeltaan se oli kaukana optimaalisesta. Ensimmäisen virtauslaskennan tulosten perusteella mallia lähdettiin parantelemaan.



Kuvio 3: Ensimmäinen versio imusarjan 3D-geometriasta

Toinen mallisarja

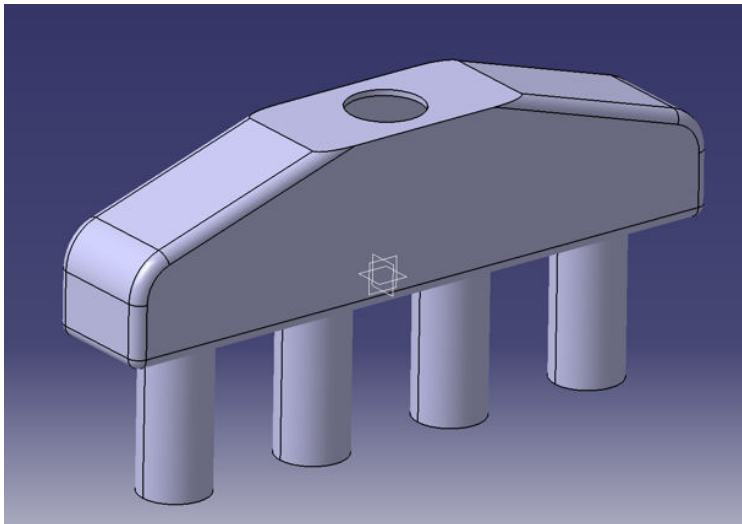
Toisessa imusarjamallistossa (kuvio 4) paineentasauskammio muutettiin tasaiseksi, sisääntuloaukko käännettiin kohtisuoraan imuputkia vastaan ja sen halkaisijaa suurennettiin 65 mm:iin. Paineentasauskammiota korotettiin ja sen pääty pyöristettiin. Imuputkia pidennettiin 145 mm:iin. Paineentasauskammion muotoa muuttamalla pyrittiin saamaan tasaisempi virtausjakauma.



Kuvio 4: Toinen versio imusarjan 3D-geometriasta

Kolmas mallisarja

Kolmannessa imusarjamallistossa (kuvio 5) sisääntuloaukko siirrettiin samansuuntaiseksi imuputkien kanssa ja imuputket eivät enää ulottuneet paineentasauskammion sisäpuolelle. Tultiin siihen johtopäätökseen, että sisääntuloaukon ollessa imuputkien yläpuolella virtauksen jakaumaa on helpompi säätää. Paineentasauskammiota korotettiin ja viistettiin sisääntuloaukon puolelta. Sisääntuloaukon ja imuputkien halkaisijat säilytettiin samoina kuin edellisessä mallissa mallienvälisen vertailun mahdollistamiseksi.



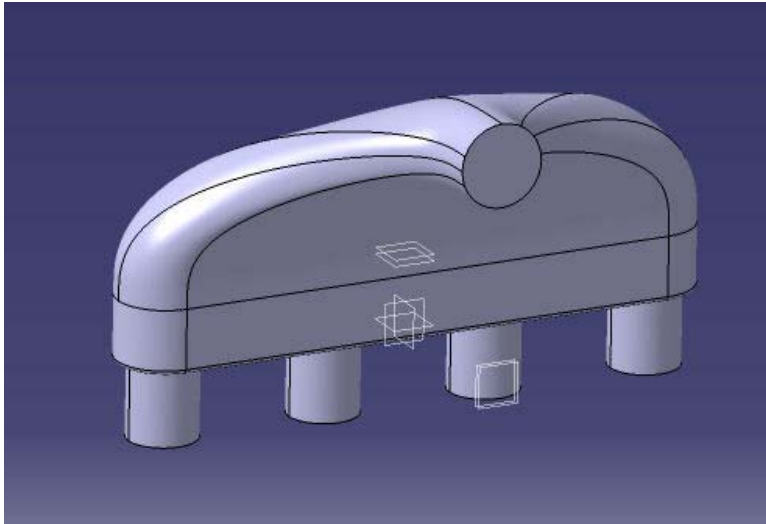
Kuvio 5: Kolmas versio imusarjan 3D-geometriasta

Lopullinen mallisarja

Viimeisimmissä malleissa 3D-geometrian luonnissa päätettiin keskittyä vain ilmatilan mallintamiseen. Imusarjahan ei ole menossa tuotantoon, joten tuotannolliset yksityiskohdat voitiin jättää vähemmälle huomiolle.

3D-geometriaa lähdettiin luomaan pintamallinnuksen kautta Generative Shape Design -moduulissa. Sisääntuloaukko käännettiin 90° kulmaan imuputkiin nähden ja sisääntuloaukon ja paineentasauskammion välinen mutka tehtiin välttämättä teräviä kulmia. Imusarjan ilmatilasta luotiin siksi kurvikas kuorimalli, joka sitten muutettiin kiinteäksi kappaleeksi Part Design -moduulissa (kuvio 6). Sisääntuloaukon ja imuputkien halkaisijat säilytettiin vakiona.

ANSYSiin siirrettäessä pelkän ilmatilan mallintamisella säästettiin monta työvaihetta geometrian määrittämisvaiheessa, koska imusarjaa ei tarvinnut erikseen täyttää ilmalla. Viimeisten mallien toimivuutta tutkittiin muuttamalla sisääntuloputken ja imuputkien pituutta.



Kuvio 6: Lopullinen versio imusarjan 3D-geometriasta

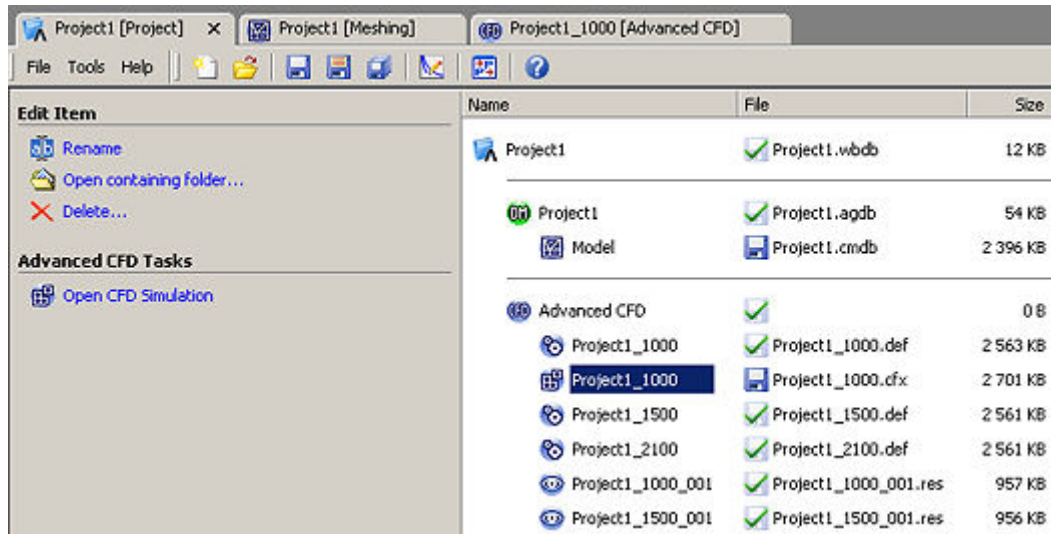
5 Virtauslaskentaprosessin kuvaus

5.1 ANSYS Workbenchin käyttö

3D-geometrian siirtäminen ANSYS-ohjelmistoympäristöön ja sen virtauslaskenta on monivaiheinen tapahtumaketju, joka alkaa geometrian muokkaamisesta johtaen verkottamisen kautta reunaehtojen asettamiseen, laskennan suorittamiseen ja tulosten jälkikäsittelyyn. Suurin osa tästä ketjusta tapahtuu ANSYS Workbench -ohjelmiston sisällä.

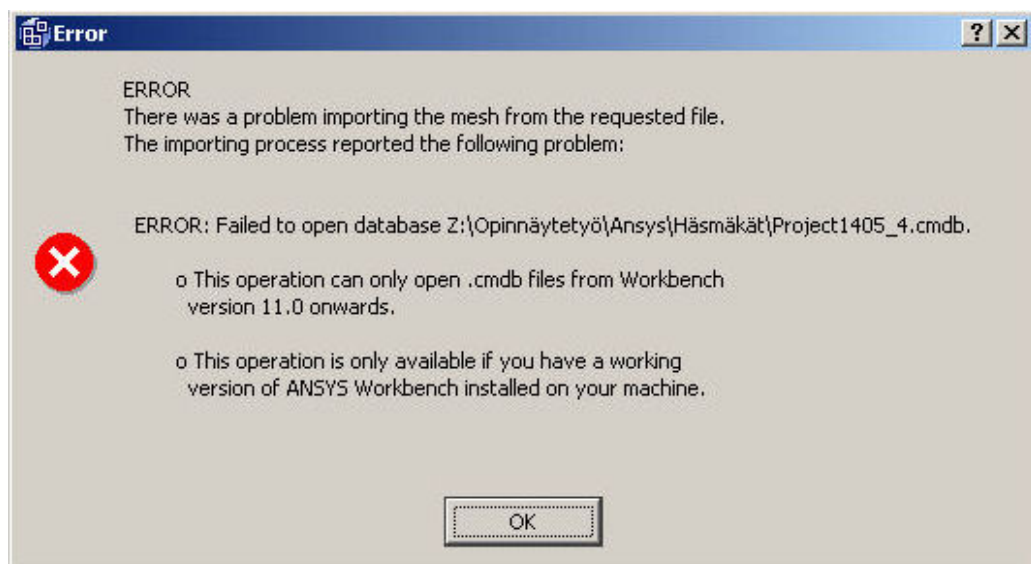
ANSYS Workbench -ohjelmistoympäristö on alusta, jolle rakentuvat kaikki Ansysin simuloititoinnot eli moduulit. Uuden projektin aloittamiselle, geometrian mallintamiselle, simuloimiselle, verkottamiselle, elementtimenetelmälle ja virtauslaskennalle on jokaiselle oma moduulinsa, joka valitaan ohjelman käynnistämisen yhteydessä. Ohjelmiston sisällä moduulien välillä liikutaan välilehtien avulla: jokainen moduuli aukeaa omalle välilehdelleen.

Aloitettaessa uutta projektia Workbench avasi [Project]-välilehden (kuvio 7). Tämän välilehden vasemmasta laidasta löytyvät käytettävissä olevat työkalut. Oikealla puolella näkyvät kaikki moduulien luomat tiedostot ja niiden koot, muokkausajankohdat, tyypit ja tallennustilat. Yläreunan työkaluriviltä löytyvät tavanomaiset kuvakkeet tiedostojen avaamiselle ja tallentamiselle.



Kuvio 7: Osa ANSYS Workbenchin [Project]-välilehdestä

Ensimmäinen toimenpide oli projektin tallentaminen paikalliselle kiintolevyllä. Tämä johtuu pääasiassa siirrettävien tallennusmedioiden rajallisesta luku- ja kirjoitusnopeudesta. Huomioonotettava seikka oli, että **tallennuspolku ei saa sisältää skandinaavisia erikoismerkkejä** (å, ä, ö...). Väärrien merkkien käyttäminen johti kuvion 8 mukaiseen harhaanjohtavaan virheilmoitukseen ja ohjelman toimimattomuuteen laskentavaiheessa.



