Norwegian University of
Science and Technology

Transient Modelling of Hydropower Plants

Simen Bomnes Valåmo

Master of Energy and Environmental Engineering
Submission date: June 2016
Supervisor: Bjørnar Svingen, EPT
Co-supervisor: Torbjørn Nielsen, EPT

Norwegian University of Science and Technology
Department of Energy and Process Engineering

NTNU
Norwegian University
of Science and Technology

Department of Energy
and Process Engineering

EPT-M-2016-148

MASTER THESIS

for

Student
Simen Bomnes Valåmo

Spring 2016

English title
Transient Modelling of Hydropower Plants

Norwegian title
Transient modellering av vannkraftverk

Background and objective

The candidate has in the project work made a generic analysis program written in Matlab. This program can be used to model arbitrary hydraulic systems used in hydropower,

The objective is to enhance and extend the program. In particular a turbine model shall be made, and more simple methods of building systems from discrete components shall be investigated. The use of "virtual surge shaft" shall be investigated; is that a feasible solution method, and if so, how shall it be done most effectively. Results from the program shall be verified with measured data.

The following tasks are to be considered:

1 A literature study of different turbine/pump models, including governor, and system modelling in general.
2 Simplifications and enhancements of the existing program. Investigate the use of "virtual surge shaft", and how to increase execution speed.
3 Make a turbine model with *power governing* and closing/opening along *arbitrary curves*, and eventually a pump model. The turbine/pump model shall be as generic as possible. Interpolations in proprietary and hard to get to characteristic curves (hill charts) shall **not** be used.
4. Verification of the program with measured data.
5. A written report.

-- " --

Within 14 days of receiving the written text on the master thesis, the candidate shall submit a research plan for his project to the department.

When the thesis is evaluated, emphasis is put on processing of the results, and that they are presented in tabular and/or graphic form in a clear manner, and that they are analyzed carefully.

The thesis should be formulated as a research report with summary both in English and Norwegian, conclusion, literature references, table of contents etc. During the preparation of the text, the candidate should make an effort to produce a well-structured and easily readable report. In order to ease the evaluation of the thesis, it is important that the cross-references are correct. In the making of the report, strong emphasis should be placed on both a thorough discussion of the results and an orderly presentation.

The candidate is requested to initiate and keep close contact with his/her academic supervisor(s) throughout the working period. The candidate must follow the rules and regulations of NTNU as well as passive directions given by the Department of Energy and Process Engineering.

Risk assessment of the candidate's work shall be carried out according to the department's procedures. The risk assessment must be documented and included as part of the final report. Events related to the candidate's work adversely affecting the health, safety or security, must be documented and included as part of the final report. If the documentation on risk assessment represents a large number of pages, the full version is to be submitted electronically to the supervisor and an excerpt is included in the report.

Pursuant to "Regulations concerning the supplementary provisions to the technology study program/Master of Science" at NTNU §20, the Department reserves the permission to utilize all the results and data for teaching and research purposes as well as in future publications.

The final report is to be submitted digitally in DAIM. An executive summary of the thesis including title, student's name, supervisor's name, year, department name, and NTNU's logo and name, shall be submitted to the department as a separate pdf file. Based on an agreement with the supervisor, the final report and other material and documents may be given to the supervisor in digital format.

☐ Work to be done in lab (Water power lab, Fluids engineering lab, Thermal engineering lab)
☐ Field work

Department of Energy and Process Engineering, 13. January 2016

Olav Bolland
Department Head

Bjørnar Svingen
Academic Supervisor

Research Advisor: Torbjørn Nielsen

Preface

This master thesis has been written at the Waterpower Laboratory, Department of Energy and Process Engineering at the Norwegian University of Science and Technology during spring 2016. The objective of this thesis has been to develop a generic simulation program, that can model any hydropower systems and simulate U-tube oscillations due to governing of the turbine opening degree.

First and foremost, I wish to acknowledge the help and guidance throughout this process from my supervisor, Dr.Ing. Bjørnar Svingen. I also show my appreciation to my co-supervisor, Prof. Torbjørn Nielsen, for good discussions and advises.

I am indebted to all the master students, Ph.D. students, Professors and all other employees at the Waterpower Laboratory, all of which have contributed to the great environment at the Laboratory. Thank you for the good discussions and for a fantastic last year as student.

Simen Bomnes Valåmo
Trondheim, June 2016

Abstract

Any changes of the volume flow through a turbine gives acceleration or deceleration of the masses of water in the penstock. This induces dynamic pressure changes in front of the turbine. A pressure transient must be controlled as it can cause severe damage on mechanical equipment, such as a turbine, if it becomes too abrupt. A widely used solution is to decrease the water inertia time constant, by placing surge shafts between the turbine and the reservoirs. Surge shafts introduces U-tube oscillations between the surge shaft and any free surface. These oscillations can give problems if the surges becomes too large. Air can be drawn into the system and cause cavitation near the turbine. Free surface flow in the tunnel can occur and expose the turbine to more stress. The surge shafts must be dimensioned according to the dynamics of the system, which is usually found through numerical simulations.

One dimensional modeling with discrete elements comprising inelastic water and pipes, is a well suited method for simulating U-tube oscillations in hydropower systems. The method is however restricted to a limited complexity of the system to be modeled.

This thesis investigates a solution that aims to enable any system to be modeled with discrete elements of inelastic water and pipes. The use of "Virtual surge shafts" is implemented in a generic simulation program written in Matlab. Four turbine models are implemented to extend the simulation possibilities.

The program is verified with the well known, fully transient simulation program LVTrans. Simulations of two hydropower systems with significantly different complexity is compared to simulation results from LVTrans.

The simulations show generally good compliance with simulations produced with LVTrans. The maximum up surge and down surge deviated with less than 5% for all simulations.

It can be concluded that the use of virtual surge shafts is a feasible solution, provided that the correct cross sectional is used. The implemented turbine models have enhanced the programs ability to simulate changes on the turbine opening degree, as the turbine characteristics are included.

The feasibility of "Virtual surge shafts" shows to be strongly dependent on the correct cross sectional area. It is recommended with further verification of the solution, to determine if an empirical value of the correct cross sectional area can be

established.

Sammendrag

Enhver endring av volumstrømmen gjennom turbinen vil akselerere eller retardere vannmassene i trykksjakta. Dette forårsaker en dynamisk trykkendring foran turbinen. Hvis det dynamiske trykket ved avslag på turbinen blir stort nok, kan det gjøre store skader. For å redusere den dynamiske trykkøkningen er det vanlig å redusere tilløpstiden for vannmassene. For de fleste høytrykkskraftverk i Norge, er dette løst ved å innføre en svingesjakt mellom turbin og øvre magasin. Svingesjakter vil innføre et nytt problem, det vil skape u-rørssvingninger mellom svingesjakt og magasin. Slike u-rørssvingninger kan skape problemer hvis de blir for store. Luft kan dras inn i tunnelen og skape kavitasjon ved turbinen, det kan oppstå vannspeilstrømning i tunnelen som kan føre til økt trykk på turbinen. Svingesjakter må dermed dimensjoneres etter dynamikken i hele vannkraftverket. Dynamikken er vanligvis funnet ved numeriske simuleringer.

En-dimensjonal modellering ved bruk av rør-elementer bestående av uelastisk vann og rør, er en mye brukt metode for simulering av u-rørssvingninger. Metoden gir en rask og nøyaktig løsning, men den har begrensninger i systemgeometrien.

Med mål om fjerne alle begrensninger i systemgeometrien for modellering ved bruk av rørelementer bestående av uelastisk vann og rør, skal det i denne masteroppgaven undersøkes en løsning på problemet ved bruk av «fiktive svingesjakter». Hvis denne løsningsmetoden fungerer, vil en-dimensjonal modellering kunne brukes til å simulere alle vannkraftsystemer, uavhengig av kompleksitet. Løsningen er implementert i et simuleringsprogram, laget av forfatteren gjennom hans fordypningsprosjekt. Fire generiske turbinmodeller er etablert og implementert i programmet, for å kunne gjøre simuleringer av ulike type turbiner med ulik turbinkarakteristikk.

Programmet er verifisert mot det velkjente, fulltransiente simuleringsprogrammet LVTrans, som er laget for beregning av trykkstøt og transienter i rørsystemer. Simuleringer gjort med to vannkraftsystemer med signifikant forskjellig kompleksitet, er sammenlignet med simuleringsresultater fra LVTrans.

Simuleringene gjort med programmet viser generelt god overensstemmelse med simuleringene fra LVTrans. Samtlige simuleringer gir et avvik på mindre enn 5% for maksimalt oppsving og nedsving i svingesjaktene.

Det kan konkluderes med at "fiktive svingesjakter" gir en løsning på problemet med begrensninger i systemgeomterien, forutsatt at det rikitge tverrsnittsarealet er brukt. De implementerte turbinmodellene har forbedret programmets evne til å

simulere endring av åpningsgraden, med reell turbinkarakteristikk.

At det rette tverrsnittsarealet blir brukt, viser seg å være avgjørende for gjennomførbarheten av "Fikitve svingesjkater" som løsningsmetode for problemene med begrensninger i systemgeometrien. Det anbefales at løsningsmetoden verifiseres mot flere vannkraftsystemer, med ulik grad av kompleks sammensetning, for å undersøke om det er mulig å finne en empirisk verdi for det rette tverrsnittsarealet.

Contents

Nomenclature

Acronyms

RK	Runge Kutta	[-]
$RK4$	Runge Kutta of order 4	[-]
$RK5$	Runge Kutta of order 5	[-]
$RKF45$	Runge-Kutta-Fehlberg Method	[-]
VSS	Virtual surge shaft	[-]

Greek Symbols

δ	Angle between generator stator and rotor	[$\circ$]
κ	Turbine opening degree	[-]
ω	Angular velocity	[rad/s]
ρ	Density of water	[kg/m^3]

Roman Symbols

A	Cross sectional area	[m^2]
a	Wave propagation speed	[m/s]
A_S	Free surface area	[m^2]
b_p	Permanent speed dropp	[-]
b_t	Transient speed droop	[-]
B_{t1}	Turbine inlet height	[m]
c	Servo motor velocity	[-]
D	Pipe diameter	[m]
D_{t1}	Turbine inlet diameter	[m^2]
D_{t2}	Turbine outlet diameter	[m^2]
f	Friction factor	[-]
g	Gravitational constant	[m/s^2]

H	Piezometric head	[m]
H_e	Effective head	[m]
H_R	Rated head	[m]
I_h	Hydraulic moment of inertia	$[kg/m^4]$
I_p	Polar moment of inertia	$[kg/m^2]$
K	Compression modulus	[Pa]
K_p	Governor proportional constant	[-]
L	Pipe length	[m]
M	Manning's number	[-]
m	Mass	[kg]
n	Rotational speed	[RPM]
Q	Volume flow	$[m^3/s]$
Q_R	Rated volume flow	$[m^3/s]$
T	Torque	[Nm]
t	Time	[s]
T_a	Acceleration time of the rotating masses	[s]
T_d	Governor derivative constant	[-]
T_i	Governor integrator constant	[-]
T_K	Servo motor time constant	[-]
T_W	Water inertia time constant	[s]
v	Mean fluid velocity	$[m/s]$
y	Servo motor position	[-]
z	Water level	[m]

List of Figures

List of Tables

1 Introduction

Hydropower systems contains large masses of water and the dynamics of the systems are extensive. Any changes of the volume flow through a turbine in a hydropower plant, accelerates or decelerates the masses of water in the pipelines. This acceleration or deceleration induces dynamic pressure changes. A pressure transient can cause severe damage on the mechanical equipment in the system. An abrupt pressure rise in front of a turbine may cause water hammers to occur. These pressure changes must therefore be controlled. The most common solution is to reduce the the water inertia time constant by inserting surge shafts between the turbine and the reservoirs. Surge shafts introduces U-tube oscillations that creates shaft surges. If the amplitude of the surges becomes too large, problems such as air entrance and free surface flow in the tunnel is introduced. This can cause cavitation near the turbine and expose the turbine to more stress. A correct dimension of the surge shaft is required and the system dynamics is of great importance. The dynamics of the system is usually found through numerical simulations.

One dimensional modeling of hydropower systems from discrete elements comprising inelastic water and pipes, is a well known method. The method provides fast and accurate solutions, and is well suited for simulation of mass oscillations in hydropower systems. Solving a dynamic system, such as a hydropower system, directly with a numerical solver requires all variables to be described with differential equations. This creates problems for the head at nodes and branches, as the head is not defined by a differential equation. For modeling of system with high complexity, this gives numerical loops which limits the methods ability to model complex systems.

This thesis is a continuance of the authors project thesis [Valaamo, 2015]. In the authors project thesis, a generic analysis program based on one dimensional modeling was made. The program solves a modeled system in the time domain and simulates U-tube oscillations in hydropower systems. The program has limitations in the system geometry and the turbine is represented as a simple valve.

The objective of this thesis is to develop an extended and enhanced program, that can model any hydropower systems and simulate U-tube oscillations due to governing of the turbine opening degree. To better reflect the behavior of a real turbine, a generic turbine model with power governing and closing/opening along arbitrary curves will implemented. To solve the problem regarding numerical loops, the use of "virtual surge shafts" is investigated. If this is a feasible solution method, how to effectively implement it in the generic simulation program shall be established. A verification of the developed program will be made. In consultation with the supervisor, Bjørnar Svingen, it was decided that the program shall be verified with results from the fully transient simulation program LVTrans.

Initially, the objective was to include a pump mode in the turbine model that would be used by a Phd. student to verify experimental results. The experiment was postponed and in consultation with the supervisor, Bjørnar Svingen, it was decided that the pump mode should not be included in the turbine model.

The thesis is structured as follows:

In **section 3**, the theory behind U-tube oscillations is described and the equations describing the different elements in a hydropower system are established. Four turbine models are presented.

In **section 4 and 5**, the numerical method and the simulation method are described. An investigation of "virtual surge shafts" is made and how to effectively implement it in the developed program is discussed.

In **section 6**, the developed simulation program uTubeOscV1 is presented. An explanation of the program structure is given and it is described how to use the program.

In **section 7**, various simulations from the developed program are verified with results from LVTrans.

2 Previous work

The Francis turbine models presented in this thesis are based on a Francis turbine model developed by Professor Torbjørn Nielsen in his Ph.D. thesis [Nielsen, 1990b]. The model is based on the Euler turbine equation and this is described by Nielsen in both the Ph.D. thesis and a paper [Nielsen, 2015].

The frequency governor model used in this thesis is presented by Professor Torbjørn Nielsen in the paper "Dynamic behaviour of governing turbines sharing the same grid" [Nielsen, 1996].

The idea of a Virtual surge shaft is based on a solution method for under-causal models, described by Forbes T. Brown in "Engineering System Dynamics" [Brown, 2001]. The solution method is related to modeling by use of the Bond Graph Method.

3 Theory

3.1 Pressure propagation

Any change of the volume flow through a turbine gives an acceleration or deceleration of the masses of water in the pressure shaft. This acceleration or deceleration induces a dynamic pressure change in front of the turbine. For a turbine running at full load, a rapid change of the guide vane opening, closing the wicket gate, results in a high pressure change due to sudden deceleration of the masses of water. For a substantial increase in the transient pressure, a water hammer will occur.

3.1.1 Water hammer

The elasticity of the water causes elastic pressure waves in the pipelines, so called water hammers. Water hammers occur when water in motion is quickly forced to change direction or is stopped, as for a turbine shut down or a sudden change of the guide vane opening. With reference to Fig. 3.1.1, if the wicket gates closes instantly a deceleration of the masses of water creates an immediate pressure rise in front of the turbine. The masses of water travels with a velocity $v = v_0$ before the wicket gates are closed. Because of the elasticity, a pressure front travels towards the upper reservoir with the speed of sound. At a given time the velocity downstream the pressure front will be $v = 0$ and the velocity upstream the pressure front will be $v = v_0$, (a). When the pressure front reaches the upper reservoir, the velocity is zero throughout the whole pipeline, giving a higher pressure in the pipeline than the pressure in the reservoir. The pressure difference makes the water flow from the pipeline into the reservoir, (b). This goes on until the pressure in the pipeline equals the pressure in the reservoir. Then the water in the pipeline have a velocity $v = -v_0$ and since the wicket gate is closed, this will give a negative pressure in front of the turbine, (c). A negative pressure wave will travel with the speed of sound and creates a negative pressure difference between the reservoir and the pipeline. Water from the reservoir starts to travel from the reservoir into the pipeline until equilibrium in pressure is reached between the reservoir and the pipeline, (d). The velocity in the pipeline is again $v = v_0$ and since the wicket gate is still closed, the whole process will repeat itself. [Nielsen, 1990a]

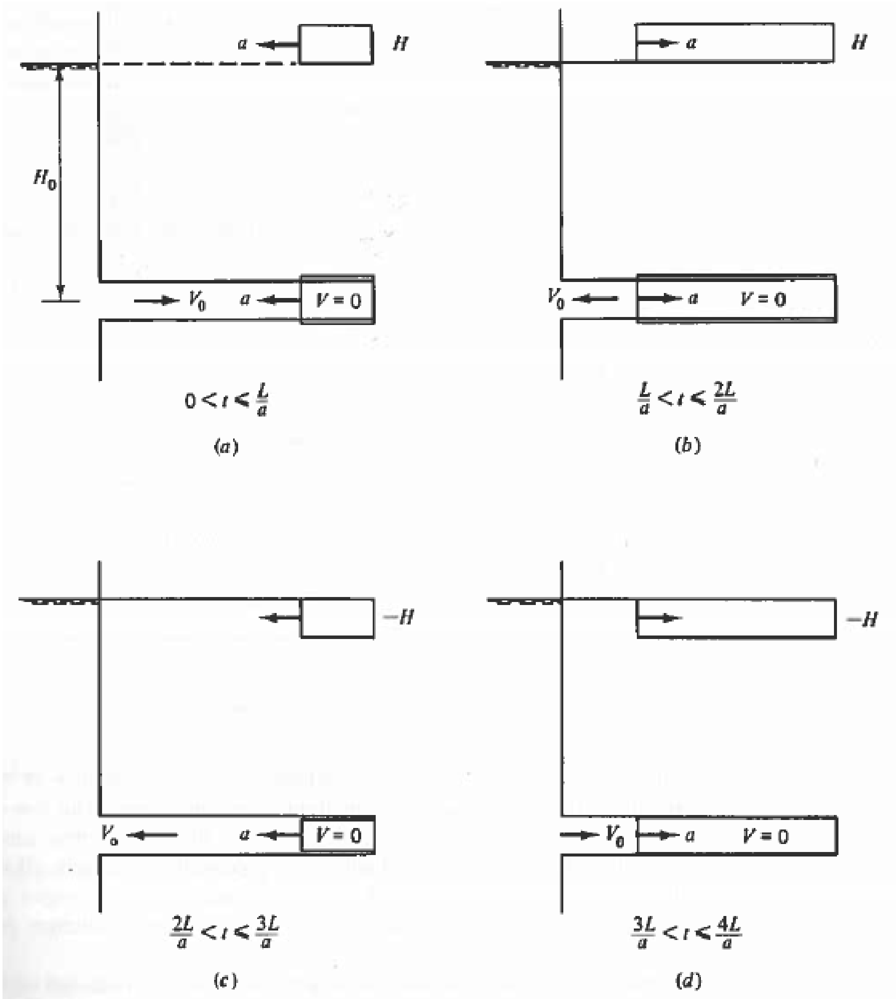


Figure 3.1.1: Sequence of events of one period after turbine shut down. Source: "Fluid Transients in Systems" [Wylie and Streeter, 1993]

3.1.2 Reducing the transient pressure

To avoid water hammer in a hydropower system the transient pressure during turbine shut down must be reduced. The transient pressure depends on two things; the closing time of the turbine, T_C, and the water inertia time constant, T_W. To reduce the transient pressure, the closing time must be increased or the water inertia time must be decreased.

The easiest way of reducing the transient pressure is to increase the closing time. However, if there is a load reduction on the power grid the turbine starts to accelerate and the rotational speed increases. To limit the increasing rotational speed the volume flow can not be reduced too slowly. For impulse turbines, such as a Pelton turbine, this is no problem as they are equipped with a deflector that interferes with the water jet and leads the water jet away from the turbine runner. Then the injector can be closed gradually and in that way reduce the transient pressure.

Reaction turbines do not have the same ability to lead the water away from the turbine runner. It is however possible to limit the increasing rotational speed by having a bypass valve that opens when the wicket gates are closing. Thus, the water flows past the turbine and the wicket gates can be closed gradually.

The other way to reduce the transient pressure is to decrease the water inertia time constant. This time constant is defined as the time it takes to accelerate the masses of water between the nearest free water surface upstream the turbine and the nearest free water surface downstream, from 0 to the rated volume flow Q_R. The water inertia time constant is defined as [Nielsen, 1990a]:

$$T_W = \frac{Q_R}{gH_R}\sum\frac{L}{A} \tag{3.1.1}$$

T_W is proportional to the length-area ratio of the pipeline between the free water surfaces. This ratio can be reduced by increasing the area of the pipeline, but that might be expensive. A more economical solution is to reduce the length of the pipeline by moving the free water surface nearer to the turbine. This means inserting a surge shaft between the reservoir and the turbine. For most of the high-head hydropower systems in Norway, systems with long pipelines, this is the chosen solution.

3.2 U-tube oscillations

Inserting a surge shaft between the reservoir and the turbine reduces the transient pressure, but it introduces another problem to be dealt with. A surge shaft is a dynamic element that causes mass oscillations between the free water surfaces. Mass oscillations occurs when the wicket gates either closes or opens due to load changes on the turbine. If there is a load rejection, the wicket gate will close and the water in the pressure shaft is decelerated. The masses of water in the head race tunnel is prevented from flowing into the pressure shaft and will flow into the surge shaft. The water level in the surge shaft will increase until the head is high enough to stop the water in the tunnel. Then the water level in the surge shaft will be higher than the reservoir level because of the inertia of the masses of water. The difference in head makes the water flow backwards from the surge shaft and into the reservoir. When the water flow stops, the water level in the surge shaft is lower than the reservoir level because of the inertia of the masses of water, and the water will flow into the surge shaft again. Thus, so called u-tube oscillations are created. This process will repeat itself until the oscillations are completely damped due to friction losses in the tunnel. Figure 3.2.1 shows the shaft surges due to load rejection.

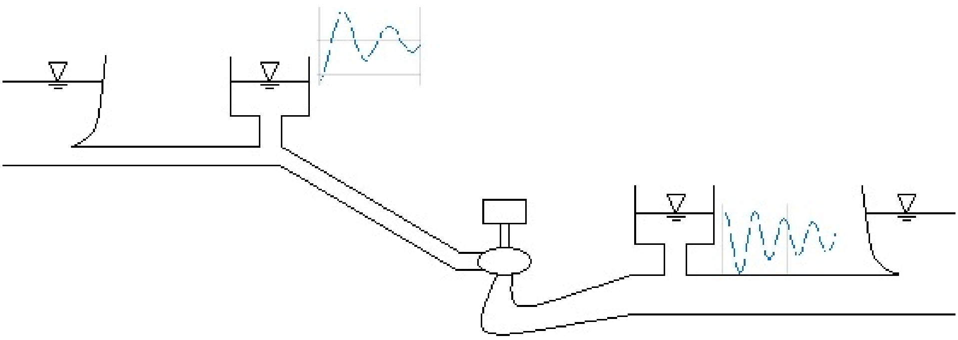


Figure 3.2.1: U-tube oscillations

As can be seen from the figure, the water level in the surge shaft downstream the turbine will have reverse oscillations. When the wicket gate closes, the water flow through the turbine will stop. The inertia of the masses of water makes the water in the tail race tunnel continue to flow into the reservoir, taking water from the surge shaft. The water level in the surge shaft drops until the head in the reservoir is high enough to stop the water in the tunnel. The reservoir level will now be higher than the water level in the surge shaft and the water starts flowing into the surge shaft.

The opposite happens when the wicket gates opens due to increased load, the oscillations will act oppositely and the upstream surge shaft will have a down surge at first, while the downstream surge shaft will have an up surge.

