REDUCTION OF PRESSURE PULSATIONS ON AUTOMOTIVE TRANSMISSION OIL VANE PUMP

REDUCTION OF PRESSURE PULSATIONS ON AUTOMOTIVE TRANSMISSION OIL VANE PUMP

By

SLOBODAN RANCIC, Dipl. Ing.

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfillment of the Requirements

for the Degree

Master of Applied Sciences

Department of Mechanical Engineering

McMaster University, Hamilton

MASTER OF APPLIED SCIENCE McMaster University

(Mechanical Engineering) Hamilton, Ontario

TITLE: Reduction of Pressure Pulsations on Automotive Transmission Oil Vane Pump

AUTHOR: Slobodan Rancic, Dipl. Ing., Ss. Cyril and Methodius University in Skopje

SUPERVISOR: Drs. Samir Ziada and Marilyn Lightstone

NUMBER OF PAGES: xiii, 151

ABSTRACT

The balanced oil pump is an important component of the automatic transmission in modern automotive vehicles. During the pump operation, pressure pulsations are generated, which negatively affect the working characteristics of a pump, including pump noise and durability. Thus, pressure pulsations must be minimized in order to ensure smooth operation of the pump, low noise and increased durability.

The current work uses Computational Fluid Dynamics (CFD) to explore the effects of port design on pump pressure pulsations for binary transmission vane pump. Several different approaches are considered, such as port length at intake and discharge, implementation of metering grooves at discharge and intake ports, along with different shapes of metering grooves.

More than 20 CFD cases are analyzed to study the effect of different port design on pressure pulsations of the pump. Validation against experimental data is performed to provide confidence in the CFD modeling results. It is found that ports can be designed more precisely using CFD than described using the simple models that are typically used for pump analysis. Implementing metering grooves helps decreasing pressure pulsations. There is no significant difference in the pressure pulsation generated with different shapes of metering grooves in the pump non-cavitation regime. Future work could consider the effects of the metering grooves geometry on feasibility and durability within the cavitation regime.

ACKNOWLEDGEMENTS

The author would like to thank the following people for their guidance and support.

Samir Ziada – Professor - McMaster University, Hamilton, Canada

Marilyn Lightstone, – Professor - McMaster University, Hamilton, Canada

Richard Muzelaar, Global Director of Innovation Group, Magna Powertrain, Concord, Canada

David Dorigo, Assistant Director for Innovation, Magna Powertrain, Concord, Canada

Wieslaw Zaton, Senior System Engineer, Magna Powertrain, Concord, Canada

Dusan Milacic, CAE Manager, Magna Powertrain, Concord, Canada

NOMENCLATURE

b	cam depth
l_v	vane lift
N	number of vanes
Q_{th}	theoretical flow rate
Rc	cam radius
R_{max}	maximum cam radius
R_{min}	minimum cam radius
R_r	rotor radius
t_v	vane thickness
V_{bv}	volume of a chamber between two consecutive vanes
V_{uv}	volume of a chamber under the vane
Y, Z	generic functions
$\Delta_1, \Delta_2, \Delta_3, \Delta_4,$	angular intervals connecting two regions of the cam
θ	current angular position
$\theta_a, \theta_b, \theta_c, \theta_d,$	characteristic angles of the cam
α_1, α_2	angular intervals characterizing the pre-compression zone of the cam

χ	angle at the rotor peripheral circumference included by a chord equal to t_v
δ_{th}	theoretical flow irregularity
ω	rotational speed
ρ	oil density
ν	dynamic viscosity of the oil
ψ	angle locating the contact point between vane edge and cam contour
$\vec{v}$	velocity vector

Table of Contents

List of Figures

List of Tables

Chapter 1 – INTRODUCTION

The automotive industry is an important component in global transportation. The volume of manufactured vehicles is continuously increasing with a current volume of manufactured motor vehicles (cars and commercial vehicles) of more than 80 million per year. Along with the increased volume of vehicles comes stricter standards for efficiency, noise and pollution. The industry strives to improve automotive designs to be more cost effective, create less noise, have robust design, produce lower emissions and consume less fuel. This need for improved design is also driven by the competition between automotive manufacturers in a global marketplace.

Transmission, which is a complicated and expensive assembly, is an element of all automotive vehicles. Development of improved transmission requires many tests to be conducted for a range of conditions and environments. Every automatic transmission must have a hydraulic pump which delivers oil to the transmission, which is required for normal operation. The transmission pump must be designed to ensure that the overall fuel efficiency of the vehicle is maintained. As such, a small, highly efficient pump is desirable. It is also expected that the pump should be quiet and create minimum level of noise.

To achieve the stated objectives, the current research is concerned with enhancing the design of the "balanced pump" concept, which is different from the conventional transmission oil pump. In contrast to the standard one inlet / one outlet pump, the balanced oil pump has two inlet and two outlet ports which are symmetrical. This allows the oil pump to have smaller dimensions due to significantly lower loads acting on the rotor. However, as with the conventional pump, the pressure and flow pulsations will remain a problem, which may be exacerbated by the reduced pump dimensions. A decrease in pressure pulsations can be achieved through proper port design. The geometries of ports and so-called metering grooves have significant effect on the pressure ripples. The effects of small design changes

can be determined with extensive experimental testing. However, testing requires significant cost for both equipment resources and time. This cost can be greatly reduced using virtual tools, such as Computational Fluid Dynamics (CFD) where simulation can greatly reduce many iterations for testing.

The process of port optimization of a transmission pump is presented in this research. The goal of this research is to optimize the port design in order to lower pressure pulsations within a balanced pump.

As a starting point, theoretical calculations of flow performance based on simplified models will be presented, followed by two dimensional model analysis. A CFD model of the initial design of the balanced pump is developed and simulation results are assessed. Experiments using a prototype created from the initial design are performed and the experimental results are compared to the predictions from the simulations. As a final step, CFD is used for evaluating different port designs in order to determine best design option for this application.

1.1. Rotary vane pump description

The hydraulic power for the steering, oil for the lubrication system or oil for the transmission is provided by a rotary-vane pump, see **Figure 1**. This pump is driven by the car's engine via a belt and pulley, or directly by a drive shaft (K. Manssouri, 2003). It contains a set of retractable vanes that spin inside a circular or oval chamber. As the vanes spin, they pull hydraulic fluid from the return line at low pressure and force it into the outlet at high pressure. The amount of flow provided by the pump depends on the number of revolutions per minute of the rotor, which is directly related to the car's engine revolutions per minute. The pump must be designed to provide adequate flow when the engine is idling. As a result, the pump transports much more fluid than necessary when the engine is running at faster speeds. The pump contains a pressure-relief valve to ensure that the pressure does not get too high, especially at high engine speeds when fluid flow rates are high.

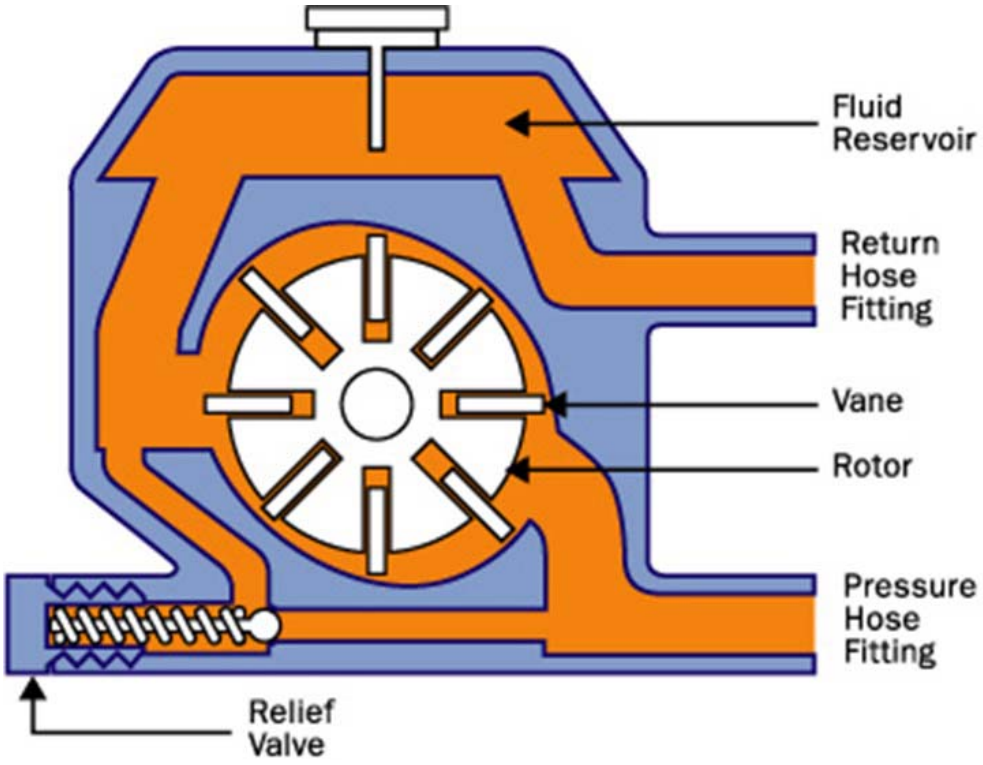


Figure 1: Rotary vane pump, principle of operation (auto.howstuffworks.com)

The vane-in-rotor pumps may be made with constant or variable displacement pumping chambers.

Figure 2 shows an engine vane oil pump. Because the rotor is shifted from the center, the vanes pull hydraulic fluid from the low-pressure chamber and force it into the high-pressure chamber. This method leads to a continuous force on the rotor, caused by the high pressure compartment. This force is parallel to the axis of rotation and creates an additional load which must be taken into consideration when calculating the bearing life. This additional force will eventually reduce the period of operation of the bearing. The transmission assembly of a motor vehicle is complicated and reliability is of high importance. Thus, the transmission pump bearing must also be long-lasting and durable since the replacement procedure of the bearing is difficult. To resolve this problem, the pumps' stator can be designed to have an elliptic, rather than circular, shape which results in the pump having two opposite low pressure and high pressure compartments (see Figure 2). Since the pairs of low-pressure and high-pressure compartments are situated symmetrical, the pressure forces acting on the rotor balance each other. As a result, they have almost no influence on the bearings, which allows for higher pressures in the pump to be created.

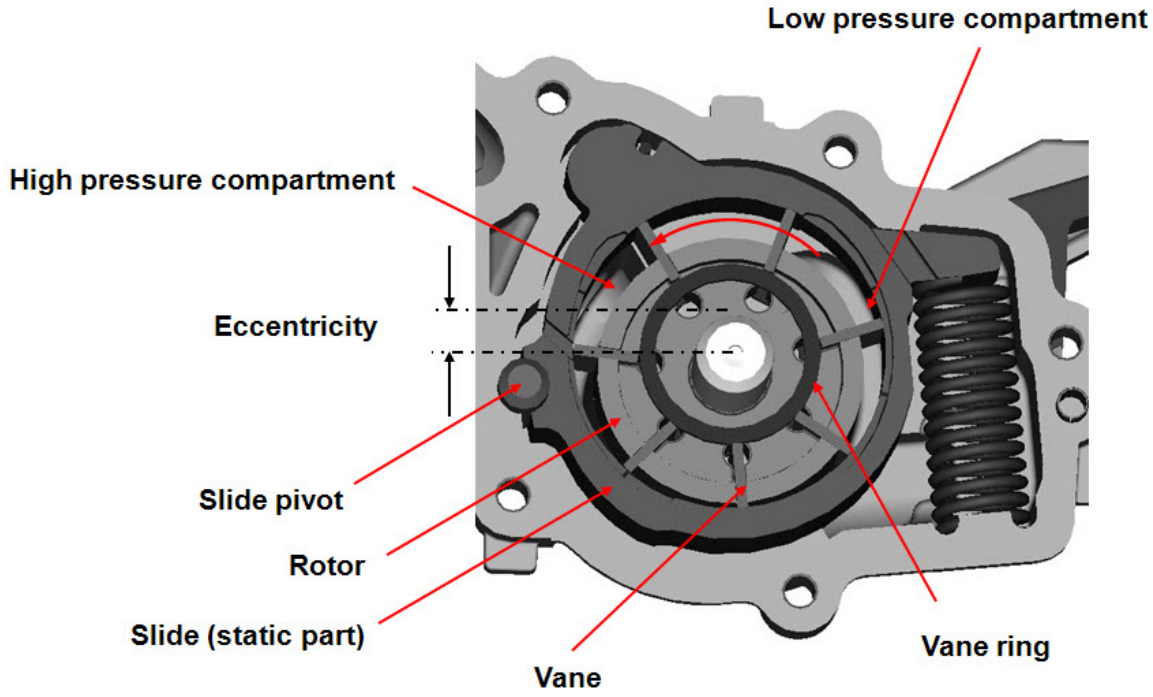


Figure 2: Engine vane pump, principle of operation (Magna Powertrain)