Kuvio 8: Vääristä tallennuspolun merkeistä johtuva virheilmoitus

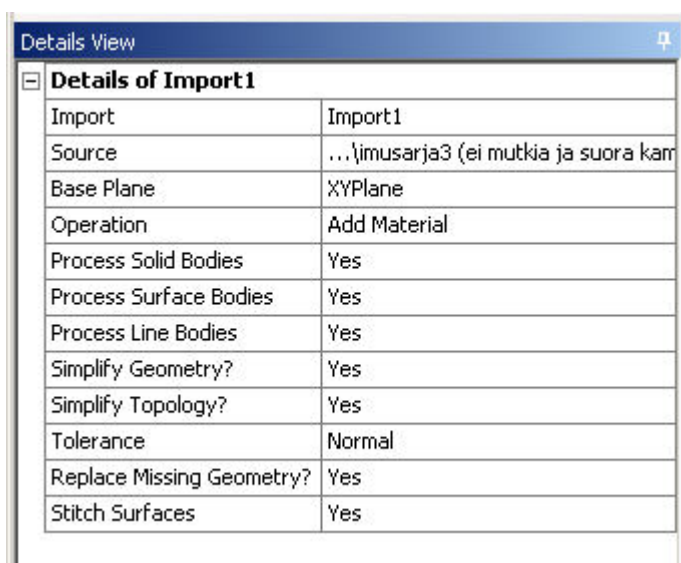
Virtauslaskennan ollessa kyseessä seuraava toimenpide oli määrittää geometria laskentaa varten käyttämällä New geometry -linkkiä sivun vasemmassa laidassa. Seuraavissa kappaleissa on käyty tarkemmin läpi laskentaprosessin vaiheita.

5.1.1 DesignModelerin käyttö

New geometry -linkin klikkaamisen jälkeen Workbench avasi [DesignModeler]-välilehden, jossa luodaan geometria laskentaa varten. Työtä varten tehty 3D-geometria oli luotu CATIAlla, joten se piti tuoda DesignModeleriin käyttäen File-valikon Import External Geometry File... -toimintoa.

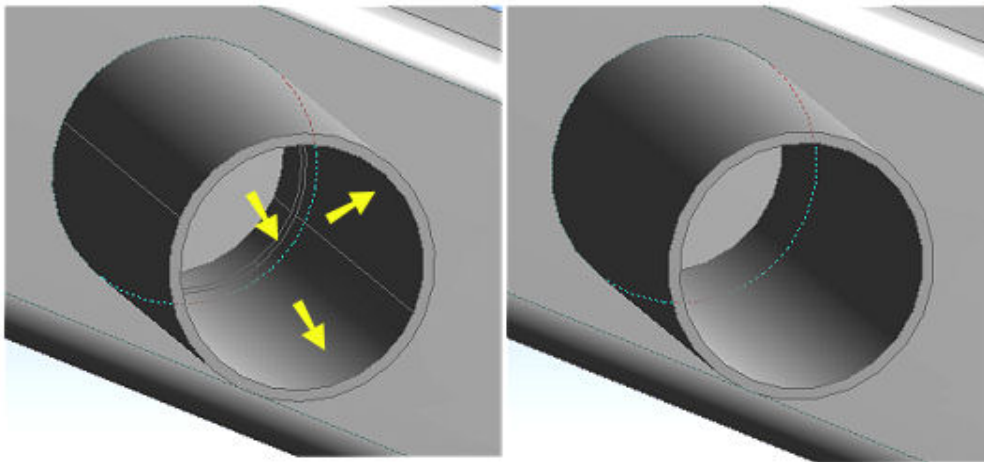
Myöhemmin tapahtuvaa verkottamista silmälläpitäen geometria pyrittiin muokkaamaan DesignModelerissa mahdollisimman yksinkertaiseksi käyttäen geometrian tuontivaiheessa alla olevan kuvion 9 mukaisia asetuksia. Geometria ja topologia yksinkertaistettiin, ja puuttuva geometria korvattiin. Näin esimerkiksi sylinterimäisen kappaleen pinta koostui useamman sijasta vain yhdestä pinnasta, kuten kuviosta 10 selviää.

Tuontiasetusten säätämisen jälkeen valinnat vahvistettiin klikkaamalla salamalla merkittyä Generate-kuvaketta. Tällöin imusarjan yksinkertaistettu geometria ilmestyi Graphics-ikkunaan.



Details View	
Details of Import1	
Import	Import1
Source	... \imusarja3 (ei mutkia ja suora kam
Base Plane	XYPlane
Operation	Add Material
Process Solid Bodies	Yes
Process Surface Bodies	Yes
Process Line Bodies	Yes
Simplify Geometry?	Yes
Simplify Topology?	Yes
Tolerance	Normal
Replace Missing Geometry?	Yes
Stitch Surfaces	Yes

Kuvio 9: Asetukset ulkoisen 3D-geometrian tuonnille



Kuvio 10: Sylinterimäisen kappaleen pinnat Simplify Geometry ja Simplify Topology asetukset pois käytöstä (vas.) ja käytössä (oik.)

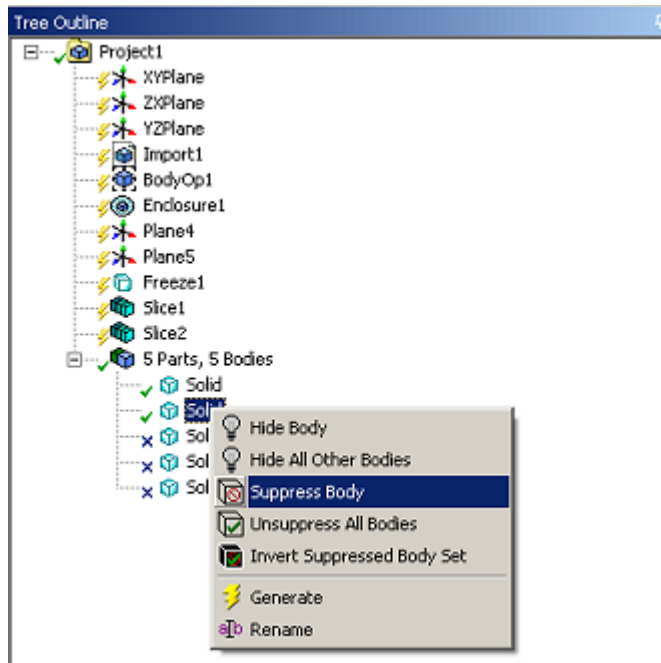
Onnistuneen geometriantuontioperaation jälkeen imusarja piti täyttää aineella, joka sitten verkotettaisiin virtauslaskentaa varten. Koska kyseessä oli imusarjan sisäisten ilmavirtausten mallintaminen, koko 3D-geometrian olisi voinut alunperin tehdä siten että olisi mallintanut vain imusarjan sisäpuolella olevan tilan. Tässä projektissa pelkän ilmatilan mallintamista kokeiltiin vain viimeisimmissä kuorimalleina toteutetuissa imusarjamalleissa.

Imusarja täytettiin aineella käyttäen Enclosure-toimintoa, joka muodostaa kappaleen ympärille tietyin ehdoin ”jäädetytyn” laatikon. Jäädettäminen mahdollistaa DesignModelerissa tiettyjen operaatioiden, kuten Slice ja Body Operation, käytön. Virtauslaskentaa varten tarvittiin vain imusarjan sisäpuolella oleva aine, joten kaikesta muusta piti päästä eroon. Tähän operaatioon käytettiin juuri Slice- ja Body Operation -toimintoja.

Imusarjan ylä- ja alapinnalle luotiin koordinaatistoon tasot käyttäen Plane-toimintoa. Ensin klikattiin kappaleen pintaa, jolle taso haluttiin ja sitten klikattiin Plane-kuvaketta. Tämän jälkeen ilmatilaa leikattiin Slice-toiminnolla luotuja koordinaatistotasoja käyttäen. Näillä toimenpiteillä ilmatila jakaantui imusarjan ulkopuoliseksi ja sisäpuoliseksi tilaksi.

Ilmatilan leikkelemisen jälkeen tarpeettomat tilavuudet poistettiin näkyvistä käyttäen Suppress Body -toimintoa. Kaikki imusarjan ulkopuoliset tilavuudet ja aluksi tuotu

imusarjan geometria poistettiin klikkaamalla hiiren oikealla painikkeella tilavuutta vastaavaa Solid-kappaletta Tree Outline näkymässä ja valitsemalla Suppress Body (kuvio 11).



Kuvio 11: Osien piilottaminen Suppress Body -toiminnon avulla.

5.1.2 Verkotuksen luominen

Oikeanlaisella kappaleen verkotuksella on ratkaisevan tärkeä vaikutus virtauslaskennan tulosten tarkkuuteen. Mitä harvempi verkko kappaleelle luodaan, sitä likimääräisempiä ja sitä kautta epätarkempia tulokset ovat. Tämän vuoksi on hyvin tärkeää, että verkko optimoidaan siten, että laskennan kannalta ratkaiseviin kohteisiin kappaleessa luodaan tiheämpi verkotus. Mikäli koko kappaleeseen luodaan tiheä verkko, saadaan tarkempia tuloksia koko kappaleen alueelta, mutta laskentamalli muuttuu todella raskaaksi käsitellä.

Imusarjan ilmatilan verkottamisprosessi aloitettiin tallentamalla ensin luotu geometria ja valitsemalla sitten [Project]-välilehden tiedostolistalta geometriatiedosto ja klikkaamalla New Mesh -linkkiä. Tällöin Workbenchiin avautui [Meshing]-välilehti ja Meshing Options -valikko. Valikosta klikattiin fysiikkamalliksi CFD ja verkotustavaksi CFX-Mesh. Tämän jälkeen valinnat vahvistettiin klikkaamalla OK, jolloin Workbench avasi [CFX-Mesh]-välilehden.

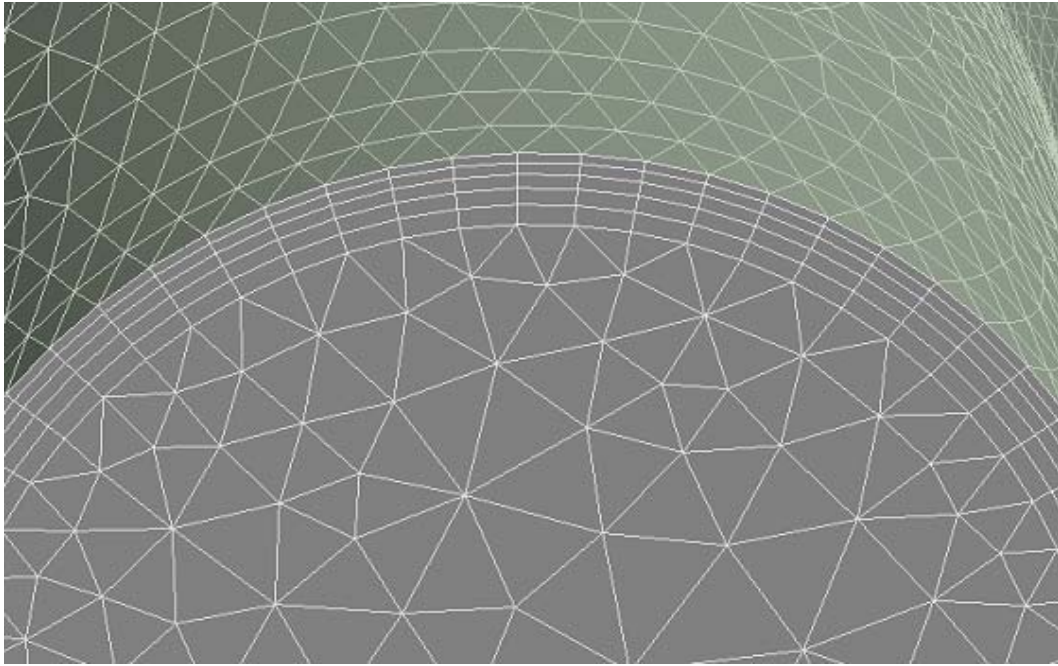
Ensimmäiseksi verkotusvaiheessa muokattiin Spacing-valikon arvoja Default Body Spacing- ja Default Face Spacing -kohdissa. Tässä projektissa käytetyt arvot on merkitty sulkeisiin. Default Body Spacingin Maximum Spacing -arvo (5 mm) määrittää millimetreinä suurimman mahdollisen elementtikoon kappaleen pinnassa ja tilavuudessa. Oletuskoko on noin 5 % kappaleen suurimmasta ulottuvuudesta, mutta tarkkaa laskentaa varten arvoa täytyy pienentää.

