

# Putkivirtaus

Hugo Tamm, Energia- ja konetekniikka, 21.02.2023

Tarkoituksena on luoda putkivirtausta simuloiva ohjelma, jossa voi asettaa alku- ja loppupään, mahdollisesti useamman pään, käyttäjän haluamaan ruudukkoon. Pääty yhdistetään ensisijaisesti suorilla putkilla ja 90:n asteen mutkaputkilla. Mahdollisesti olisi tarjolla myös muita variaatioita, mm. T-putkia sekä pumppuja. Putket ovat ruudukossa 1 metrin pituisia laskujen helpottamiseksi.

Tavoitteena on saavuttaa vaativa vaikeustaso ja ylittää se esimerkiksi luomalla ohjelmaan myös pieni putkien yhdistely peli, jossa pelaajan täytyy yhdistää kaikki putket toisiinsa kääntämällä ne oikeinpäin tietyn aikarajan sisällä. Bernoullin lain avulla pystyy laskemaan putken sisällä kulkevan nesteen (tässä tilanteessa veden) nopeutta. Lisäämällä esimerkiksi eri paksuisia putkia, voidaan lisätä vähän enemmän monipuolisuutta ja käytännön iloa ohjelmaan.

Ohjelma toteutetaan graafisella käyttöliittymällä, jolla olisi tarkoitus kerätä käyttäjän antamia syötteitä sekä tulostaa tietoa samalle ikkunalle.

Pääikkunassa näkyy iso ruudukko, jossa on alku- ja päätepiiste, joille voi molemmille asettaa käyttäjän haluaman paineen. Ikkunassa on oma lokero putkille, joita voi asettaa ruudukkoon. Ruudukosta voi poistaa putkia klikkaamalla niitä. Ikkunassa on syöttökentät paineille, kuvalliset nappulat putkille sekä nappi nopeuden- ja paineen laskemiselle. Kaikki nappulat antavat ohjelmalle komennon tehdä jotain.

Ohjelmaa käytetään siis ensisijaisesti putkiston sisällä virtaavan aineen nopeuden laskemiseksi, kun tiedetään paineet. Ohjelmassa on myös pieni putkipeli, jolla voi haastaa itseään. Pääikkunassa on nappi kyseiselle pelille, joka avaa uuden ikkunan, johon muodostuu pelikenttä täynnä putkia. Putket täytyy yhdistää toisiinsa kääntämällä niitä niin, ettei missään ole avoimia päitä.

Ohjelma on jaettu moneen osaan alla olevan UML-diagrammin mukaan.



Ohjelma käyttää .png kuvatiedostoa, jolla esittää *GridElementtejä*. *PipeGridin* taustana voi toimia .png tai .jpg kuvatiedosto, jotta ohjelma olisi helposti tulkittavissa. Muunlaisille ohjelman ulkopuolisille tiedostoille ei ole varsinaista tarvetta, sillä käyttäjä voi antaa kaikki syötteet helposti käyttöliittymässä eikä ohjelma tallenna tuloksia mihinkään.

Laajennettu Bernoullin-yhtälö on muotoa

$$\frac{p_1}{\rho g} + \frac{v_1^2}{2g} + z_1 + h_{loss} = \frac{p_2}{\rho g} + \frac{v_2^2}{2g} + z_2$$

Tämän voisi saattaa muotoon, jossa muuttujat ovat kaikki korkeuden (m) yksikössä. En vielä ole selvittänyt miten.

Toinen tapa laskea nopeus paineiden avulla on

$$v = \sqrt{\frac{2(p_2 - p_1)}{\rho}}$$

Yksi sovellus tästä on pitot-putki, jolla voidaan mitata putken läpikulkevan aineen nopeutta mittaamalla paine-eroa kahdessa eri pisteessä. Kaava ei välttämättä sovellu täysin tehtävään, mutta sitä voi kokeilla.

Testauksessa olisi hyvä testata:

- virran nopeuden laskeminen
  - Nopeuden pitäisi muuttua ruudukossa olevan putkiston sekä käyttäjän antamien paineiden mukaisesti. Tätä voidaan testata laskemalla tietyn tilanteen tulos etukäteen ja tarkistamalla onko ohjelman antama tulos sama.
- paineen laskeminen
- ruudukossa olevan putkiston eheys
  - Jokaisen putken pään täytyy olla liitoksissa johonkin toiseen putkeen tai pääty pisteeseen. Tälle voisi luoda jonkun oman funktion, joka tarkistaa, että putken päät ovat liitettyinä.

- nappien toiminta
  - Putkien valikossa olevien nappien täytyy vaihtaa aktiivisena olevaa putken osaa, jotta ruudukkoon tulisi oikea putki tyhjää ruutua painaessa.
  - Laskuja suorittavien nappien tulisi käynnistää tarpeelliset funktiot *Bernoulli*-luokan sisällä ja tulostaa niiden tulos käyttöliittymään.
- syötteet
  - Syötteiden formaatti ja pätevyys täytyy tarkistaa, ettei laskuja yritetä suorittaa merkkijonoilla tai muilla sopimattomilla arvoilla. Yksinkertainen assertRaises-komento pitäisi toimia tässä tilanteessa.
- ruudukkoon putkien laittaminen ja poistaminen
  - Putkien täytyy ilmestyä tyhjään ruutuun, kun ruutua painaa ja poistua, kun putkea painaa.
- ruudukon tyhjentäminen
  - Ruudukon tulisi tyhjentyä, kun painaa "Clear"-nappia, joka tyhjentää ruudukon.
- päätepisteiden siirtäminen
  - Päätepisteitä pitäisi pystyä siirtämään vetämällä niitä. Tätä vetämistä ei välttämättä pysty testaamaan yksinkertaisesti, koska tällaisen ominaisuuden lisääminen ei ehkä ole edes mahdollista. Mutta päätepisteiden sijainnit pystytään tarkistamaan aina siirtojen jälkeen ja näin voidaan testata ovatko ne oikeissa paikoissa.

Math-kirjasto saattaa olla tarpeellinen laskujen muodostamisessa. Muuten ohjelman toteutus onnistuu pelkällä PyQt:llä.

Ohjelman tekeminen aloitetaan siitä, että opitaan PyQt:llä tekemään sovelluksen pääikkuna, sisältäen putket, napit sekä ruudukon. Putkivirtauksen laskuja täytyy vähän vielä hioa ja varmistaa, että ne ovat oikein. Heti, kun on ohjaustapaamiset hoidettu 3.3.2023 mennessä, niin aloitan ohjelmoinnin.

Ensimmäinen tavoite on saada käyttöliittymä toimimaan niin, että ruudukko ja putket ovat näkyvillä ja niitä voisi käyttää. Aikaisemmin esitetystä UML-

diagrammissa on luokkia, joita luon tarpeen mukaan. Ensin pääikkuna, sitten ruudukko, sitten *GridElement*-luokka, joka jakautuu putkiin ja päätepisteisiin. Seuraavaksi pitää varmistaa toimintakyky näille luokille. Sitten seuraa matemaattinen osuus, kun putket ovat ruudukossa ja niitä voi siirtää, lisätä ja poistaa.

Tiedonhaussa on käytössä Google, Youtube, vanhat Aallon omat virtausmekaniikan kurssimateriaalit. Esimerkiksi sivustot:

<https://phys.libretexts.org/>

<https://courses.lumenlearning.com/>

<https://mycourses.aalto.fi/course/view.php?id=28999&section=3>

Michel van Biezen - youtube kanava

The Efficient Engineer – youtube kanava