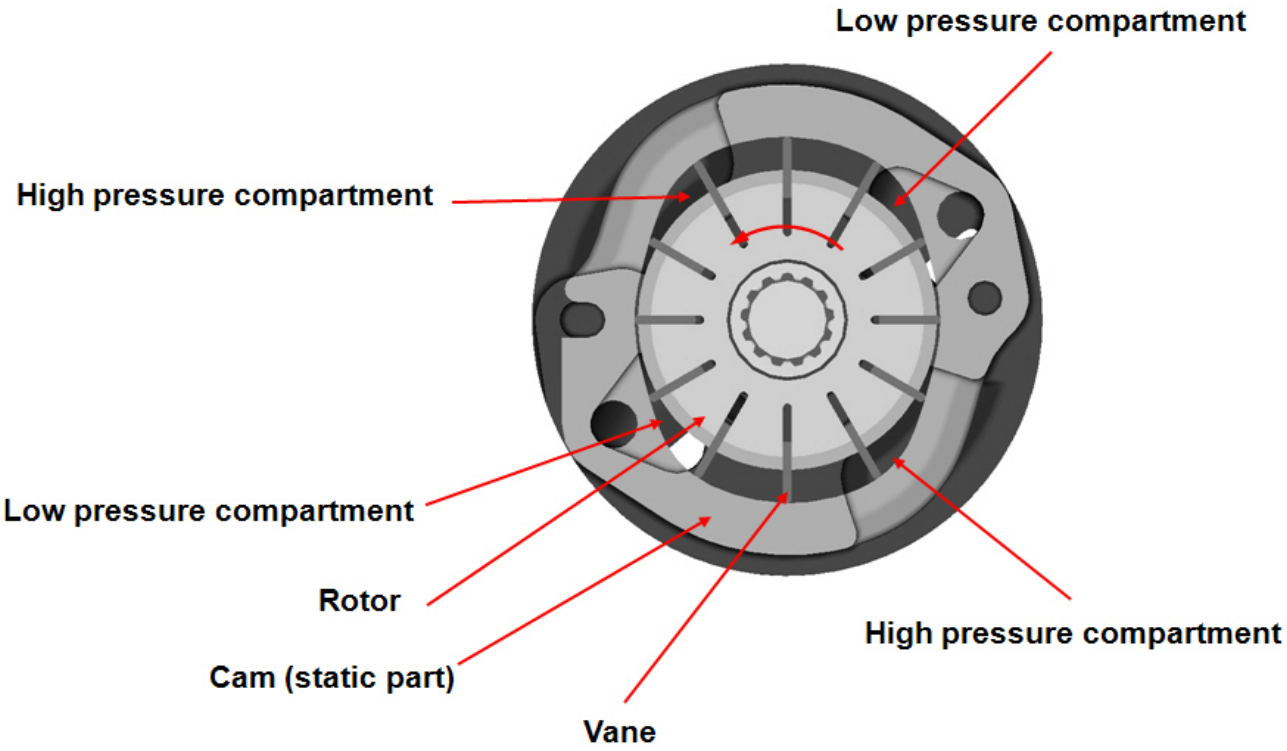


Figure 3: Transmission balanced vane pump, principle of operation (Magna Powertrain)

Most rotary-vane pumps as in **Figure 1** are constant displacement pumps. This means that they provide a constant flow per rotation, which results in a proportional growth of the flow rate with increasing pump speed. This pump characteristic can result in overall inefficiencies because of the constant flow per rotation; the pump is not always working efficiently and therefore wastes power. This power translates into wasted fuel. A method to overcome this issue is to have rotating slide (shown on **Figure 2**), in which the slide position changes with the operation needed of the pump. Changing the slide position changes the eccentricity of the pump, which changes the flow performance of the pump. The rotating slide changes position at high pressure and the eccentricity becomes lower. As a result of this movement, pump delivers reduced amount of oil at high operating points, which saves energy. While this principal is used in engine oil pumps, which are relatively large and complex, the pumps which are used in transmission must be compact and the rotating slide cannot be implemented due to packaging

constraints. Therefore, transmission pump analysis and design presented in this research will focus on optimization of the port shape in order to develop best performance solution in terms of reduced flow and pressure pulsations.

Chapter 2 - Literature review

As a part of this research, an extensive study of the available literature was conducted. A limited number of publications which describe the details of the design and operation of transmission vane pumps were found, because they are considered proprietary materials and are not published. Therefore, modeling of balanced pump is still a challenge in many areas of principles of operation.

The chapter begins by providing a general description of a balanced pump. A simplified model as well as the AmeSim model which is a more precise, two-dimensional model, are then presented. These models are used as a starting point for calculating the pump operating characteristics and are used to guide the initial pump design. Both approaches, however, have limitations due to assumptions inherent in the model environment. For example, these simplified models do not account for leakage between chambers or fluid cavitation. Moreover, they are not capable of providing details on the pressure and velocity field distributions. Computational Fluid Dynamic (CFD), in contrast, has the capability to provide a full three-dimensional description of the transient pressure and velocity fields throughout the pump. As such, it can be used to provide analysis basis for the operating performance of a balanced transmission pump, albeit with much greater computational effort than the simplified methods.

2.1. Balanced pump description

A balanced vane pump, now widely used in many hydraulic power systems because of its compactness, lightweight, and low cost, is especially suitable for a hydraulic power source in a power steering system and transmission of a vehicle, in which low pressure pulsation and low noise are required. When designing a hydraulic pump including the vane pump, the mechanical efficiency as well as the volumetric efficiency constitutes a key factor in evaluating the pump performance (Y. Inaguma & A. Hibi 2005). Vane pumps change the energy level of the operating fluid, like any other positive displacement pump, by converting mechanical input to hydraulic output. Balanced vane pumps are fixed displacement pumps consisting of an ellipsoidal shape cam ring with two pressure and suction quadrants opposite each other, as shown in **Figure 4** (A. Guiffrida & R. Lanzafame, 2004). Thus, the rotor is hydraulically balanced and bearing loads are reduced greatly. The pump is driven by the drive shaft and carries the vanes in radial slots.

The vanes follow the cam ring through their cycle, two cycles per revolution for each vane: suction-seal-discharge-seal. The rotor-vanes assembly is positioned within the cam ring. The center of the rotor coincides with the center of the ring which leaves a fixed shaped void between the rotor and the ring.

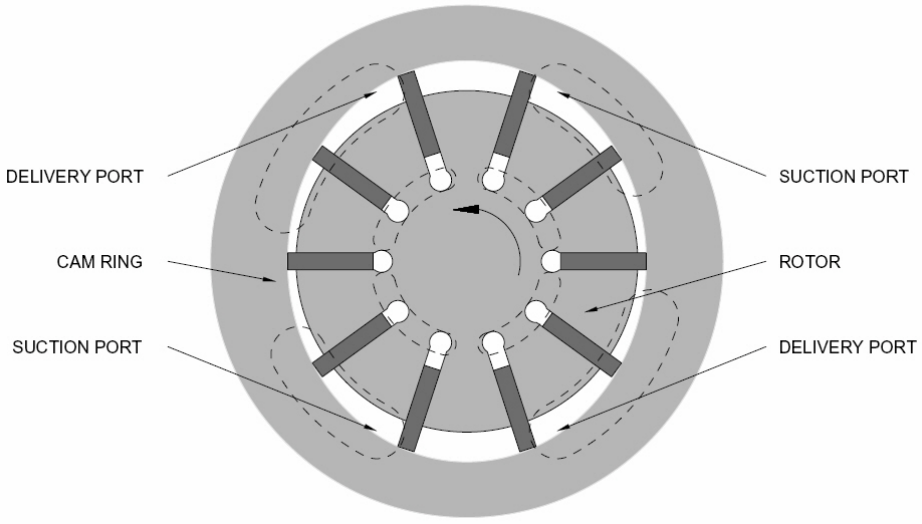


Figure 4: Typical balanced vane pump configuration (A. Guiffrida & R. Lanzafame, 2004)

The components shown in **Figure 4** are held in place axially by two flat plates which engage the end faces of the rotor vanes-ring assembly. These port plates, in conjunction with the cam ring, provide sealed chambers. With properly situated ports, fluid can be directed to and from the pump in a direction perpendicular to the page. As the rotor turns, the vanes are forced against the inner surface of the cam ring by centrifugal force and pressure acting on the back side of the vanes. In this research project, vanes are hydraulically loaded outward against the ring with oil pressure from the discharge side of the pump. In engine oil pump applications, vanes are forced to the cam by vane ring, as shown on **Figure 2**. **Figure 5** shows a cross-section of the pump modeled with the two plates, vanes, shaft, rotor and the cam ring.

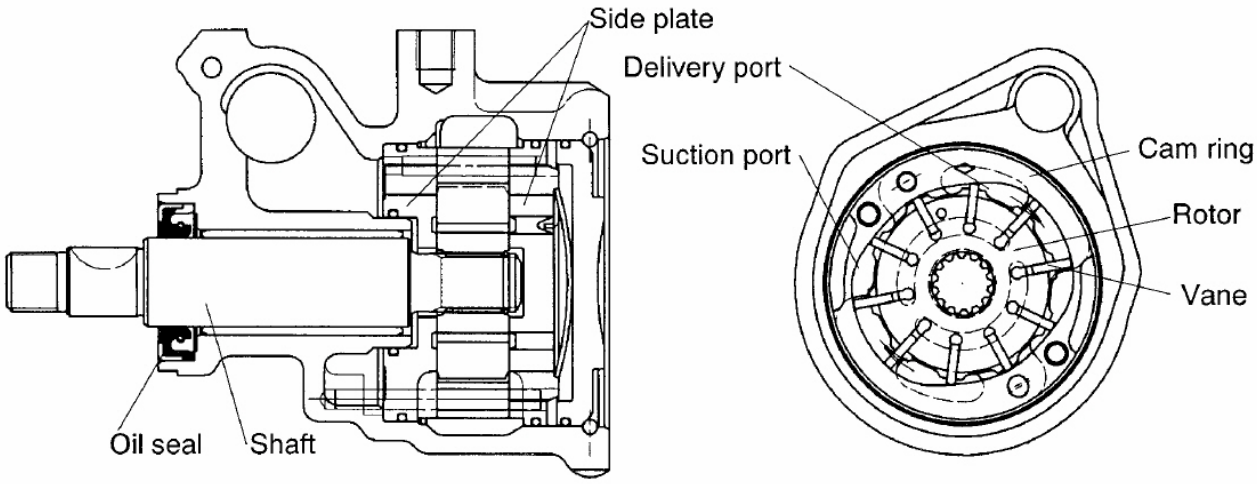


Figure 5: Cross-section of a balanced vane pump (Y. Inaguma & A. Hibi 2005)

A firm but light force of the vane against the cam is provided by cavity pressure, which comes from pressurized system from transmission. This force assures smooth cam tracking by the vane. Thus, the vanes are held outward in a light but steady contact against the cam ring. Balanced vane pumps are the quietest hydraulic pumps.

The most important component which determines pump performance and overall dimensions of the pumps is the cam profile. The mathematical description of such a profile, which presents the shape of a particular cam, is fundamental for any mathematical model of the pump. This research gives guidelines for the design of the inner contour of the cam ring, which is a base starting point for balanced pump design. Simplifying hypotheses will be introduced, including the method for calculating the rate of change of pumping chamber volume with rotating angle, leading to the determination of the theoretical flow rate of the pump. This approach is commonly known as one-dimensional (1D) analysis, since there is no geometry, except for chamber volumes, involved in this type of calculation. In addition, a two-dimensional (2D) calculation will be presented, which provides a more precise way of description of pump performance. Finally, Computational Fluid Dynamics (CFD) analysis will be used to determine details for flow and pressure pulsations as well as to optimize ports to reduce pressure pulsations for balanced pump.

2.2. Calculation of cam ring contour

In balanced vane pumps, the cam ring is the most characteristic and interesting machine element. Its inner contour impacts the chamber volume which changes as the vane rotate and determines the flow delivered by the pump. Designing such a contour is a basic calculation for a balanced pump, because flow performance of the balanced pump is directly related to basic dimensions of the cam shape and its height (Ref: A. Guiffrida & R. Lanzafame, 2004).