Default Face Spacingin kohta Angular Resolution (7 °) määrittää asteina, kuinka tarkasti kappaleen kaarevat osuudet verkotetaan. Mitä pienempi kulma, sitä tiheämpi verkko. Minimum ja Maximum Edge Length (1 mm, 5 mm) määrittävät elementin reunan ääriimitat.

Verkottaminen aloitettiin klikkaamalla Generate Surface Meshes- ja Generate Volume Mesh -nappuloita. Imusarjan verkottamiseen kului aikaa alle minuutti ja tuloksena syntyi yli puoli miljoonaa tetrahedraelementtiä. Verkotus sujui siis suhteellisen nopeasti. Mikäli verkotus muutetaan erittäin pieneksi ja elementtejä syntyy miljoonia, verkotukseen voi kulua useita tunteja. Tietokonekapasiteettia vaaditaan siis jo ennen varsinaista virtauslaskentaa.

Imuputkien ilmatila oli erityisen kiinnostavaa aluetta virtauslaskennan kannalta, sillä virtaukset imuputkissa määrittelivät pitkälti imusarjan toimivuuden. Tämän vuoksi verkotusta imuputkien sisäpuolella haluttiin tihentää käyttäen Inflated Boundary -toimintoa. Toiminto saatiin käyttöön klikkaamalla hiiren oikealla nappulalla Inflation > Insert > Inflated Boundary. Tämän jälkeen valittiin imuputket klikkaamalla yksitellen kutakin imuputken sylinteripintaa pitäen samalla Ctrl-näppäintä alaspainettuna.

Inflated Boundary -toiminto määritettiin käyttäen Maximum Thickness -arvoa 5 mm. Maximum Thickness tulee olla maksimissaan yhtäsuuri kuin Default Body Spacingin suurin arvo. Inflationin säädöt jätettiin vakioksi, koska tihennys vaikutti onnistuneelta, kuten kuvioista 12 näkyy. Valinnat vahvistettiin luomalla pinta- ja tilavuuselementit uudelleen.



Kuvio 12: Tihennetty elementtiverkko sylinteripinnan tuntumassa imuputken päädyistä katsottuna

Verkotuksen valmistuttua lopullinen verkko käytiin tallentamassa [Project]-välilehdellä ja tämän jälkeen siirryttiin ANSYSin virtauslaskentapuolelle CFX:ään.

5.2 Virtauslaskennan suorittaminen ANSYS CFX:llä

CFX on ANSYSin virtauslaskentamoduuli. Se koostuu CFX-Pre-, CFX-Solver- ja CFX-Post-osiosta. CFX-Pre:ssä annetaan virtauslaskennan reunaehdot, CFX-Solver on varsinainen laskentamoduuli ja CFX-Postilla suoritetaan tulosten graafinen jälkikäsittely.

5.2.1 Reunaehdot virtauslaskennalle

AGCO Sisu Power toimitti työtä varten reunaehdot Excel-taulukkona. Kaaviosta selvisivät tarvittavat suureet laskennan suorittamiseksi. Laskenta päätettiin suorittaa taulukon 1 kolmesta eri pisteestä: suurimman kierrosnopeuden (2100 1/min), suurimman vääntömomentin (1500 1/min) ja pienimmän kierrosnopeuden kohdalta (1000 1/min).

Jotta laskenta pysyisi riittävän yksinkertaisena, imusarjan virtauksien luomiseksi käytettiin reunaehtona massavirtaa sisääntuloaukon päässä. Massavirrat moottorin eri pyörimisnopeuksille on esitetty taulukossa 1.

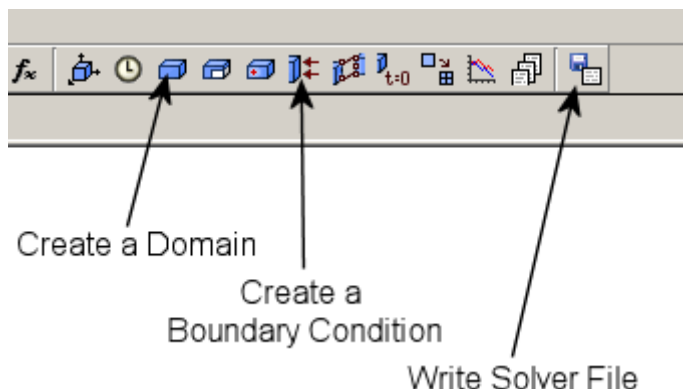
Taulukko 1: Virtauslaskennan reunaehdot (AGCO Sisu Power)

DynoSpeed _Avg rev/min	Dyno Torque_ Avg Nm	Brake Power_A vg kW	IntakeAir Temp_Avg °C	ChargeAir Temp2_Avg °C	ChargeAir Press2_Avg kPa	IntakeAir MassFlow_ Avg kg/s
2100	339	74,6	23,9	54,7	108,2	0,149
1900	392	78,1	23,4	51,4	110,8	0,136
1500	428	67,2	23,4	40,3	96,3	0,103
1000	414	43,4	23,8	27,6	46,0	0,049

5.2.2 CFX-Pren käyttö

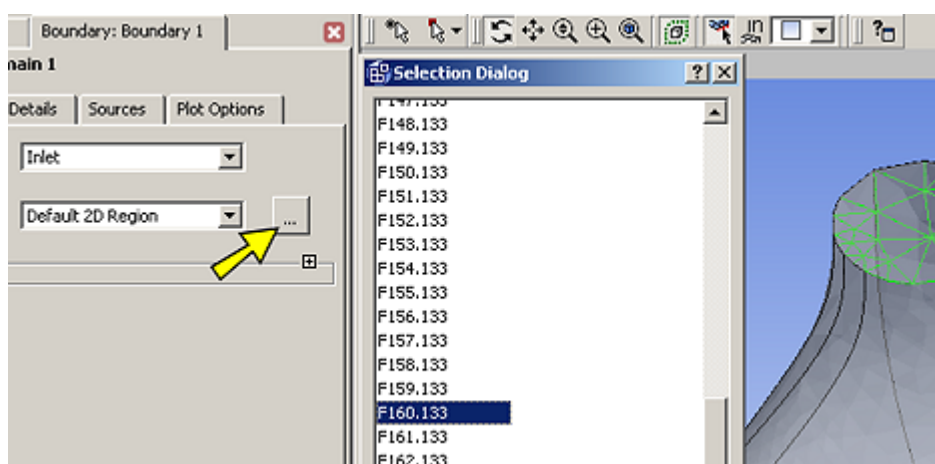
ANSYS CFX:n käyttö aloitettiin klikkaamalla ensin projektin verkotustiedostoa [Project]-välilehden tiedostolistalta ja sitten vasemmasta reunasta Create CFD Simulation with Mesh -linkkiä. Tämä aukaisi [Advanced CFD]-välilehden joka sisältää CFX:n moduulit eri alareunan välilehdillä. Oletuksena ensimmäisenä aukesi CFX-Pre, jossa luotiin virtauskenttä ja asetettiin reunaehdot laskennalle.

Ensimmäiseksi luotiin virtauskenttä (Domain) imusarjaan käyttäen kuvaruudun yläreunan työkalurivin Create a Domain -toimintoa (kuvio 13). Tämä aukaisi virtauskentän nimeämisen jälkeen ruudun vasempaan reunaan Virtauskentän yksityiskohdat. Näistä voitiin säätää esimerkiksi nesteen tyyppi, virtauskentän liikkuvuus, verkon muodonmuutos ja lämmönsiirrossa ja turbulenssissa käytettävä malli. Tässä projektissa virtauskentän yksityiskohtia ei tarvinnut muokata lainkaan. Ne piti vain hyväksyä painamalla alareunasta Apply- ja OK-painikkeita.



Kuvio 13: CFX-Pren työkalurivi

Seuraavaksi piti asettaa reunaehdot virtauslaskennalle. Tämä tapahtui käyttäen työkalurivin Create a Boundary Condition -painiketta (kuvio 13). Reunaehdot nimettiin siten, että sisääntuloaukko oli in ja imuputkien reiät yhdessä out. Molemmat reunaehdot luotiin aikaisemmin luotuun virtauskenttään. Nimeämisen jälkeen reunaehdon yksityiskohtat aukesivat ruudun vasempaan reunaan. Tästä päästiin valitsemaan In-reunaehdon tyyppiä Inlet. Reunaehdon sijainti täytyi etsiä käyttäen Location-rivillä sijaitsevaa painiketta, jossa oli kolme pistettä (kuvio 14). Tästä aukesi Selection Dialog, josta päästiin selaamaan listalta imusarjan pintoja. Oikea sijainti löytyi tarkkailemalla imusarjan 3D-mallia listan oikealla puolella.



Kuvio 14: Reunaehdon sijainnin määrittäminen Basic Settings –välilehdellä. Selection Dialogin avaava painike on osoitettu nuolella.

Reunaehdon yksityiskohtien Boundary Details -välilehdeltä määritettiin Mass and Momentum -kohdan asetukseen ehdoksi massavirta Mass Flow Rate -toiminnolla. Turbulenssin ehdoksi otettiin Medium (Intensity = 5 %), koska imusarjassa esiintyvän

turbulenssin voimakkuudesta ei ollut etukäteistietoa. Muita muutoksia yksityiskohtiin ei tehty. Valinnat hyväksyttiin jälleen Apply- ja Ok-painikkeilla. (ANSYS CFX-Pre User's Guide 2006, 58)

Out-reunaehto määritettiin kuten aikaisempikin, mutta tyyppiksi valittiin Opening, joka sallii virtauksen Outlet-tyypistä poiketen myös vastakkaiseen suuntaan.