These u-tube oscillations can give problems if the amplitude of the surges becomes too large. If the water level in the surge shaft reaches the tunnel ceiling, free surface flow occur and the local velocity of the water increases. Thereby the mass transport increases and the turbine is exposed to more stress. If the water level drops below the tunnel ceiling, air is drawn into the system and can cause cavitation near the turbine. Likewise, if an up surge surpasses the ground level water will flow out into the nature and energy is lost. It is therefore necessary to calculate the maximum and minimum surge level in the surge shafts.

3.3 Stability criteria for U-tube oscillations

The U-tube between a surge shaft and the reservoir constitutes a spring-mass system that gives oscillations when the turbine opening degree is changed. For the spring-mass system to be stable the cross sectional area of the surge shaft must be larger than the Thoma cross section area. The Thoma cross section area is defined as [Nielsen, 1990a]:

$$A_{th} = 0.0085 \frac{M^2 A_{tunnel}^{5/3}}{H_e} \tag{3.3.1}$$

For the U-tube oscillations to be stable, the shaft area A_s must be greater than A_{th} and for safety the requirement is usually increased to ensure stability:

$$A_s > 1.5 A_{th} \tag{3.3.2}$$

The Thoma criteria assumes ideal governing, $Q \times H =$ constant through the turbine. The criteria yields for all free water surfaces in a system.

3.4 Basic differential equations

The basic differential equations for transient flow are described according to ¨Fluid Transients¨ [Wylie and Streeter, 1983].

3.4.1 The equation of motion and the continuity equation for pipes

The equation of motion for a pipe in terms of piezometric head and average velocity can be described as:

$$\frac{\partial H}{\partial x} + \frac{1}{g}\frac{\partial v}{\partial t} + f\frac{v|v|}{2gD} = 0 \tag{3.4.1}$$

The piezometric head, from now on referred to as head, is the sum of the hydraulic pressure and geostatic head, $H = h + z$. The first term of Eq. (3.4.1) is the change of head along the pipeline, the second term gives the velocity changes over time and the last term is the frictional loss.

The continuity equation for a pipe can be described as:

$$g\frac{\partial H}{\partial t} + a^2\frac{\partial v}{\partial x} = 0 \tag{3.4.2}$$

The elasticity of water is defined by the compression modulus, K, and is connected to the propagation speed by $a = \sqrt{\frac{K}{\rho}}$. The elasticity is included in the continuity equation through a^2, the propagation speed squared. [Wylie and Streeter, 1983]

3.4.2 Inelastic pipe and water

U-tube oscillations are very slow, the frequency is much lower than the frequency of a water hammer. Therefore the elastic effects are of insignificant importance for calculations of U-tube oscillations and it can be assumed inelastic pipes and water. For modeling and simulation of hydropower systems comprising inelastic pipes and water, the compression modulus goes toward infinity. For the equation of motion and the continuity equation for pipes, this means that the speed of sound goes toward infinity.

Dividing the equation of motion for a pipe by a^2 and let $a \to \infty$ gives $\frac{\partial v}{\partial x} = 0$ and thereby $Q = vA = constant$. With $\frac{\partial H}{\partial x} = \frac{H_2 - H_1}{L}$ the equation of motion for a pipe element comprising inelastic pipe and water becomes:

$$\frac{L}{gA}\frac{dQ}{dt} = H_1 - H_2 - \Delta h \tag{3.4.3}$$

where Δh is the steady state friction head loss.

3.4.3 Steady state friction head loss

The friction loss in Eq. (3.4.1) is based on head loss given by the Darcy-Weisbach equation:

$$\Delta h = \frac{f\Delta x v|v|}{2gD} \tag{3.4.4}$$

With $Q = vA$, assuming inelastic pipes, and $\Delta x = L$, the head loss can be rewritten using the volume flow as the state variable:

$$\Delta h = \frac{fL}{2gA^2D}Q|Q| \tag{3.4.5}$$

It can be seen that the head loss takes the same form as the steady state turbulent loss, $\Delta h = kQ^2$, and it is assumed that the frictional loss has the same characteristics as steady state turbulent loss. In non-stationary operation the velocity might change direction during the performance and the head loss will always work against the movement, no matter flow direction. This is expressed by the absolute value of Q.

This formulation of the loss is suitable for the calculation models used in this thesis, but it is not a perfect formulation. The formula assumes a fully developed turbulent velocity profile for all operation points. This means that the velocity maintains a turbulent profile when it change direction, which is unlikely. In reality this is only a good assumption when the flow is well turbulent in an operating point, when it is small fluctuations around a large flow. [Nielsen, 1990a]

Another objection is that this loss formulation do not count for the surging frequency, which has shown to have an impact on the loss. However, in state space calculations, the frequency is impossible to include. [Nielsen, 1990a]

In general, the steady state loss formulation has sufficient compliance with oscillations around an operating point where the steady state flow is large. For most purposes the friction is modeled sufficient enough. For oscillations around zero flow however, the formulation provides far too low, or almost no dampening. [Nielsen, 1990a]

3.4.4 Shaft surging

The surge shaft may be treated as a simple reservoir if it has a large cross sectional area and low velocities. Then the inertial forces and the frictional losses are neglected and it is assumed a hydrostatic pressure variation. [Wylie and Streeter, 1983]

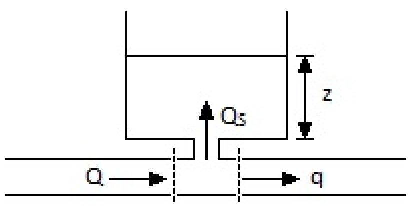


Figure 3.4.1: Surge shaft

A principle drawing of a surge shaft is shown in Fig. 3.4.1. The volume flow into the surge shaft, Q_S, is a product of the velocity, given by the change in water level in the surge shaft, and the cross sectional area:

$$Q_S = \frac{dz}{dt} A_S \tag{3.4.6}$$

The continuity equation in the branch is:

$$Q_S = Q + q \tag{3.4.7}$$

Eq. (3.4.6) and Eq. (3.4.7) combined gives the differential equation for shaft surging:

$$\frac{dz}{dt} = \frac{(Q - q)}{A_S} \tag{3.4.8}$$

3.4.5 Differential equations describing the turbine and rotating masses

The turbine converts hydraulic energy to rotating energy, shown in Fig.3.4.2 with a power balance over the turbine and the generator.

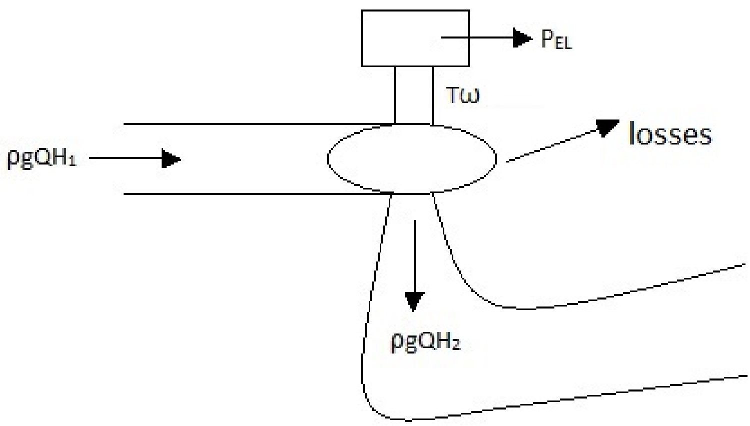


Figure 3.4.2: Power balance over turbine and generator

The hydraulic power goes into acceleration of the rotating masses, electric power to the grid and losses:

$$P_h = \rho g Q(H_1 - H_2) = T\omega + P_{EL} + losses \tag{3.4.9}$$

Newtons second law for a rotating system, also known as the angular momentum, states that the net external torque is equal to the polar moment of inertia multiplied with the angular acceleration, the angular momentum equation can be described as [Cengel and Cimbala, 2010]:

$$T = I_p \frac{d\omega}{dt} \tag{3.4.10}$$

where T is the torque and I_p is the polar moment of inertia. I_p can be expressed with the acceleration time of the rotating masses, T_a, the maximum power output, P_{max}, and the angular speed of the turbine at full load, ω_0, as [Nielsen, 1990a]:

$$I_p = \frac{T_a P_{max}}{\omega_0} \tag{3.4.11}$$

The differential equation describing the turbine and the rotating masses will then be:

$$I_p \omega \frac{d\omega}{dt} = P_h - P_{EL} - losses \tag{3.4.12}$$

For a turbine implemented in a system, the linear momentum equation is needed to describe the flow through the turbine. Newtons second law for a system of mass m subjected to the net force $\sum F$, gives the linear momentum equation [Cengel and Cimbala, 2010]:

$$\sum F = m\frac{dv}{dt} = \rho A L \frac{dQ}{dt}\frac{1}{A} \tag{3.4.13}$$

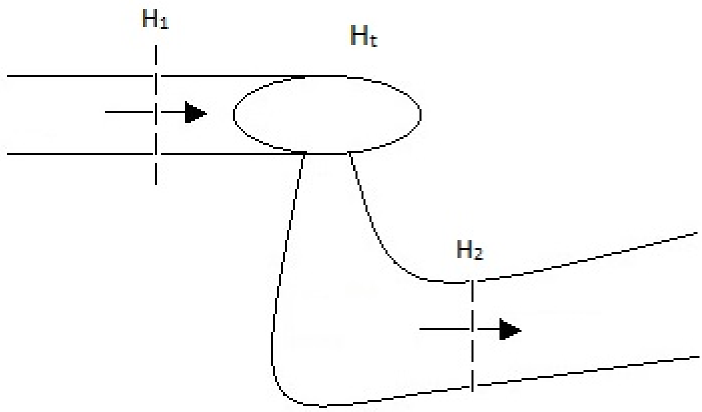


Figure 3.4.3: The head over a turbine

With reference to Fig.3.4.3, the net force acting on the system is:

$$\sum F = \rho g(H_1 - H_2 - H_t)A \tag{3.4.14}$$

Substituting Eq.(3.4.14) into Eq.(3.4.13) and adding losses gives the differential equation describing the flow through a turbine:

$$I_h \frac{dQ}{dt} = g(H_1 - H_2) - gH_t - losses \tag{3.4.15}$$

where I_h is the hydraulic inertia through the turbine and H_t is the head over the turbine.

3.5 Turbines

3.5.1 Turbine opening degree

The volume flow in a hydropower system is defined by the opening degree of the turbine. Normally, the change in volume flow is decided from an efficiency diagram. In this case, however, the only phenomena studied is the u-tube oscillations and it can be assumed that the turbine acts as a valve with an varying opening degree, κ, given by the dimensionless valve equation [Wylie and Streeter, 1983]:

$$\kappa = \frac{Q}{Q_R} \frac{\sqrt{2gH_R}}{\sqrt{2gH}} \tag{3.5.1}$$

where Q_R and H_R is the rated volume flow and the rated head. H is the pressure difference over the turbine.

3.5.2 One dimensional Euler turbine equation

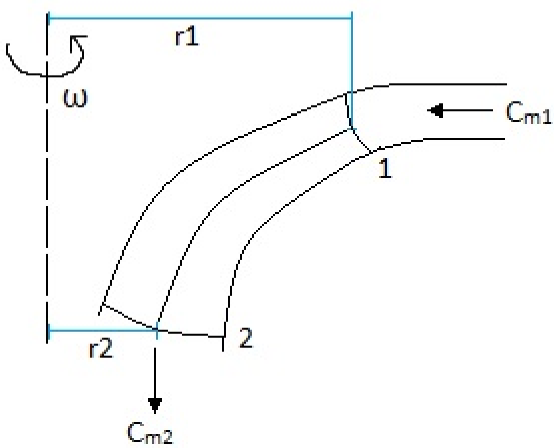


Figure 3.5.1: Axial cut of turbine runner: main geometry at inlet and outlet

By looking at the main geometry at inlet and outlet of a reaction turbine, Fig. 3.5.1, and applying Bernoulli's equation on the stream-line between inlet and outlet, the difference in hydraulic energy between the inlet and outlet of the turbine runner can be expressed as:

$$gH_t = u_1 c_{u1} - u_2 c_{u2} \tag{3.5.2}$$

H_t is the head difference between the inlet and outlet of the runner. $u = \omega r$ is the velocity of the runner blade and c is the water's absolute velocity, which will be explained further in the next section.

The euler turbine equation expresses how the hydraulic energy is transformed to mechanical energy through the turbine runner. By multiplying Eq.(3.5.2) with the mass flow and set $u = \omega r$, the Euler turbine equation is formed:

$$\rho g Q H_t = \rho Q (r_1 c_{u1} - r_2 c_{u2})\omega = T\omega \tag{3.5.3}$$

It can be seen that the expression on the left hand side is the hydraulic power and the expression on the right hand side is the rotational power.

3.5.3 Velocity diagrams for a Francis turbine runner

The velocity diagram at inlet and outlet of a Francis runner is shown in Fig. 3.5.2. The velocity diagram at the inlet, denoted 1, is seen from above looking down on the runner and the velocity diagram at the outlet, denoted 2, is seen from the side of the runner.

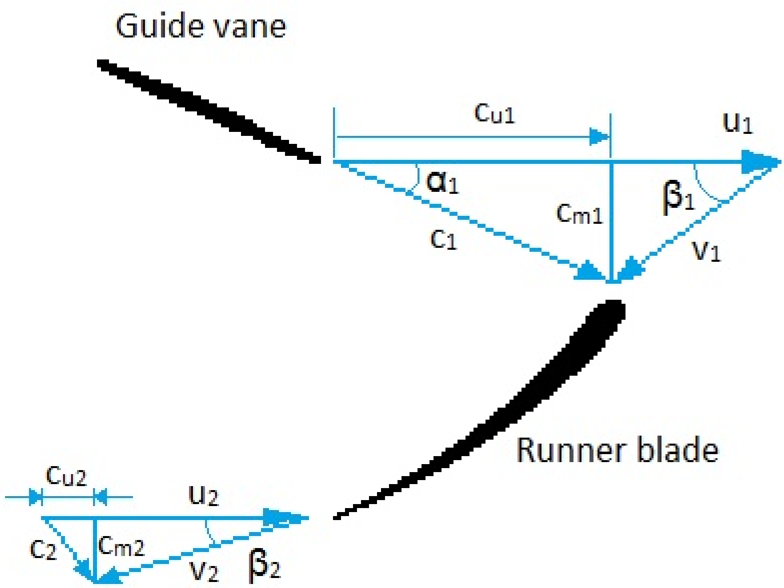


Figure 3.5.2: Velocity diagrams at inlet(1) and outlet(2) of a Francis runner

In the figure:

u	-	peripheral velocity
c	-	absolute velocity of water
c_{m1}	-	component of absolute velocity in direction perpendicular to the runner inlet cross section
c_{m2}	-	component of absolute velocity in direction perpendicular to the runner outlet cross section
c_u	-	component of absolute velocity in peripheral direction
v	-	relative velocity
α_1	-	inlet guide vane angle
β_1	-	inlet relative angle
β_2	-	outlet blade angle

At the design point, also called the Best Efficiency Point, BEP, the runner angles are perfect for transforming the hydraulic power to rotating mechanical power. The inlet relative angle is equal to the runner vane angle $\beta_1 = \beta_{1r}$. At the outlet the water will flow in direction perpendicular to the runner outlet cross sectional area, in other words $c_2 = c_{m2}$, this means hat there is no flow in peripheral direction, $c_{u2} = 0$.

3.5.4 The momentun equation for a reaction turbine

In section 3.4.5, the differential equation describing the flow through a turbine was given as:

$$I_h \frac{dQ}{dt} = g(H_1 - H_2) - gH_t - losses \tag{3.5.4}$$

The head difference between the inlet and outlet of the runner is given by Eq.(3.5.2). Applying the Cosines sentence on the inlet and outlet velocity diagrams and implementing this in the equation gives:

$$gH_t = \frac{1}{2}(c_1^2 - c_2^2) - \frac{1}{2}(v_1^2 - v_2^2) + s\omega^2 \tag{3.5.5}$$

where

$$s = \frac{1}{8}D_1^2(1 - \frac{D_2^2}{D_1^2}) \tag{3.5.6}$$

A derivation of Eq.(3.5.5) and Eq.(3.5.6) is given in Appendix A. Solving the equation describing the turbine opening degree, Eq.(3.5.1), with respect to the head gives:

$$H = H_R\Big(\frac{Q}{\kappa Q_R}\Big)^2 \tag{3.5.7}$$

which at the design point is the head difference between the inlet and outlet of the runner:

$$H_R\Big(\frac{Q}{\kappa Q_R}\Big)^2 = \frac{1}{2}(c_{1R}^2 - c_{2R}^2) - \frac{1}{2}(v_{1R}^2 - v_{2R}^2) + s\omega_R^2 \tag{3.5.8}$$

or

$$\frac{1}{2}(c_{1R}^2 - c_{2R}^2) - \frac{1}{2}(v_{1R}^2 - v_{2R}^2) = H_R\Big(\frac{Q}{\kappa Q_R}\Big)^2 - s\omega_R^2 \tag{3.5.9}$$

The head difference between the inlet and outlet of a reaction turbine runner may then be expressed by implementing Eq.(3.5.9) in Eq.(3.5.5):

$$gH_t = gH_R\Big(\frac{Q}{\kappa Q_R}\Big)^2 + s(\omega^2 - \omega_R^2) \tag{3.5.10}$$

The last term on the right hand side of Eq.(3.5.10) represents the self-governing of a turbine. The turbine represents a valve in the system and it is a function of the angular speed of rotation, defined by the turbine geometry s. For a Low head Francis turbine, the inlet diameter of the runner is smaller than the outlet diameter, s is positive, and the flow will increase as the rotational speed increases. For a High head Francis turbine the inlet diameter is larger than the outlet diameter, s is negative, and the flow will decrease as the rotational speed increases.

Neglecting the hydraulic losses through the turbine, the momentum equation for a reaction turbine can be expressed as:

$$I_h\frac{dQ}{dt} = g(H_1 - H_2) - gH_R\Big(\frac{Q}{\kappa Q_R}\Big)^2 - s(\omega^2 - \omega_R^2) \tag{3.5.11}$$

3.5.5 The torque equation for a reaction turbine

From the differential equation describing the turbine and rotating masses, given in section 3.4.5, the torque equation can be formed. By dividing both side of Eq.(3.4.12) with the angular velocity and neglecting the losses the torque equation can be described as:

$$I_p\frac{d\omega}{dt} = T_t - T_g \tag{3.5.12}$$

The shaft torque on a turbine is equal to the change in moment of momentum from the turbine inlet to the turbine outlet and is defined by the Euler turbine equation:

$$T_t = \rho Q(r_1 c_{u1} - r_2 c_{u2}) \tag{3.5.13}$$

By examining the velocity diagrams in Fig.3.5.2, the torque may be expressed as [Nielsen, 1990b]:

$$T_t = \rho Q(r_1 c_1 cos\alpha_1 + r_2 A_z c_1 sin\alpha_1 cot\beta_2 - r_2^2 \omega) \tag{3.5.14}$$

where A_z is the ratio of the runner outlet and inlet cross sectional area.

From the velocity diagram at the outlet of the runner, it can be seen that $c_1 = \frac{c_{m1}}{sin\alpha_1}$ and the component of absolute velocity in meridional direction, c_{m1}, must be equal to the flow through the inlet cross sectional area divided by the area, $c_{m1} = \frac{Q}{A_1}$. If this is included in Eq. (3.5.14), the torque can be rewritten using the volume flow as the state variable:

$$T_t = \rho Q(\frac{r_1}{A_1} Q cot\alpha_1 + \frac{r_2}{A_2} Q cot\beta_2 - r_2^2 \omega) \tag{3.5.15}$$

The start torque is found when there is no rotation of the runner. With $\omega = 0$, Eq.(3.5.15) gives:

$$t_s = \rho Q \frac{r_1}{A_1} Q cot\alpha_1 + \frac{r_2}{A_2} Q cot\beta_2 \tag{3.5.16}$$

where t_s is the start torque.

3.5.6 Generator

For dynamic simulation of hydropower systems only the generator torque's dependency of the electrical load is of interest. Therefore, it is sufficient with a simplified generator model for these types of simulations. In this thesis, a generator model suggested in "Dynamic Behaviour of governing turbines sharing the same electrical grid" by Professor Torbjørn K. Nielsen, has been used. [Nielsen, 1996]

The relation between generator torque and the grid frequency can be described by the angle between the generator stator and rotor in a rotating reference frame. The angle is related to the angular grid frequency [Fitzgerald et al., 2003]:

$$\omega_{grid} = 2\pi f_{grid} \tag{3.5.17}$$

When synchronous generators are connected to the grid, their speed is determined by the grid frequency. The synchronous angular speed of rotation for a generator is a function of the number of generator poles, P [Nielsen, 1996]:

$$\omega_s = \frac{2}{P}\omega_{grid} \tag{3.5.18}$$

In steady state operation, the angular speed of rotation of the turbine, ω_t, must be equal to the synchronous speed. During a load change the change in angle between the generator stator and rotor can be described by the following differential equation [Nielsen, 1996]:

$$\frac{d\delta}{dt} = \frac{P}{2}\omega_t - \omega_{grid} \tag{3.5.19}$$

When the electrical load increases, the grid frequency decreases which gives a positive change of δ, thus the angle will increase. When the electrical load decreases, the grid frequency increases and the angle will decrease.

With good approximations the generator torque can be modeled as a sinus function of the angle, δ, according to [Nielsen, 1996]:

$$T_g = T_{gR}\frac{\sin\delta}{\sin\delta_R} \tag{3.5.20}$$

where subscript R denotes the rated values.

A damping of the angular movement must be introduced in the torque equation:

$$I_p\frac{d\omega}{dt} = \rho Q(t_s - r_2^2\omega) - T_g - m_d\frac{d\delta}{dt} \tag{3.5.21}$$

where m_d is the angular movement damping coefficient.

3.5.7 Grid mode and Island mode

A hydropower plant is either connected to an isolated grid or to the joint grid. If the plant is connected to an isolated grid, it is running in Island mode. Then the grid frequency is determined by the hydropower plants connected to the grid. If the plant is connected to the joint grid, it is running in grid mode. In grid mode, each turbine connected to the grid sees the grid frequency as fixed. All hydropower plants connected to the grid has synchronous generators. The turbine rotational speed is then determined by the grid frequency.

For a reaction turbine, this means that the angular grid frequency in Eq.(3.5.19) is either constant or it is a variable that depends on the output voltage. When transferring mechanical rotating power to electrical power, the output voltage is primarily a function of the angular grid frequency and the magnetic flux, $k\Phi$. The angular grid frequency can be expressed as [Nielsen, 1996]:

$$\omega_{grid} = \frac{E}{k\Phi} \tag{3.5.22}$$

According to Ohm's law the voltage is equal to the product of the electric current and the resistance on the grid, $E = R_{grid}I$. The generator torque is a function of the electric current, which can be described as [Nielsen, 1996]:

$$I = \frac{T_g}{k\Phi \cos\phi} \tag{3.5.23}$$

where ϕ is the phase angle which is dependent on the property of the grid. The output voltage can then be expressed as:

$$E = R_{grid}\frac{T_g}{k\Phi \cos\phi} \tag{3.5.24}$$

and when a hydropower plant is running in Island mode, the change in angle between the generator stator and rotor is described by the following differential equation:

$$\frac{d\delta}{dt} = \frac{P}{2}\omega_t - R_{grid}\frac{T_g}{(k\Phi)^2 \cos\phi} \tag{3.5.25}$$

The change in output voltage can be expressed with forward difference:

$$\frac{dE}{dt} = E^{n+1} - E^n \tag{3.5.26}$$

3.6 Governors

3.6.1 Power governing

A hydropower plant delivers electric power to a grid, and together with all hydropower plants connected to the same grid, they shall ensure that the power demand of the consumers are covered at all time. It is also demanded that the turbine keeps a synchronous speed of rotation so that the grid frequency, 50 Hz in the Nordic power grid, is maintained. The governor is implemented in a hydropower system to make sure that there is a balance between the delivered power and the power demand. If the power demand is increasing, there will be a deficit of hydraulic power, which decelerates the rotational masses and the speed of rotation decreases. The governor will sense a speed deviation from synchronous speed and responds by changing the wicket gate position so that the volume flow is changed and the balance between the delivered power and the power demand is obtained. The governor will act in the same way if the power demand decreases. Then there will be a surplus of the hydraulic power and the rotational masses are accelerated. The speed of rotation increases and when the governor senses the speed deviation from synchronous speed, it changes the flow by closing the wicket gates until the hydraulic power and the power demand is balanced. [Nielsen, 1990a]

To make sure that the rotational speed of the turbine is equal to the synchronous speed at any time, the governor acts on a deviation from the reference speed of rotation. The block diagram describing the governing process is shown in Fig.3.6.1.