Five regions in the angular interval from 0 to π (other side, from π to 2π, is symmetrical) can be defined. These regions are shown on **Figure 6** and are described below:

1. First, for $\vartheta \in [0, \vartheta_a]$, where R_c is constant and equal to its minimum value (R_{min});
2. Second, for $\vartheta \in [\vartheta_a, \vartheta_b]$, which R_c presents a positive gradient. This region is related to the suction zone, then its design is fundamental to produce the correct pressure drop across the suction ports which

causes the fluid to flow into the pump to fill the variable volume pumping chambers;

3. Third, for $\vartheta \in [\vartheta_b, \vartheta_c]$, which gives the pre-compression for the trapped chamber. Here R_c decreases very slowly;
4. Fourth, for $\vartheta \in [\vartheta_c, \vartheta_d]$, characterized by a radius decreasing with a gradient greater than the previous one. This is the delivery zone and characterizes the irregularity of the theoretical flow rate;
5. Fifth, $\vartheta \in [\vartheta_d, \pi]$, in where R_c is equal to its minimum value (R_{min}) again.

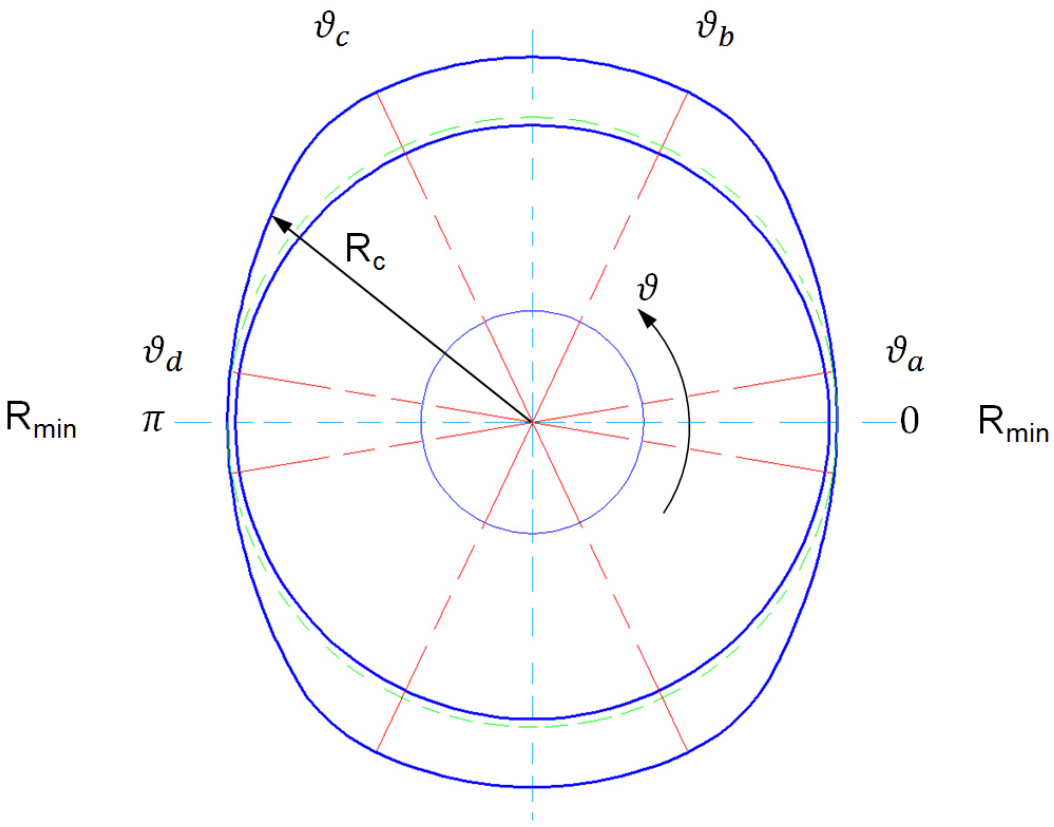


Figure 6: Regions description in a pump profile

Figure 7 shows a qualitative development on R_c—ϑ plot of the radius describing the inner contour of the cam ring. The asymmetry of the curve shown in **Figure 7** constrains the rotor to rotating in one direction only.

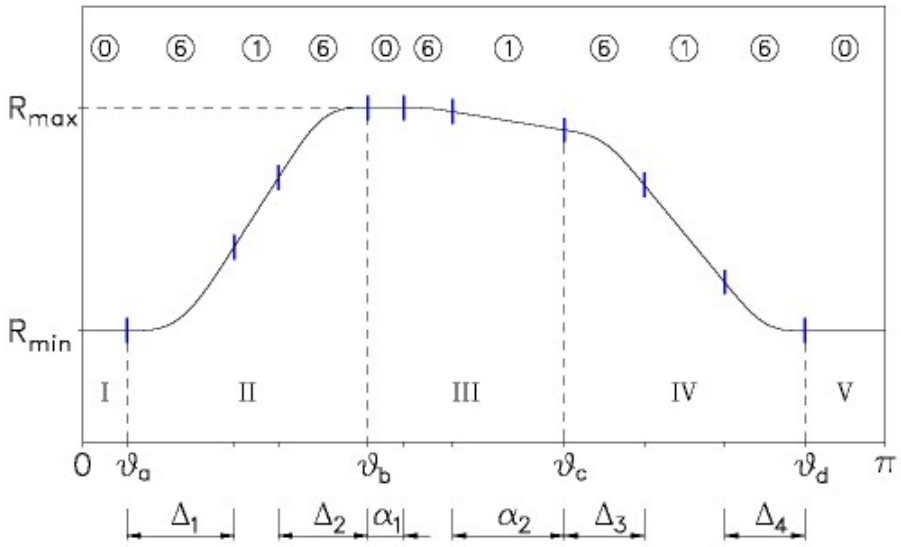


Figure 7: Qualitative radial development of the cam contour

2.3. Equations used to design the cam

The zones characterizing the cam contour are described by functions that are to be properly connected at those points where certain function ends and another begins. This connection is necessary if a mathematical continuity is to be assured. Thus, possible discontinuities in the profile are eliminated and smoothness is ensured. The design of the cam, though, will be more rigorous if the derivatives of the functions describing the cam are equal in these connecting points. Imposing such conditions, it is possible to write:

$$Y(\tilde{\vartheta}) = Z(\tilde{\vartheta}) = R_c(\tilde{\vartheta}) \quad \textbf{(1)}$$

$$\frac{dY(\tilde{\vartheta})}{d\vartheta} = \frac{dZ(\tilde{\vartheta})}{d\vartheta} = R_c'(\tilde{\vartheta}) \quad \textbf{(2)}$$

$$\frac{d^2Y(\tilde{\vartheta})}{d\vartheta^2} = \frac{d^2Z(\tilde{\vartheta})}{d\vartheta^2} = R_c''(\tilde{\vartheta}) \quad \textbf{(3)}$$

$$\frac{d^3Y(\tilde{\vartheta})}{d\vartheta^3} = \frac{d^3Z(\tilde{\vartheta})}{d\vartheta^3} = R_c'''(\tilde{\vartheta}) \quad \textbf{(4)}$$

Here $\tilde{\vartheta}$ is the angle where the function *Y(ϑ)* ends and the function Z(ϑ) begins.

The functions adopted to design the cam are simple polynomials whose order does not exceed the sixth and depend on the current angle ϑ, so the continuity conditions required with **Eqs. (**1**)-(**4**)** are truly respected. In fact, the sixth order polynomial allows continuity in the fillet points up to the third derivatives. The numbers shown in the circles in **Figure 7** refer to the order of polynomial functions. The values of $R_c(\vartheta)$ for the various ϑ_a, ϑ_b, ϑ_c and ϑ_d (**Figure 7**) are chosen in order to obtain the desired displacement of the pump.

As seen in **Figure 7**, the first and the fifth zones are described by the same circumference radius. In contrast, the second and the fourth zones, are realized by means of a combination of one sixth order polynomial, followed by one of the first order, then another of the sixth order. The zone where the pre-compression of the trapped fluid occurs is realized using once again three functions: one constant, then a sixth order polynomial and finally another polynomial of the first order. The necessity of splitting the third zone into three functions, with particular reference to a constant zone, is easily demonstrable. Referring to **Figure 7,** if $R_c(\vartheta)$ presents only one maximum value, the vane will touch the inner contour of the cam ring at first with one edge, then, almost immediately, with the other one. Therefore it should be desirable to extend the angular interval where $R_c(\vartheta)$ keeps its maximum value to avoid any possible collisions, at least from ϑ_b to $\vartheta_b + \alpha_1$ owing to the foregoing specification. The angular interval α_1 is related to the vane thickness and to the maximum value of $R_c(\vartheta)$ according to the relation

$$\alpha_1 > 2 \cdot \arcsin \frac{t_v}{2 \cdot R_{max}}$$

The edge touching the cam contour presents certain, even if little, curvature radius, but the latter is obviously less than the local curvature radius of the cam contour in the contact point (at least two orders of

magnitude). For the sake of simplicity in the following sections, the tip of the vane is assumed to present a sharp edge at the contact point with the cam contour.

2.4. Analysis of volumes variation

The volumes derivatives of the chambers between two consecutive vanes are difficult to evaluate, essentially because of the vane thickness. Vane thickness for transmission pump presented in this research is 1 mm. Therefore, an approximation regarding the shape of the vane is introduced. Considering that the vane touches the cam contour almost always with only one edge, the little nail of oil at the tip of the vane (represented by the triangle shaped area $P_1P_2P_3$ on **Figure 8**) may be neglected.

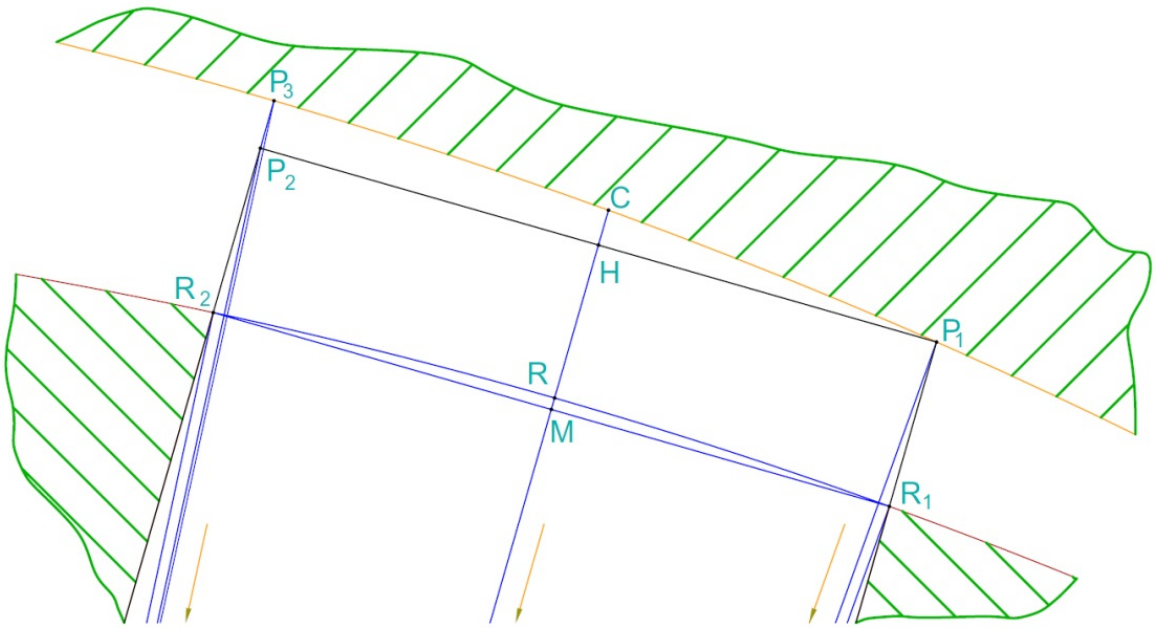


Figure 8: Vane touching the inner contour of the cam ring in P1

Thus, the tip of the vane is considered to be in contact with cam. With reference to the configuration shown on the left of **Figure 9**, it is sufficient to consider the segments P_1R_1 and P_3R_2,

portions of vanes sides emerging out from the rotor. Both these segments depends on ϑ. The chamber between these two consecutive vanes occupies the angular position $\vartheta - \frac{\pi}{N}$, where N is number of vanes. The pump presented in this research has 12 vanes.

During rotation, the segments P_1R_1 and P_3R_2 at the left of **Figure 9** have moved to $P_{1n}R_{1n}$ and $P_{3n}R_{2n}$ respectively, at the right of the same figure (the subscript n indicates the new current position). The prolongations of these segments are always tangent to a circumference whose diameter is equal to t_v, vane thickness. This circumference is concentric to the rotor.