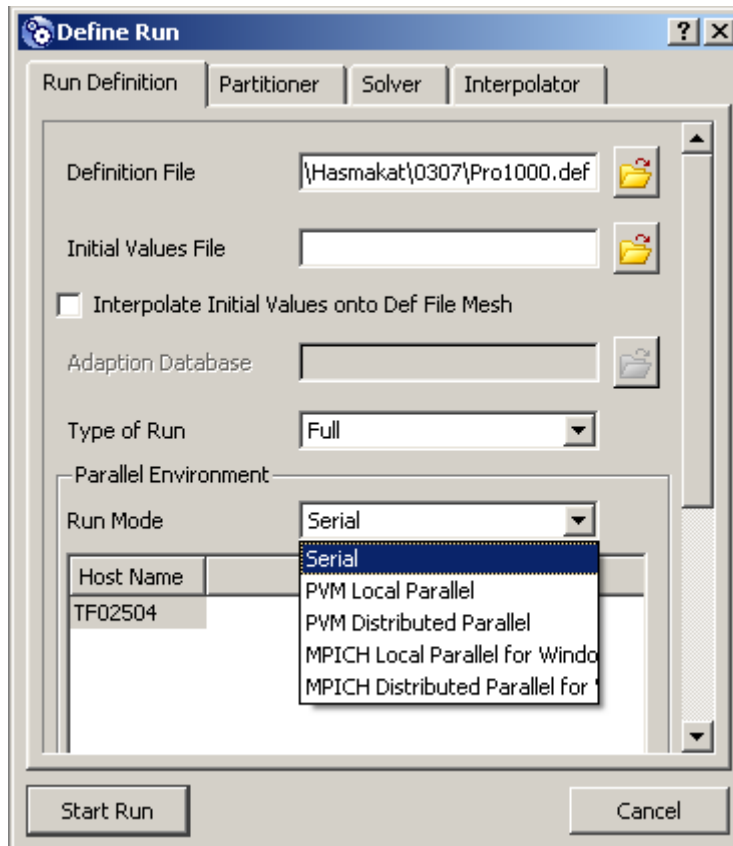
Virtauslaskennan aikana Solver varoittaa, jos Outlet-tyyppisessä reunaehdossa tapahtuu virtausta väärään suuntaan ja estää sen laittamalla seinää virtauksen tielle. Sijainti valittiin kuten sisääntuloaukollekin, mutta ulostuloon valittiin kaikki neljä imuputken päätä. Boundary Details -välilehdeltä valittiin Mass and Momentum -kohdan vaihtoehdoksi Static Pres. (Entrain). Siihen määritettiin paineeksi nolla.

Turbulenssimalliksi valittiin Zero Gradient, joka soveltuu täysin kehittyneen turbulenssin olosuhteisiin. (ANSYS CFX-Pre User's Guide 2006, 58)

Lopuksi laskennan reunaehdoista kirjoitettiin laskentaa varten .def-päätteinen määrittelytiedosto käyttäen Write Solver File -painiketta työkalurivillä (kuvio 13). Tiedostoa tallennettaessa voi valita tallennusrudun yläreunasta toimenpiteen, jonka CFX suorittaa tallennuksen yhteydessä. Vaihtoehtona on tallentaa pelkkä Solver-tiedosto, aloittaa laskenta CFX-solverissa tai käynnistää CFX-Post.

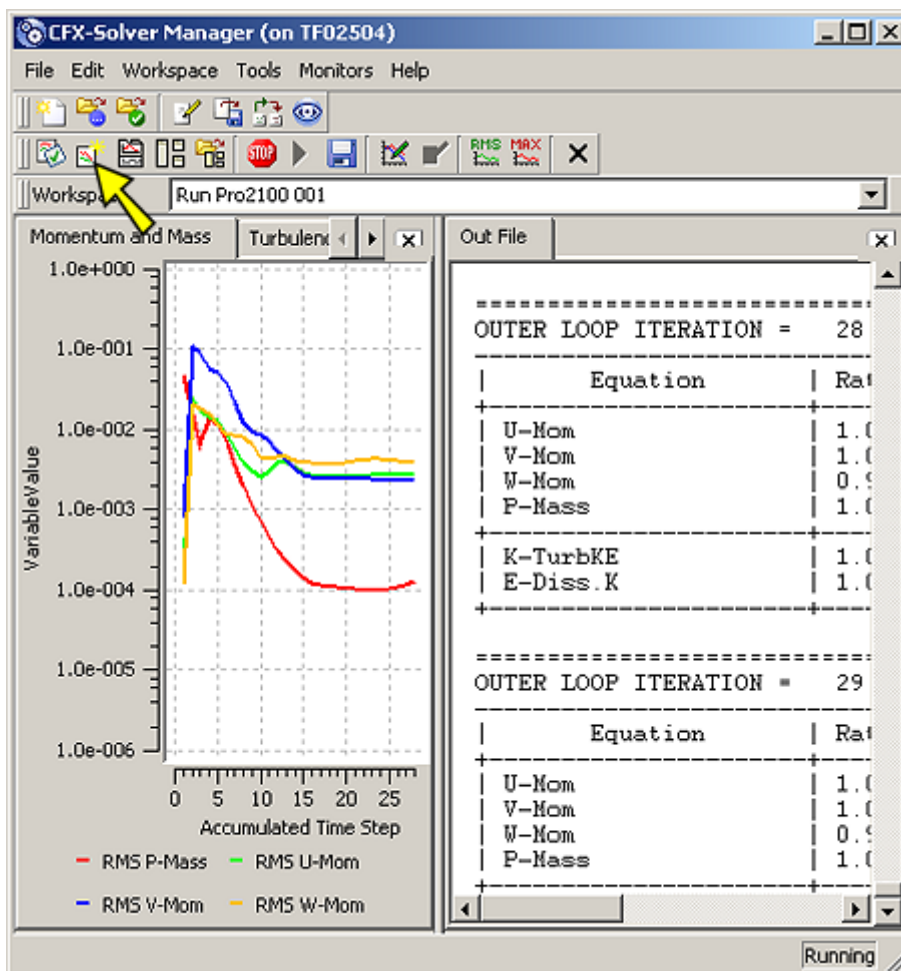
5.2.3 CFX-Solverin käyttö

CFX-Solver on melko automaattinen laskentamoduuli. Sen voi käynnistää joko laskennan määrittelytiedoston tallentamisen yhteydessä, tai tuplaklikkaamalla määrittelytiedostoa Windowsin puolella. Workbenchissä ajettuna Solver avautuu samalla välilehdellä kuin CFX-Pre. Määrittelytiedoston avaamisen jälkeen avautui Define Run -ikkuna, josta päästiin määrittämään laskenta-ajon asetukset (kuvio 15).



Kuvio 15: CFX-Solverin Define Run -ikkuna

Ajoasetukseksi valittiin Run Mode -valikosta Serial eli sarja-ajo. Laskenta-ajan kutistamiseksi laskenta olisi ollut mahdollista jakaa myös rinnakkaiseksi, eli useammalla koneella tapahtuvaksi, mutta tähän ei ollut mahdollisuutta. Laskenta käynnistettiin Start Run -painikkeella, jolloin CFX-Solver aukesi avaten oikealle teksti-ikkunan ja vasemmalle suppenemiskäyrät kuvion 16 mukaisesti.



Kuvio 16: CFX-Solverin ikkuna jossa oikealla teksti-ikkuna ja vasemmalla suppenemiskäyrät. New Monitor -painike on osoitettu nuolella.

Laskenta-ajon aikana on mahdollista seurata laskennan kannalta oleellisten suureiden muuttumista ja tarvittaessa muokata niitä. Tähän käytetään New Monitor -painiketta, joka on osoitettu kuviossa 16. Esimerkiksi painetta imusarjan sisääntuloaukolla voidaan monitoroida, jotta saadaan asetettua oikea ulostulopaine imuputkien päähän.

Suppenemiskäyrät osoittavat Solverin käyttämien yhtälöiden suppenemista. RMS-suppenemiskäyrät muodostetaan hakemalla jäännökset koko virtauskentän alueelta ja muodostamalla niistä neliöllinen keskiarvo. Laskennan monimutkaisuudesta ja alkuarvojen asettamisesta johtuen käyrät eivät välttämättä suppene aina loppuun asti.

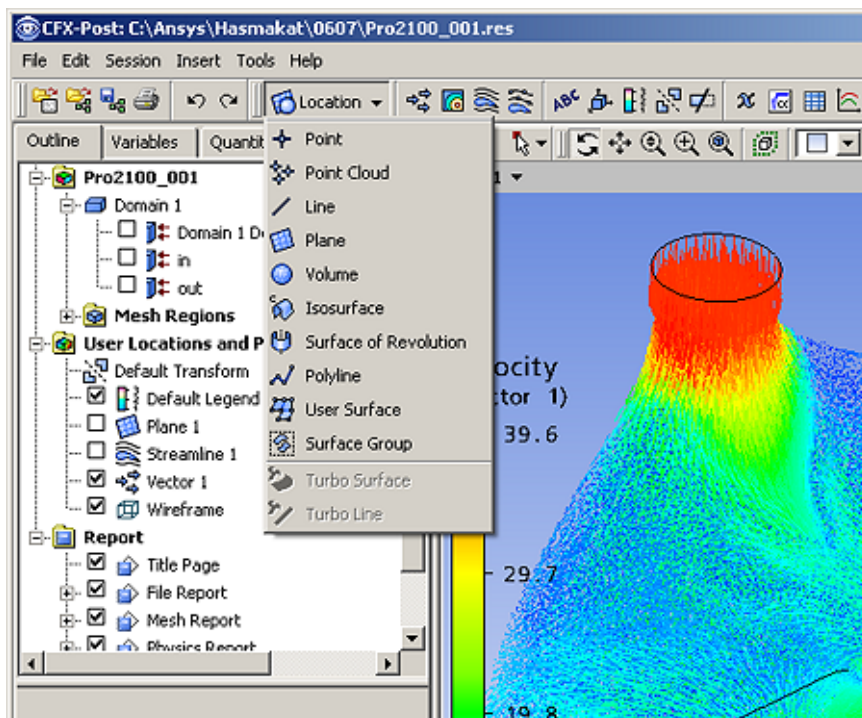
Suhteellisen suppenevaan tulokseen vaaditaan enintään $5,0E-4$ jäännöstaso. Jos tätä tasoa ei aliteta, voidaan tutkia kuinka hyvin käyrät tasoittuvat laskennan aikana. Hyvin tasoittuvat käyrät ovat merkki siitä, että laskennan tulos ei enää merkittävästi muutu.

Tämä voidaan varmentaa monitoroimalla haluttua suuretta laskennan aikana. (ANSYS CFX-Solver User's Guide 2006, 66)

Laskennan valmistuttua CFX-Solver tuottaa tuloksista .res- ja .out-päätteiset tulostiedostot, joissa on tiedostonimen lopussa lisäksi laskennan järjestysnumero. Out-tiedostosta selviää tilastotietoa ajosta ja sen yksityiskohdista. Res-tiedosto on varsinainen tulostiedosto, jota CFX-Post käsittelee. Laskennan suppenemiskäyriä ja teksti-ikkunaa voi tarkastella jo valmistuneista laskennoista jälkeempään avaamalla Solverilla .res-tulostiedoston. Tämä tapahtuu painamalla työkalurivin Monitor Finished Run –painiketta, tai painamalla näppäinyhdistelmää Ctrl-O. Kyseinen nappula näkyy myös kuvioista 16, kolmantena ylemmällä työkalurivillä.