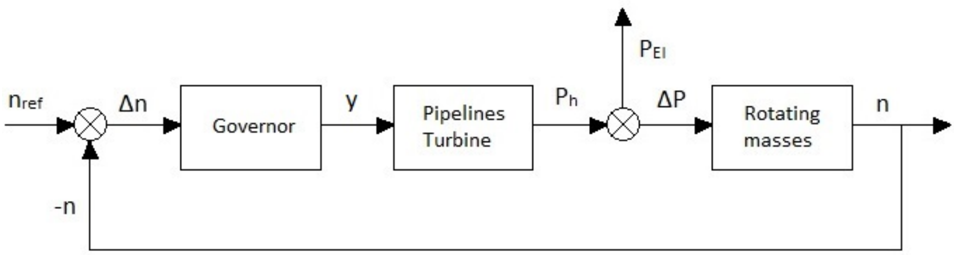


Figure 3.6.1: Block diagram for a governing system

The hydraulic power is transformed to mechanical rotating power which is transformed to electrical power by the generator. If the electrical power demand decreases or increases, $\Delta P \neq 0$, the speed will increase or decrease according to the angular momentum equation. The speed of rotation is measured by the governor and compared with the reference speed of rotation. The deviation causes the governor to move the wicket gate position in either a closing direction or an opening direction. Then the hydraulic power is decreased or increased and a power balance is obtained.

At first, the turbine will respond opposite to the governors control signals. If the wicket gates are moved in closing direction, the masses of water in the pressure shaft will decelerate and the pressure in front of the turbine will increase for a short period of time. Then the velocity of the water going through the vanes increases and the power output will increase. The opposite happens if the wicket gates are moved in opening direction, the power output will decrease. This will only last for a short time and the effect is usually not observable on Francis and Pelton turbines. For other turbine types, such as a Kaplan turbine, this effect is more noticeable.

3.6.2 Types of governors

The different governor types are described according to "Reguleringsteknikk" [Balchen et al., 2003].

There are mainly four types of governors, P-, PI-, PD- and PID-governors. The P-governor, proportional governor, is the simplest version and it measures the speed of rotation and changes the wicket gate position proportional to the deviation between the measured speed of rotation and the reference speed of rotation:

$$\frac{dy}{dt} = -K_p \frac{dn}{dt} \tag{3.6.1}$$

Often this governor gives a stationary deviation from the reference speed of rotation, as it changes the wicket gate position until the speed becomes constant. The stationary deviation is dependent on the proportional constant, K_p. There will be less deviation if K_p is large, but this will lead to poorer stability characteristics.

By including an integrator term, PI-governor, the stationary deviation will be removed. The output of a PI-governor equals the integrated input signal and there is no sudden reaction on the output signal when the input signal is changed quickly. The speed of rotation will be adjusted until there is no deviation from the reference speed. The PI-governor can be described with the following differential equation:

$$\frac{dy}{dt} = -K_p \frac{dn}{dt} + \frac{K_p}{T_i}(n_{ref} - n) \tag{3.6.2}$$

The first term is the proportional action and the second term changes the wicket gate position until the measured speed of rotation equals the reference speed. The problem with the integrator term is that it aggravates the system stability, as it gives a negative contribution to the phase. However, the proportional term acts much faster than the integrator term and for reasonable T_i-values the PI-governor will achieve stable governing.

The PD- and PID-governor have a derivative term, which gives fast governing and it improves the stability as it has a positive contribution to the phase. The differential equation for a PID-governor is:

$$\frac{dy}{dt} = -K_p\frac{dn}{dt} + \frac{K_p}{T_i}(n_{ref} - n) - K_pT_d\frac{d^2n}{dt^2} \tag{3.6.3}$$

With the derivative term, the last term, the change in deviation is taken into account when the output signal is computed.

3.6.3 Permanent speed droop and governor parameters

There are two types of speed droop, transient and permanent. The transient speed droop represents the proportional action of the governor. The transient speed droop is represented by:

$$b_t = \frac{1}{K_pn_{ref}} \tag{3.6.4}$$

The turbine governor has a yielding resetting that makes the frequency stationary dependent on the load. This resetting is adjustable on the governor and it is determined by the permanent speed droop. Dependent on the load, the permanent speed droop allows a certain frequency deviation. All synchronous generators connected to the same grid must have the same stationary frequency. If several power plants supply the same grid, the permanent speed droop decides how the total load change will be distributed to each of the connected generators.

There are three governing parameters for a general turbine PI-governor, the integral gain, T_i, the transient speed droop, b_t, and the permanent speed droop, b_p. As can be seen in section3.6.2, the governors have a proportional constant, K_p. This is included in the transient droop as described in Eq.(3.6.4). The PI-governor expressed with transient droop instead of the proportional constant is:

$$\frac{dy}{dt} = -\frac{1}{b_tn_{ref}}\frac{dn}{dt} + \frac{1}{b_tT_in_{ref}}(n_{ref} - n) - \frac{b_p}{b_tT_i}y \tag{3.6.5}$$

The permanent speed droop is included as the last term.

The governing parameters must be tuned according the dynamic behaviour of the whole hydropower system. As a starting point, the parameters can be set according to Stein's empirical formulas [Nielsen, 1990a]:

$$T_i = 6T_w \tag{3.6.6}$$

$$b_t = 2.6\frac{T_w}{T_a} \tag{3.6.7}$$

Dependent on the system requirements, the permanent speed droop can be adjusted within 0.00 - 0.06 [Nielsen, 1990a].

3.6.4 Governor model

For the governor model used in this thesis, the PI-governor is found to be sufficient. The frequency governor used in the developed Matlab program, *uTubeOscV2*, is based on a governor model presented by Torbjørn K. Nielsen in the paper "Dynamic behaviour of governing turbines sharing the same grid" [Nielsen, 1996]. The frequency governor is described by the two following differential equations:

$$\frac{dy}{dt} = c \tag{3.6.8}$$

$$\frac{dc}{dt} = \frac{y_{ref}}{T_K}\left[-\frac{1}{b_t n_{ref}}\frac{dn}{dt} + \frac{1}{b_t T_i}\frac{(n_{ref}-n)}{n_{ref}} - \frac{b_p T_K + b_t Ti}{b_t Ti}c - \frac{b_p}{b_t T_i}(y_{ref}-y)\right] \tag{3.6.9}$$

where y is the servo motor position and c is the servo motor velocity. The equations are derived from the transfer function for a PI-governor with permanent speed droop and servo motor time constant. The transfer function is transformed to time domain and hence Eq.(3.6.9) has an additional term compared to the PI-governor described in section 3.6.2. The second last term on the right hand side of Eq.(3.6.9) is a direct consequence of the transformation from frequency domain to time domain. The first term on the right hand side is the proportional action, the second term is the integrator term and the last term gives the contribution to the total power change on a grid, the permanent speed droop.

The servo motor position is proportional to the turbine opening degree as shown in Fig. 3.6.2.

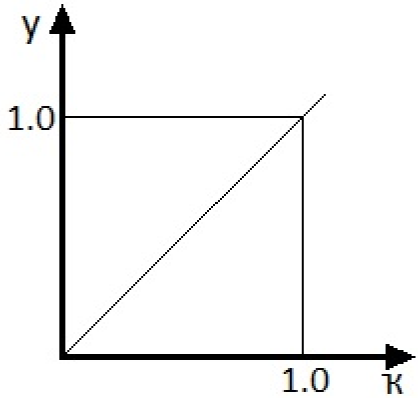


Figure 3.6.2: Servo motor position related to the opening degree

The differential equations describing the frequency governor can then be expressed with the turbine opening degree instead of the servo motor position, by replacing y with κ directly.

3.7 Pelton turbine model

Solving Eq. (3.5.1) with respect to the head difference over the turbine gives:

$$H_t = H_R \left(\frac{Q}{\kappa Q_R} \right)^2 \tag{3.7.1}$$

The flow change through a Pelton turbine is controlled by an injection needle, acting as a valve. Therefore, Eq.(3.7.1) is sufficient for simulations of hydropower systems comprising Pelton turbines. The head difference over the turbine included in the momentum equation describes a Pelton turbine element:

$$I_h \frac{dQ}{dt} = H_1 - H_2 - H_R \left(\frac{Q}{\kappa Q_R} \right)^2 \tag{3.7.2}$$

3.8 Francis turbine models

In reaction turbines, such as a Francis turbine, the flow transients is also influenced by the speed of rotation and it is therefore necessary to include the torque equation and expand the momentum equation to cover self-governing due to changes in the speed of rotation. The torque equation and the momentum equation for a reaction turbine was described in section 3.5. When the torque equation is included, the varying generator torque gives another differential equation that must be included in the Francis turbine model. The change in generator torque is expressed by the change in the angle between the generator stator and rotor, Eq.(3.5.19). This gives three differential equations that describes a Francis turbine element:

$$I_h \frac{dQ}{dt} = g(H_1 - H_2) - gH_R \Big(\frac{Q}{\kappa Q_R}\Big)^2 - s(\omega^2 - \omega_R^2) \tag{3.8.1}$$

$$I_p \frac{d\omega}{dt} = \rho Q(t_s - r_2^2 \omega) - T_g - m_d \frac{d\delta}{dt} \tag{3.8.2}$$

$$\frac{d\delta}{dt} = \frac{P}{2}\omega_t - \omega_{grid} \tag{3.8.3}$$

where the generator torque is given by:

$$T_g = T_{gR} \frac{\sin \delta}{\sin \delta_R} \tag{3.8.4}$$

and the angular grid frequency, ω_{grid} is either constant or given by Eq.(3.8.5), dependent on whether the turbine is running in grid mode or Island mode.

$$\omega_{grid} = R_{grid} \frac{T_g}{(k\Phi)^2 \cos \phi} \tag{3.8.5}$$

For the developed analysis program uTubeOscV1, three different Francis turbine models is implemented. Dependent on whether the user will simulate manual change of the opening degree or a change in power demand on the grid, there are different turbine models for each case. The last turbine model presented, is for Francis turbines running in Island mode. This model is not in use, but is included in the program for further development.

3.8.1 Simple Francis turbine, grid mode

For a hydropower plant running in grid mode, simulation of manual change on the opening degree requires only the momentum equation, the torque equation and the differential equation describing the change in generator torque, when the turbine is of Francis type. The turbine element is then described by Eq.(3.8.1), Eq.(3.8.2) and Eq.(3.8.3). The angular grid frequency is then constant.

3.8.2 Francis turbine with frequency governor, grid mode

When the hydropower plant is running in grid mode and it is desired to simulate U-tube oscillations due to changes in power demand on the grid, the frequency governor must be included in the model. Then the turbine element is described by Eq.(3.8.1), Eq.(3.8.2), Eq.(3.8.3) and the differential equations for the frequency governor, Eq.(3.6.8) and Eq.(3.6.9). A change in power demand on the grid will give an increased or decreased grid frequency. The simulations are done by changing the constant grid frequency to the grid frequency that reflects the change in power demand.

3.8.3 Francis turbine with frequency governor, Island mode

When the simulated hydropower plant is running in Island mode, the angular grid frequency is no longer constant and is given by Eq.(3.8.5). The change in voltage must then be included in the model. The turbine element is then described by Eq.(3.5.26), in addition to the differential equations describing a Francis turbine with frequency governor running in grid mode. This turbine model is implemented in uTubeOscV1 for further development, but is not in use.

4 Calculation methods

4.1 Analytical calculations

The up-surge and down-surge from the steady state level in the surge shafts, due to turbine shut down and start up, can be calculated with estimate formulas. In 1972, Dr. R. Svee presented two estimate formulas that include the effect of head loss [Svee, 1972]. Due to turbine shut down the up-surge from steady level in the shaft was presented as:

$$\Delta z_{up} = \Delta Q \sqrt{\frac{L/A_{tunnel}}{gA_s}} + \frac{1}{3}h_f \tag{4.1.1}$$

where h_f is the head loss in the tunnel at steady state before turbine shut down.

The down surge from steady state level in the shaft due to turbine start up was presented as:

$$\Delta z_{down} = -\Delta Q \sqrt{\frac{L/A_{tunnel}}{gA_s}} - \frac{1}{9}h_f \tag{4.1.2}$$

where h_f is the head loss at steady state after turbine start up.

The natural frequency and the time period of the U-tube oscillations can be determined by looking at the U-tube between a reservoir and a surge shaft as a spring-mass system. Without losses, the equation of motion for the U-tube is:

$$\frac{L_{tunnel}}{gA_{tunnel}}\frac{dQ}{dt} = \Delta z \tag{4.1.3}$$

Taking the derivative of the differential equation for shaft surging, Eq.(3.4.8), gives;

$$\frac{1}{A_s}\frac{dQ}{dt} = \frac{d^2z}{dt^2} \tag{4.1.4}$$

Eq.(4.1.4) substituted in Eq.(4.1.3) gives the following equation, which is on the form describing a spring-mass system:

$$\frac{L_{tunnel}}{gA_{tunnel}}\frac{d^2z}{dt^2} - \frac{1}{A_s}\Delta z \tag{4.1.5}$$

For a spring-mass system $m\frac{d^2x}{dt^2} - kx = 0$ the solution of natural frequency is well known as $\omega = \sqrt{\frac{k}{m}}$ [Young et al., 2004]. The natural frequency of the U-tube system will then be:

$$\omega = \sqrt{\frac{gA_{tunnel}}{LA_s}} \tag{4.1.6}$$

and its time period is:

$$T = \frac{2\pi}{\omega} \tag{4.1.7}$$

4.2 Numerical method

4.2.1 1D-modeling

When calculating U-tube oscillations numerically, the dynamic behavior of the whole system must be taken into account. This can be done by one dimensional modeling of hydropower systems, using tube elements comprising inelastic pipes and water. The dynamics of a given hydropower system is then defined by the differential equations describing the elements that makes the system. The modeling is based on three element types; pipe, surge shaft and turbine. These elements are described according to the differential equations given in section 3. A pipe element is described by Eq.(4.2.1), which defines the volume flow through the element.

$$\frac{L}{gA}\frac{dQ}{dt} = H_1 - H_2 - \frac{fL}{2gA^2D}Q|Q| \tag{4.2.1}$$

The equation describing a surge shaft element defines the water level in the surge shaft. It is assumed mass-less shafts, which means that the head in the nodes attached to the element is equal to the water level in the shaft. Eq.(4.2.2) describes a surge shaft element.

$$\frac{dz}{dt} = \frac{(Q-q)}{A_S} \tag{4.2.2}$$

Depending on which turbine model used, the turbine element is described by one or several equations. For a Pelton turbine, only one equation is needed to define the volume flow through the element and the element is described according to the Pelton model presented in section 3.7. To define the volume flow through a Francis turbine, several equations are needed to describe the dynamic behaviour of the turbine. The Francis turbine elements is described according to the Francis turbine models presented in section 3.8.

Solving a dynamic system such as a hydropower system directly with a numerical solver, requires all variables to be described with a differential equation. For a hydropower system modeled with tube elements comprising inelastic pipes and water, there are three variables; volume flow, Q, water level in surge shafts, z, and head, H. The differential equations that defines the volume flow through each element is connected by the head in the nodes and branches between the different elements. As the head at nodes and branches are not described by a differential equation for pipe-pipe connections and pipe-turbine connections, the varying head must be described as a function of the volume flow and the head given by the elements connected to the node or the branch. For a generic analysis program that can model an arbitrary hydropower system, such as uTubeOscV1, it requires so many loops and if-/else-statements that there has to be some limitations in the system geometry. To avoid limitations, the numerical loops must be avoided. A well known solution method to numerical loops for systems of spring-mass behavior, is to introduce a virtual inertance [Brown, 2001]. In the case of one dimensional modeling of hydropower systems, this means to insert "virtual surge shafts".

4.2.2 Virtual surge shaft

A "virtual surge shaft", from now on referred to as VSS, is a surge shaft element with a significantly small cross sectional area that acts like a narrow tube with a varying water level. Inserting a VSS in nodes and branches allows the head to be described by a differential equation, when assuming mass-less shaft. If the use of VSS is a feasible solution, then one dimensional modeling can be used for modeling of any systems, regardless of complexity and composition. This will give more correct models of any real system, as there will be no need for simplifications.

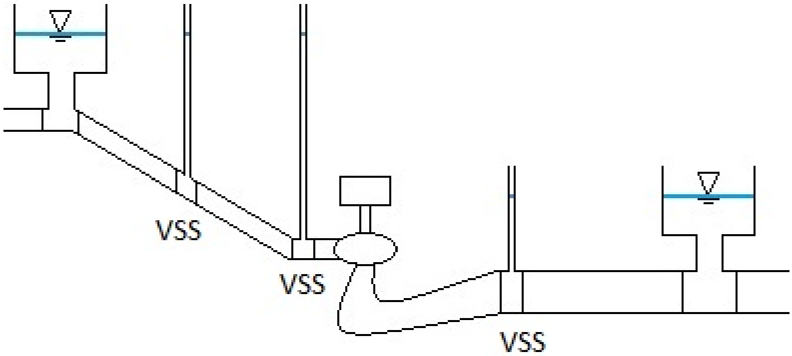


Figure 4.2.1: Virtual surge shafts

As shown in Fig. 4.2.1, the VSS is placed in between pipes and in pipe-turbine connections. The water level in the VSSs varies as the volume flow changes, in the same way as the water level in the surge shafts. The immediate thought might be that this will cause U-tube oscillations between the surge shafts and the VSSs. However, if the cross sectional area is small enough, the oscillations in the VSS will not have a significant impact on the volume flow in the system and affect the shaft surging. As the VSS contains small masses of water, the frequency of the oscillations becomes high. Since U-tube oscillations are very slow, the water level in the VSSs should fluctuate around the water level in the surge shafts without affecting the U-tube oscillations.

The stability criteria for U-tube oscillations states that the Thoma criteria applies to all free surfaces in a system. With a significantly small cross sectional area, the masses of water in the VSSs will be of such small amount that there will be no U-tube oscillations between the surge shafts and the VSSs.

The differential equation describing a VSS is the same equation that describes a surge shaft. For a node, pipe-pipe connection or pipe-turbine connection, the differential equation is:

$$\frac{dz_{node}}{dt} = \frac{(Q_{in} - Q_{out})}{A_{VSS}} \tag{4.2.3}$$

where Q_{in} is the volume flow through the first element and Q_{out} is the volume flow through the second element. For a branch or bound Q_{in} and Q_{out} will be the sum of the volume flow entering and leaving the branch or bound. The differential equation describing a branch will then be:

$$\frac{dz_{branch}}{dt} = \frac{(Q_{in} - \sum Q_{out})}{A_{VSS}} \tag{4.2.4}$$

and the differential equation describing a bound will be:

$$\frac{dz_{bound}}{dt} = \frac{(\sum Q_{in} - Q_{out})}{A_{VSS}} \tag{4.2.5}$$

4.3 Simulation method

4.3.1 Matlab

For modeling and simulation the multi-paradigm programming language Matlab has been used. Matlab has several built-in numerical solvers for ordinary differential equations, which all handles first-order ODEs. For the developed simulation program uTubeOscV1, the numerical solver used is Ode45.

4.3.2 Ode45 Solver

Ode45 is a numerical solver for non-stiff differential equations. The solver uses the following syntax; [T,Y] = ode45(odefun,tspan,y0). Ode45 takes three arguments as input and gives two parameters as output, all described in the table below. [Inc., 2015]

Table 4.3.1: Arguments for Ode45

odefun	A function that evaluates the right hand side of a system of differential equations.
tspan	A vector that specifies the interval of integration, [t0, tmax].
y0	A vector that specifies the initial conditions for the system of equations.
T	A vector with the evaluation points, time values.
Y	An array with the solutions. Each row in Y correspond to the solution at the time value returned in the corresponding row of T.

Ode45 takes a function $odefun$ with a system of differential equations on the form $\frac{dy}{dt} = f(t, y)$, together with a time span and a set of initial conditions as input. The system of differential equations is integrated from t_0 to t_{max} with the initial conditions y0. The numerical method used to solve the system of equation is the Runge-Kutta-Fehlberg Method.

4.3.3 The Runge-Kutta-Fehlberg Method(RKF45)

RKF45 is a numerical method that solves differential equations explicitly. It combines the solution of a RK method of fourth order,RK4, and a RK method of fifth order,RK5, to achieve a high accuracy in the solution. At each step the solution of a given initial value problem is approximated with the two RK methods. The approximations are compared to determine if the proper step size h is used. The approximation is accepted if the two results are in close agreement. If the the two answers differs with more digits than a specified accuracy, the step size is reduced. If the answers coincides with more significant digits than the specified accuracy, the step size is increased.

For each step the following six values are used:

$$
\begin{aligned}
k_1 &= hf\Big(t_n, y_n\Big)\\
k_2 &= hf\Big(t_n + \frac{1}{4}h, y_n + \frac{1}{4}\Big)\\
k_3 &= hf\Big(t_n + \frac{3}{8}h, y_n + \frac{3}{32}k_1 + \frac{9}{32}k_2\Big)\\
k_4 &= hf\Big(t_n + \frac{12}{13}h, y_n + \frac{1932}{2197}k_1 - \frac{7200}{2197}k_2 + \frac{7296}{2197}k_3\Big)\\
k_5 &= hf\Big(t_n + h, y_n + \frac{439}{216}k_1 - 8k_2 + \frac{3680}{513}k_3 - \frac{845}{4104}k_4\Big)\\
k_6 &= hf\Big(t_n + h, y_n - \frac{8}{27}k_1 + 2k_2 - \frac{3544}{2565}k_3 + \frac{1859}{4104}k_4 - \frac{11}{40}k_5\Big)
\end{aligned}
\tag{4.3.1}
$$

With these values an approximation to the solution of the IVP is made using the RK4 method:

$$y_{n+1} = y_n + \frac{25}{16}k_1 + \frac{1408}{2565}k_3 + \frac{2197}{4101}k_4 - \frac{1}{5}k_5 \tag{4.3.2}$$

An even better approximation to the solution is made using the RK5 method:

$$z_{n+1} = y_n + \frac{16}{135}k_1 + \frac{6656}{12825}k_3 + \frac{28561}{56430}k_4 - \frac{9}{50}k_5 + \frac{2}{55}k_6 \tag{4.3.3}$$

With the two approximations the optimal step size sh is determined by multiplying a scalar s with the step size h used in the approximations. The scalar is:

$$s = \left(\frac{\epsilon h}{2|z_{n+1} - y_{n+1}|} \right)^{1/4} \tag{4.3.4}$$

where ϵ is the specified error tolerance. [Mathews and Fink, 2004]

5 Investigation of virtual surge shafts

The aim of this section is to investigate the use of VSS in one dimensional modeling of hydropower systems. The dependency on the cross sectional area is investigated with simulations of a hydropower system modeled with and without VSSs. In the first part, the behavior of a VSS is described and possible problems are addressed. In the second part, results from simulations with different cross sectional areas are presented and discussed. Whether the use of VSS is a feasible solution will be discussed and a conclusion is given. At last it is discussed how to effectively implement the use of VSS in uTubeOscV1.

5.1 The behavior of a VSS

A general description of a VSS was given in section 4.2.2. A VSS is a surge shaft with a significantly small cross sectional area. As the volume flow through the turbine is changed, the water level in a VSS will oscillate with a high frequency around the same water level as the surges in the attached surge shaft. The stationary water level is equal to the water level of the nearest free surface minus the head loss in the pipes between the VSS and the free surface. When the volume flow through the system is changed, the water level in any VSSs placed between a turbine and a surge shaft will oscillate according to the surges in the surge shaft. Any VSS placed between a reservoir and a surge shaft will oscillate with the same frequency as the surges in the surge shaft, but with a reduced amplitude. The VSSs will reflect the head at the node they are attached to.

If the cross sectional area of a VSS is too large, the oscillations affects the volume flow through the system and the shaft surges will start to fluctuate. For a significantly large cross sectional area, the VSSs will act as surge shafts inserted in the system.