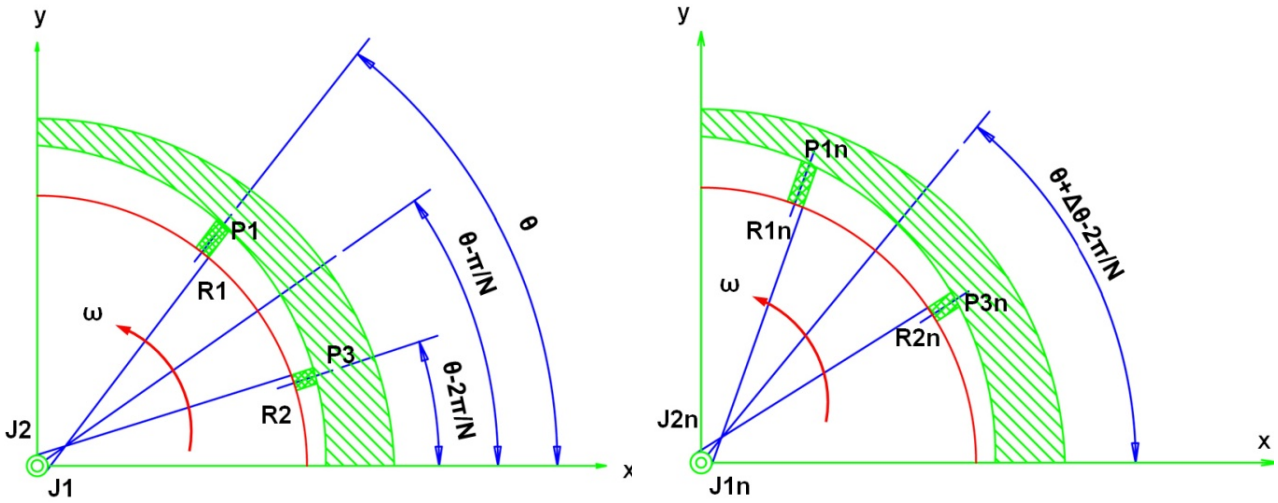


Figure 9: Positions of two consecutive vanes and next configuration after a Δθ rotor rotation

The volume of fluid entering a chamber between two consecutive vanes can be evaluated from the geometrical calculations and is governed by:

$$\frac{dV_{bv}\left(\vartheta - \frac{\pi}{N}\right)}{d\vartheta} = \frac{b}{2} \cdot \left[(\overline{P_1 J_1}(\vartheta))^2 - \left(\overline{P_3 J_2} \left(\vartheta - \frac{2 \cdot \pi}{N} \right) \right)^2 \right]$$

These segments do not depend on the current angular position ϑ. Thus, two segments *(P_1J_1* and *P_3J_2)* are sufficient for the calculation of the variation of the volume of a chamber between two consecutive vanes.

Additional details on the analysis of volume calculations are given in **Appendix A**.

2.5. Theoretical flow rate according to one dimensional analysis

One of the goals of this analysis is to determine the pump flow rate as a function of the rotational velocity. This section will present a method for calculating the theoretical flow rate, using the previous calculation for the volume variations for a balanced pump.

As shown in **Figure 9**, if ϑ locates the axis of the leading vane, then the middle position of the chamber between this vane and the trailing one is located by the angle $\vartheta - \pi/N$ and $\vartheta - 2\pi/N$ locates the axis of the trailing vane; the rotation is counter-clockwise.

The ideal theoretical flow rate can be expressed as follows (Ref: A. Guiffrida & R. Lanzafame, 2004):

$$Q_{th}(\vartheta) = \omega \cdot \sum_{k=1}^{N} \frac{1}{2} \cdot \left[\frac{dV_{bv,k}\left(\vartheta - k \cdot \frac{\pi}{N}\right)}{d\vartheta} \cdot \left(\operatorname{sgn} \frac{dV_{bv,k}\left(\vartheta - k \cdot \frac{\pi}{N}\right)}{d\vartheta} - 1 \right) \right]$$

$$+ \omega \cdot \sum_{k=1}^{N} \frac{1}{2} \cdot \left[\frac{dV_{uv,k}\left(\vartheta + (k-1) \cdot \frac{2 \cdot \pi}{N}\right)}{d\vartheta} \cdot \left(\operatorname{sgn} \frac{dV_{uv,k}\left(\vartheta + (k-1) \cdot \frac{2 \cdot \pi}{N}\right)}{d\vartheta} - 1 \right) \right] \quad \textbf{(5)}$$

Here it is to be highlighted that if the vane thickness is negligible and the cam radius R_c does not vary in angular intervals of 0, i.e. *p,* and P/2, equal to *2p/N,* one can easily demonstrate that the foregoing equation can be simplified to:

$$Q_{\text{th}} = \omega \cdot b \cdot \left(R_{\max}^2 - R_{\min}^2\right) \quad \textbf{(6)}$$

Where ω is rotational velocity and b is depth into the page.

As shown on **Eq. (6)**, theoretical flow rate is independent of ϑ. It is clear that the theoretical flow rate irregularity defined as (Ref: A. Guiffrida & R. Lanzafame, 2004):

$$\delta_{th} = \frac{Q_{th}(\vartheta)|_{max} - Q_{th}(\vartheta)|_{min}}{\frac{N}{2 \cdot \pi} \cdot \int_0^{\frac{2 \cdot \pi}{N}} Q_{th}(\vartheta) d\vartheta}$$

is zero. If the vane thickness is not negligible, the time histories of *N* volumes for the chambers between two vanes and of *N* volumes for the chambers under each vane are to be analyzed. According to the design of the cam contour, with reference to the function $R_c(\vartheta)$, the theoretical flow rate actually depends on ϑ, but the flow rate irregularity is expected to be very low if the cam parameters are opportunely chosen.

The actual instantaneous flow rate delivered by a positive displacement pump is a periodic function which is a combination of three main categories of waveforms or ripples:

1. Kinematic flow ripple, which is a function of the rotating group geometry (given by **Eq. (5)**),
2. Compression flow ripple, which is a result of compression or de-compression of a trapped fluid volume,
3. Leakage flow ripple, which is a result of pressure differentials across leakage paths.

These generalized waveforms are typical of all positive displacement pumps. For a balanced vane pumps, as the vanes pass the suction ports, the volume of fluid in the pumping chamber increases, thereby drawing in fluid. Next, this fluid reaches the end of the suction ports and enters the pre-compression zone, so named because the fluid is typically compressed over this region which is closed off from both the suction and delivery ports. Within the pre-compression zone the fluid pressure rises according to the extent of volume reduction (as well as the degree of leakage). If this pressure does not match the discharge pressure, once the chamber opens to the delivery port, a flow pulse is generated and lasts until the pressures equalize. This pulse is referred to above as the compression ripple. As the compression ripple subsides, the flow rate is dictated by the rate of change of the fluid chamber volume, which is a function of the cam geometry. This is referred to above as the kinematic ripple. The resulting flow in this region has a mean value plus an unsteady flow induced by the cam shape and the vane thickness. The leakage ripple is driven by pressure differential across leakage paths throughout the pumping cycle.

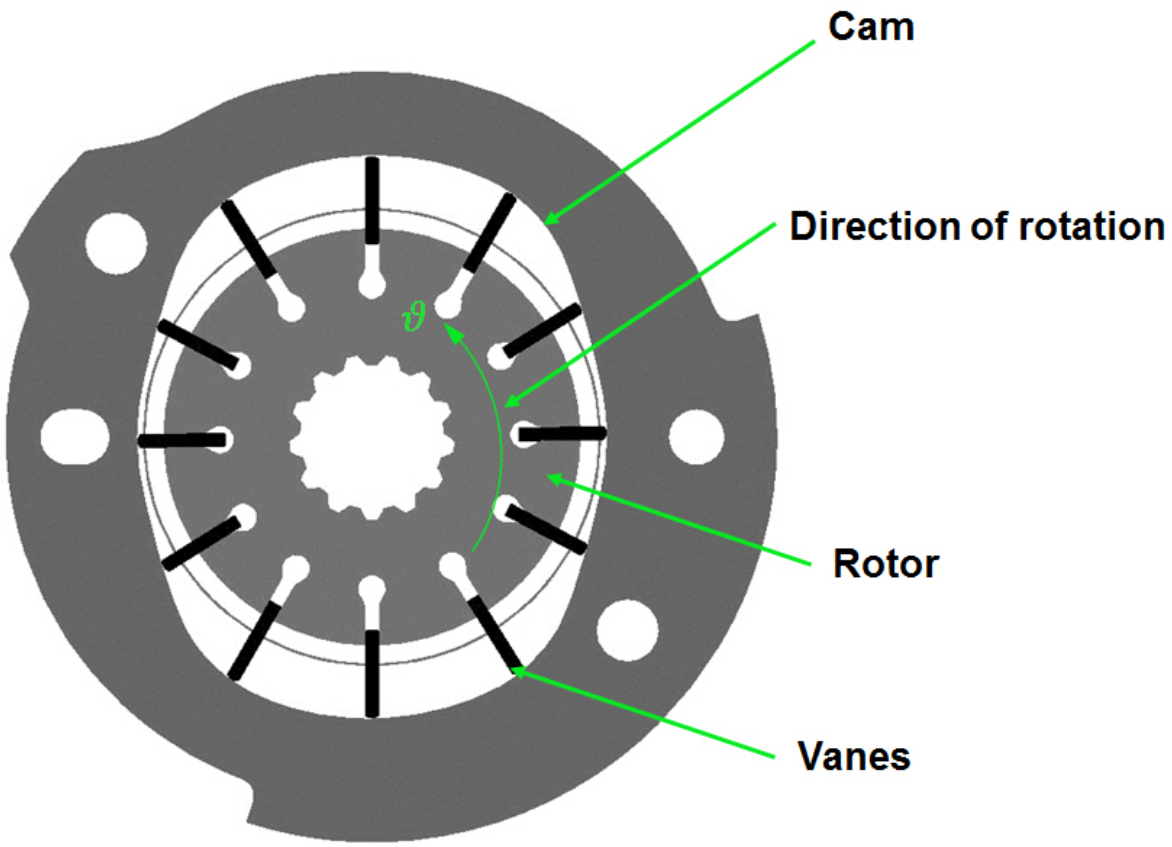


Figure 10: Cam, rotor and vanes used in balanced pump for this study

Cam, rotor and vanes which are used in balanced pump for purpose of this research are depicted on **Figure 10**.

Using the geometry for the 'basic' balanced pump used in this research, a sample plot of the pump flow rate is shown in **Figure 11.** This was calculated using the equation for the variation of flow (**Eq. (5)**), taking into account the cam width (or cam thickness) and finite thickness of the vanes. The graph presented on **Figure 11** has 12 peaks (which occur during one revolution of the pump), where each peak represents flow from both chambers. The shape of the curve can be explained by considering the terms that make up **Eq. (5).** The first term of that equation considers the derivative of volume over angle for chamber between two consecutive vanes. The second term of the equation is the derivative of the volume of a chamber under the vane. The "sign" function is used to double the value at certain

position, while at all other positions it yields a value of 0. For example, if we take a look at first chamber only, **Eq. (5)** will have much simpler form of:

$$Q_{th}(\vartheta) = \omega \cdot \frac{1}{2} \cdot \left[\frac{dV_{bv,k}\left(\vartheta - \frac{\pi}{12}\right)}{d\vartheta} \cdot \left(\operatorname{sgn} \frac{dV_{bv,k}\left(\vartheta - \frac{\pi}{12}\right)}{d\vartheta} - 1 \right) \right]$$

$$+ \omega \cdot \frac{1}{2} \cdot \left[\frac{dV_{uv,k}(\vartheta)}{d\vartheta} \cdot \left(\operatorname{sgn} \frac{dV_{uv,k}(\vartheta)}{d\vartheta} - 1 \right) \right]$$

The sinusoidal shape of the presented curve strictly depends on the derivative of volume as a function of the angle ϑ which depends on the design of the elliptical shape of the cam. Moreover due to the elliptical cam shape, the curve plotted on **Figure 11** represent chambers passing the discharge ports twice during one revolution. This plot is created at angular velocity of 4000 RPM which is used for all simulations conducted in this research. This rotational speed is adopted because it is in the middle of the operating range for balanced pumps, in order to be away from cavitation regime.

The pump flow rate depicted on **Figure 11** is similar to that obtained from the simplified theoretical calculation from **Eq. (6)** of 44.2 liters per minute. It must be noted that the calculation for flow in the one-dimensional analysis does not include the effects of fluid properties, such as density and viscosity. Thus, flow calculated by one-dimensional analysis depends on rotor and cam geometry only.