5.2.4 CFX-Postin käyttö

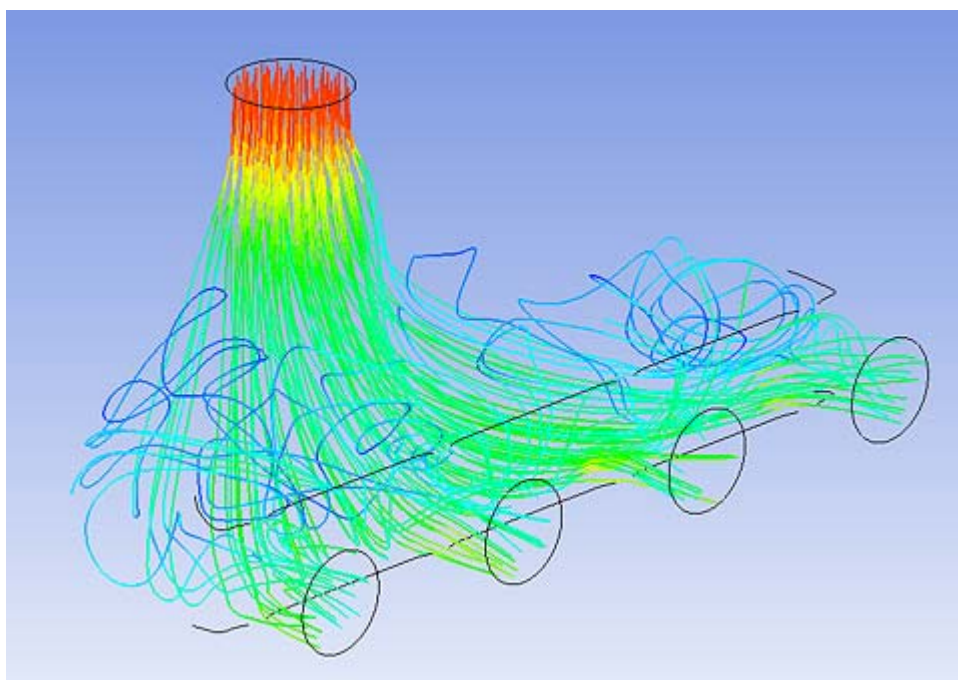
CFX-Post aukeaa Workbenchissä avattuna [Advanced CFD]-välilehdelle. Sen voi avata myös erikseen tuplaklikkaamalla Windowsissa .res-tulostiedostoa. CFX-Postin käyttöliittymässä työkalurivit ovat tutusti ruudun yläreunassa, graafinen ikkuna oikealla, ja rakennepuu vasemmalla (kuvio 17).



Kuvio 17: CFX-Postin käyttöliittymä Location-valikko avattuna

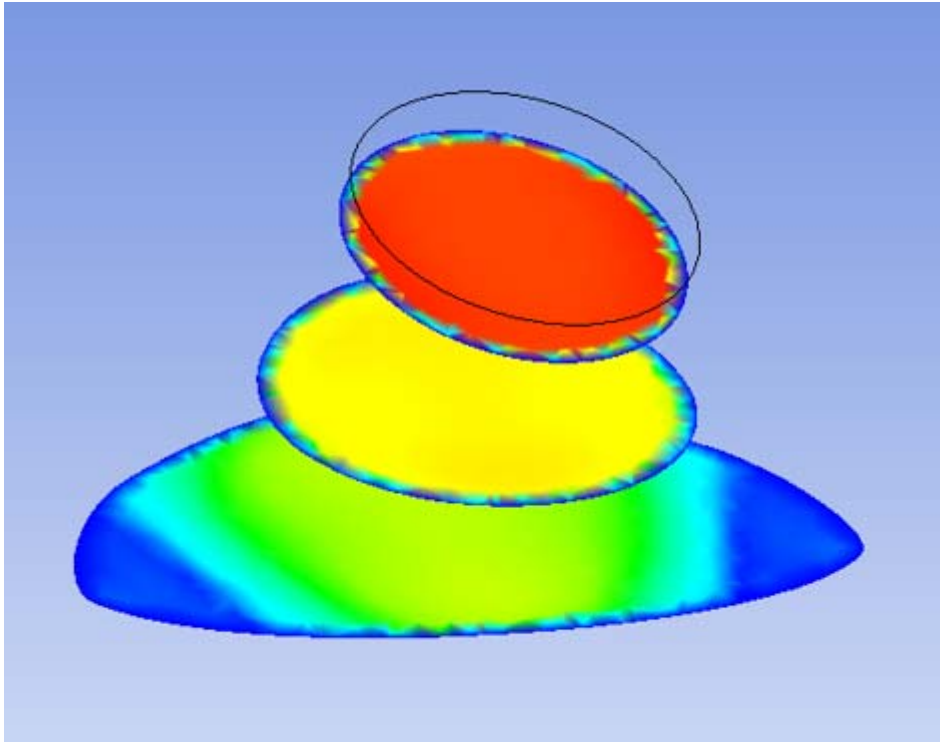
Tulosten analysointiin käytettiin pääasiassa työkaluriviltä löytyviä Vector- ja Streamline-toimintoja sekä kuviossa 17 näkyvästä Location-valikosta löytyvää Plane-toimintoa. Kaikkia näitä työkaluja käyttäessä tulee huolehtia siitä, että säätää työkalun yksityiskohdista virtauskentäksi eli Domainiksi sen kentän, jonka itse on luonut CFX-Pressä. Samoin työkalun käyttämä väri kannattaa muuttaa Variableksi, jotta pääsee valitsemaan haluamansa suureen tarkkailtavaksi. Työkalun yksityiskohdat ilmestyvät ruudun vasempaan alareunaan työkalun valitsemisen jälkeen.

Vector-toiminnolla luodaan virtauskenttään halutun suureen mukaisia vektoreita. Kuviossa 17 on käytetty virtauskentän nopeusvektoreita. Streamline-toiminnolla luodaan virtauskenttään haluttu määrä virtausviivoja, joista voi karkeasti tulkita virtauksen luonnetta. Kuviossa 18 on käytetty Streamline-toimintoa osoittamaan virtauskentän virtausnopeuksia sadalla virtausviivalla.



Kuvio 18: Esimerkki Streamline-toiminnolla luoduista virtausviivoista virtauskentässä

Plane-toiminnolla luodaan virtauskenttään taso, jonka voi säätää näyttämään haluttua suuretta. Tason voi asettaa mihin koordinaatiston suuntaan tahansa yksityiskohtia muokkaamalla. Esimerkki Plane-toiminnon käytöstä on esitetty kuviossa 19, jossa näkyy kolme imusarjan sisääntuloaukon virtausnopeustasoa.



Kuvio 19: Plane-toiminnolla luotuja virtausnopeustasoja sisääntuloaukolta

Haluttujen analysointityökalujen luomisen jälkeen asetukset kannattaa tallentaa Save State As -toiminnolla joka löytyy sekä File-valikosta että työkaluriviltä. Tämän jälkeen luotuja asetuksia voi käyttää samantapaisissa simulaatioissa lataamalla tallennetun tilan Load State -toiminnolla.

5.3 CSC:n supertietokoneen käyttäminen

CSC:n supertietokoneita pääsi käyttämään ottamalla yhteyden CSC:n serveriin suojatulla yhteydellä käyttäen SSH Clientiä. Ohjelmaan löytyi pikakuvake Windowsin työpöydältä nimellä SSH Secure Shell Client.

Ohjelman aukaisemisen jälkeen päästiin kirjautumisikkunaan painamalla Enter- tai Space-näppäintä. Kohtaan Host Name syötettiin CSC:n serverin osoite eli esimerkiksi Murska-serveriä käytettäessä murska.csc.fi. Kohtaan User Name syötettiin oma CSC:n extranetin käyttäjätunnus. Port Number jätettiin oletukseksi. Authentication Methodiksi valittiin Password. Tämän jälkeen painettiin Connect-painiketta ja syötettiin aukeavaan ikkunaan CSC:n extranetin salasana.

CSC:n serverit toimivat Linux-käyttöjärjestelmäympäristössä, joten kaikki käytettävät komennot olivat sen mukaisia. Salasanan syöttämisen jälkeen päädyttiin kotihakemistoon eli \$HOME-hakemistoon. Tähän hakemistoon päästiin käsiksi sekä CSC:n extranetistä että avaamalla SSH Clientin New File Transfer –ikkuna.

Supertietokoneella suoritettavaa laskentaa varten tarvittavat tiedostot piti siirtää ensin paikalliselta tietokoneelta CSC:n serverin kotihakemistoon ja sieltä työkansioon nimeltään \$WRKDIR. Nopeaa interaktiivista laskentaa varten ei tarvinnut siirtää muuta kuin CFX-Prellä luotu .def-päätteinen Solver File. Monimutkaisempaa sarja- tai rinnakkaisajoa varten tarvitsi siirtää myös niin sanottu jobfile eli työtiedosto, joka määritteli laskennassa tarvittavat parametrit.

Jobfilellä ei ole tiedostopäätettä ja se on luotu Linux-ympäristössä. Tästä seuraa että esimerkiksi Windowsin Notepad näyttää koko monirivisen tiedoston yksirivisenä.

Jobfile-tiedostoa ei pidä rivittää uudestaan, sillä Windowsin rivinvaihto ei vastaa Linuxin rivinvaihtoa. **Rivinvaihtoa osoittavat neliöt tulee siis jättää rauhaan** tiedostoa muokatessa.

5.3.1 Interaktiivinen laskenta

Tässä projektissa CSC:n laskentakapasiteettia ruvettiin käyttämään melko myöhäisessä vaiheessa, sillä Linux-komentojen opettelussa ja toimivan jobfilen luomisessa kesti yllättävän kauan. Tämän vuoksi turvauduttiin ensihädässä interaktiiviseen laskenta-ajoon.

Interaktiivinen ajo on tarkoitettu lähinnä pieniä testilaskentoja varten. Se on aikarajoitettu oletuksena yhteen tuntiin ja yhdelle ytimelle, joten kovin massiivisia laskentoja sillä ei ehdi suorittaa. Interaktiivisessa ajossa työ aloitetaan heti, ilman jonotusta, mikäli serverillä on ytimiä käytettävissä.

Tämän projektin imusarjamallien laskettamisessa kului koulun tietokoneella noin 45 minuuttia per laskenta. Aluksi CSC:n koneilla laskentoihin kului lähes sama aika, koska laskennan määrittelytaidoissa oli puutteita ja laskenta suoritettiin yhdellä ytimellä.

Interaktiivista ajoa varten .def-päätteinen tiedosto kopioitiin \$HOME-kansiosta \$WRKDIR-kansioon komennolla

```
cp tiedostonimi.def $WRKDIR
```

Tämän jälkeen serverille piti ladata CFX-moduuli komennolla

```
module load ansys/ansys_11.0_cfx
```

Laskenta aloitettiin siirtymällä ensin työhakemistoon komennolla

```
cd $WRKDIR
```

ja käynnistämällä sitten CFX-Solver komennolla

```
cfx5solve -def tiedostonimi.def
```

Interaktiivisen ajon aikana yhteyttä CSC:n serverille ei saanut katkaista, koska muuten laskenta katkesi. Laskennan päättyessä normaalisti, \$WRKDIR-hakemistoon ilmestyivät normaalit .res- ja .out-päätteiset tulostiedostot. Jos laskenta päättyi virheeseen, samaan hakemistoon ilmestyi kuitenkin .out-päätteinen tiedosto, jota pääsi tarkastelemaan komennolla

```
less tiedostonimi.out
```

Jotta tulostiedostoja pääsi tarkastelemaan koulun koneella, ne piti kopioida ensin kotihakemistoon komentoa

```
cp tiedostonimi.res $HOME
```

```
cp tiedostonimi.out $HOME
```

ja sen jälkeen imuroimalla ne SSH:n New File Transfer -ikkunassa omalle tietokoneelle.

Lopuksi \$WRKDIR-hakemistosta tyhjennettiin kaikki tiedostot, jotta CSC:n resursseja ei käytetty turhan datan säilyttämiseen. Tähän käytettiin komentoa

```
rm *
```

Komennon syöttämisen jälkeen kysyttiin vielä varmistus erikseen jokaisen tiedoston poistamiselle. Näihin vastattiin painamalla y-kirjainta ja Enteriä.

5.3.2 Ei-interaktiivinen laskenta

Ei-interaktiivisessa ajossa laskenta laitetaan serverille työjonoon ja työ aloitetaan kun resursseja on riittävästi käytettävissä. Työ on mahdollista ajaa sarja- tai rinnakkaisajona eli yhdellä tai useammalla ytimellä. Positiivista yhden ytimen käytössä on, että resursseja riittää lähes aina yhden ytimen käyttämiseksi, mutta laskennassa kestää yhtä kauan kuin paikallisella koneella laskiessa. Moniydinajossa säästetään ajassa moninkertaisesti, mutta serverin käyttöasteesta riippuen työn aloittamisessa saattaa kestää oma aikansa.