5.2 Different cross sectional area

5.2.1 Hydropower system

The hydropower system used for simulation of different VSS cross sectional areas, comprises surge shafts between the turbine and the reservoirs. The head race tunnel and the penstock consist of two pipe elements each, with the same geometry, connected with a VSS. To connect the turbine element with the surrounding pipe elements, a VSS is placed in the node before and in the node after the turbine. The layout of the hydropower system is depicted in Fig. 5.2.1.

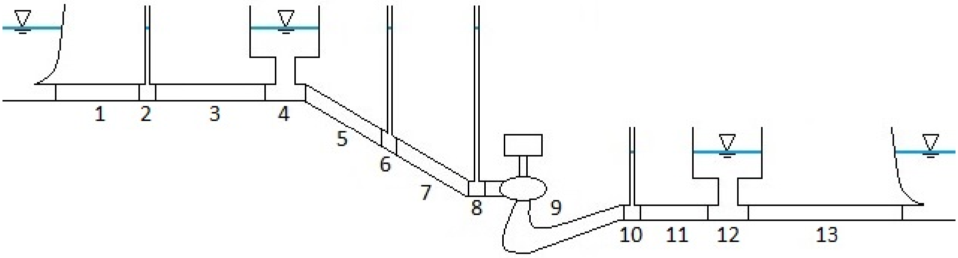


Figure 5.2.1: Hydropower system with VSS

The simulations are compared with results from simulations made with a modeled hydropower system without VSSs.

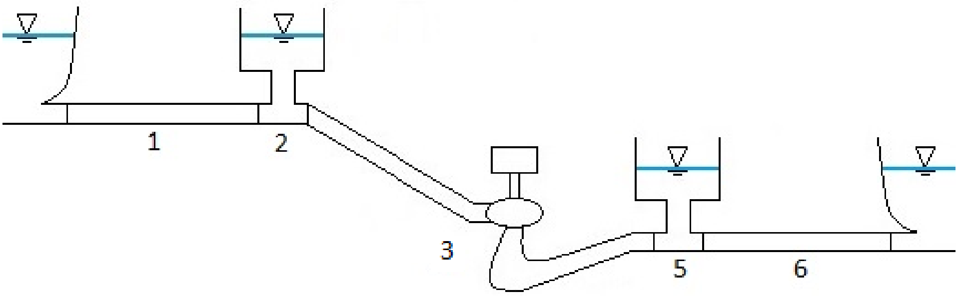


Figure 5.2.2: Hydropower system without VSS

In the reference model, the penstock and the pipe in between the turbine and the downstream surge shaft, are included in the turbine element. Hence, the different elements are connected through the head at the surge shafts. All data for both models are given in Appendix D.1.

A turbine shut down is simulated, from full opening to five percent opening, with the Simple Francis turbine model. The turbine shut down is simulated with six different values of the VSS cross sectional area.

5.2.2 Simulation results and discussion

In this section, it is only given the plots of shaft surges in the upstream surge shaft and oscillations in the VSS placed in the penstock, element 6. For plots of oscillations in the remaining VSSs and surges in the downstream surge shaft, the reader is referred to Appendix D.2.

The shaft surges in the upstream surge shaft is plotted with cross sectional areas of 1 to 10 square meters in Fig. 5.2.3. The shaft surges shows, as expected, an increased deviation from the reference surges with increasing cross sectional area. This is due to the growing impact on the volume flow through the system as more water will enter the VSSs. With a cross sectional area of 1 m^2 or larger the shaft surges starts to fluctuate and the amplitude and phase of the surges decreases with an increasing area.

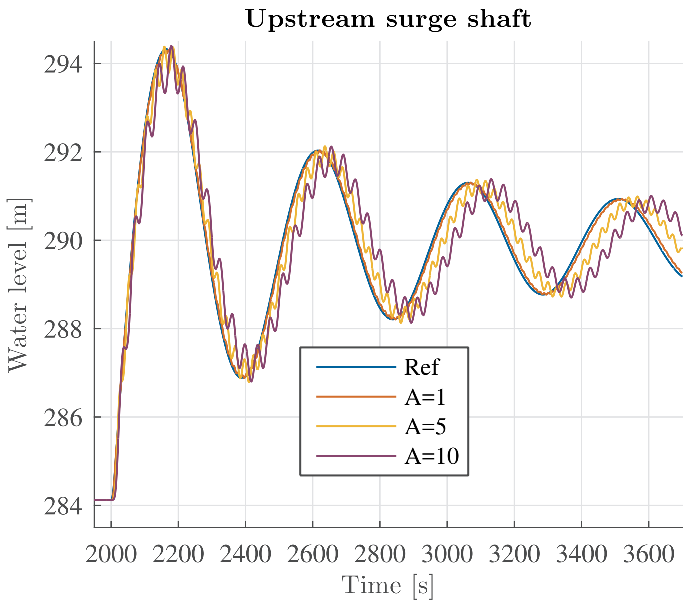


Figure 5.2.3: Shaft surges in upstream surge shaft: Areas of 1 to 10 m^2

The shaft surges seen in relation with the oscillations in the VSSs explains the fluctuations around the shaft surges when the cross sectional area is of 1 m^2 or larger. Oscillations in the VSS connecting the pipe elements in the penstock, element 6, occurring at the start of the shut down, is plotted with cross sectional areas of 1 to 10 square meters in Fig. 5.2.4.

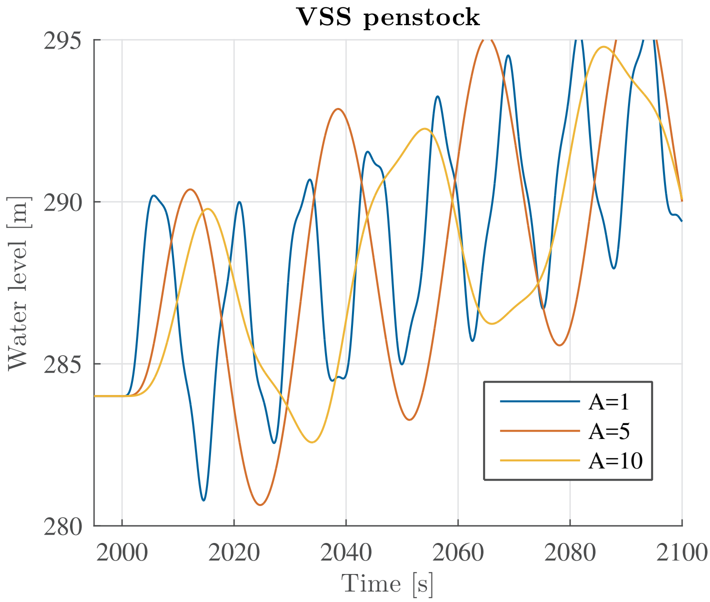


Figure 5.2.4: Oscillations in VSS: Areas of 1 to 10 m^2

The oscillations shows a decreased frequency with increasing cross sectional area. The amplitude of the oscillations increases until the cross sectional area is significantly large, 10 m^2. Then the amplitude of the oscillation decreases as the volume flow entering the VSS becomes significantly large and the VSS will act as a regular surge shaft. For all cross sectional areas above 1 m^2, the frequency is significantly low and will impact the shaft surges.

The shaft surges in the upstream surge shaft is plotted with cross sectional areas of 0.1 to 0.001 square meters in Fig. 5.2.5.

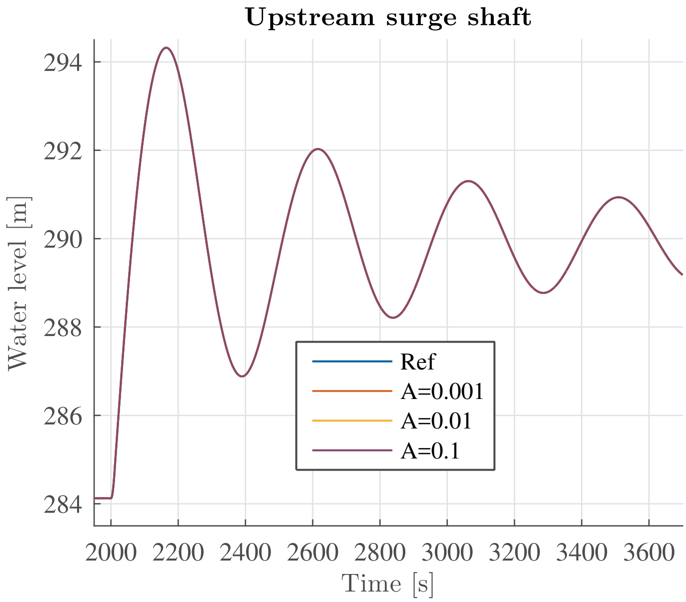


Figure 5.2.5: Shaft surges in upstream surge shaft: Areas of 0.1 to 0.001 m^2

The shaft surges shows nearly no affection from the VSSs if the cross sectional area is significantly small. With values below 1 m^2 the impact seems to be irrelevant for the surges. However, with a closer look the shaft surges reveal some fluctuations with a cross sectional area of 0.1 m^2. To be able to detect any deviation from the reference surges, the surges have to be studied on a small time scale. The maximum up surge is plotted with cross sectional area of 0.1 to 0.001 square meter in Fig. 5.2.6.

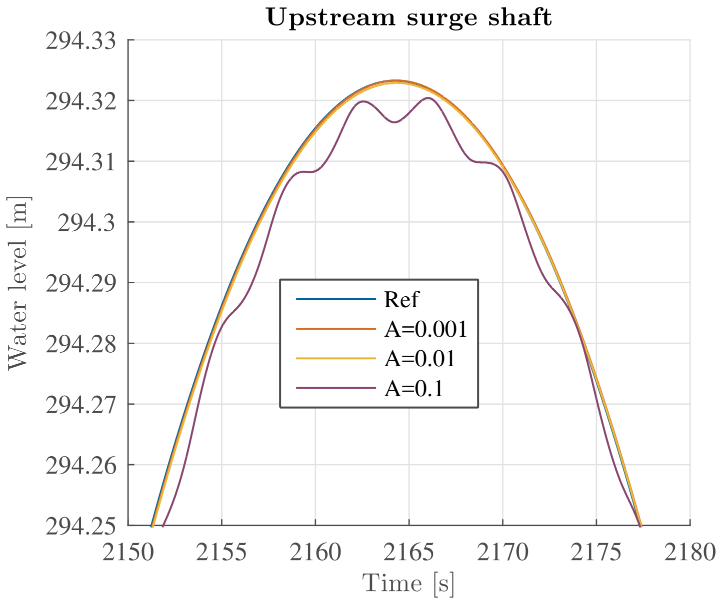


Figure 5.2.6: Maximum up surge: Areas of 0.1 to 0.001 m^2

Compared to the reference surge, the maximum up surge with a cross sectional area of 0.1 m^2 deviates with only 0.068 %. However, the surge shows small fluctuations, and to obtain a smooth curve it seems like the cross sectional area must be even smaller. With a cross sectional area of 0.01 and 0.001 m^2, the surges coincides completely with the reference surge.

The oscillations occurring at the start of the shut down in the VSS connecting the pipe elements in the penstock, element 6, is plotted with cross sectional areas of 0.1 to 0.001 m^2 in Fig. 5.2.7.

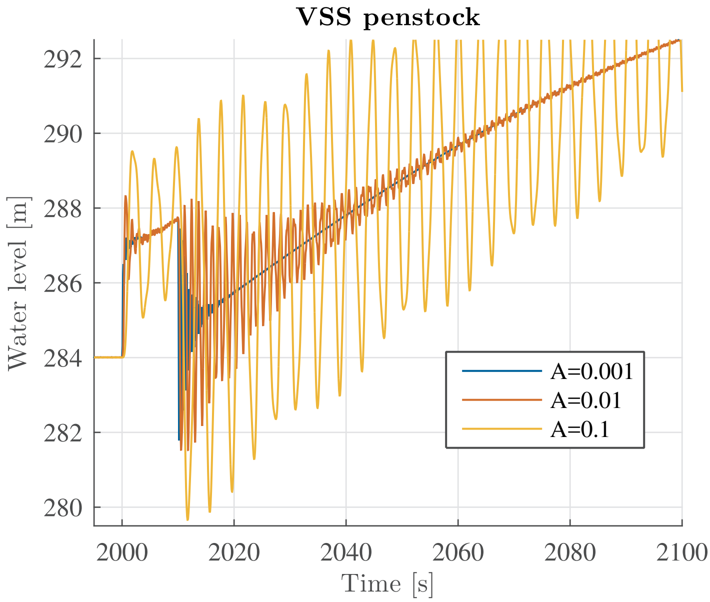


Figure 5.2.7: Oscillations in VSS: Areas of 0.1 to 0.001 m^2

The oscillations shows a change of behavior as the cross sectional area is reduced from 0.1 m^2 to 0.01 m^2. Then the frequency becomes significantly high and the oscillations fades out rather quickly. Hence, the oscillations have almost no impact on the volume flow in the system and the VSSs acts as intended.

For the VSSs placed downstream the turbine (See Appendix D.2), the water level will fluctuate around the stationary level. This is most likely because the system is not at equilibrium when the simulation is started. The results indicate that the reason why these fluctuations only occurs at the VSSs placed downstream the turbine, is because of the low head. Seen in relation to a mass-spring system, a low pressure gives a weaker spring compared to a high pressure. The amplitude of the fluctuations around the stationary water level increases as the cross sectional area decreases, but they are stable fluctuations.

It is possible to include a friction model that dampens out the fluctuations. Most friction models requires the head at the node as a variable. The intention of using VSS in one dimensional modeling is to get a differential equation for the head in nodes and branches. Therefore, these friction model will create the same problem as the use of VSS is intended to solve. However, it should be possible to use steady state friction loss. Then, the friction model is only dependent on the volume flow entering the VSS.

The results shows, as expected, that the cross sectional area is of great importance for the VSSs impact on the rest of the system. For areas larger than 1 m^2, the VSS causes fluctuations in the surge shafts and the use of VSS is not a feasible solution. When the cross sectional area is reduced to values below 1 m^2, the shaft surges shows almost no influence from the oscillations in the VSSs. With an area of 0.01 and 0.001 m^2, the shaft surges coincides completely with the reference surge. An area of 0.1 m^2 gives small fluctuations in the surge. Even though the surge have small fluctuations, the simulations shows a negligible deviation from reference surge. The fluctuation is insignificant and this area could have been used for modeling with VSS, but to obtain a smooth curve the author recommend a smaller cross sectional area. How much the area must be reduced to avoid these fluctuations may be determined by simulation of several values between 0.1 and 0.01. This area is most likely dependent on the complexity of the modeled system and the correct value might be less than 0.01 m^2 if the system geometry is significantly complex.

Provided a significantly small cross sectional area, it can be argued that the use of VSS is a feasible solution for one dimensional modeling using tube elements comprising inelastic water and pipes. With an cross sectional area of 0.01 and 0.001 m^2, the fluctuations shows no impact on the rest of the system. It should be stated that these values are not verified with complex hydropower systems and the exact cross sectional area that avoids fluctuations has not been determined. It might be necessary with an even smaller cross sectional area for models that are complex. But the solution method gives good results and is used in the developed simulation program.

The amplitude of the stable fluctuations in VSSs downstream the turbine, increases as the cross sectional area decreases. These fluctuations will not impact the volume flow in the system and for simulations done with a turbine model without frequency governor, the increased amplitude is of little importance. However, for simulations with a turbine model that includes a frequency governor, these fluctuations will cause the governor to adjust the opening degree. If the amplitude is too large, the system will eventually become unstable. The author recommend to have a slightly higher cross sectional area if the system is complex and the governor is included in the turbine model.

The correct cross sectional area is dependent on the system complexity and whether or not a turbine model with governor is used. Therefore, the VSSs should be used wisely.

5.3 VSS implemented in uTubeOscV1

In uTubeOscV1, the problems regarding head at nodes and branches for one dimensional modeling is solved with the use of VSS. The VSS is implemented as node, branch and bound elements. The elements are defined by the differential equations describing a VSS, Eq.(4.2.3), (4.2.4) and (4.2.5). The cross sectional area of the VSSs is specified after the program is started and it is the same for all VSSs in the system.

If the turbine model without governor is used, the cross sectional area is recommended to be 0.01 m^2 or 0.001 m^2 dependent on the system complexity. If the frequency governor is included, the cross sectional area is recommended to be slightly higher. The execution speed decreases as the cross sectional area decreases and it is desirable to have a high execution speed. The area should be as large as possible without affecting the system. In the verification of uTubeOscV1, a complex hydropower system is modeled with a cross sectional area of both 0.09 and 0.001 m^2 for the VSSs and the results are satisfying.

6 uTubeOscV1

uTubeOscV1 is based on one dimensional modeling, as described in section 4.2. In uTubeOscV1, nine different element types are used for modeling of hydropower systems; pipes, surge shafts, nodes, branches, bounds and the four turbine models given in section 3. The dynamic behavior of a given hydropower system is defined by the differential equations describing the different elements.

6.1 Layout

A given hydropower system is described in a text file. The Matlab program scans the file and stores the data in an array. A developed function *dynsysV1* uses the data stored in the array to create a system of differential equations that describes the dynamic behavior of the hydropower system. The function containing the dynamic equations is taken as input for the built-in numerical solver, Ode45, together with initial values and a time span. The solver gives data for U-tube oscillations between the surge shafts and the reservoirs, which is plotted and displayed in figures.

The function *dynsysV1* evaluates the right hand side of the system of differential equations. In the program-code for *dynsysV1*, the system of differential equations is expressed with a function f that can be both the volume flow, Q, and the water level in surge shafts and VSSs, z.

The program-code for uTubeOscV1 and the function *dynsysV1* are given in Appendix B.

6.2 Assumptions and restrictions

There are made some assumptions for the hydropower systems that are modeled with uTubeOscV1. For simplicity, it is assumed mass-less surge shafts and VSSs so that the head in the nodes attached to a surge shaft or VSS element is equal to the water level in the shafts. A turbine element is assumed to act as a valve with a varying opening degree, κ. The losses in the turbine is neglected for the turbine models. As the only phenomena studied is u-Tube oscillations, the turbine losses is assumed to be of limited importance.

The program has some restrictions on the composition of the elements that makes the hydropower system:

- A system must start with a tunnel(pipe element) attached to the upper reservoir.

- A system must end with a tunnel(pipe element) attached to the lower reservoir.
- A surge shaft element must be connected to pipe elements
- A turbine element must be connected to pipe elements. The node before and after a turbine is included in the turbine element.
- A branch or bound must be connected to pipe elements
- The minimum turbine opening degree allowed is $\kappa = 0.01$. This is due to how the opening degree is implemented in the equation of motion for a turbine. In Eq.(3.5.7), κ is included in the denominator. Hence, it is not possible to set the opening degree equal to zero.

6.3 Text file

The text file describing a given hydropower system must be of .txt format. The elements given in the text file must be structured according to how the program reads the file, described in section 6.3.2.

6.3.1 Input

The dynamic equations describing the different elements requires different information. Table 6.3.1 provides the necessary input for the respective elements.

Table 6.3.1: Required input for the different element types

Pipe	1, prev.El., nextEl., f_0, A, L, D, f
S.shaft	2, prev.El., nextEl., f_0, A_S
Pelton turbine	31, prev.El., nextEl., f_0, I_h, H_R, Q_R, H_{t1}, H_{t2}
Francis turbine, grid mode	32, prev.El., nextEl., f_0, I_h, H_R, Q_R, D_{t1}, D_{t2}, B_{t1}, α_{1R}, β_2, Pole-pairs, P_{max}, f_{grid}, T_a, ω_R, H_{t1}, H_{t2}
Francis turbine w/freq.governor, grid mode	33, prev.El., nextEl., f_0, I_h, H_R, Q_R, D_{t1}, D_{t2}, B_{t1}, α_{1R}, β_2, Pole-pairs, P_{max}, f_{grid}, T_a, ω_R, T_K, T_i, b_t, b_p, H_{t1}, H_{t2}
Francis turbine w/freq.governor, island mode	34, prev.El., nextEl., f_0, I_h, H_R, Q_R, D_{t1}, D_{t2}, B_{t1}, α_{1R}, β_2, Pole-pairs, P_{max}, f_{grid}, T_a, ω_R, T_K, T_i, b_t, b_p, E, H_{t1}, H_{t2}
Node	4, prev.El., nextEl., f_0
Branch	5, f_0, # el. after, prev.El., nextEl.1, nextEl.2, ..., nextEl.n
Bound	6, f_0, # el. before, nextEl., prev.El.1, prev.El.2, ..., prev.El.n

f_0 is the initial value for the differential equation describing the element. For a pipe element and a turbine element $f_0 = Q_0$ and for the other elements $f_0 = H_0$. The reservoirs are modeled with constant water level and do not constitute separate elements. The first element in the hydropower system, a pipe element, is attached to the upper reservoir. For the program to understand that this is the first element, the previous element must be set to zero. The last element is attached to the lower reservoir and the next element must then be set to zero.

6.3.2 The text file structure

A template of the text file that describes a given hydropower system is given in Appendix C. The template describes how the elements should be structured and what information is needed for the different element types. The elements are listed with associated input as shown in figure 6.3.1. One line in the file represents one element in the hydropower system and the associated input is separated with comma, colon, semicolon or blank space. The elements must be listed in the correct order such that element number one is defined in the first line, element number two is defined in the second line and so on.

```
% The elements are listed below:

1;prev.El;nextEl;Q_0;A;L;D;f

2;prev.El;nextEl;z_0;A_s

1;prev.El;nextEl;Q_0;A;L;D;f

31;prev.El;nextEl;Q_0;I_h;H_R;Q_R;H_t1;H_t2

1;prev.El;nextEl;Q_0;A;L;D;f

2;prev.El;nextEl;z_0;A_s

1;prev.El;nextEl;Q_0;A;L;D;f
```

Figure 6.3.1: How the elements are structured in the text file

6.4 Import data to Matlab

The data that describes all the elements in a given hydropower system is imported from the text file to Matlab. The data is stored in an array X, which is declared as an array of all zeros. Each row i represents an element and each column j contains the data for the element corresponding to row i. The turbine element representing a Francis turbine with frequency governor running in Island mode, is the element that requires most input, with 24 values. Hence the number of columns in array X is 24. For a hydropower system consisting of n elements, the size of array X will be (n,24):

$$\text{X(i,j)} = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,24} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,24} \\ \vdots & \vdots & & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,24} \end{bmatrix}$$

The height level at the upper and lower reservoir, and the characteristics of the varying opening degree for the turbines are taken as input from the user when the program is started. These values are stored in another array, A.

6.5 System of dynamic equations

The function *dynsysV1* uses the data from array X to create the system of differential equations that describes the system. Before the system of equations is formed, a for-loop iterates through all the rows in X and creates a vector of initial values that is used as input in the numerical solver. The differential equations describing each element are then created by iteration in array X. The first value in each row gives the element type. With if-/else-statements the program determines what differential equation that should be used for the element represented by row i.

A pipe element can be described with different equations, based on what position the element has in the system. If row i represents a pipe element, the attached elements are checked to determine if the pipe element is the first or the last element in the system. Then the differential equation is dependent on the height level at the upper or lower reservoir, hence the pipe element is described with a differential equation containing the head at the upper or lower reservoir.

As the modeled hydropower system must start and end with a pipe element, the other element types are not directly connected to the head at the upper and lower reservoir and it is therefore not necessary to check for which elements that are connected to the given element.

There is no need for a node element at turbine-pipe connections, they are created within a turbine element.

6.6 Program abilities

uTubeOscV1 is a program for simulation of U-tube oscillations due to changes in the turbine opening degree. The program can model an arbitrary hydropower system and has no limitations regarding the system geometries. The program is able to model hydropower plants with turbines connected to generators that are running in grid mode.

Most of the turbines used in Norway are closed and opened with intervals. Typically, a turbine shut down is divided in two intervals. In the first interval, the wicket gates are closed fast and in the second interval the closing is slower. For manual change of the opening degree in uTubeOscV1, the closing/opening can be done in two intervals. The user must specify the time intervals, at what time and to what values the turbine opening degree should be changed. This is described further in section 6.7.

When a power plant is connected to the joint grid, the turbine rotational speed is synchronous and the speed is determined by the grid frequency. Any changes in power demand on the grid will cause a change in the grid frequency. The program is able to simulate how a given hydropower plant acts on a change in power demand. The grid frequency is then set to change to a given value at a given time during the simulation.