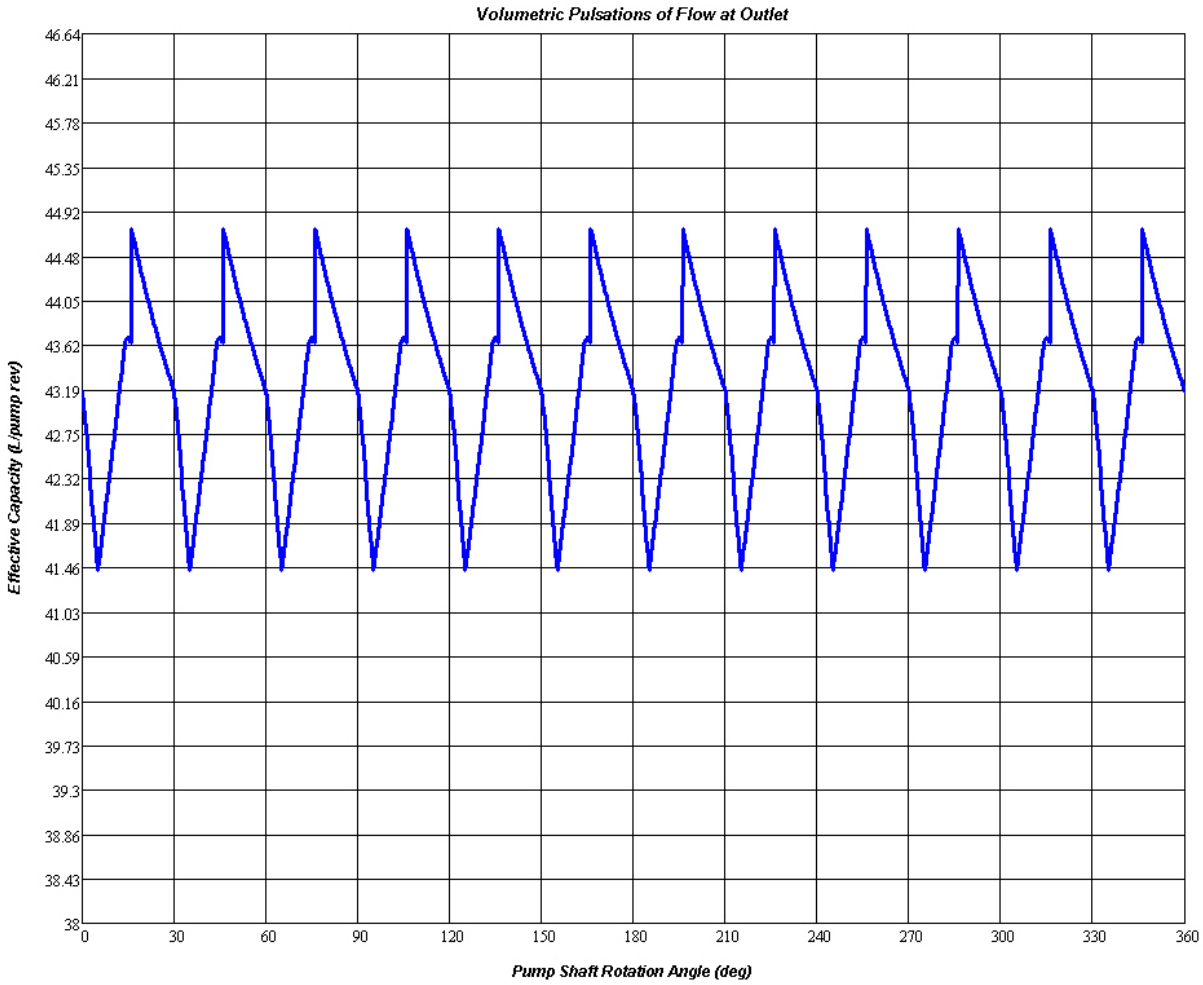


Figure 11: Flow performance of the pump, calculated with basic dimensions

2.6. Limitations of one dimensional analysis

The above calculations represent a so called one dimensional calculation, where only the basic dimensions of radii and rotor width are considered. This calculation gives an approximate dimension of the pump itself, taking into account requirement for flow performance, given by the customer. The geometry of the ports and leakage across the vane tips are not considered. The one dimensional analysis provides a starting point for a design of a balanced pump.

2.7. Methodology used for AmeSim model for two dimensional analysis

2.7.1 Introduction

The theoretical calculation presented in the previous chapter is considered to be a one dimensional approach, since the flow is determined by the cam shape and rotational rate only. This section describes a two-dimensional preliminary calculation of vane pumps, which uses a two-dimensional geometry. As such, this method is referred to as a two-dimensional analysis.

The advantage of a two dimensional (2D) over a three dimensional (3D) method lies in its ability to efficiently simulate the variation of flow rates and pressures within a network of interconnected components such as pumps, pipes, valves, junctions and controllers, particularly as the operating conditions of the components change. These conditions can change, for example, by closing or opening valves or varying the pump speed. The efficiency of the much faster 2D simulation has to be weighed against significantly less detailed understanding of local, flow-related phenomena that results from a 3D approach. Some, subtle aspects of the complex flow behavior are simply beyond the ability of the 2D analysis, which uses fluid mechanics formulae and equations as well as empirical or semi-empirical correlations.

The benefits of both approaches (2D and 3D), can be combined in co-simulations, where sharing of the boundary data between 2D and 3D models enables exploring the behavior of complex fluid systems in which only the most critical components are built in 3D detail.

There exist a number of software suites offering 2D CFD capabilities. For example, there are: MapleSim, Matlab-Simulink and LMS AmeSim. The AmeSim software is particularly popular among the North American automakers, since it was the first software offering several libraries of diverse components for quickly building simulation models of complex, multi-physics systems, such as the complete lubrication system of an automobile.

2.7.2 Methodology

The pump design will first be studied using a two-dimensional model of the pump developed within LMS AmeSim simulation software, which is capable of reasonably accurate and fast assessment of the pump behavior and has been extensively tested at Magna Powertrain. The analysis with more accurate, CFD-based tools would be too time-consuming at this preliminary stage in the pump design.
A simulation model of the balanced pump was built using the Hydraulic Library of the AmeSim software (Ref. AmeSim Manual). The graphical representation of the model is shown in **Figure 12**. The model is comprised of several interconnected icons, with each icon representing a discrete component. Each icon object encapsulates the physical formulation of the component. The transient behavior of each component depends on a few parameters and is described by a few variables. After the complete model is assembled the software formulates equations of the power flow and power conservation within the system. The equations are time-dependent and are used to assess the transient evolution of the system.

The graphical complexity of a model can become overwhelming. In order to manage it, in addition to the standard library icons, the system model can contain user-defined icons of subsystems, which internally can comprise other standard icons and/or user-defined subsystems, in several consecutive layers, if necessary.

The red icons in **Figure 12** represent the control components of the pump model. The input to the model is provided by the variable speed source on the left hand side, which is piecewise linear signal source. The vane pump presented in this study rotates with rotational speed of 4000 RPM. The speed signal is fed into an integrator and is converted into angular position of the pump input shaft. From the shaft position the red Mod (Modulus) components calculate the angular position of each vane. The modulus component computes an output signal between a minimum and a maximum values from an input signal using modulo function. In this particular model, the output signal is between 0 and 360

degrees. Each icon in the vertical column of twelve black, clipped rectangles is a user-defined subsystem representing a single inter-vane (IV) chamber. The chambers are connected in a ring-like fashion. Details of a single IV chamber model are shown in **Figure 13**.

In **Figure 12**, the blue lines symbolize hydraulic connections. The suction tank, exposed to ambient pressure, provides the IV chambers with oil via the suction line. The oil properties, such as density, viscosity, bulk modulus etc., are defined in the blue drop component. The blue drop component is used to set the characteristics of the hydraulic fluid. The fluid is identified by an "index of hydraulic fluid" (integer parameter) that must be used by all the other submodels having hydraulic properties and using this particular fluid. There are several assumptions used while defining oil properties:

- The absolute viscosity is assumed to be constant.
- The air/gas release phenomenon is not considered.
- The bulk modulus at pressures above the saturation pressure is assumed to be constant and is set using the bulk modulus parameter.
- Below the saturation pressure, the bulk modulus is also assumed to be constant and its value is equal to the bulk modulus above saturation pressure divided by 1000.
- The liquid density at atmospheric pressure is adjusted using the density parameter. The fluid density values are consistent with the bulk modulus in order to account for mass conservation.

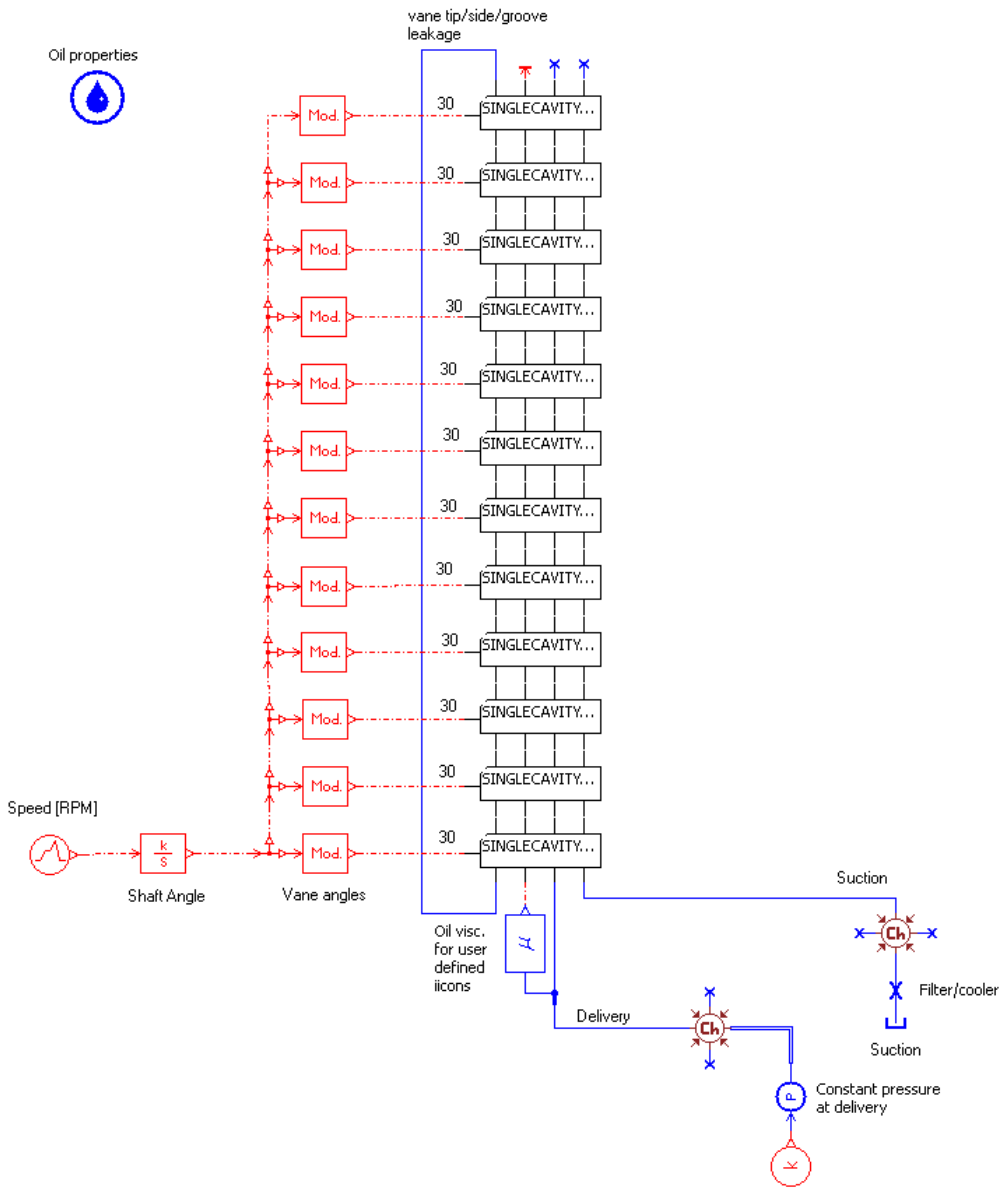


Figure 12: AmeSim model of transmission vane oil pump

From the oil tank, oil flows through an orifice which accounts for oil filter and cooler losses, to a chamber symbolizing the volume of the passages, including suction ports, that oil has to pass through prior entering the IV chambers. The oil tank is exposed to atmospheric pressure of 0 bar gauge. Flow in the orifice representing the oil filter is modeled using laminar flow. The pressure drop is specified for a given flow rate, which is provided by the filter manufacturer.

Pressurized oil, leaving the volumes, passes the delivery port chamber which, in the displayed version of the model, is under constant pressure of 8 bar for the particular balanced pump used in this study. The hydraulic line at the delivery port is modeled as a pipe with friction in order to have a more realistic model which reflects real pump environment conditions.