Ei-interaktiivisen ajon määrittelyssä käytettiin jo mainittua jobfilea, josta on esitetty esimerkki liitteessä 1. Jobfilen määrittely aloitettiin muokkaamalla esimerkkitiedoston tiedot työn tietoja vastaaviksi. Työn kestoksi (#BSUB -W) määritettiin varmuuden vuoksi kaksi tuntia. Työn valmistumisesta tai katkeamisesta ilmoittavan sähköpostin osoitteeksi vaihdettiin oma osoite, koska CSC:n työntekijälle on turha ilmoittaa oman työn valmistumisesta.

Jobfilen lopussa käytettiin lähes samoja komentoja kuin käynnistäessä interaktiivista ajoa. Eroina oli, että työkansiona käytettiin itse luotua kansiota työkansion sisällä ja ajon aloittavan cfx5solve-komennon perässä käytettiin hieman monimutkaisempaa määrittelyjonoa. Jos määrittelyjonon komentojen tarkoitusta ei ymmärtänyt, ne kannatti jättää rauhaan, koska oletusasetuksilla laskenta onnistui hyvin.

Työn valmistelua jatkettiin luomalla työhakemistoon oma hakemisto työtä varten komennolla

```
mkdir $WRKDIR/käyttäjänimi
```

Hakemiston nimenä käytettiin CSC:n extranetin käyttäjänimeä, koska se oli helppo muistaa. Toki mikä tahansa nimi olisi käynyt. Pääasia oli, että se vastasi työhakemistoa jobfilessä.

Tämän jälkeen .def-määrittelytiedosto kopioitiin juuri luotuun hakemistoon ja jobfile kopioitiin \$WRKDIR-hakemistoon.

Työ aloitettiin siirtymällä \$WRKDIR-hakemistoon ja kirjoittamalla komento

```
bsub < jobfile
```

Tällä komennolla työ siirtyi jonoon. Työn alkaessa itse luotuun kansioon ilmestyi työn nimellä ja järjestysnumerolla nimetty kansio, josta löytyi työn etenemisestä kertova .out-päätteinen tiedosto. Sitä pääsi lukemaan komennolla

```
less tiedostonimi.out
```

Laskennan päättyessä tulostiedostot ilmestyivät jobfilessä määritettyyn työkansioon.

6 Tulosten analysointi

6.1 Tulosten tarkkuuteen vaikuttavat tekijät

Virtauslaskennan tulosten tarkkuus riippuu hyvin monesta tekijästä. Suurin vaikutus syntyy oikeanlaisten laskentamenetelmien ja reunaehtojen määrittämisestä. Tarkkoja tuloksia ei takaa ainoastaan se, että laskennassa käytettävä verkotus ja virtauskenttä on riittävän tiheä. Haluttaessa mahdollisimman todenmukaisia tuloksia, kaikki virtaukseen vaikuttavat seikat on syötettävä malliin mahdollisimman tarkasti, käyttäen mahdollisimman tarkkaan määritettyjä reunaehtoja.

Myös virtauslaskennassa käytettävä 3D-geometria vaikuttaa tulosten tarkkuuteen. Esimerkiksi tässä projektissa olisi kannattanut mallintaa geometriaa imusarjasta jonkin matkaa venttiilikannen puolelle ja sisääntuloaukolta tuloilmaputken suuntaan. Tällöin laskennassa olisi huomioitu myös mahdolliset geometriamuutokset mallin ulkopuolella.

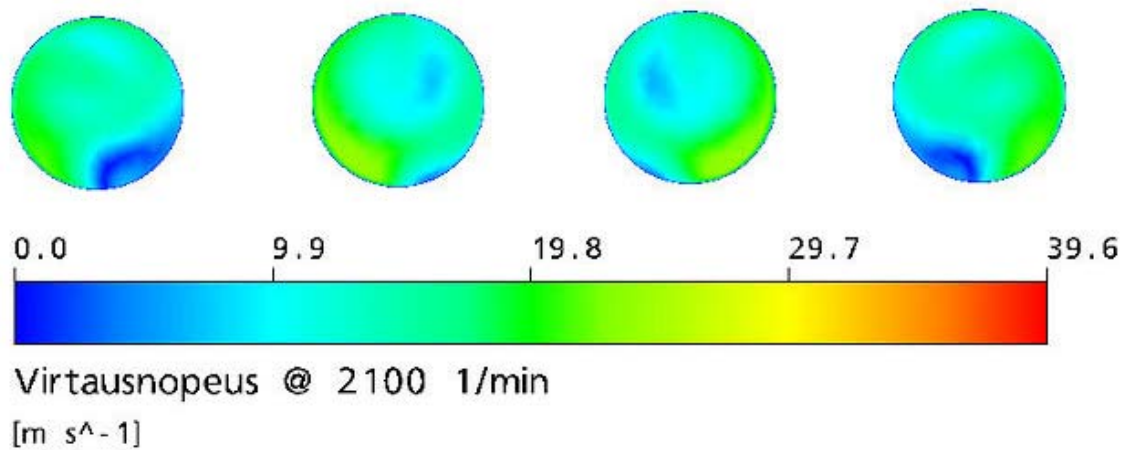
Virtausilmiöiden yleisluontoisiin tutkimuksiin riittävät väljemmät virtauslaskentamallin määrittäykset. Hyvin likiarvoisiin tuloksiin päästään löysemmillä reunaehdoilla ja harvemmalla verkotuksella. Tämä soveltuu hyvin esimerkiksi ensimmäisiin koelaskentoihin virtauslaskentaan tutustuessa.

6.2 Virtausnopeudet eri kierrosnopeuksilla

Seuraavissa kappaleissa imuputkiin on viitattu numeroin 1–4, vasemmalta oikealle. Moottorin eri kierrosnopeuksilla suoritettuja virtauslaskennan kuvioita tarkasteltaessa ei voi välttyä huomioimasta, kuinka samanlaisia virtaukset eri kierrosnopeuksilla ovat. Nopeasti katsottuna ainoa asia, joka kuvissa on erilaista, on nopeusasteikon lukemat.

Kierrosnopeus 2100 1/min

Kuviossa 20 esitetyllä kierrosnopeudella 2100 1/min virtausnopeudet imuputkien ulostuloaukolla vaihtelivat välillä 0–26 m/s. Imuputkien 1 ja 4 alasisäreunassa esiintyi selvä matalamman virtausnopeuden alue, jossa virtausnopeus oli nollan ja noin 10 m/s välillä. Samoin putkien 2 ja 3 ylä- ja alasisäreunalla oli hitaamman virtauksen alue, jolla virtausnopeus oli noin 5–10 m/s.

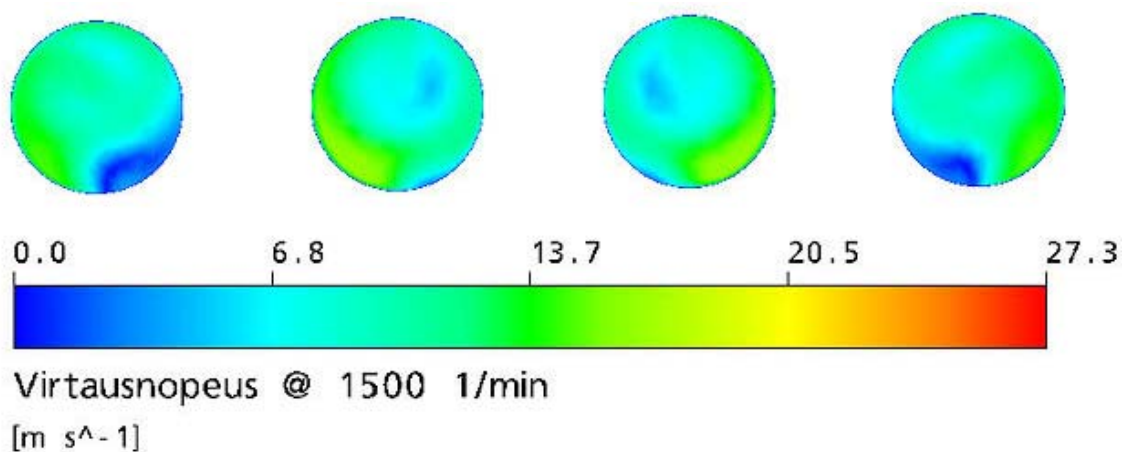


Kuvio 20: Imuputkien virtausjakauma moottorin kierrosnopeudella 2100 1/min

Vastaavasti suurimmat virtausnopeudet osuivat imuputkissa 1 ja 4 keskeltä ulkoreunalle, ollen noin 14–20 m/s. Imuputkissa 2 ja 3 suurimmat virtausnopeudet löytyivät alaulkoreunasta, jossa nopeus oli noin 20–26 m/s.

Kierrosnopeus 1500 1/min

Kuviossa 21 esitetyllä suurimman väännön tuottavalla kierrosnopeudella virtausnopeudet ulostuloaukoilla vaihtelivat nollan ja noin 18 m/s välillä. Hitaimmat virtaukset löytyivät edellisen tapauksen tavoin putkista 1 ja 4 alhaalta sisäreunasta. Virtausnopeus oli noin 0–5 m/s. Putkien 2 ja 3 keskeltä sisäreunaan hitain virtausnopeus oli luokkaa 5–7 m/s. Suurimmat virtausnopeudet olivat putkissa 1 ja 4 noin 8–14 m/s ja putkissa 2 ja 3 noin 15–18 m/s.

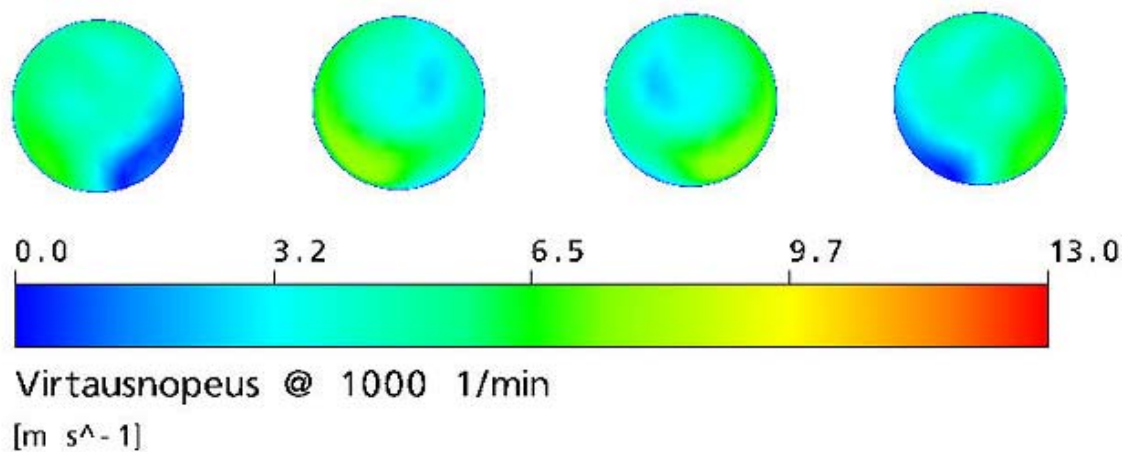


Kuvio 21: Imuputkien virtausjakauma moottorin kierrosnopeudella 1500 1/min

Kierrosnopeus 1000 1/min

Pienimmällä tarkastellulla kierrosnopeudella virtausnopeudet olivat välillä 0–9 m/s. Pienimmät virtaukset löytyivät edellisten tapausten tavoin uloimmista imuputkista, joissa hitain virtaus oli luokkaa 0–3 m/s. Sisemmissä imuputkissa virtaus oli hitaimmillaan noin 2–3 m/s. Suurimmat virtausnopeudet olivat ulommissa putkissa noin 4–6 m/s ja sisemmissä imuputkissa noin 7–9 m/s.