As default, uTubeOscV1 plots the U-tube oscillations between all free surfaces in the given hydropower system. It is also possible to plot the volume flow through each element, the change in turbine opening degree, the angular frequency of the turbine and the angle between generator stator and rotor.

With the use of VSSs, the program can model any hydropower systems regardless of the system composition.

A turbine model for simulation in Island mode is implemented in the program. This turbine element is not in use and is included for further development of the program.

6.7 Run the program

The first thing to do is to put up the desired hydropower system in the text file. To be able to use the text file, it must be saved in the same folder as the program files. Once this is done the program is ready to run. When the program is started the user will be asked to specify which text file the program should scan through, as shown in the figure below.

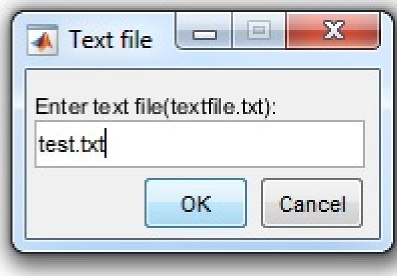
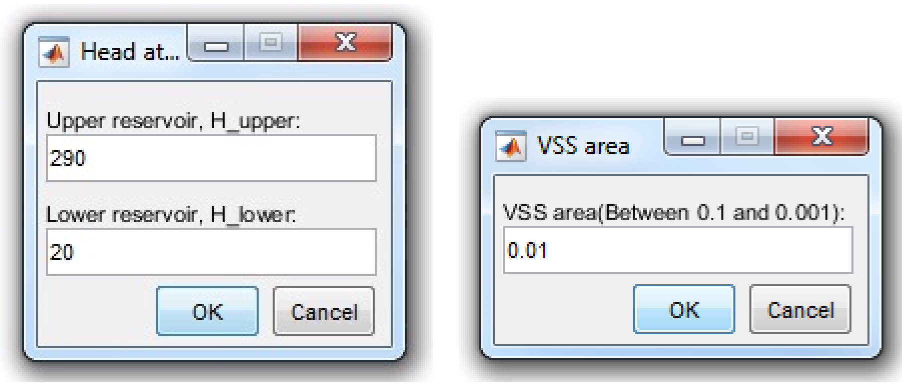


Figure 6.7.1: uTubeOscV1 input; Specify text file

Next, the user will be asked to specify the height level at the upper and lower reservoir. The cross sectional area of the VSSs is also specified.



(a) Head at reservoirs

(b) VSS area

Figure 6.7.2: uTubeOscV1 input; Head and cross sectional area

The program is able to simulate different scenarios and the turbine opening degree can be adjusted either by changing the opening degree manually or by changing the grid frequency. Since there are different turbine models for the different scenarios, the user will be asked how many turbines the hydropower system contains of and what type of turbines that are used.

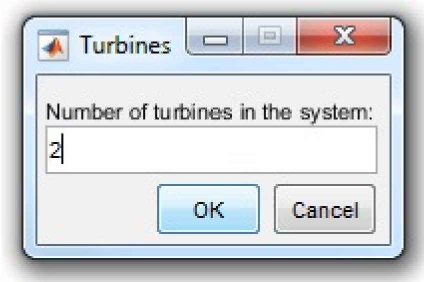


Figure 6.7.3: uTubeOscV1 input; Specify how many turbines in the system

For each turbine the user will be asked what type of turbine it is.

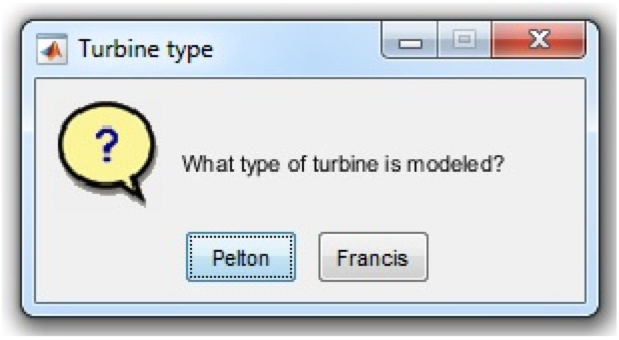
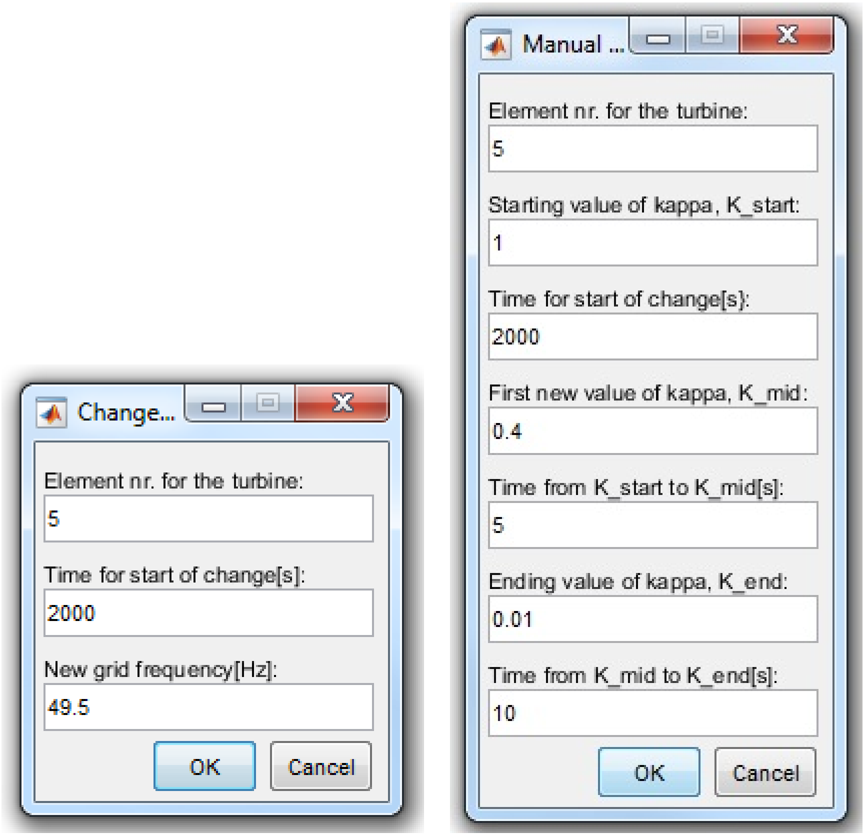


Figure 6.7.4: uTubeOscV1 input; Specify turbine type

If the turbine is a Francis turbine, the user must specify what Francis turbine model that is used:

Figure 6.7.5: uTubeOscV1 input; Specify Francis turbine model

For a Francis turbine with frequency governor, the turbine opening degree is adjusted by changing the grid frequency. As shown in Fig. 6.7.6a, the user will be asked to specify when and to what value the grid frequency should be changed during the simulation. If the turbine is a Pelton turbine or a Simple Francis turbine, the turbine opening degree is adjusted by manual change of the opening degree. Then the user will be asked to specify how the opening degree should be changed during the simulation, as shown in Fig. 6.7.6b.



(a) Grid frequency (b) κ manually

Figure 6.7.6: uTubeOscV1 input; Change of turbine opening degree

When all the turbine input is determined, the total simulation time must be specified and the simulation will start.

7 Verification of uTubeOscV1

This section aims to verify the simulation program, uTubeOscV1. The verification is performed by comparing the simulation results to simulations produced with LVTrans. Two hydropower systems with significantly different system complexity are modeled. To verify the different turbine models presented in section 3.7 and 3.8, hydropower system 1 is simulated with all three models. To verify the programs ability to do simulations of a complex system, it is simulated manual change of the turbine opening degree and change in the power demand with hydropower system 2.

7.1 LVTrans

LVTrans is an object oriented simulation program for pipe systems with fluid. The program is general and can be used to compute all systems comprising fluid, however, the program is specially designed for dynamic calculations of hydropower systems. LVTrans is programmed in LabVIEW and it uses LabVIEWs interface for modeling and simulation. Each element in a hydropower system is predefined and the model of a real system is created by connecting the different elements together, as they are connected in reality. The dynamic behavior of the hydropower system is described by numerical models, using the method of characteristics to solve the differential equations. The program has been verified with measured data from several hydropower plants. Tonstad Power Plant and Fortun Power plant, owned by Statkraft, are some examples.

LVTrans is developed by Dr.Ing. Bjørnar Svingen for SINTEF Energy Research, in cooperation with Statkraft. The source code is open and editable. This means that the user is able to make adjustment on the different elements or create new ones if needed. For general calculations on hydropower systems, there is no need to make adjustments as all the necessary elements are defined.

For more information about LVTrans the reader is referred to the user manual [Svingen, 2015] and ¨Documentation for LVTrans(LabVIEW Transient Pipe Analysis) ¨ [Svingen, 2003].

The layout of the hydropower systems modeled in LVTrans are depicted in Appendix E.2 together with the governing parameters.

7.2 Hydropower system 1

Hydropower system 1 is a simple system with an upstream surge shaft and a downstream surge shaft. The system consists of one turbine that is connected to the penstock and the downstream pipe by VSSs. The power plant has a gross head of $H_G = 270$ meter and a flow rate of $Q_0 = 20.7649m^3/s$. The layout of the system is depicted in Fig. 7.2.1 along with a table presenting the geometrical data.

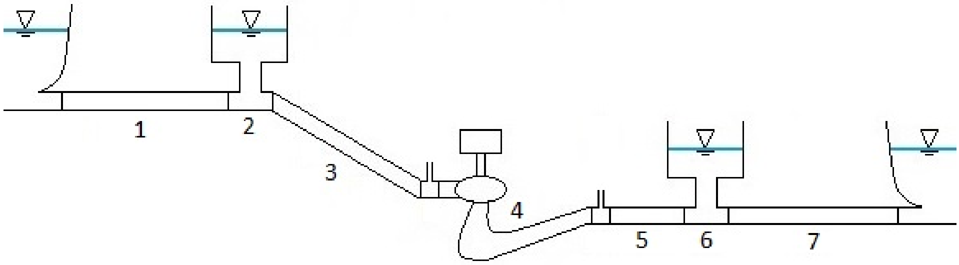


Figure 7.2.1: Hydropower system 1 layout

Table 7.2.1: Geometrical data for Fig. 7.2.1

Element	**A**$[m^2]$	**L[m]**	**Friction factor**
1	12.57	3500	0.05
2	177		
3	13.19	350	0.02
5	13.19	20	0.02
6	78		
7	12.57	2000	0.05

All of the turbine models have a rated head $H_R = 270m$ and a rated volume flow $Q_R = 20.7649\frac{m^3}{s}$. The geometrical data for the Francis turbines and the governing parameters are presented in Appendix E.1. To allow the system to reach equilibrium at steady state, any changes of the opening degree is done after 2000 seconds for all simulations.

7.2.1 Pelton: Manual change of the opening degree

With the Pelton turbine model, it is simulated manual change of the opening degree. Initially, the turbine has an opening degree $\kappa = 0.05$ and after 2000 seconds the opening degree is changed to $\kappa = 1$, with an opening time $T = 10$ seconds. The oscillations in the upstream surge shaft is plotted in Fig. 7.2.2.

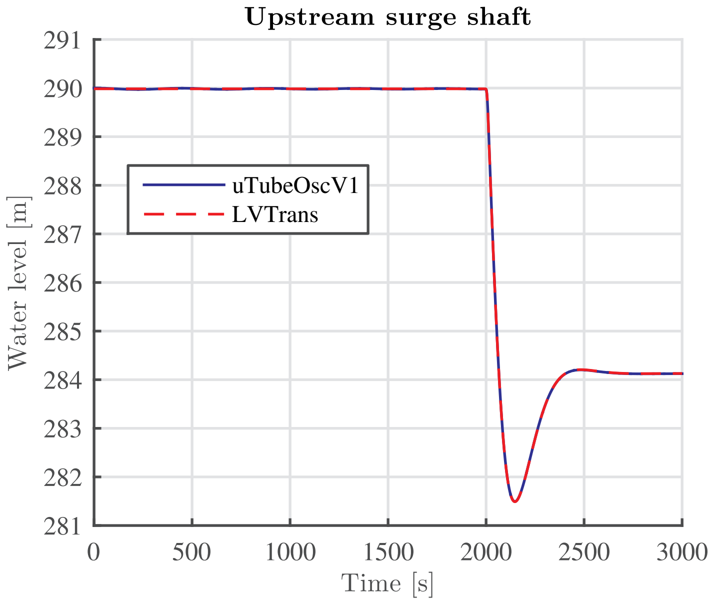


Figure 7.2.2: Upstream surge shaft: Manual change of the opening degree

Table 7.2.2: Down surge in upstream surge shaft

	Stationary level[m]	**Lowest level[m]**	**Max down surge[m]**
uTubeOscV1	289.9848	281.4928	8.4920
LVTrans	289.9848	281.4897	8.4951
Deviation	0.0000	0.036 %	0.036 %

From the oscillations it can be seen that a down surge occurs in the upstream surge shaft as the turbine opening degree is increased. Compared to the simulation with LVTrans, uTubeOscV1 gives good simulation results. The oscillations coincides completely. The tabulated results of the maximum down surge shows that the uTubeOscV1 simulation deviates with only 0.036 % compared to the simulation with LVTrans.

The increased opening degree causes an up surge in the downstream surge shaft. The oscillations in the downstream surge shaft are plotted in Fig. 7.2.3 along with a table presenting the numerical results of the maximum up surge.

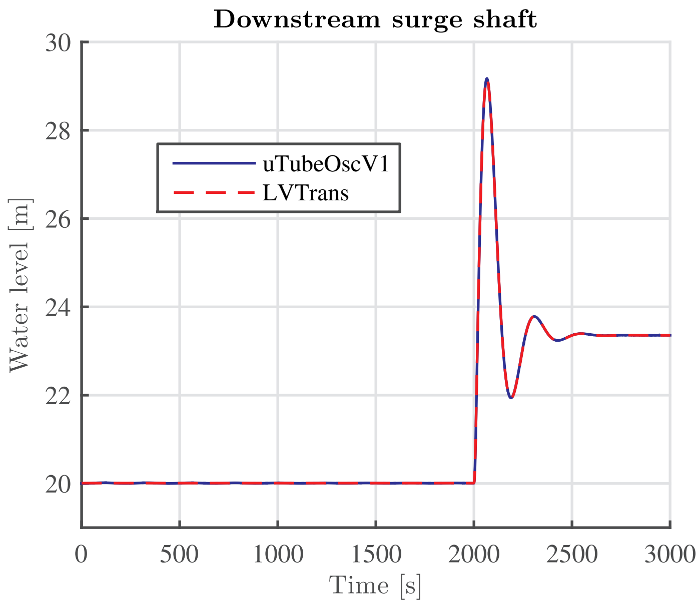


Figure 7.2.3: Downstream surge shaft: Manual change of the opening degree

Table 7.2.3: Up surge in downstream surge shaft

	Stationary level[m]	**Highest level[m]**	**Max up surge[m]**
uTubeOscV1	20.0087	29.1738	9.1651
LVTrans	20.0087	29.1633	9.1546
Deviation	0.0000	0.115 %	0.115 %

The oscillations in the downstream surge shaft coincides completely. The maximum up surge is insignificantly higher when simulating with uTubeOscV1 compared to LVTrans. The up surge deviates with only 0.115 %.

7.2.2 Francis: Turbine shut down

A turbine shut down is simulated with the Francis model without frequency governor. The turbine shut down has a closing time $T_C = 15$ seconds and the wicket gates are closed in two intervals with different speed. In the first interval, the opening degree is changed from $\kappa = 1$ to $\kappa = 0.5$ in 5 seconds. From $\kappa = 0.5$ to fully closed the turbine uses 10 seconds. Since it is not possible to set the opening degree, κ, equal to zero in uTubeOscV1, the closed value is set to $\kappa = 0.01$.

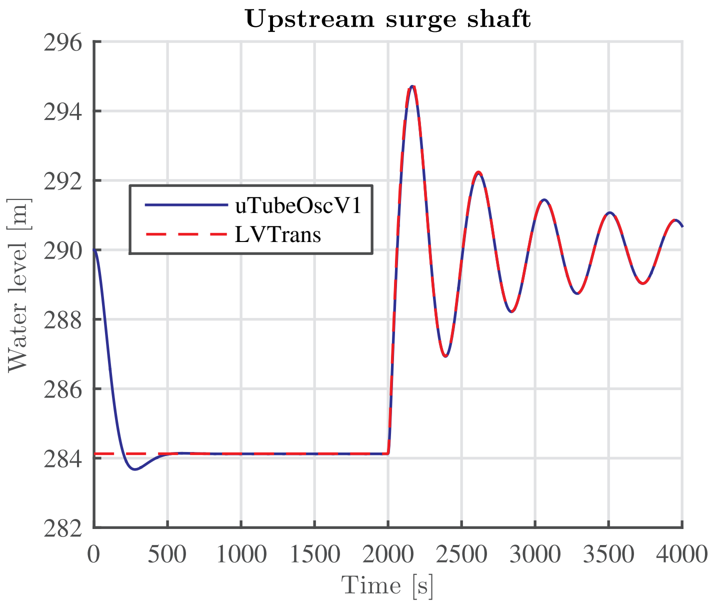


Figure 7.2.4: Upstream surge shaft: Shut down with different speed

Table 7.2.4: Up surge in upstream surge shaft

	Stationary level[m]	**Highest level[m]**	**Max up surge[m]**
uTubeOscV1	284.1246	294.7131	10.5885
LVTrans	284.1283	294.8247	10.6964
Deviation	0.0013 %	0.04 %	1.01 %

The oscillations in the upstream surge shaft due to turbine shut down are plotted in Fig. 7.2.4. The simulated oscillations with uTubeOscV1 coincides with the simulation results from LVTrans. The numerical results of the maximum up surge shows, as expected, a deviation of 1.01%. This is due to the value of the opening degree for fully closed wicket gates in uTubeOscV1. In LVTrans, the opening degree $\kappa = 0$ at fully closed wicket gates and this will give a slightly higher up surge as the retardation pressure becomes higher.

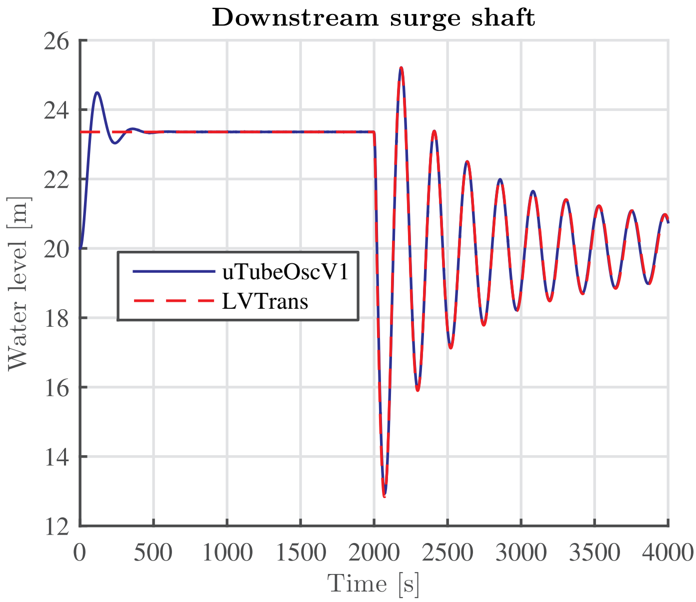


Figure 7.2.5: Downstream surge shaft: Shut down with different speed

Table 7.2.5: Down surge in downstream surge shaft

	Stationary level[m]	**Lowest level[m]**	**Max down surge[m]**
uTubeOscV1	23.3575	12.9305	10.4270
LVTrans	23.3562	12.8363	10.5199
Deviation	0.0056 %	0.73 %	0.88 %

The oscillations in the downstream surge shaft are plotted in Fig. 7.2.5. Here as well, the simulations done with uTubeOscV1 coincides with the simulations with LVTrans. Due to the closing value of the opening degree, the maximum down surge deviates with 0.88%.

7.2.3 Francis w/governor: Decrease in power demand on the grid

For the Francis turbine model with frequency governor, it is simulated an increase in grid frequency due to a decrease of power demand. The grid frequency is constant at first, $f_{grid} = 50Hz$, then a decrease of power demand gives a grid frequency $f_{grid} = 50.5Hz$. The governor acts on the increased grid frequency by changing the turbine opening degree, as shown in Fig. 7.2.6.

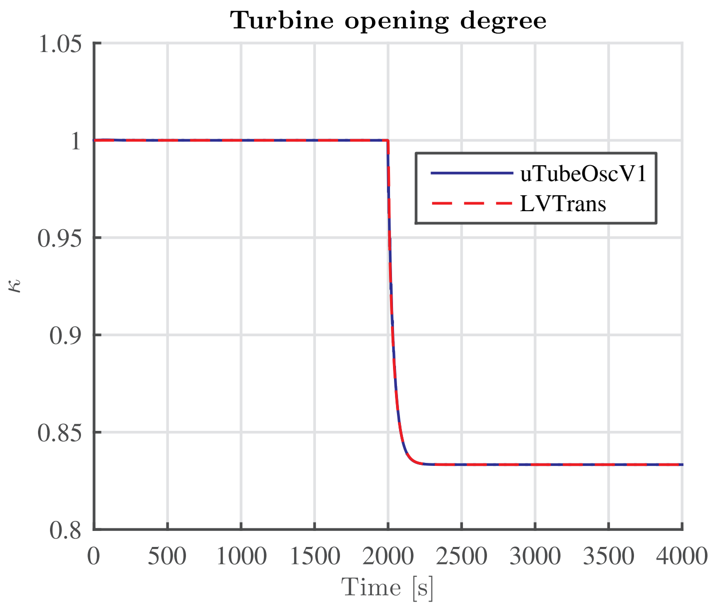


Figure 7.2.6: Turbine opening degree: increased grid frequency

As the power demand decreases the turbine will reduce the volume flow through the turbine by reducing the turbine opening degree. Then the produced power meets the power demand and a balance is obtained. The increased grid frequency gives a decreased generator torque and an increased angular velocity(See Appendix E.3.1). The governor model in uTubeOscV1 seems to regulate the opening degree in a satisfying way compared to the governor used in LVTrans.

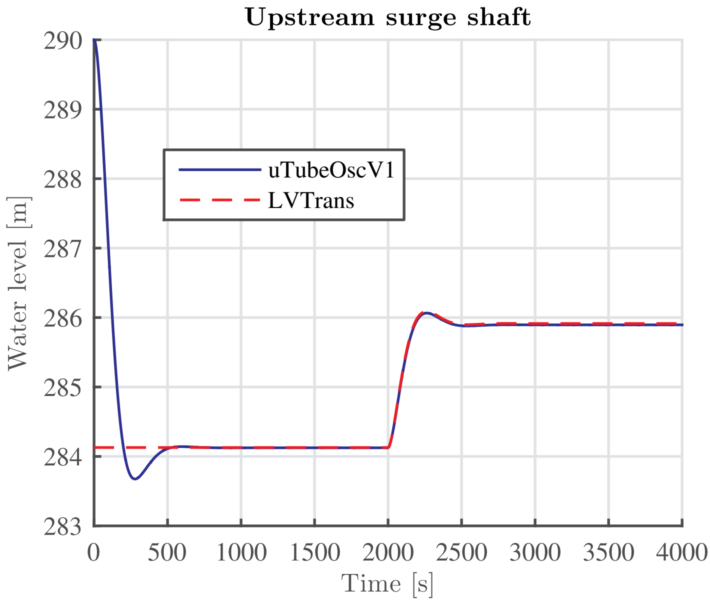


Figure 7.2.7: Upstream surge shaft: increased grid frequency

Table 7.2.6: Up surge in upstream surge shaft

	Stationary level[m]	**Highest level[m]**	**Max up surge[m]**
uTubeOscV1	284.1246	286.0645	1.9399
LVTrans	284.1283	286.0869	1.9586
Deviation	0.0013%	0.0078%	0.95%

The surges in the upstream surge shaft due to the change of opening degree is plotted in Fig. 7.2.7. The simulations done with LVTrans coincides with the simulations produced with LVTrans. The maximum up surge has a deviation of 0.95%.