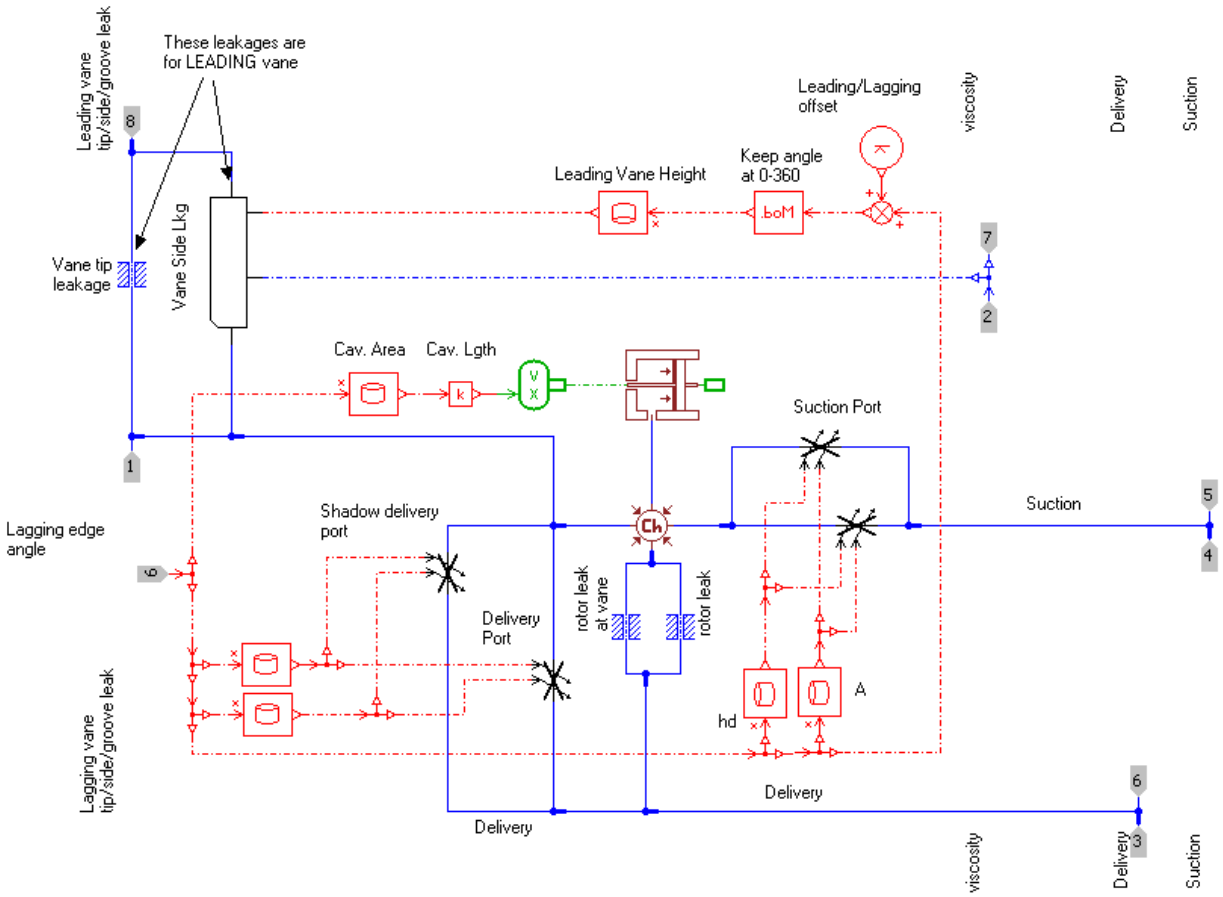


Figure 13: Model of a single inter-vane chamber

The central component of the IV chamber model from **Figure 13** is a piston/cylinder icon. Label 9 on the left side of the figure indicates the connection point of the red control lines to the IV chamber angular position from **Figure 12**. The instantaneous volume of the IV chamber depends on the chamber angular position. Although the instantaneous volume can be calculated from the pump geometry and the chamber position, it is much more efficient to pre-calculate the relationship (**Figure 16** and **Figure**

17) and pass it to the model as a look-up table (LUT). The LUTs are symbolized in AmeSim models as red cylinder icons. During simulation the instantaneous IV chamber volume is interpolated from LUT data. The IV chamber instantaneous volume value is fed to a green VX icon which, differentiating the input, calculates the flow rate to and from the IV chamber and thus "drives" the piston of the following icon. The green VX icon converts a dimensionless signal input to a linear displacement with the same value with a linear velocity. This velocity is obtained by an approximate differentiation of the angle using a first order lag method with a user supplied time constant.

The cylinder is connected to a chamber, with varying volume defined by the preceding cylinder and, further, via hydraulic lines to the suction and delivery ports. Both ports have variable cross sections, which in the model are represented with black icons of variable orifices. Two control inputs into each orifice correspond to the hydraulic diameter of the orifice and to its cross section area, calculated/interpolated from their own LUTs. The double orifices represent the presence of two paths for the flow: one for the port in the pump main body and the other for the so-called shadow port in the pump cover.

The model shown in **Figure 13** contains a few fixed rectangular orifices (blue-hatched icons) that provide for leakage flows across several paths displayed in **Figure 15**. The rectangular black icon in upper left corner of **Figure 13** represents the leakage path at the vane side. This user-defined icon represents flow through a narrow, rectangular orifice with its width varying as the exposed length of the vane between the rotor and the cam.

The leakage orifice parameters can be used to either allow or to block the leakage path. Leakages are modeled as a flat, fixed hydraulic orifice, with laminar resistance. The basic parameters of this orifice are shown on **Figure 14** and **Figure 15**.

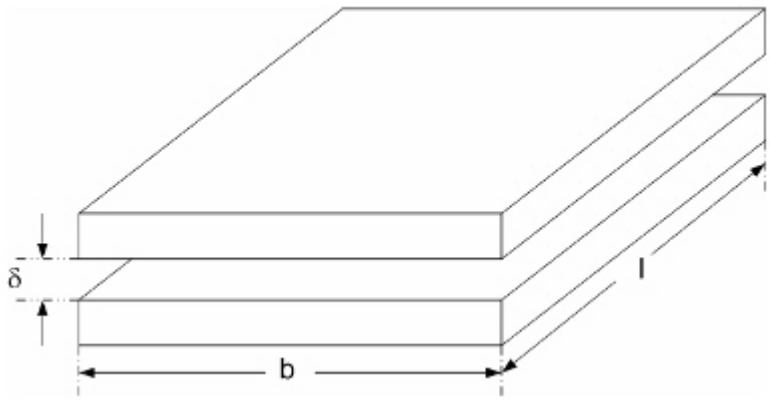


Figure 14: Fixed hydraulic orifice parameters

Flow through orifice can be represented with following equation (Ref: AmeSim Reference Manual):

$$q_b = \frac{b\delta^3}{12\mu}.\frac{p_a - p_b}{l}.\frac{\rho.(\frac{p_a - p_b}{2})}{\rho(0)}$$

Where

- q_b and p_b are resp. the flow rate and pressure at outlet port of orifice
- p_a is and pressure at inlet port of orifice
- b is the width of the rectangular geometry
- l is the length of the orifice
- ρ is the fluid density
- μ is the fluid viscosity

The AmeSim model presented in this study has the capability to include leakages in the pump; however, leakages are neglected in this model. Leakages are neglected only because of lack of a knowledge of them during initial stage of design.

Matlab scripts were developed to generate LUTs from the pump basic geometry, as shown on **Figure 16** and **Figure 17**.

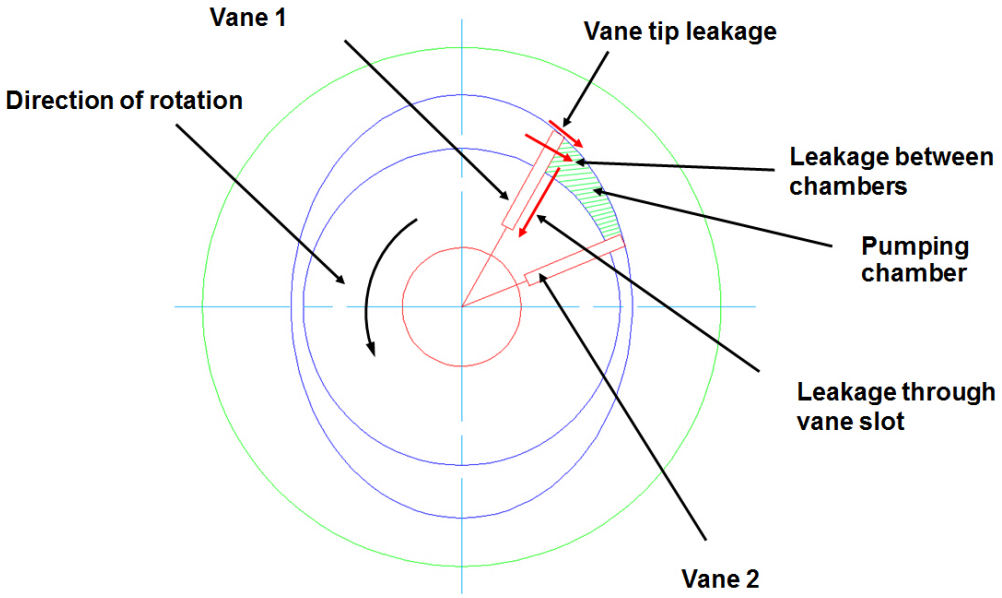


Figure 15: Leakage Paths within a Single Chamber

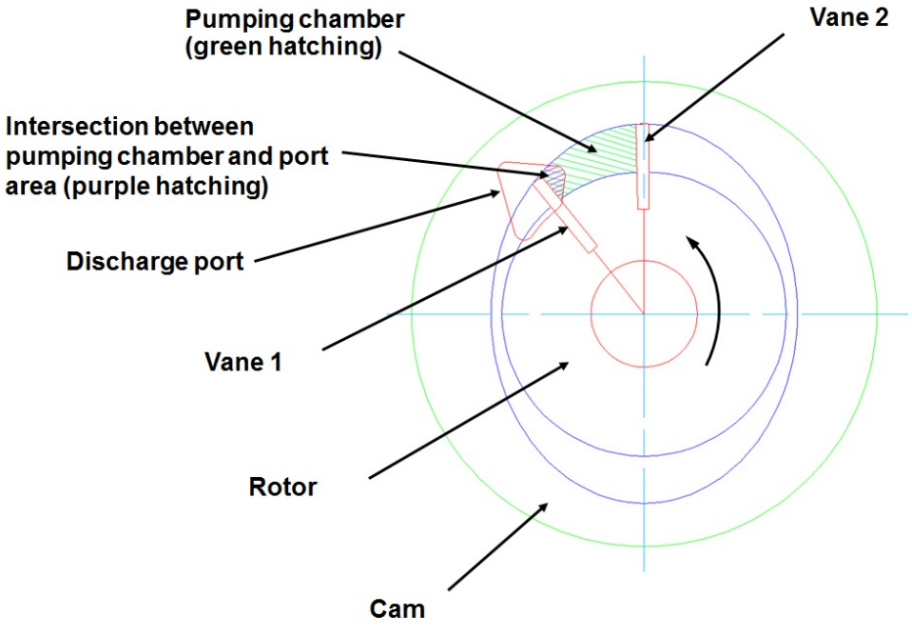


Figure 16: Establishing single cavity area at discharge port

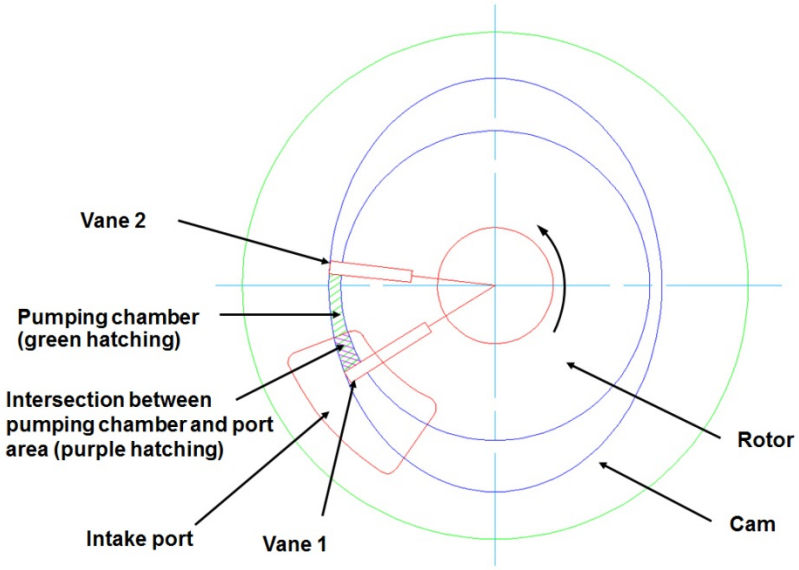


Figure 17: Establishing single cavity area at intake port

Figure 16 displays plots of the IV chamber area (a) as well as the suction and delivery port opening area (b) versus the chamber angular position. The plots were generated using data from the AmeSim model LUTS. Port opening areas with respect to rotor angle are depicted on **Figure 18**. The cross sectional area of the chamber with respect to rotor angle is depicted on **Figure 19**.