Kuvioita tarkkaan vertailemalla huomaa, että virtausnopeuksien pienentyessä erot virtausnopeuksissa tasoittuvat hieman.

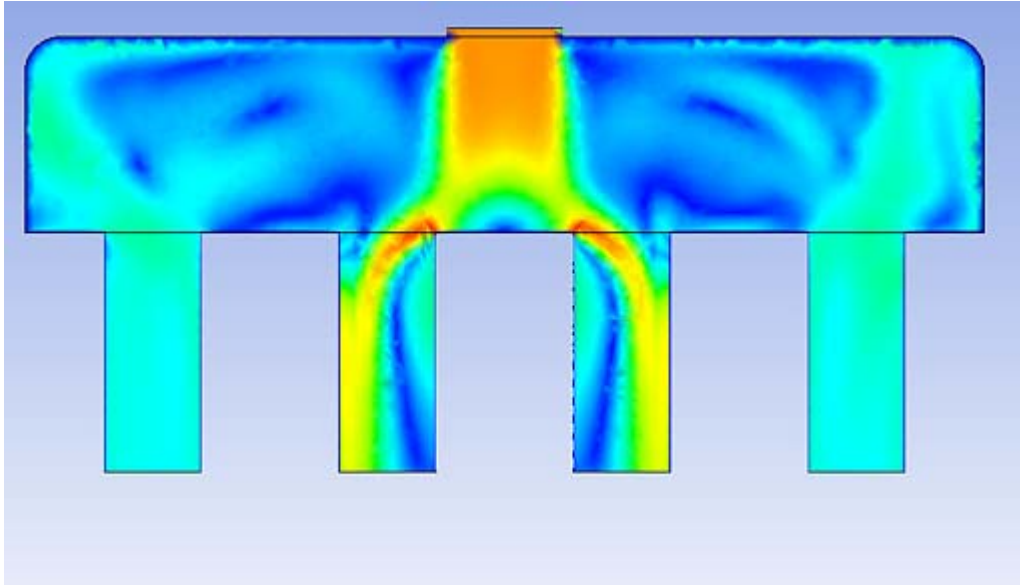


Kuvio 22: Imuputkien virtausjakauma moottorin kierrosnopeudella 1000 1/min

6.3 Tekijät jotka vaikuttavat virtauksen tasaiseen jakautumiseen

Hyvää imusarjaa suunniteltaessa täytyy hyvän virtaavuuden lisäksi ottaa huomioon monia muitakin asioita. Paineen tulisi jakautua paineentasauskammiossa siten, että moottorin käynnistä aiheutuvat paineiskut imusarjassa parantaisivat moottorin täytöstä ja parantaisivat näin moottorin toimintaa. Ajoneuvoa suunniteltaessa myös imusarjan ulkomuoto voi olla moottorin tilan komponenttisijoittelusta riippuen tiukkaan rajoitettu. Suunnittelijan on siis tehtävä kompromisseja toimivuuden ja käytännöllisyyden suhteen. Tässä projektissa kompromisseja ei tehty, sillä huomio kohdistettiin ainoastaan imusarjan hyvään virtaavuuteen.

Suurin asia, mikä kymmeniä imusarjageometrioita tutkiessa nousi esiin, oli että virtauksen poikkipinta-ala ei saa muuttua liian äkkinäisesti. Tästä on otettu esimerkiksi toisen mallisarjan mallin virtausjakauma kuviossa 23.

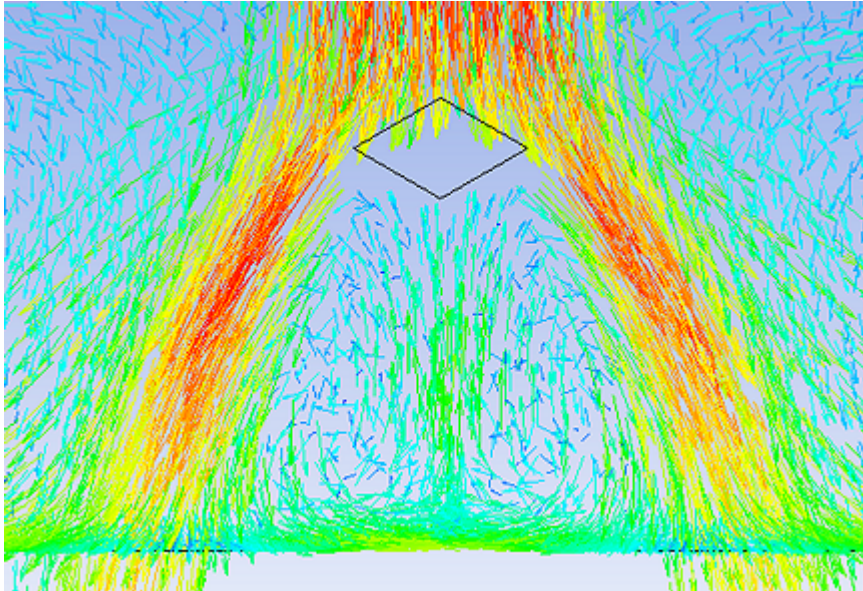


Kuvio 23: Esimerkki virtauksen poikkipinta-alan nopeasta muutoksesta virtauksessa sisääntuloaukosta paineentasauskammioon

Kuviossa sisääntuloaukon virtaus saapuu paineentasauskammioon ylhäältä alaspäin. Oranssilla värillä kuvattu suuri virtausnopeus tasaantuu huonosti imusarjan sisällä. Poikkipinta-alaan muuttuessa nopeasti virtaus ei ehdi tasaantua, vaan keskittyy keskimmäisiin imuputkiin. Virtauksen ympärille syntyy suuria pyörteitä, jotka puolestaan johtavat suuriin paine- ja nopeuseroihin imusarjassa. Nämä tekijät johtavat huonon virtausjakauman lisäksi epädulliseen moottorin täytökseen ja palamisreaktioon.

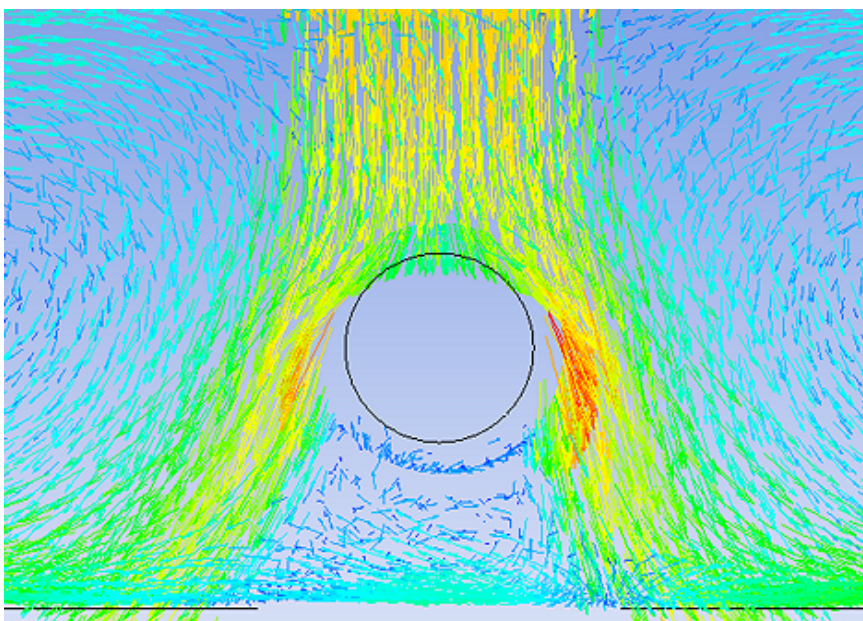
Parempi virtausjakauma imuputkien välillä saavutetaan kasvattamalla sisääntuloaukon profiilia hitaasti kohti paineentasauskammiota, kuten lopullisessa 3D-geometriassa tehtiin. Tästä on esimerkki kuviossa 18. Virtausviivat jakautuvat melko tasaisesti imuputkien kesken ja vain paineentasauskammion yläosaan syntyy pyörteilyä. Todettiin myös, että ilman sisääntuloaukon ollessa symmetrisesti imusarjan keskilinjalla, virtauksen jakautumista on helpompi kontrolloida.

Toinen tärkeä huomio tuloksia analysoidessa oli, että terävät kulmat imusarjassa aiheuttavat helposti pyörteisiä virtauksia. Tästä esimerkkinä virtaukset kuvioissa 21 ja 22.



Kuvio 24: Esimerkki terävien kulmien aiheuttamista pyörteistä

Kuviossa 24 virtaus tulee ylhäältä sisääntuloaukosta paineentasauskammioon ja osuu teräväkulmaiseen esteeseen. Virtaus kääntyy jyrkästi alkuperäisestä suunnastaan ja esteen sivuilla olevien terävien kulmien alapuolelle syntyy voimakasta pyörteilyä.



Kuvio 25: Esimerkki pyöreän esteen aiheuttamasta virtauksen muuttumisesta

Kuviossa 25 imusarjaan on sijoitettu teräväkulmaisen sijasta pyöreä este. Kuten virtauksesta huomataan, pyöreä pinta aiheuttaa huomattavasti vähemmän muutoksia virtaukseen. Esteen alapuolelle ilmaantuu kyllä pyörteilyä, mutta se ei ole yhtä selvää kuin teräväkulmaisella esteellä. Virtauksen suunta muuttuu vain vähän, ja mikäli este olisi korkeammalla paineentasauskammiossa, virtaus voisi jatkua lähes samansuuntaisena ennen kammion pohjaan osumista. Ero teräväkulmaiseen esteeseen on kuitenkin huomattava.

Imusarjaa mallinnettaessa myös imuputkien pituudella on vaikutusta virtausjakaumaan. Pitemmillä imuputkilla virtausnopeus tasaantuu paremmin, mutta imusarjan toimivuuden kannalta imuputkien pidentäminen ei kannata.

7 Päätelmät

7.1 3D-mallinnuksen onnistuminen

3D-mallinnus onnistui työn sujuvuuden kannalta hyvin. Ohjelmiston kanssa ei tullut mainittavia toimivuusongelmia ja geometrioista saatiin halutun muotoisia. Ensimmäiset imusarjamallistot luotiin ainoastaan sillä ajatuksella, että lopputuloksen tulee näyttää ajoneuvotyypiseltä imusarjalta. Tämä tavoite saavutettiin.

Seuraavia mallisarjoja lähdettiin kehittelemään sitten enemmän virtauslaskennan tulosten perusteella. Kehitys oli hidasta ja pienetkin muutokset geometriassa vaativat uuden virtauslaskennan. Tämä johti siihen, että malleja syntyi yhteensä yli sata. Geometriamuutosten tekeminen oli kyllä suhteellisen nopeaa, mutta kun puutteellisista koodaustaidoista johtuen kaikki ensimmäisten viikkojen virtauslaskennat jouduttiin suorittamaan koulun tietokoneella, kaikkea työhön käytettävissä ollutta aikaa ei voitu käyttää täysin hyödyksi.

Lopullisen imusarjamalliston monimutkaisemmasta geometriasta johtuen mallit luotiin vain imusarjan sisätilasta. Pyrittäessä eroon kaikista turhista terävistä kulmista mallin luominen ei onnistunut enää Part Design -moduulissa, vaan geometria luotiin pintamallinnuspuolella. Tämä mahdollisti monimutkaisten pyöreiden muotojen luomisen.