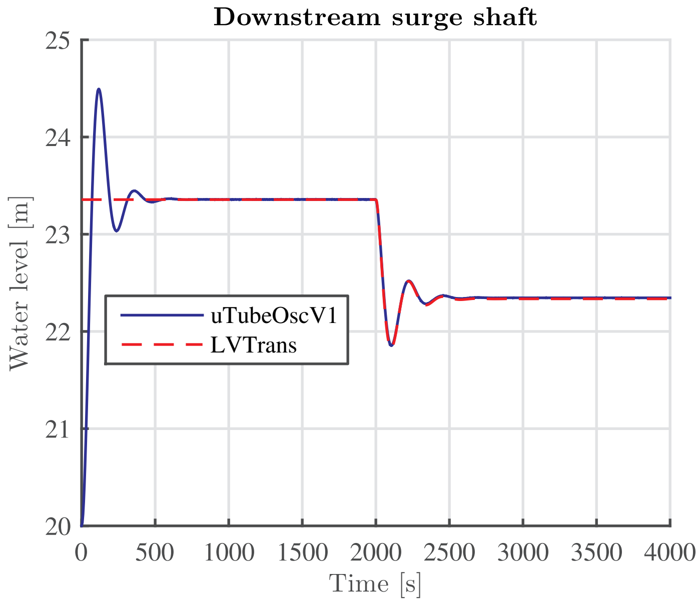


Figure 7.2.8: Downstream surge shaft: increased grid frequency

Table 7.2.7: Down surge in downstream surge shaft

	Stationary level[m]	**Lowest level[m]**	**Max down surge[m]**
uTubeOscV1	23.3575	21.8535	1.5040
LVTrans	23.3562	21.8306	1.5256
Deviation	0.0057 %	0.1049 %	1.42 %

In the downstream surge shaft, a down surge will occur as the turbine opening degree is decreased. Compared to the simulations with LVTrans, the simulation results from uTubeOscV1 shows an deviation of 1.42% for the maximum down surge.

7.3 Hydropower system 2

Hydropower system 2 is a complex system that includes all types of VSS elements. The system consists of two equal turbines placed in parallel. Each turbine has a rated volume flow $Q_R = 20.7649 m^3/s$ and a rated head $H_R = 270$ meter. The penstock constitutes two pipes with different area and length. The system have both branch and bound elements and the system volume flow is $Q_0 = 41.5298 m^3/s$. The system layout is given in Fig. 7.3.1 and the geometrical data is tabulated below. The geometrical data for the Francis turbines are the same as in the simulations of hydropower system 1.

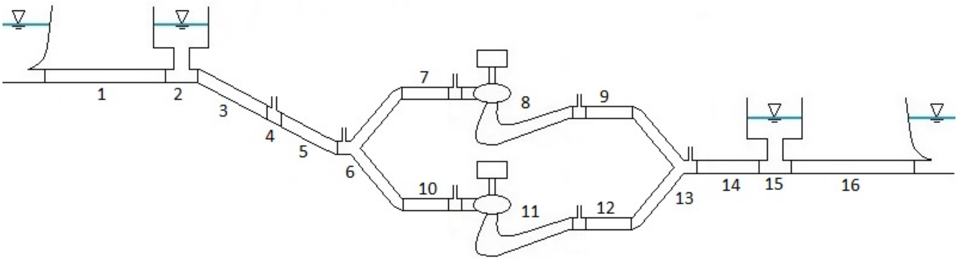


Figure 7.3.1: Hydropower system 2 layout

Table 7.3.1: Geometrical data for Fig. 7.3.1

Element	**A**$[m^2]$	**L[m]**	**Friction factor**
1	19.64	3000	0.05
2	200		
3	12.57	300	0.02
5	7.07	200	0.02
7	4.91	50	0.02
9	4.91	20	0.02
10	4.91	30	0.02
12	4.91	30	0.02
14	7.07	100	0.02
15	100		
16	15.90	2000	0.05

The cross sectional area of the VSSs is set to 0.001 m^2 for the simulation done with the Francis turbine model without governor. For the simulation done with governor, the cross sectional area of the VSSs is set to 0.09 m^2.

7.3.1 Manual change of opening degree on one of two turbines

Manual change of the turbine opening degree is simulated with Francis turbines without governor. Initially, the opening degree on both turbines are $\kappa = 1$. After 2000 seconds the opening degree on turbine 1, element 8, is changed to $\kappa = 0.05$. Due to the change of opening degree, the volume flow through the system is reduced and causes an up surge in the upstream surge shaft.

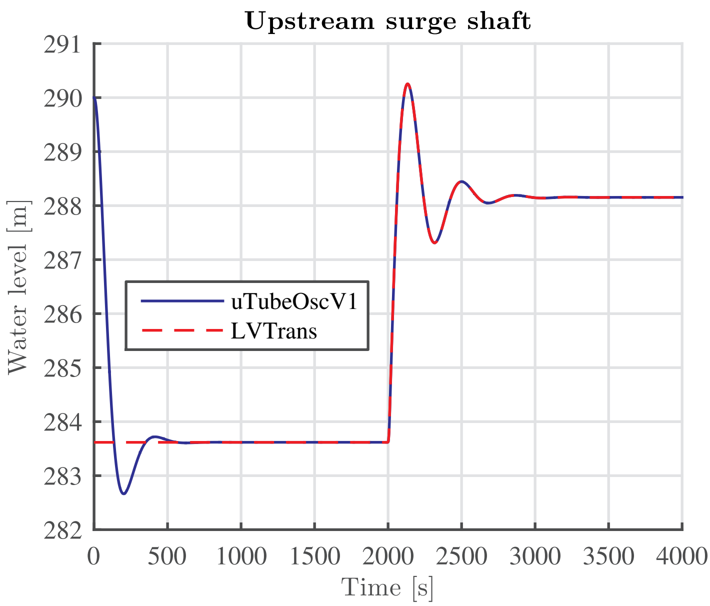


Figure 7.3.2: Up surge due to manual change of opening degree on one of two turbines

Table 7.3.2: Up surge in upstream surge shaft

	Stationary level[m]	**Highest level[m]**	**Max up surge[m]**
uTubeOscV1	283.618	290.2541	6.6361
LVTrans	283.617	290.2554	6.6384
Deviation	0.00%	0.00%	0.03%

The oscillations in the upstream surge shaft is plotted in Fig. 7.3.2. The simulation with utubeOscV1 gives a maximum up surge of 6.6361 meter, which deviates with 0.03% from the simulation done with LVTrans. In the downstream surge shaft a down surge will occur, as shown in Fig. 7.3.3.

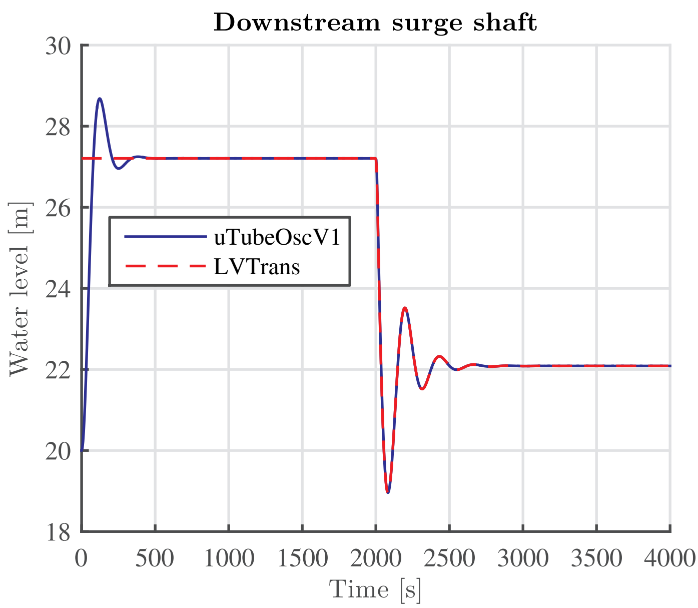


Figure 7.3.3: Down surge due to manual change of opening degree on one of two turbines

Table 7.3.3: Down surge in downstream surge shaft

	Stationary level[m]	**Lowest level[m]**	**Max down surge[m]**
uTubeOscV1	27.2054	18.9589	8.2465
LVTrans	27.2054	18.9606	8.2448
Deviation	0.000 %	0.009 %	0.021 %

From the results on the maximum down surge, tabulated above, it can be seen that the maximum down surge results are almost equal for the simulations done with uTubeOscV1 and LVTrans.

7.3.2 Increased power demand on the grid

The governor is included in the turbine models to simulate a change in grid frequency due to increased power demand on the grid. The grid frequency is changed from 50 Hz to 49 Hz on turbine 1 after 2000 seconds. As the frequency drops, the governor increases the opening degree on turbine 1 to meet the increased power demand. Fig. 7.3.4 shows that the opening degree is adjusted almost equally in uTubeOscV1 and LVTrans. In LVTrans the opening degree changes slightly faster than what the governor in uTubeOscV1 does.

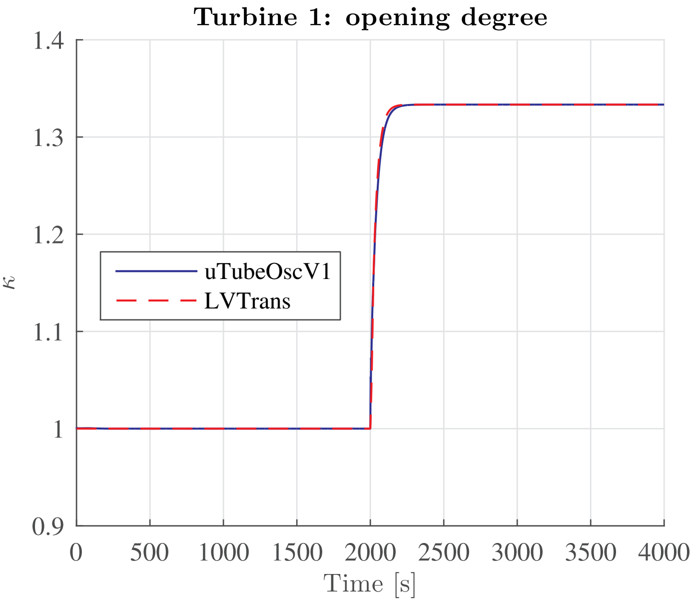


Figure 7.3.4: Turbine opening degree: decreased grid frequency

The surges in the upstream surge shaft are plotted in Fig. 7.3.5. The increased opening degree gives a down surge in the upstream surge shaft and the plot shows that the simulation done with uTubeOscV1 gives a slightly less down surge compared the simulation produced with LVTrans. The maximum down surge deviates with 3.99% and the water level stabilizes a bit higher for the uTubeOscV1 simulation.

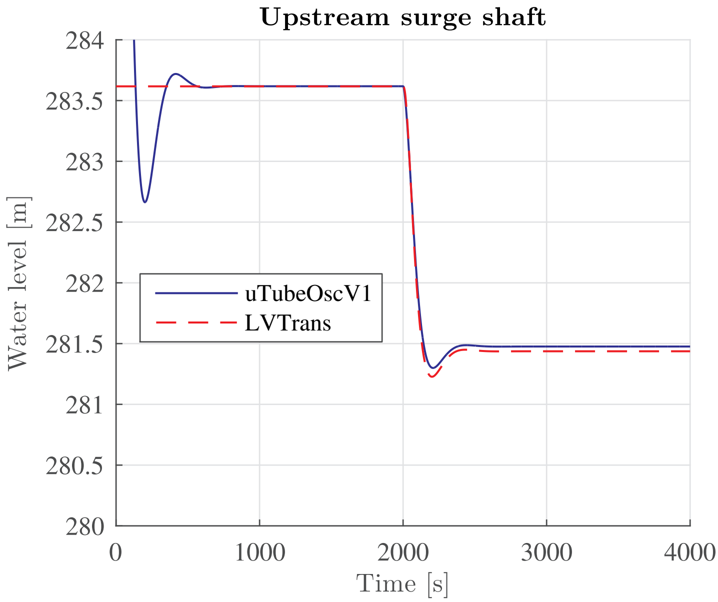


Figure 7.3.5: Upstream surge shaft: decreased grid frequency

Table 7.3.4: Down surge in upstream surge shaft

	Stationary level[m]	**Lowest level[m]**	**Max down surge[m]**
uTubeOscV1	283.6179	281.2995	2.3184
LVTrans	283.6170	281.2272	2.3898
Deviation	0.00 %	0.26 %	2.99 %

In the downstream surge shaft, an up surge occurs as the opening degree is increased. The downstream shaft surges are plotted in Fig. 7.3.6.

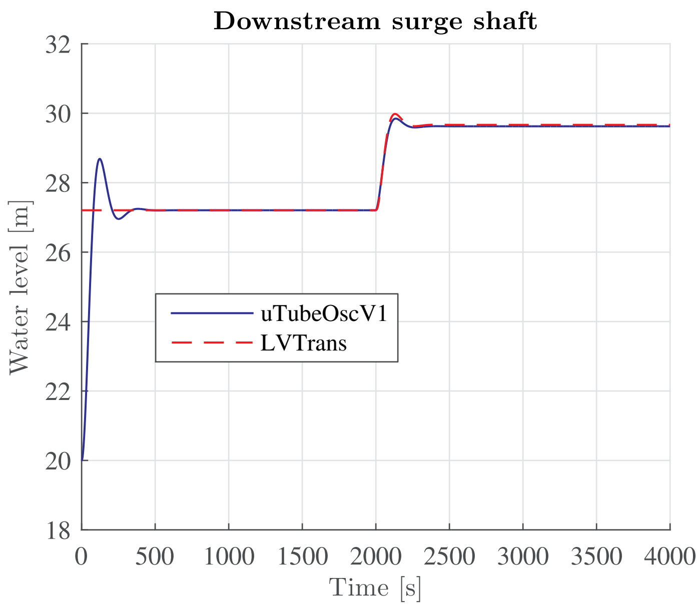


Figure 7.3.6: Downstream surge shaft: decreased grid frequency

Table 7.3.5: Up surge in downstream surge shaft

	Stationary level[m]	**Highest level[m]**	**Max up surge[m]**
uTubeOscV1	27.2055	29.8456	2.6401
LVTrans	27.2054	29.9799	2.7745
Deviation	0.00%	0.45%	4.84%

Compared to LVTrans, the increased opening degree gives a lower up surge with uTubeOscV1. The maximum up surge deviates with 4.84%.

8 Discussion

The investigatioin of virtual surge shafts has been discussed throughout section 5. The aim of the investigation was to determine whether the use of VSS is a feasible solution and it is therefore in place with some finale remarks on the results.

The results show that the cross sectional area is of great importance for the VSSs impact on the dynamics of the system. The simulations show that with an area between 0.1 m^2 and 0.001 m^2, the system is not affected by the oscillations in the VSSs. After a change on the turbine opening degree, the water level in the VSSs will fluctuate around the surges in the attached surge shafts. As the cross sectional area is decreased, the simulations show that these fluctuations is decreased and dampens out faster. The initial conditions of the system is not stated correctly, oscillations in the system cause fluctuations around stationary water level at any VSSs placed downstream the turbine. The results indicate that this is due to the low head and for the VSSs placed upstream the turbine, these fluctuations do not occur as the head is much higher. With a decreased cross sectional area, the amplitude of the fluctuations is increased. The fluctuations are stable and for simulations with a turbine model without governor, they do not affect the system. If the governor is included and the amplitude of the fluctuations is significantly large, the turbine opening degree will be adjusted according to the fluctuations and the system will eventually become unstable. The cross sectional area is therefore suggested to be slightly higher if the system is complex and the governor is included in the turbine model. The correct area is dependent on the modeled systems complexity and composition. The VSS solution should therefore be used wisely.

The different turbine models were verified with hydropower system 1. The simulations show excellent correspondence with the simulations produced with LVTrans. For all simulations, the results of the maximum up surge and down surge deviates with less than 1.5 %. The three turbine models seems to act as intended. For the manual change of the opening degree with the Pelton turbine model, the simulated shaft surges coincides completely with the simulations from LVTrans. Even though the opening degree is restricted to 0.01 for fully closed wicket gates in uTubeOscV1, the turbine shut down with closing over two intervals provides good results. The Francis turbine model with governor adjusts the opening degree with satisfying accuracy compared to LVTrans, the κ response is virtually identical. The maximum down surge deviates with 1.42 % and the maximum up surge deviates with 0.95 %, which are relatively small deviations.

Hydropower system 2 was modeled and simulated to verify the program's ability to handle complex system geometries. The simulation of manual change on the opening degree shows that uTubeOscV1 handles the complexity of the system. The shaft surges is nearly equivalent and there is almost no deviation in the maximum up and down surge. The simulation of a change in power demand shows some small deviations from the simulation done with LVTrans. The turbine opening degree is changed slightly faster in LVTrans. The maximum down surge deviates with 2.99% and the maximum up surge deviates with 4.84%. The changes on the grid frequency were quite moderate, from 50Hz to 49Hz, and it is expected that the deviation, to some extend, will be larger if the grid frequency is changed to lower frequencies. Still, the deviations are relatively small. Therefore, it can be argued that the turbine model with frequency governor copes with a system of high complexity.

In general, the results show that uTubeOScV1 simulates with a high accuracy. The simulations show that the use of VSS is a feasible solution, provided the correct cross sectional area. The VSS cross sectional area was adjusted to the complexity of the two hydropower systems. For the simulations of hydropower system 1, the area was set to 0.01 m^2 and 0.001 m^2. For the simulations of system 2, the cross sectional area was set to 0.001 m^2 for the simulation with a Francis turbine model without governor. As the governor will act on large fluctuations in the VSSs downstream the turbine, the cross sectional area was set to 0.09 m^2 for the simulation with the Francis turbine model with frequency governor. For a system that has a different complexity and geometrical composition, the correct area might be different. However, the area is most likely to lay between 0.1 and 0.001 square meters.

A weakness of the program is that is has to reach equilibrium at steady state before any changes on the turbine opening degree can be made. This requires an unnecessary high simulation time and it is difficult to predict how long it will take before the equilibrium is reached.

The execution speed decreases as the cross sectional area of the virtual surge shafts decreases. Due to lack of time, any investigation on how to increase the execution speed was not performed.

The governor model used in uTubeOscV1 has different governing parameter types, compared to the governor in LVTrans. The parameters were tuned to match the behavior of the governor in LVTrans. It should be taken into account that these parameters might differ from parameters used in real governors. However, the intention was to verify the turbine models and the results are satisfying.

9 Conclusion and further work

9.1 Conclusion

Three generic turbine models are established for use in a developed Matlab simulation program. The models allow simulation of manual change on the turbine opening degree for Pelton and Francis turbines, as well as governing of changes in the power demand for a Francis turbine connected to the grid. The manual change of the opening degree can be done in intervals with different closing/opening speed. A fourth turbine model is suggested for further development. This model allows simulations of hydropower plants running on isolated grid.

An investigation on virtual surge shafts in one dimensional modeling shows that the use of virtual surge shafts is a feasible solution, provided the correct cross sectional area. Simulations with different cross sectional area shows that the impact on the system dynamics is strongly dependent on the size of the virtual surge shafts. A too large area causes oscillations between the virtual surge shafts and any free surfaces in the system. Virtual surge shafts exposed to a low head may experience fluctuations around stationary water level and with a significantly small area, a large amplitude of the fluctuations can make simulations with governing turbine models unstable. The results indicates that the correct cross sectional area is dependent on the system complexity and the system parameters. The virtual surge shafts should therefore be used with care. A cross sectional area between 0.1 and 0.001 square meters is suggested as a guideline.

The turbine models is implemented in the simulation program and virtual surge shafts are used to remove limitations in the system geometry. The program allows modeling of any hydropower systems, regardless of system composition and complexity.

Simulations with the program show good results compared to simulations produced with LVTrans. Out of five simulated scenarios with two hydropower systems of significantly different complexity, all simulations of the maximum up surge and down surge due to changes on the turbine opening degree, deviated with less than 5%. The turbine models act as intended and the results indicates that the program do handle systems with high complexity.

9.2 Recommendations

The feasibility of the use of virtual surge shafts, shows to be strongly dependent on the correct cross sectional area. Further investigation is recommended to determine if a empirical value of the correct cross sectional area can be established.

The method was verified with two hydropower systems of significantly different complexity, it should also be verified with systems of moderate complexity.

Based on what was emerged in this thesis, it is recommended to do a study on a friction model that is only dependent on the volume flow into the virtual surge shafts. This may reduce fluctuations around the steady state water level in virtual surge shafts exposed to a low head. With a friction model, a small cross sectional area can be used for simulations with turbine models that include governing.

9.3 Further work

For further development of the simulation program, the following is suggested:

- Include losses in the turbine models.
- Implement an algorithm to obtain steady state solution before a simulation is started.
- Utilize the suggested turbine model, for simulations of hydropower plants running on isolated grid.
- Investigate how to increase execution speed .

10 References

References

[Balchen et al., 2003] Balchen, J. G., Andresen, T., and Foss, B. A. (2003). *Reguleringsteknikk*. Institutt for teknisk kybernetikk.

[Brown, 2001] Brown, F. T. (2001). *Engineering System Dynamics*. Marcel Dekker Inc.

[Cengel and Cimbala, 2010] Cengel, Y. A. and Cimbala, J. M. (2010). *Fluid Mechanics Fundamentals and Applications*. McGraw-Hill Education.

[Fitzgerald et al., 2003] Fitzgerald, A. E., Jr., C. K., and Umans, S. D. (2003). *Electric Machinery, Sixth Edition*. McGraw-Hill Companies.

[Inc., 2015] Inc., M. (2015). ode45 - matlab r2015b documentation.

[Mathews and Fink, 2004] Mathews, J. H. and Fink, K. K. (2004). Runge-kutta methods. In *Numerical Methods Using Matlab*. Prentice-Hall Inc., Upper Saddle River, New Jersey, USA, 4th edition.

[Nielsen, 1990a] Nielsen, T. K. (1990a). Dynamisk dimensjonering av vannkraftverk. Technical report, NTNU Vannkraftlaboratoriet.

[Nielsen, 1990b] Nielsen, T. K. (1990b). *Transient Characteristics of high head Francis turbines*. PhD thesis, NTH.

[Nielsen, 1996] Nielsen, T. K. (1996). Dynamic behaviour of governing turbines sharing the same grid. Technical report, Kvaerner Energy AS.

[Nielsen, 2015] Nielsen, T. K. (2015). Simulation model for francis and reversible pump turbines. Technical report, Department of Energy and Process Engineering NTNU.

[Svee, 1972] Svee, R. (1972). Fordelingsbasseng ved vannkraftverk: Formler til bestemmelse av svingegrenser og kritiske tverrsnitt. Technical report, NTH Vassdrags- og havnelaboratoriet.

[Svingen, 2003] Svingen, B. (2003). Documentation for lvtrans(labview transient pipe analysis). Technical report, SINTEF Energy Research AS.

[Svingen, 2015] Svingen, B. (2015). *Manual for LVTrans*, 2014-1.9.11 edition.

[Valaamo, 2015] Valaamo, S. B. (2015). Modelling and simulation of hydropower plants. Project thesis, NTNU.

[Wylie and Streeter, 1983] Wylie, E. B. and Streeter, V. L. (1983). *Fluid Transients*. FEB PRESS.

[Wylie and Streeter, 1993] Wylie, E. B. and Streeter, V. L. (1993). *Prentice-Hall Inc.* FEB PRESS.

[Young et al., 2004] Young, H. D., Freedman, R. A., and Ford, A. L. (2004). *University Physics with Modern Physics*. Pearson Education Inc., 11th edition.