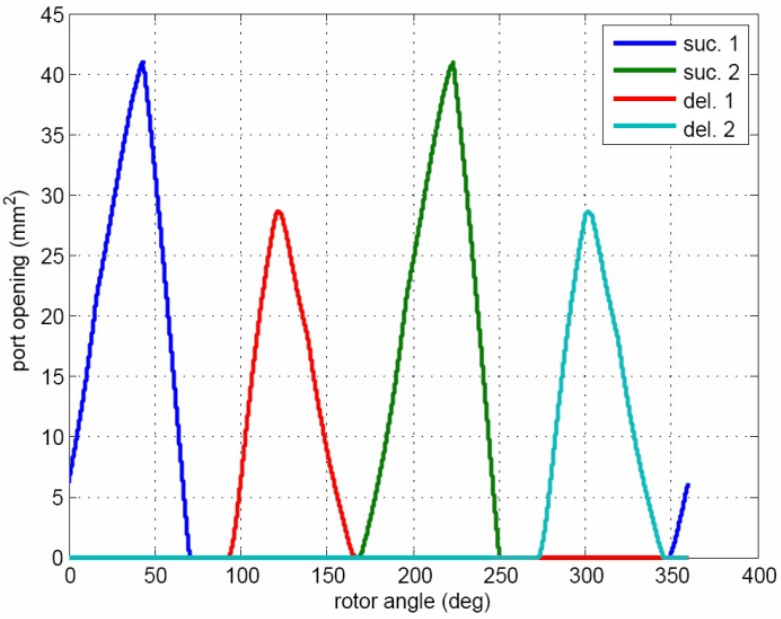


Figure 18: Port opening areas with respect to rotor angle

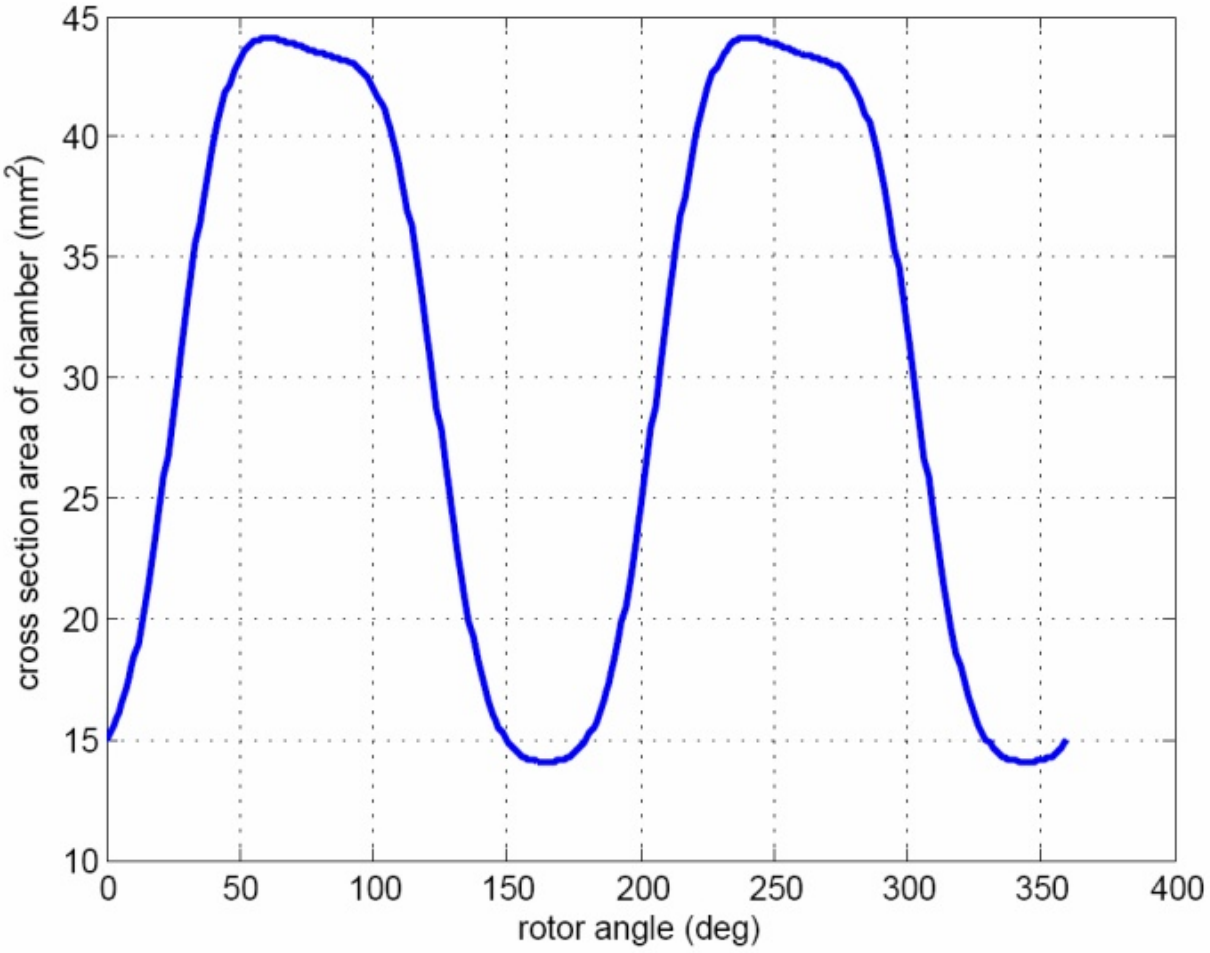


Figure 19: Cross section area of chamber with respect to rotor angle

The AmeSim model considers the impact of fluid properties such as density and viscosity, compared to the one-dimensional method where those parameters are not used. Aeration, leakages and cavitation are neglected in AmeSim model. AmeSim model also considers a more precise description of the port shape of transmission pump. Using this two-dimensional method, more complicated port shapes can be investigated, which would be not feasible using one-dimensional method.

2.8. Results from AmeSim model for two dimensional analysis

Due to approximations and lack of detailed knowledge of AmeSim model regarding leakages, pressure pulsations will not be reviewed in details. The flow performance as an output from AmeSim model is shown on **Figure 20**.

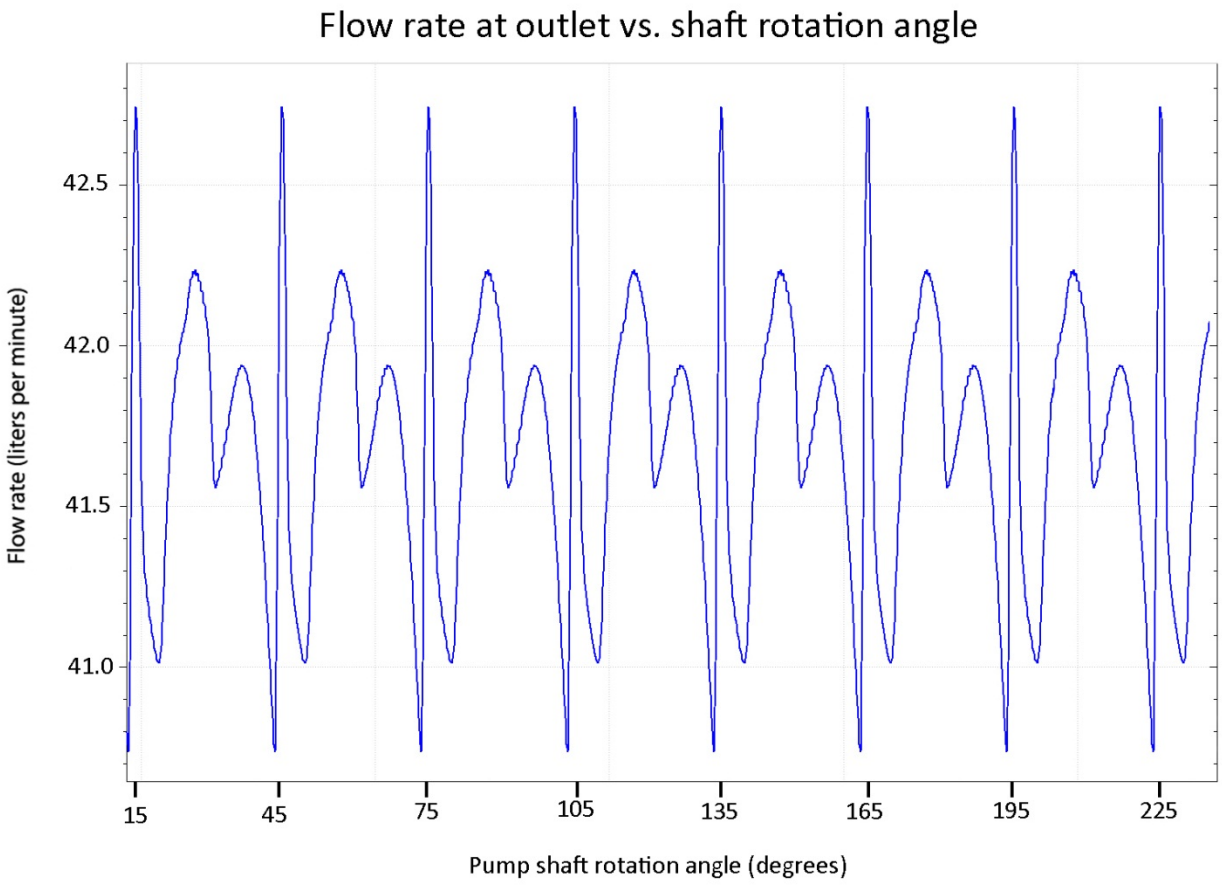


Figure 20: Flow performance from AmeSim model for balanced pump

Comparing the results obtained with the two-dimensional method shown on **Figure 20** with results obtained with one-dimensional method shown on **Figure 11**, it is seen that the results for flow are different: 43 liters per minute from the one-dimensional method and 42 liters per minute from the two-dimensional method. The two-dimensional method is considered to be more precise and the prediction should be closer to experimental results.

2.9 Limitations of the AmeSim model

The above calculations represent so called two dimensional calculations, where only two dimensional cam and ports descriptions are used to calculate the pump performance. There are many advantages with this approach. The simulation can be completed in a time frame of several minutes providing predictions for flow and pressure pulsations. The effect of the angular speed is also easily assessed. However, there are limitations to this two dimensional approach:

1. The port height along the axis is not taken into consideration, which can affect the pressure pulsations.
2. Leakages are not taken into account, which affects the final flow performance of the pump. Neglecting the effect of leakage leads to over prediction of the pressure pulsation in comparison to real pumps with leakage.
3. Aeration cannot be modeled in two dimensional models since the fluid is modeled as only homogenous, with specified properties. Aeration affects the pressure pulsations.
4. Cavitation is not modeled. As such, degradation of flow performance due to severe cavitation cannot be detected.
5. Pressure contours throughout the 3D geometry are not calculated. As such, dead zone areas (zone pockets where fluid flow is abrupt and where pressure builds up) cannot be identified with two dimensional approaches.

In contrast to the 1D and 2D models described above, Computational Fluid Dynamics (CFD) provides a detailed description of the velocity and pressure field throughout the domain. CFD modeling can allow for the prediction of the effects of aeration of the fluid and leakages between chambers and is also capable of predicting cavitation. The current research will use CFD analysis to improve the design of a balanced pump.

Chapter 3 - Model for balanced pump

3.1. CFD usage for Automobile and Engine applications

To improve the performance of modern cars and trucks (environmental quality, fuel economy, etc.), the automobile industry has accelerated its use of high-technology research and design tools (J. Anderson,1995). One of these tools is Computational Fluid Dynamics (CFD). Whether it is the study of the external flow over the body of a vehicle, or the internal flow through the engine, CFD is helping automotive engineers to better understand the physical flow processes, and in turn to design improved vehicles.

CFD has proven to be an excellent tool for the automotive industry, since it lowers costs for testing, it can be executed in a relatively short period of time, turnaround is quick and engineering data can be obtained early in the design process. CFD analysis gives comprehensive information about flow, turbulence, pressure gradient, velocity field and cavitation for different regions of interests.

CFD code used in this research is Ansys Fluent, version 12.1. It is installed on Lenovo D20 workstation on XP operating system with dual Xeon CPU X5560 @ 2.80 GHz and 16 GB of RAM memory.