Kaiken kaikkiaan 3D-mallinnuksen onnistumiseen oltiin varauksin tyytyväisiä. Löysistä mallinnuksen reunaehdoista johtuen mielikuvitukselle annettiin sijaa kukoistaa. Toisaalta tämä vapauttaa suunnittelijan tekemään toisinaan toiminnallisesti puutteellisia kokeiluja. Jatkuvan muokkaamisen ja testaamisen ansiosta tuloksia saatiin kuitenkin hitaasti parannettua. Ohjelmiston käyttö oli suhteellisen ongelmaton ja geometriat saatiin helposti tallennettua ANSYSin hyväksymään muotoon.

7.2 Virtauslaskennan onnistuminen

Virtauslaskenta ei ole helppoa. Tämän sai todeta toistuvasti projektin aikana. Tutoriaalitehtävissä käytetyt suorakaiteen muotoiset kappaleet laminaarisessa virtauksessa olivat hyvää esittelyä virtauslaskentaohjelmiston käytössä. Tutoriaalitehtävistä omaan projektiin siirryttäessä haasteet nousivat kuitenkin kymmenkertaisiksi.

Turbulenssin mallintamisessa, verkotuksen onnistuneessa luomisessa, reunaehtojen oikein asettamisessa ja laskennan tulosten analysoimisessa olisi tarvittu runsasta perustietämystä virtaustekniikan perusteista jo alusta lähtien. Erityisesti tulosten analysointi vaatisi runsaampaa tietotaitoa virtauslaskennan alalta.

Suurin osa työn aikana tehdyistä tärkeistä päätöksistä liittyi siihen, mitä mallinnusta mihinkin virtauslaskennan operaatioon käytettäisiin. Tässä turvauduttiin lähes poikkeuksetta ohjelmiston ohjedokumentaatioon, joka yllätti laajuudellaan positiivisesti. Jokaisen toiminnon ja mallin toiminta ja taustateoriat oli selvitetty todella tarkasti. Silti päätösten suhteen oltiin toisinaan hakoteillä ja välillä vertailtiin eri mallien käyttämisen luomaa eroa tuloksissa.

Projektin tulosten paikkansapitävyydestä ei voida lausua selvää argumenttia suuntaan tai toiseen. Lähinnä pinnallinen tietämys virtauslaskennasta johtaa siihen, että tulosten tarkkuudessa eri laskentojen välillä voi olla huomattaviakin eroja. Osaamattomissa käsissä ohjelmistolla saaduilla tuloksilla on siis hyvin vähän vastinetta tosielämässä tapahtuvalle virtaukselle ja tulosten jälkikäsitteilyssä päästään ihastelemaan kauniita kuvia joilla ei, epätarkkuudesta johtuen, ole varsinaista funktiota.

Virtauslaskennan suorittaminen projektin aikana oli alusta asti kokeilemisen ja epäonnistumisen sekä kokeilemisen ja onnistumisen vuorottelua, jossa yllättävän pienistä asioista, kuten skandinaavisten merkkien tai rivinvaihtojen käytöstä, koitui yllättävän suuria vaikeuksia, joita ratkottiin useiden asiantuntijoiden voimin.

Loppujen lopuksi virtauslaskennassa onnistuttiin vähintään keskinkertaisesti. Tuloksia saatiin parannettua alun kokeiluista huomattavasti ja tulevaisuutta ajatellen kaikesta

projektin aikana opitusta on varmasti hyötyä. Tarkempien tulosten saaminen vaatisi kuitenkin huomattavasti laajempaa ohjelmiston käyttökokemusta ja teoriataustaa.

7.3 ANSYS CFX -ohjelmiston soveltuvuus suunnittelutyöhön

Ohjelmiston käytön opettelu oli muutamaa opastussessiota lukuun ottamatta hyvin pitkälti itsenäistä työskentelyä, jossa toimintojen vaikutusta lopputulokseen mietittiin ensin maalaisjärjellä, toiseksi kokeilemalla ja kolmanneksi ohjeita selaamalla.

Ohjelman peruskäytön voi opettaa täysin aiheesta tietämättömälle päivässä. Tämä tarkoittaa kuitenkin sitä, että käyttäjä tekee asioita, joiden tarkoitusta hän ei tarkemmin ymmärrä, kuten on asian laita varmaankin useimpien tietokoneohjelmien käytön kanssa nykypäivänä. Virtauslaskentaa tehdessä on kuitenkin tärkeää tietää mitä seurauksia jonkin tietyn toiminnon tai asetuksen valitsemisella on. Lopputuloksen tarkkuus voi huonontua dramaattisesti väärän ratkaisumallin tai asetuksen takia.

Virtauslaskennalla elantonsa tekevän täytyy tietää mitä tekee. Optimitilanne on, että työntekijä ymmärtää ohjelmiston taustalla vaikuttavat periaatteet ja kaavat pintaa syvemmillä. Yliopistopohjainen koulutus on siis suotava edellytys teorioiden ymmärtämiselle ja soveltamiselle.

ANSYS virtauslaskentaohjelmisto on hyvin ammattimainen työkalu. Sillä onnistuu hyvin monimutkaisten simulaatioiden luominen ja analysointi. Käyttöliittymä on hyvin CAD-mainen ja suhteellisen looginen. Toiminnot ja ulkonäkö moduulien välillä ovat yhtenäisiä ja helpottavat ohjelman peruskäyttöä. Tarvittaessa koko virtauslaskentaprojektin voi suorittaa geometrian luonti mukaan luettuna samalla ohjelmistolla. Tämä on taloudellisesti suuri etu, koska erillisiä 3D-ohjelmistoja ei tarvita.

Parannettavaa ohjelmistossa on lähinnä pikkuasioissa. Yksi turhautumisia aiheuttanut tekijä oli ohjelmiston kyvyttömyys arvioida verkotuksen määrittämiseen kuluvaa aikaa. On täysin mahdollista kuluttaa tehokas työpäivä siihen että odottaa verkotuksen valmistumista.

Toinen verkotukseen liittyvä pulma oli, kun monimutkaista verkkoa luodessa ohjelmisto kaatui tietokoneen muistin loppuessa kesken. Ohjelmisto voisi siis antaa jonkinlaisen valistuneen arvauksen prosessiin kuluva muistista ja osoittaa sen käyttäjälle ennen verkotuksen aloittamista. Tällä säästettäisiin tehokasta työaikaa. Verkotus on ilmeisesti mahdollista suorittaa komentokehotteesta superklusterillakin, mutta tavalliselle käyttäjälle se on liian monimutkaista.

Kaiken kaikkiaan ANSYS-ohjelmiston käyttö oli alkututustumisen jälkeen suoraviivaista toimintaa, jossa huomasi ohjelmiston käyttömukavuuden olevan hyvällä pohjalla. Ohjelman jatkokäyttö onnistuu varmasti pitemmänkin tauon jälkeen, koska ohjelman moduulit ja käyttöliittymä ovat hyvin loogisia.

7.4 Ajatuksia projektin jatkokehittämisestä

Tieteellisesti projektin anti jäi melko vaatimattomaksi. Opinnäytetyön ajallinen laajuus, tai tässä tapauksessa suppeus, esti tarkemman tieteellisen analyysin tekemisen. Jatkossa ANSYSin ohjelmiston peruskäyttö opetetaan toivottavasti jo jollain kurssilla kouluaikana, jolloin olisi mahdollista laajentaa imusarjaprojektia huomattavasti.

Esimerkiksi moottoria voitaisiin mallintaa siihen pisteeseen asti, että päästäisiin mallintamaan palotapahtuma ja siitä johtuvat painevaihtelut imusarjassa. Tällöin voitaisiin tutkia tarkemmin paineentasauskammion optimaalista muotoilua ja paineiskujen vaikutusta moottorin täytökseen.

Asiantuntemuksesta projektin laajentaminen ei ainakaan jäisi kiinni. Process Flow tarjosi ainakin tätä projektia varten hyvää tukea malliesimerkkien ja opetuskäynnin muodossa.

Lähteet

- A Brief History of Computational Fluid Dynamics (CFD). [www-sivu] [viitattu 8.7.2009]
 Saatavissa: <http://www.fluent.com/about/cfdhistory.htm>
- ANSYS CFX Introduction 2006 [pdf] [viitattu 6.7.2009] Saatavissa CSC:n asiakkaille:
https://hotpage.csc.fi/appl/cfd/manuals/cfx/cfx11/AnsysCFX11_Introduction.pdf
- ANSYS CFX-Pre User's Guide 2006. [pdf] [viitattu 8.7.2009] Saatavissa CSC:n asiakkaille:
https://hotpage.csc.fi/appl/cfd/manuals/cfx/cfx11/AnsysCFX11_Pre.pdf
- ANSYS CFX-Solver Modeling Guide 2006 [pdf] [viitattu 6.7.2009] Saatavissa CSC:n asiakkaille:
https://hotpage.csc.fi/appl/cfd/manuals/cfx/cfx11/AnsysCFX11_SolverModeling.pdf
- ANSYS CFX-Solver Theory Guide 2006 [pdf] [viitattu 7.7.2009] Saatavissa CSC:n asiakkaille:
https://hotpage.csc.fi/appl/cfd/manuals/cfx/cfx11/AnsysCFX11_Theory.pdf
- ANSYS CFX-Solver User's Guide 2006. [pdf] [viitattu 9.7.2009] Saatavissa CSC:n asiakkaille:
https://hotpage.csc.fi/appl/cfd/manuals/cfx/cfx11/AnsysCFX11_SolverManager.pdf
- Bhaskaran, Rajesh & Collins, Lance. Introduction to CFD Basics. [online] [viitattu 6.7.2009].
<http://courses.cit.cornell.edu/fluent/cfd/intro.pdf>
- Bohl, Willi 1971. Teknillinen virtausoppi, toinen painos 1988. SONATOR OY: Tampere.
- NASA. Navier-Stokes Equations. [www-sivu] [viitattu 7.7.2009] Saatavissa:
<http://www.grc.nasa.gov/WWW/K-12/airplane/nseqs.html>
- Shaw, C.T. Using Computational Fluid Dynamics 1992. Prentice Hall. [pdf][viitattu 8.7.2009] Saatavissa: <http://www.topajka-shaw.co.nz/UCFD.pdf>
- TKK, Rakennusfysiikan perusteiden luentomateriaali 2008 [pdf] [viitattu 28.7.2009] Saatavissa:
<http://users.tkk.fi/~talesone/TKK/Kurssit/Rakennusfysiikan%20perusteet/Luentomateriaali/Konvektio.pdf>

Liitteet

Liite 1: Esimerkki jobfile-tiedoston rakenteesta rinnakkaisajossa

```
#!/bin/csh
###
### parallel job script example
###
# Initializes the execution environment
#BSUB -L /bin/csh

## name of your job, %J will show as your jobID
#BSUB -J my_job%J

## system error message output file
#BSUB -e err_%J

## system message output file
#BSUB -o out_%J

## a per-process (soft) memory limit
## limit is specified in KB
## example: 1 GB is 1048576
#BSUB -M 1048576

## how long a job takes, wallclock time hh:mm
#BSUB -W 00:30

##the number of processes (number of cores)
#BSUB -n 4

## when job is finished you will get email notification
#BSUB -u email@address.fi
#BSUB -N

## run my MPI executable
module load ansys/ansys_11.0_cfx
cd $WRKDIR/username
cfx5solve -def filename.def -part 4 -part-mode metis-kway -par-local -start-method
'PVM Local Parallel'

## bjobs will save some information about my job
bjobs -l $LSB_JOBID
```