A Derivations

A.1 Head difference between turbine inlet and outlet

From the main geometry at inlet and outlet of a reaction turbine, the difference in hydraulic energy can be expressed as:

$$gH_t = u_1 c_{u1} - u_2 c_{u2} \tag{A.1.1}$$

The cosines sentence applied on the turbine inlet velocity diagram gives:

$$v_1^2 = u_1^2 + c_1^2 - 2u_1 c_1 \cos(\alpha_1) \tag{A.1.2}$$

From the diagram:

$$\sin(\alpha_1) = \frac{c_{u1}}{c_1} \tag{A.1.3}$$

This, inserted in the previous equation gives:

$$v_1^2 = u_1^2 + c_1^2 - 2u_1 c_{u1} \tag{A.1.4}$$

Sorting the equation:

$$u_1 c_{u1} = \frac{1}{2}c_1^2 - \frac{1}{2}v_1^2 + \frac{1}{2}u_1^2 \tag{A.1.5}$$

With the same procedure, the expression for the outlet is found as:

$$u_2 c_{u2} = \frac{1}{2}c_2^2 - \frac{1}{2}v_2^2 + \frac{1}{2}u_2^2 \tag{A.1.6}$$

Inserting Eq.(A.1.5) and Eq.(A.1.6) in Eq.(A.1.1) gives:

$$gH_t = \frac{1}{2}(c_1^2 - c_2^2) - \frac{1}{2}(v_1^2 - v_2^2) + \frac{1}{2}(u_1^2 - u_2^2) \tag{A.1.7}$$

with $u = \omega r = \omega \frac{D}{2}$, the last term can be expressed as:

$$\frac{1}{2}(u_1^2 - u_2^2) = \frac{\omega^2}{8}(D_1^2 - D_2^2) = \frac{\omega^2}{8}D_1^2(1 - \frac{D_2^2}{D_1^2}) = s\omega^2 \tag{A.1.8}$$

where

$$s = \frac{1}{8}D_1^2(1 - \frac{D_2^2}{D_1^2}) \tag{A.1.9}$$

B Program code

B.1 uTubeOscV1

```
%% uTubeOscV1 - Transient modeling of hydropower plants and simulation of U-tube oscillati
% By Simen Bomnes Valaamo, Hydropower Laboratory NTNU

%% Constants
global n X Hupper Hlower deltaR A g rho md TYPE numEq Anode

g = 9.81;
rho = 1000;
md = 0.01;
deltaR = 15;
deltaStart = 12.45557;

%% Input from user

% Textfile
prompt1 = {'Enter text file(textfile.txt):'};
name1 = 'Text file';
ans1 = inputdlg(prompt1,name1);
textfile = char(ans1);

% Water level at reservoirs
promptH = {'Upper reservoir, H_upper:','Lower reservoir, H_lower:'};
nameH = 'Head at reservoirs';
ansH = inputdlg(promptH,nameH);
tempH = char(ansH);
ANSH = str2num(tempH);

Hupper = ANSH(1,1);
Hlower = ANSH(2,1);

% Cross sectional area of VSSs
promptNode = {'VSS area(Between 0.1 and 0.001):'};
nameNode = 'VSS area';
ansNode = inputdlg(promptNode,nameNode);
tempNode = char(ansNode);
ANSNode = str2num(tempNode);

Anode = ANSNode(1,1);

% Number of turbines
promptNT = {'Number of turbines in the system:'};
nameNT = 'Turbines';
ansNT = inputdlg(promptNT,nameNT);
tempNT = char(ansNT);
ANSNT = str2num(tempNT);
A = zeros(ANSNT,7);
```

```
TYPE = zeros(1,ANSNT);

% Turbine type
for iter = 1:1:ANSNT
    choice1 = questdlg('What type of turbine is modeled?',...
        'Turbine type','Pelton','Francis','Pelton');
    switch choice1
        case 'Pelton'
            prompt2 = {'Element nr. for the turbine:',...
                'Starting value of kappa, K_start:',...
                'Time for start of change[s}:',...
                'First new value of kappa, K_mid:',...
            'Time from K_start to K_mid[s]:', ...
            'Ending value of kappa, K_end:', ...
            'Time from K_mid to K_end[s]:'};
            name2 = 'Manual change of kappa';
            answer = inputdlg(prompt2,name2);
            temp = char(answer);
            a = str2num(temp);
            A(iter,:) = a;
            TYPE(iter) = 1; % kappa manual
        case 'Francis'
            choice2 = questdlg('What type of turbine model?',...
                'Francis turbine',...
                'Simple-Grid(32)','W/Freq.gov-Grid(33)',...
                'W/Freq.gov-Island(34)','Simple-Grid(32)');
            switch choice2
                case 'Simple-Grid(32)'
                    prompt2 = {'Element nr. for the turbine:',...
                        'Starting value of kappa, K_start:',...
                        'Time for start of change[s]:',...
                        'First new value of kappa, K_mid:',...
                        'Time from K_start to K_mid[s]:', ...
                        'Ending value of kappa, K_end:', ...
                        'Time from K_mid to K_end[s]:'};
                    name2 = 'Manual change of kappa';
                    answer = inputdlg(prompt2,name2);
                    temp = char(answer);
                    a = str2num(temp);
                    A(iter,:) = a;
                    TYPE(iter) = 1; % kappa manual
                case 'W/Freq.gov-Grid(33)'
                    prompt2 = {'Element nr. for the turbine:',...
                        'Time for start of change[s]:',...
                        'New grid frequency[Hz]:'};
                    name2 = 'Change of grid frequency';
                    answer = inputdlg(prompt2,name2);
                    temp = char(answer);
                    a = str2num(temp);
                    A(iter,1:3) = a;
                    TYPE(iter) = 3; % grid frequency
                case 'W/Freq.gov-Island(34)'
```

```
                    prompt = msgbox('The model is not in use',...
                        'Error','error');
            end
    end
end

% Simulation time
promptTime = {'Total simulation time, tmax:'};
nameTime = 'Simulation time';
TIME = inputdlg(promptTime,nameTime);
tempTime = char(TIME);
tmax = str2num(tempTime);

%% Import data from text file

fID = fopen(textfile);
C = textscan(fID, '%f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f %f
             'Delimiter',{',',';',' '}, 'CommentStyle', '%',...
             'TreatAsEmpty',{'NaN'}, 'CollectOutput',1);
fclose(fID);

Y = cell2mat(C);
[totR,totC] = size(Y);

X = zeros(totR,totC);

for i = 1:totR
    X(i,:) = Y(i,:);
end

[n,m] = size(X);

%% Decide the total number of eqns.

Extra = 0; % Counter, additional eqns
for i = 1:1:n
    if (X(i,1)==31)
        Extra = Extra + 2;
    elseif (X(i,1)==32)
        Extra = Extra + 4;
    elseif (X(i,1)==33)
        Extra = Extra + 6;
    elseif (X(i,1)==34)
        Extra = Extra + 7;
    end
end

numEq = n + Extra; % Total number of eqns

%% ode45 solver

% Initial values for dynamic eqns
```

```
figVal = zeros(n,2);
f0 = zeros(1,numEq);
j = 1;
for i = 1:1:n
    if (X(i,1)==1) % Pipe
        f0(j) = X(i,4);
        j = j+1;
    elseif (X(i,1)==2) % S.shaft
        figVal(i,1) = i;
        figVal(i,2) = j;
        f0(j) = X(i,4);
        j = j+1;
    elseif (X(i,1)==4) % Pipe-pipe connection
        f0(j) = X(i,4);
        j = j+1;
    elseif (X(i,1)==5) % Branch
        f0(j) = X(i,2);
        j = j+1;
    elseif (X(i,1)==6) % Bound
        f0(j) = X(i,2);
        j = j+1;
    elseif (X(i,1)==31) % Pelton turbine
        f0(j) = X(i,4);
        f0(j+1) = X(i,8);
        f0(j+2) = X(i,9);
        j = j+3;
    elseif (X(i,1)==32) % Francis turbine, grid mode
        qt = 1;
        for q = 1:1:ANSNT
            if (A(qt,1)==i)
                deltaConstant = A(qt,2);
                break
            else
                qt = qt+1;
            end
        end
        f0(j) = X(i,4);
        f0(j+1) = X(i,18);
        f0(j+2) = X(i,19);
        f0(j+3) = X(i,17);
        f0(j+4) = deltaConstant*deltaStart;
        j = j+5;
    elseif (X(i,1)==33) % Francis turbine w/freq.governor, grid mode
        f0(j) = X(i,4);
        f0(j+1) = X(i,22);
        f0(j+2) = X(i,23);
        f0(j+3) = X(i,17);
        f0(j+4) = deltaStart;
        f0(j+5) = 1;
        f0(j+6) = 0;
        j = j+7;
    elseif (X(i,1)==34) % Francis turbine w/freq.governor, island mode
```

```
        f0(j) = X(i,4);
        f0(j+1) = X(i,23);
        f0(j+2) = X(i,24);
        f0(j+3) = X(i,17);
        f0(j+4) = deltaStart;
        f0(j+5) = 1;
        f0(j+6) = 0;
        f0(j+7) = X(i,22);
        j = j+8;
    end
end

% Solver for the set of dynamic eqns, Runge-Kutta-Fehlberg

tspan = [0 tmax];
[T,f] = ode45(@dynsysV1,tspan,f0);

%% Plots

figN = 0;

for p=1:1:n

    if(X(p,1)==2)
        tempF = figVal(p,2);
        figN = figN + 1;
        figure(figN)
        title(['Surge shaft: Element ' num2str(p)])
        axis auto
        xlabel('Time [s]')
        ylabel('Water level [m]')
        grid
        hold on
        plot(T,f(:,tempF))
    end
end
```

B.2 dynsysV1

```
%% uTubeOscV1: dynsysV1 - Function giving the set of dynamic equations
% By Simen Bomnes Valaamo, Hydropower Laboratory NTNU

function df=dynsysV1(t,f)
%% Constants and variables

global n X Hupper Hlower A TYPE deltaR g rho md numEq Anode

%% Initialize dfCount and PMAX
c = 1; % Initialize counter, df
turb = 0; % Initialize counter, PMAX
```

```
PMAX = zeros(1,n);
dfCount = zeros(n,9);
for i = 1:1:n
    if (X(i,1)==31)
        turb = turb+1;
        dfCount(i,1) = i;
        dfCount(i,2) = c;
        dfCount(i,3) = c+1;
        dfCount(i,4) = c+2;
        c = c+3;
    elseif (X(i,1)==32)
        turb = turb+1;
        dfCount(i,1) = i;
        dfCount(i,2) = c;
        dfCount(i,3) = c+1;
        dfCount(i,4) = c+2;
        dfCount(i,5) = c+3;
        dfCount(i,6) = c+4;
        c = c+5;
    elseif (X(i,1)==33)
        turb = turb+1;
        dfCount(i,1) = i;
        dfCount(i,2) = c;
        dfCount(i,3) = c+1;
        dfCount(i,4) = c+2;
        dfCount(i,5) = c+3;
        dfCount(i,6) = c+4;
        dfCount(i,7) = c+5;
        dfCount(i,8) = c+6;
        PMAX(turb) = X(i,14);
        c = c+7;
    elseif (X(i,1)==34)
        turb = turb+1;
        dfCount(i,1) = i;
        dfCount(i,2) = c;
        dfCount(i,3) = c+1;
        dfCount(i,4) = c+2;
        dfCount(i,5) = c+3;
        dfCount(i,6) = c+4;
        dfCount(i,7) = c+5;
        dfCount(i,8) = c+6;
        dfCount(i,9) = c+7;
        PMAX(turb) = X(i,14);
        c = c+8;
    else
        dfCount(i,1) = i;
        dfCount(i,2) = c;
        c = c+1;
    end
end

%% Change of opening degree
```

```
kappa = zeros(1,n);
P = zeros(1,n);
kapparef = zeros(1,n);
fgrid = zeros(1,n);
for m = 1:1:turb
    if (TYPE(m)==1)
        tnr = A(m,1);
        if t < A(m,3)
            kappa(tnr) = A(m,2);
        elseif t < (A(m,3)+A(m,5))
            dkappa = (A(m,2)-A(m,4))/A(m,5);
            kappa(tnr) = A(m,2) - (t-A(m,3))*dkappa;
        elseif t < (A(m,3)+A(m,5)+A(m,7))
            dkappa = (A(m,4)-A(m,6))/A(m,7);
            kappa(tnr) = A(m,4) - (t-A(m,3)-A(m,5))*dkappa;
        else
            kappa(tnr) = A(m,6);
        end

    elseif (TYPE(m)==2)
        tnr = A(m,1);
        if t < A(m,2)
            P(tnr) = PMAX(m);
        elseif t < A(m,4)
            P(tnr) = A(m,3);
        else
            P(tnr) = A(m,5);
        end
        kapparef(tnr) = P(tnr)/PMAX(m);

    elseif (TYPE(m)==3)
        tnr = A(m,1);
        if t < A(m,2)
            fgrid(tnr) = X(tnr,15);
        else
            fgrid(tnr) = A(m,3);
        end
    end
end

%% Set of dynamic eqns.

df = zeros(numEq,1);
for i = 1:1:n
    next = X(i,3);
    prev = X(i,2);
    iEl = dfCount(i,2);
   if (X(i,1)==1) % Element i: Pipe
       if (prev==0) % First element
           if (X(next,1)==31 ||X(next,1)==32 ||...
                   X(next,1)==33 || X(next,1)==34) % Next el: turbine
               b = dfCount(next,3);
```

```
            else % Next el: not turbine
                b = dfCount(next,2);
            end
            k = X(i,8)*X(i,6)/(2*g*X(i,7)*((X(i,5))^2));
            df(iEl) = (g*X(i,5)/X(i,6))*...
                (Hupper - f(b) - k*f(iEl)*abs(f(iEl))); % dQ/dt
        elseif (next==0) % Last element
            if (X(prev,1)==31 || X(prev,1)==32 ...
                    || X(prev,1)==33 || X(prev,1)==34) % Prev el: turbine
                a = dfCount(prev,4);
            else % Prev el: not turbine
                a = dfCount(prev,2);
            end
            k = X(i,8)*X(i,6)/(2*g*X(i,7)*((X(i,5))^2));
            df(iEl) = (g*X(i,5)/X(i,6))*(f(a) - ...
                Hlower - k*f(iEl)*abs(f(iEl))); % dQ/dt
        else
            if (X(prev,1)==31 || X(prev,1)==32 ||...
                    X(prev,1)==33 || X(prev,1)==34) % Prev el: turbine
                if (X(next,1)==31 || X(next,1)==32 ...
                        || X(next,1)==33 || X(next,1)==34) % Next el: turbine
                    a = dfCount(prev,4);
                    b = dfCount(next,3);
                else % Next el: not turbine
                    a = dfCount(prev,4);
                    b = dfCount(next,2);
                end
            else  % Prev el: not turbine
                if (X(next,1)==31 || X(next,1)==32 || X(next,1)==33 ...
                    || X(next,1)==34) % Next el: turbine
                    a = dfCount(prev,2);
                    b = dfCount(next,3);
                else % Next el: not turbine
                    a = dfCount(prev,2);
                    b = dfCount(next,2);
                end
            end
            k = X(i,8)*X(i,6)/(2*g*X(i,7)*((X(i,5))^2));
            df(iEl) = (g*X(i,5)/X(i,6))*(f(a) - f(b) -...
                k*f(iEl)*abs(f(iEl))); % dQ/dt
        end
    elseif (X(i,1)==2) % Element i: S.shaft
        a = dfCount(prev,2);
        b = dfCount(next,2);
        df(iEl) = (f(a)-f(b))/X(i,5);
    elseif (X(i,1)==4) % Element i: Pipe-pipe connection
        a = dfCount(prev,2);
        b = dfCount(next,2);
        df(iEl) = (f(a)-f(b))/Anode;
    elseif (X(i,1)==5) % Element i: Branch
        prevBranch = X(i,4);
        aBranch = dfCount(prevBranch,2);
```

```
    nOut = X(i,3);
    fOut = zeros(1,nOut);
    for k = 1:1:nOut
        temp = X(i,4+k);
        btemp = dfCount(temp,2);
        fOut(k) = f(btemp);
    end
    sumfOut = sum(fOut);
    df(iEl) = (f(aBranch)-sumfOut)/Anode;
elseif (X(i,1)==6) % Element i: Bound
    nextBound = X(i,4);
    bBound = dfCount(nextBound,2);
    nIn = X(i,3);
    fIn = zeros(1,nIn);
    for k = 1:1:nIn
        temp = X(i,4+k);
        atemp = dfCount(temp,2);
        fIn(k) = f(atemp);
    end
    sumfIn = sum(fIn);
    df(iEl) = (sumfIn-f(bBound))/Anode;
elseif (X(i,1)==31) % Element i: Pelton turbine
    a = dfCount(prev,2);
    b = dfCount(next,2);
    before = dfCount(i,3);
    after = dfCount(i,4);
    Ht = X(i,6)*((f(iEl)./(kappa(i)*X(i,7)))^2); % Pressure over turbine
    df(iEl) = (1/X(i,5))*(f(before) - f(after) - Ht); % dQ/dt
    df(before) = (f(a)-f(iEl))/Anode; % Node before
    df(after) = (f(iEl)-f(b))/Anode; % Node after
elseif (X(i,1)==32) % Element i: Francis turbine, grid mode
    a = dfCount(prev,2);
    b = dfCount(next,2);
    before = dfCount(i,3);
    after = dfCount(i,4);
    Omega = dfCount(i,5);
    Delta = dfCount(i,6);
    wgrid = 2*pi*X(i,15);
    s = (X(i,8)^2)*(1 - (X(i,9)^2)/(X(i,8)^2))/8; % Turbine geometry
    rA1 = 1/(2*pi*X(i,10)); % rt1/At1
    rA2 = 2/(pi*X(i,9)); % rt2/At2
    alpha = abs(asind(kappa(i)*sind(X(i,11)))); % Guide vane angle
    TgR = 1.2*(rho*X(i,7)*(rA1*X(i,7)*cotd(X(i,11)) ...
        + rA2*X(i,7)*cotd(X(i,12)) -...
        (X(i,9)^2)*X(i,17)/4)); % Rated generator torque
    Tg = TgR*sind(f(Delta))/sind(deltaR); % Generator torque
    Ht = X(i,6)*((f(iEl)./(kappa(i)*X(i,7)))^2) + s*(f(Omega)^2 ...
    - X(i,17)^2)/g; % Pressure over turbine
    df(iEl) = (1/X(i,5))*(f(before) - f(after) - Ht); % dQ/dt
    df(before) = (f(a)-f(iEl))/Anode; % Node before
    df(after) = (f(iEl)-f(b))/Anode; % Node after
    df(Omega) = (X(i,17)^2)/(X(i,14)*(10^6)*X(i,16))*...
```

```
        (rho*f(iEl)*(rA1*f(iEl)*cotd(alpha) +...
        rA2*f(iEl)*cotd(X(i,12)) - (X(i,9)^2)*f(Omega)/4)...
        - Tg - md*(X(i,13)*f(Omega) - wgrid)); % d(omega)/dt
    df(Delta) = X(i,13)*f(Omega) - wgrid; % d(delta)/dt

elseif (X(i,1)==33) % Element i: Francis turbine w/freq.governor, grid mode
    a = dfCount(prev,2);
    b = dfCount(next,2);
    before = dfCount(i,3);
    after = dfCount(i,4);
    Omega = dfCount(i,5);
    Delta = dfCount(i,6);
    KAPPA = dfCount(i,7);
    Servo = dfCount(i,8);
    wgrid = 2*pi*fgrid(i);
    s = (X(i,8)^2)*(1 - (X(i,9)^2)/(X(i,8)^2))/8; % Turbine geometry
    rA1 = 1/(2*pi*X(i,10)); % rt1/At1
    rA2 = 2/(pi*X(i,9)); % rt2/At2
    alpha = abs(asind(f(KAPPA)*sind(X(i,11)))); % Guide vane angle
    TgR = 1.2*(rho*X(i,7)*(rA1*X(i,7)*cotd(X(i,11)) + rA2*X(i,7)...
        *cotd(X(i,12)) - (X(i,9)^2)*X(i,17)/4)); % Rated generator torque
    Tg = TgR*sind(f(Delta))/sind(deltaR); % Generator torque
    Ht = X(i,6)*((f(iEl)./(f(KAPPA)*X(i,7)))^2) + s*(f(Omega)^2 ...
    - X(i,17)^2)/g; % Pressure over turbine
    df(iEl) = (1/X(i,5))*(f(before) - f(after) - Ht); % dQ/dt
    df(before) = (f(a)-f(iEl))/Anode; % Node before
    df(after) = (f(iEl)-f(b))/Anode; % Node after
    df(Omega) = (X(i,17)^2)/(X(i,14)*(10^6)*X(i,16))*...
        (rho*f(iEl)*(rA1*f(iEl)*cotd(alpha) +...
        rA2*f(iEl)*cotd(X(i,12)) - (X(i,9)^2)*f(Omega)/4)...
        - Tg - md*(X(i,13)*f(Omega) - wgrid)); % d(omega)/dt
    df(Delta) = X(i,13)*f(Omega) - wgrid; % d(delta)/dt
    df(KAPPA) = f(Servo); % d(kappa)/dt
    df(Servo) = (1/X(i,18))*(-(1/X(i,20)/X(i,17))*((X(i,17)^2)...
        /(X(i,14)*(10^6)*X(i,16))*...
        (rho*f(iEl)*(rA1*f(iEl)*cotd(alpha) +...
        rA2*f(iEl)*cotd(X(i,12)) - (X(i,9)^2)*f(Omega)/4) - ...
        Tg - md*(X(i,13)*f(Omega) - wgrid))) +...
        (1/X(i,20)/X(i,19))*((X(i,17)-f(Omega))/X(i,17)) -...
        (X(i,21)*X(i,18)+X(i,20)*X(i,19))/...
        (X(i,20)*X(i,19))*f(Servo) -...
        X(i,21)/(X(i,20)*X(i,19))*(1-f(KAPPA))); % dc/dt

elseif (X(i,1)==34) % Element i: Francis turbine w/freq.governor, island mode
    a = dfCount(prev,2);
    b = dfCount(next,2);
    before = dfCount(i,3);
    after = dfCount(i,4);
    Omega = dfCount(i,5);
    Delta = dfCount(i,6);
    KAPPA = dfCount(i,7);
    Servo = dfCount(i,8);
```

```
        Volt = dfCount(i,9);
        wgrid = 2*pi*X(i,15); % Angular grid frequency
        s = (X(i,8)^2)*(1 - (X(i,9)^2)/(X(i,8)^2))/8; % Turbine geometry
        rA1 = 1/(2*pi*X(i,10)); % rt1/At1
        rA2 = 2/(pi*X(i,9)); % rt2/At2
        kphi = X(i,22)/wgrid; % Magnetic flux
        T0 = X(i,14)*(10^6)/X(i,17); % Start torque
        phi = 0; % Phase angle
        Rgrid = X(i,22)*kphi*cos(phi)/T0; % Ohmic resistance on the grid
        alpha = abs(asind(f(KAPPA)*sind(X(i,11)))); % Guide vane angle
        TgR = 1.2*(rho*X(i,7)*(rA1*X(i,7)*cotd(X(i,11)) + ...
            rA2*X(i,7)*cotd(X(i,12)) - ...
            (X(i,9)^2)*X(i,17)/4)); % Rated generator torque
        Tg = TgR*sind(f(Delta))/sind(deltaR); % Generator torque
        Ht = X(i,6)*((f(iEl)./(f(KAPPA)*X(i,7)))^2) + s*(f(Omega)^2 ...
        - X(i,17)^2)/g; % Pressure over turbine
        df(iEl) = (1/X(i,5))*(f(before) - f(after) - Ht); % dQ/dt
        df(before) = (f(a)-f(iEl))/Anode; % Node before
        df(after) = (f(iEl)-f(b))/Anode; % Node after
        df(Omega) = (X(i,17)^2)/(X(i,14)*(10^6)*X(i,16))*...
            (rho*f(iEl)*(rA1*f(iEl)*cotd(alpha) +...
            rA2*f(iEl)*cotd(X(i,12)) - (X(i,9)^2)*f(Omega)/4) - Tg...
            - md*(X(i,13)*f(Omega) - f(Volt)/kphi)); % d(omega)/dt
        df(Delta) = X(i,13)*f(Omega) - f(Volt)/kphi; % d(delta)/dt
        df(KAPPA) = f(Servo); % d(kappa)/dt
        df(Servo) = (kapparef(i)/X(i,18))*(-(1/X(i,20)/X(i,17))*...
            ((X(i,17)^2)/(X(i,14)*(10^6)*X(i,16))*...
            (rho*f(iEl)*(rA1*f(iEl)*cotd(alpha) +...
            rA2*f(iEl)*cotd(X(i,12)) - (X(i,9)^2)*f(Omega)/4)...
            - Tg - md*(X(i,13)*f(Omega) - f(Volt)/kphi))) +...
            (1/X(i,20)/X(i,19))*((X(i,17)-f(Omega))/X(i,17))...
            - (X(i,21)*X(i,18)+X(i,20)*X(i,19))/...
            (X(i,20)*X(i,19))*f(Servo) -...
            X(i,21)/(X(i,20)*X(i,19))*(kapparef(i)-f(KAPPA))); % dc/dt
        df(Volt) = Rgrid*Tg/(kphi*cos(phi)) - f(Volt); % dE/dt

    end
end
end
```

C Textfile template

```
 ________________________________________________________________________________
%
%                              SIMULATION OF U-TUBE OSCILLATIONS WITH uTubeOscV1
%
%   The hydropower system is described in this file, with all the elements forming the system. Each line
%   represents one element.
%
%   The different elements are listed as described below, remember one element per line.
%   The elements must be listed in the correct order such that element number one is defined in the
%   first  line, element two in the second line and so on.
%
%   Pipe(=1)              ->  1; prev.El; nextEl; Q_0; A; L; D; f
%   S.shaft(=2)           ->  2; prev.El; nextEl; z_0; A_s
%   Pelton(=31)           ->  31; prev.El; nextEl; Q_0; I_h; H_R; Q_R; H_t1; H_t2
%   Francis,
%   grid mode(=32)        ->  32; prev.El; nextEl; Q_0; I_h; H_R; Q_R; D_t1; D_t2; B_t1; alpha_1R; beta_2;
%                             Pole-pairs; P_max; T_a; omega_R; H_t1; H_t2
%   Francis
%   w/governor,
%   grid mode(=33)        ->  33; prev.El; nextEl; Q_0; I_h; H_R; Q_R; D_t1; D_t2; B_t1; alpha_1R; beta_2;
%                             Pole-pairs; P_max; T_a; omega_R; T_K; T_i; b_t; b_p; H_t1; H_t2
%   Francis
%   w/governor,
%   Island mode(=32)  ->  34; prev.El; nextEl; Q_0; I_h; H_R; Q_R; D_t1; D_t2; B_t1; alpha_1R; beta_2;
%                             Pole-pairs; P_max; T_a; omega_R; T_K; T_i; b_t; b_p; E; H_t1; H_t2
%   Node(=4)              ->  4; prev.El; nextEl; H_0
%   Branch(=5)            ->  5; H_0; numberOfElAfter; prev.El; nextEl.1; nextEl.2; ...; nextEl.n
%   Bound(=6)             ->  6; H_0; numberOfElBefore; nextEl; prev.El.1; prev.El.2; ...; prev.El.n
%
%   The data is separated with comma, colon, semicolon or space
% ________________________________________________________________________________


% The elements are listed below:

1; prev.El; nextEl; Q_0; A; L; D; f
2; prev.El; nextEl; z_0; A_s
1; prev.El; nextEl; Q_0; A; L; D; f
31; prev.El; nextEl; Q_0; I_h; H_R; Q_R; H_t1; H_t2
1; prev.El; nextEl; Q_0; A; L; D; f
2; prev.El; nextEl; z_0; A_s
1; prev.El; nextEl; Q_0; A; L; D; f
```

D Virtual surge shaft simulations

D.1 Geometrical data for the hydropower system

Table D.1.1: Geometrical data for the system with VSSs

Element	**A[m^2]**	**L[m]**	**Friction factor**
1	12.57	1750	0.05
3	12.57	1750	0.05
4	177		
5	13.19	200	0.02
7	13.19	150	0.02
11	13.19	20	0.02
12	78		
13	12.57	2000	0.05

Table D.1.2: Geometrical data for reference system

Element	**A[m^2]**	**L[m]**	**Friction factor**
1	12.57	3000	0.05
2	177		
3	13.19	370	0.02
5	78		
6	12.57	2000	0.05

D.2 Plots

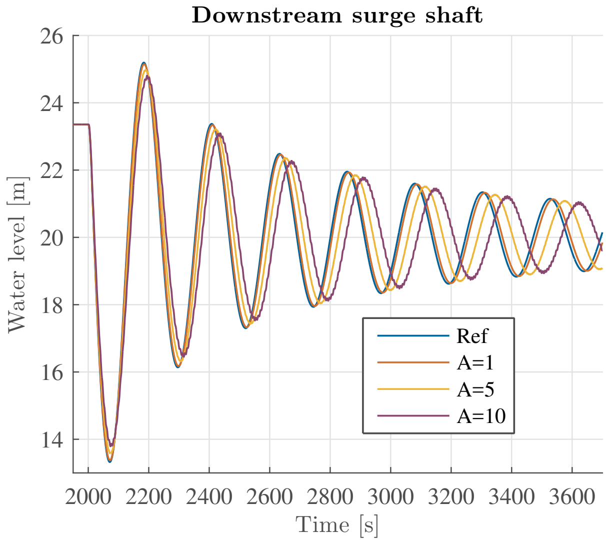


Figure D.2.1: Downstream surge shaft: Areas of 1 to 10 m^2

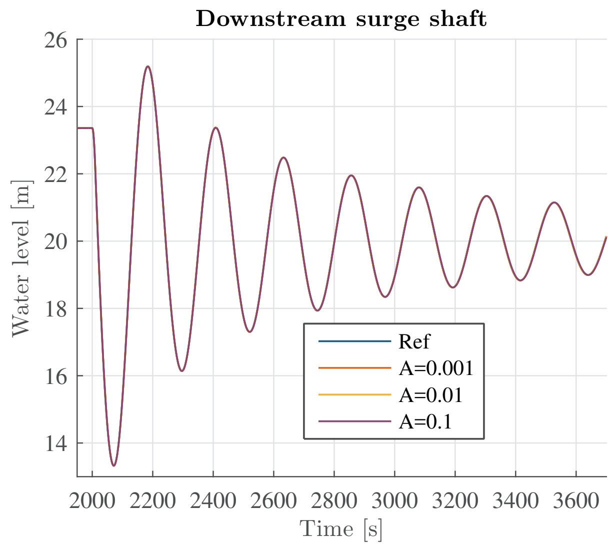


Figure D.2.2: Downstream surge shaft: Areas of 0.1 to 0.001 m^2

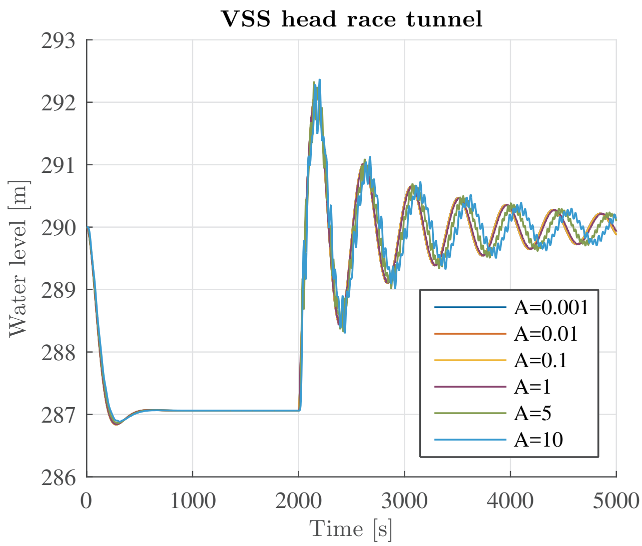


Figure D.2.3: Areas of 0.001 to 10 m^2

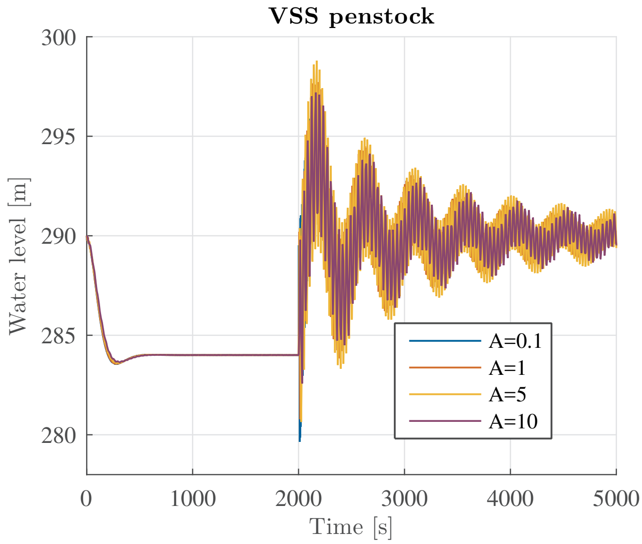


Figure D.2.4: Areas of 0.1 to 10 m^2

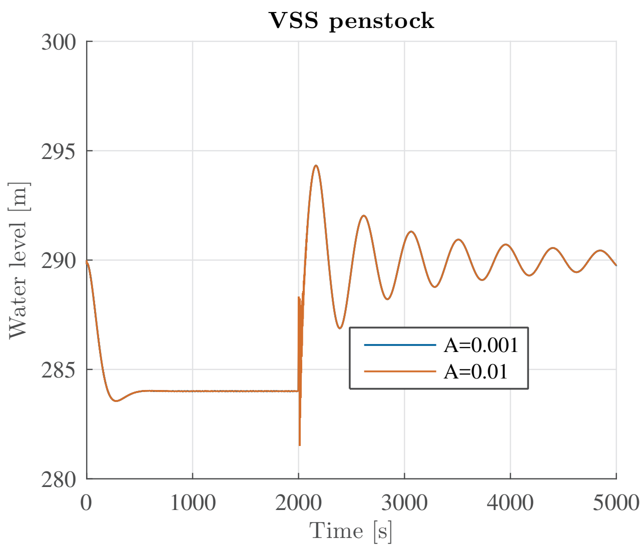


Figure D.2.5: Areas of 0.01 to 0.001 m^2

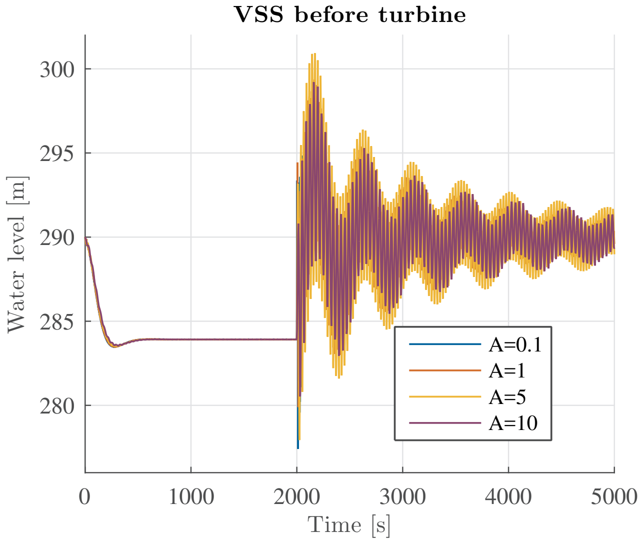


Figure D.2.6: Areas of 0.1 to 10 m^2

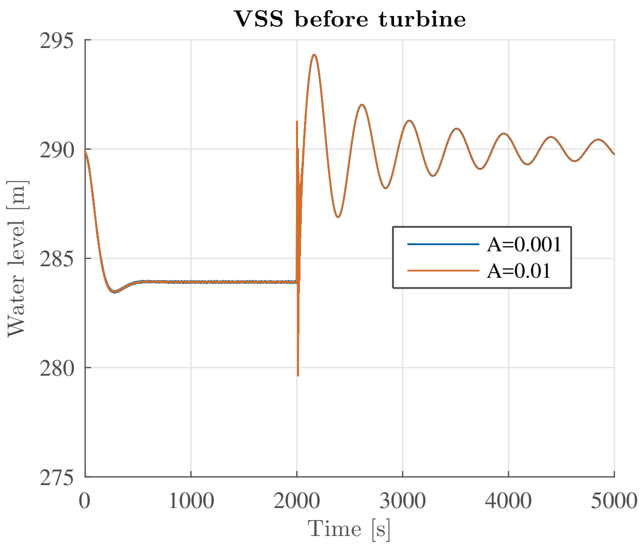


Figure D.2.7: Areas of 0.01 to 0.001 m^2

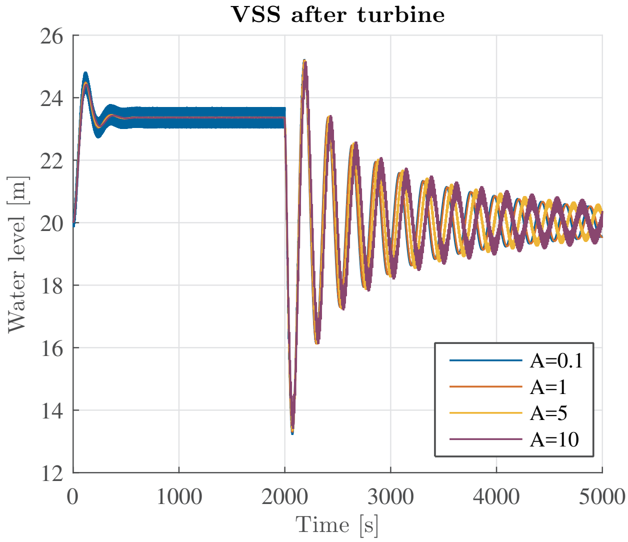


Figure D.2.8: Areas of 0.1 to 10 m^2

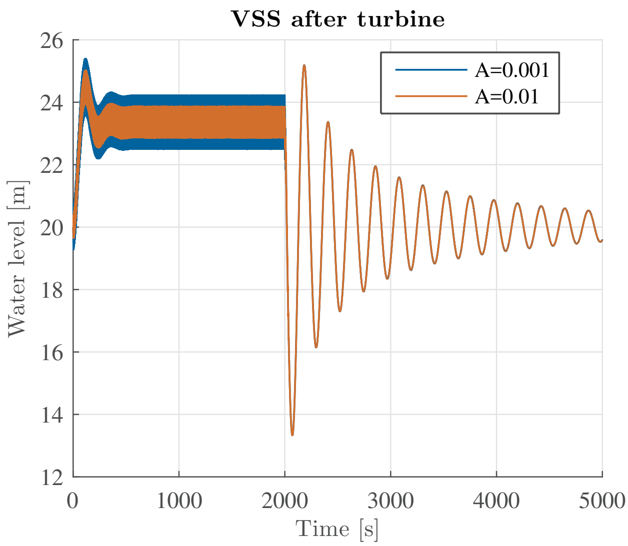


Figure D.2.9: Areas of 0.01 to 0.001 m^2

E Verification uTubeOscV1

E.1 Geometrical data for Francis turbines and governor

Table E.1.1: Turbine geometry

D_{t1}	D_{t2}	B_{t1}	α_{1R}	β_2
2.0305	1.5279	0.3163	12.1287	15.8088

Table E.1.2: Turbine and governor parameters

Turbine		Governor	
Poles	12	f_{grid}	50Hz
n	500RPM	b_t	0.6
P_{max}	52.85MW	b_p	-0.06
T_a	6 s	T_i	4
Q_R	20.7649 m^3/s	T_K	0.85
H_R	270 m		

E.2 LVTrans

E.2.1 Governing parameters

Table E.2.1: Governing parameters

P	T_i	T_d	Droop	P_R [MW]
5.5	8	0	0.06	52.85

E.2.2 System layout

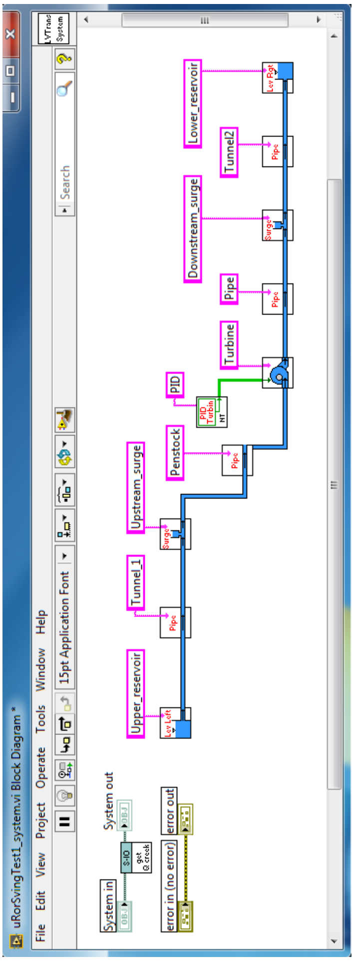


Figure E.2.1: Hydropower system 1

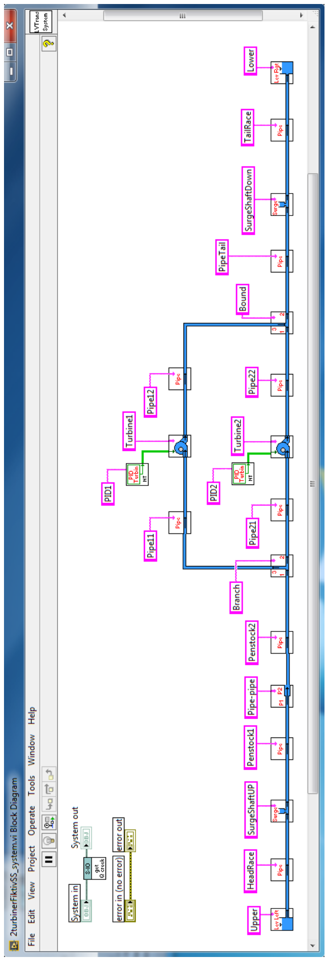
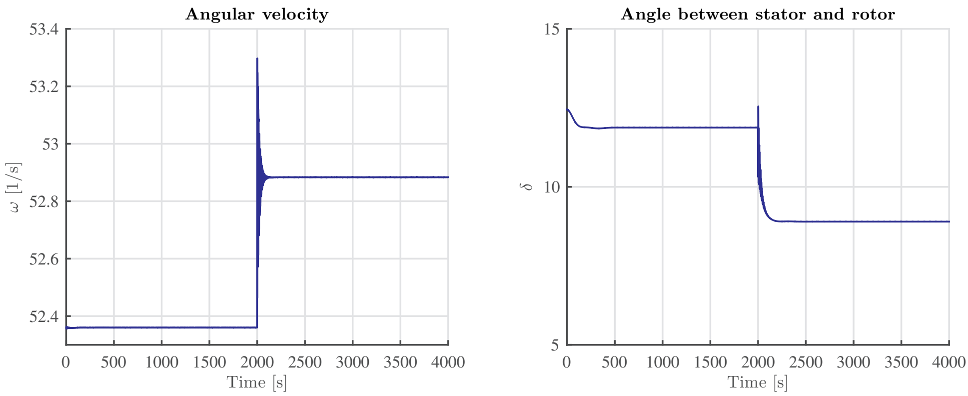


Figure E.2.2: Hydropower system 2

E.3 Hydropower system 1

E.3.1 Decreas in power demand on the grid

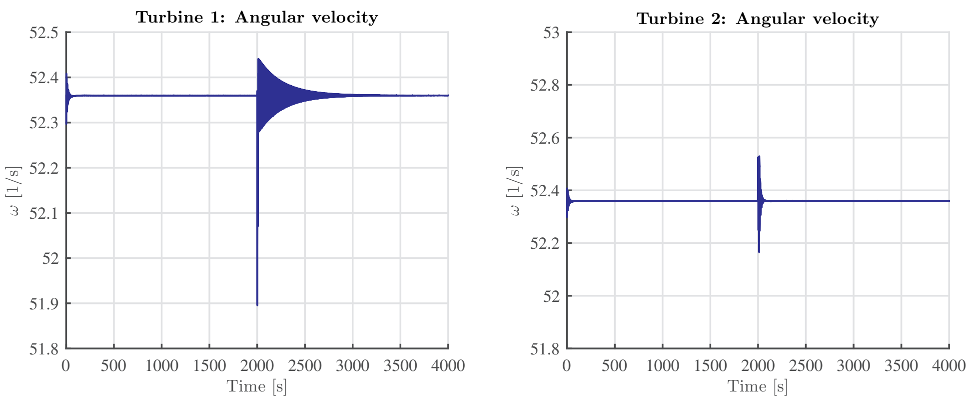


(a) Angular velocity (b) Torque angle

Figure E.3.1: Turbine characteristics

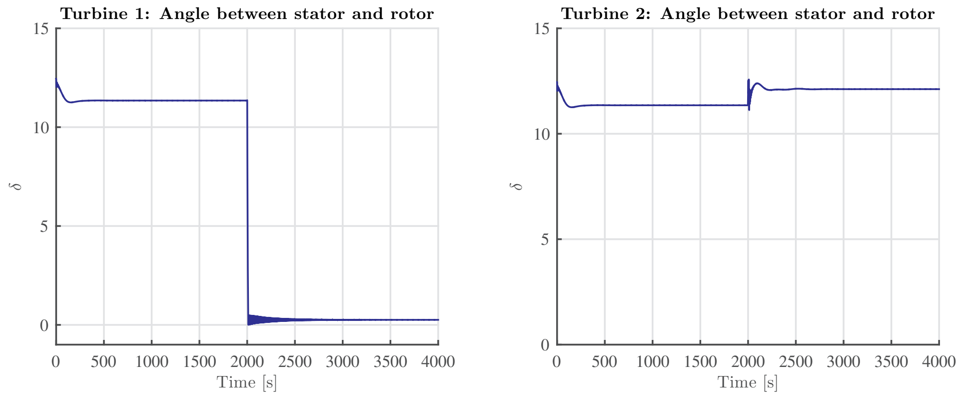
E.4 Hydropower system 2

E.4.1 Turbine shut down



(a) Turbine 1 (b) Turbine 2

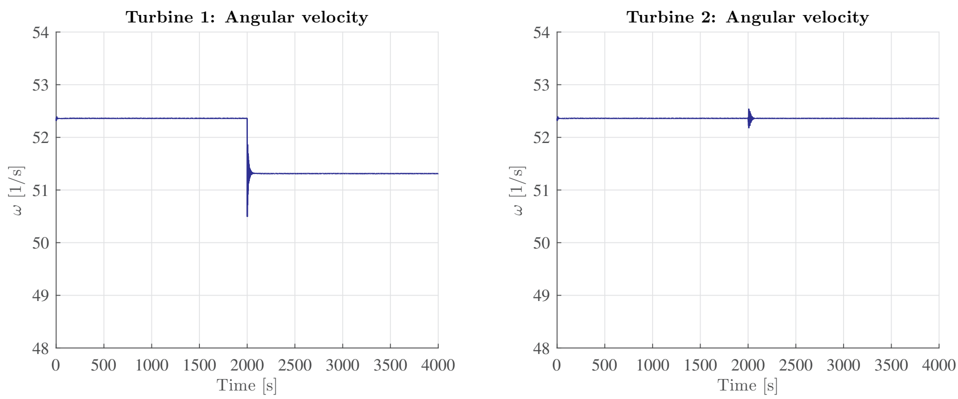
Figure E.4.1: Angular velocity



(a) Turbine 1 (b) Turbine 2

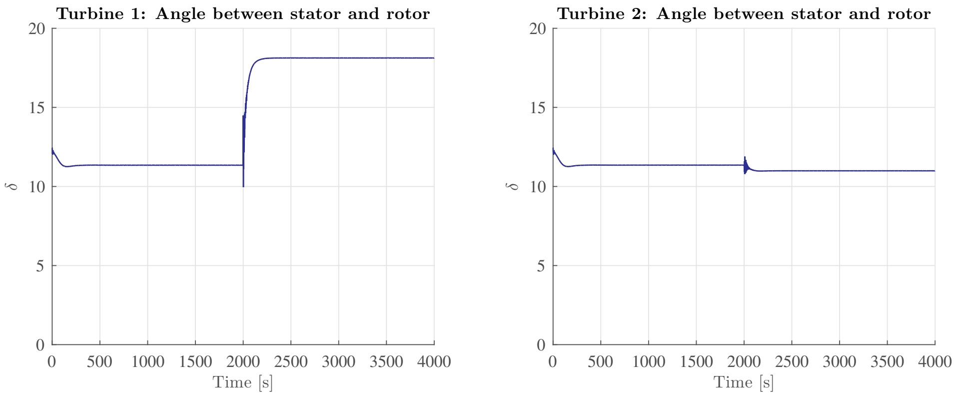
Figure E.4.2: Angle between generator stator and rotor

E.4.2 Increased power demand on the grid



(a) Turbine 1 (b) Turbine 2

Figure E.4.3: Angular velocity



(a) Turbine 1

(b) Turbine 2

Figure E.4.4: Angle between generator stator and rotor