3.2. Description of the components of a balanced transmission pump

This section provides a description of the pump that is studied in this research. The work consists of both numerical and experimental research on a physical prototype. The pump is designed so that both suction and delivery ports in the pump are diametrically opposed to provide complete balance of all internal radial forces (Ref: A. Guiffrida, R. Lanzafame, 2004). In this pump, the side plates have vane backpressure grooves at the sides facing the rotor to introduce the delivery pump pressure to all of the bottoms of the vane slot of the rotor. During the pump operation, the vane is always pushed from the bottom by the delivery pressure and rotates with the loads imposed by the vane tip on the cam

contour. The rotor is driven by a shaft through a very loose-fitting spline and the rotor and vanes rotate between two-side plates with running clearance. The housing and cover were made from machined aluminum. Production intent components will be made with die-casting process. The cam ring was made of ferro-sintered alloy and the inner surfaces of the cam rings were ground with roughness of 1–2 mm Rz. The rotor is made of hardened steel. The vanes were finished by the barrel polishing and the roughness of the vane tip was ~0.3 µm Rz. The expression of roughness of Rz, average peak-to-valley height, is the average value of the individual peak-to-valley heights in five continuous individual measurement sections. The hydraulic fluid used was commercial mineral oil and its density ρ and viscosity ν at 90°C are 804 kg/m^3 and 0.005885 Pas, respectively. Pins are used in the assembly to prevent relative movement of the cam, housing and cover during the pump operation. A c-clip is used to prevent axial movement of components relative to shaft. An exploded assembly of the balanced pump is shown in **Figure 21**. The housing shown on the assembly on **Figure 21** is for production. Since the pump must be instrumented with dynamic pressure transducers, the housing was modified in order to be able to accommodate the pressure transducers. The housing used for testing for this research is shown on **Figure 22**.

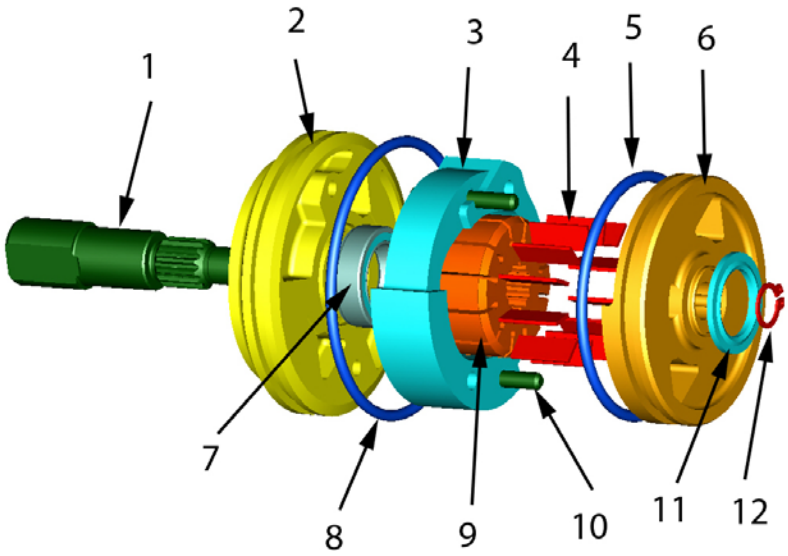


Figure 21: Components description for balanced pump

1. Drive shaft
2. Housing (production intent), housing used in prototype has place for pressure transducers
3. Cam
4. Vanes
5. Back o-ring
6. Cover
7. Seal
8. Front o-ring
9. Rotor
10. Pins
11. Bottom seal
12. C-clip

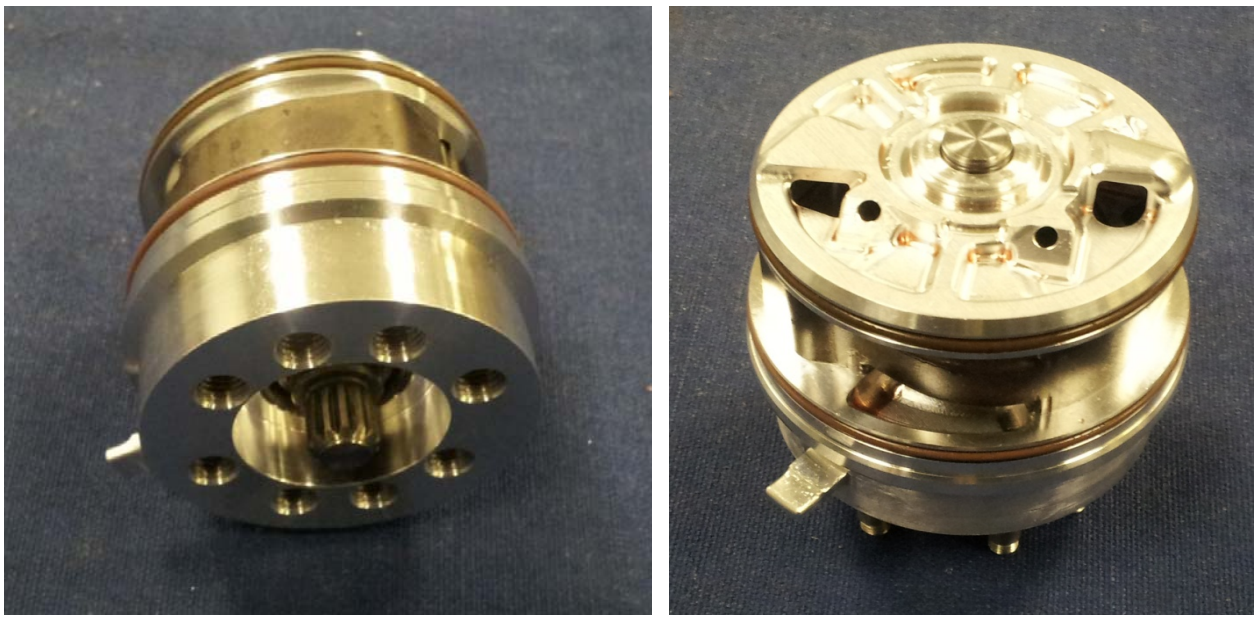


Figure 22: Actual prototype tested on a test stand

3.3. Balanced pump meshing approach, structured and unstructured finite volume schemes

As previously introduced, this research presents a computational fluid dynamic analysis of a balanced transmission vane pump. The geometry for this pump is designed using CAD software. Fluid domains are extracted from the geometry and are shown on **Figure 23**.

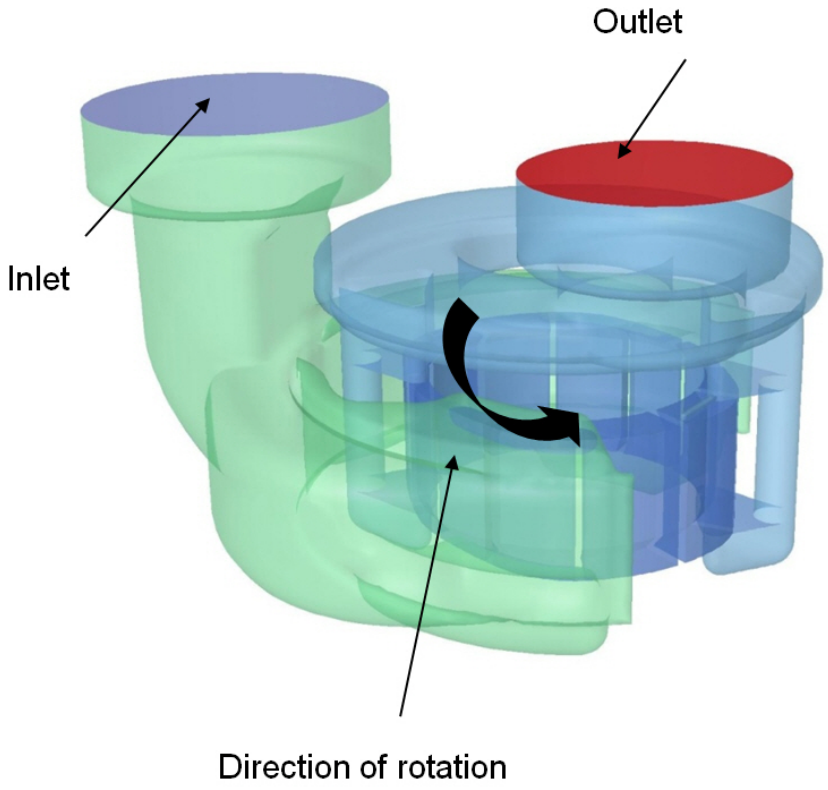


Figure 23: Geometry of extracted fluid domains of the pump

To conduct CFD simulations, a hexahedral and tetrahedral mesh was generated by using Gambit and T-Grid mesh generator. In particular, inlet and outlet fluid domains were discretized by tetrahedral elements while rotating chambers and end face clearances were discretized by hexahedral elements. The total number of cells in CFD model used in this research was about 430 000.

In order to check the effect of grid density to results of CFD simulations, several CFD models with different grids were analyzed. Initially, a CFD model with a total cell count of about 326 000 elements

was analyzed. Then, the grid was gradually increased to 430 000, 887 000 and 1 490 000 elements. The initial grid density of 326 000 elements gave rough results for geometry representation, in comparison to other CFD models. Results for other models than the initial are identical, thus all consecutive CFD simulation were conducted with grid size of about 430 000 elements.

The considered computational domain was comprised of four main parts (**Figure 24** and **Figure 25**):

- Fluid domain for pump inlet (shown on **Figure 24a**);
- Fluid domain for pump outlet (shown on **Figure 24b**);
- Fluid domain made of 12 pump chambers (shown on **Figure 25b**);
- Fluid domain for end face clearances, located at the top and bottom of pump chambers (shown on **Figure 25a**);

In particular, inside the cam ring, fluid volumes are identified and bounded by the cam, the rotor and the vanes. Each vane rotates off-center with respect to the housing guaranteeing the expansion of the volumes during the suction phase and the contraction of the volumes during the discharge phase. This expansion/contraction of the volumes generates the pumping action. **Figure 26** shows a top view of the pump rotor ring and the inlet and outlet ports.

There are several interfaces in the pump:

- Between inlet domain, outlet domain and end face clearances;
- Between end-face clearances and pump chambers on both sides;
- Between pump chambers and inlet domain at the side, shown on **Figure 27**.

The interface between the inlet fluid domain and the end face clearances is shown in green, whereas the interface between the inlet fluid domain and chambers is shown in magenta.

The finite volume mesh of the full CFD model of the pump is show on **Figure 28**.

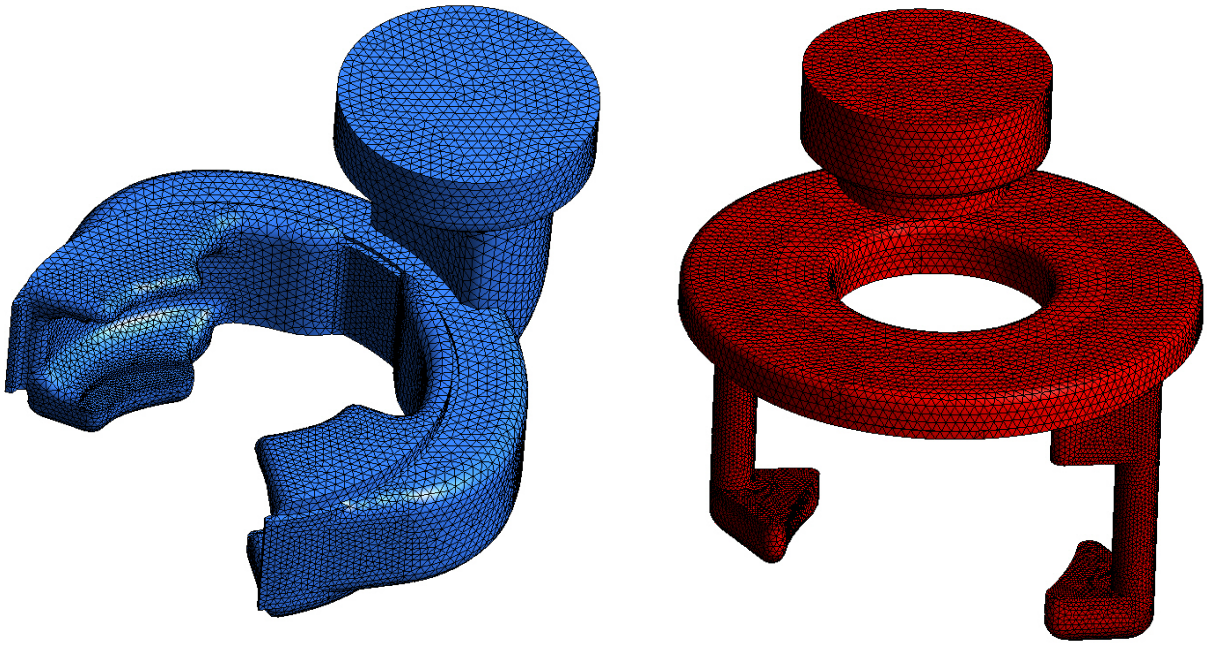


Figure 24: Inlet and outlet fluid domains of the pump

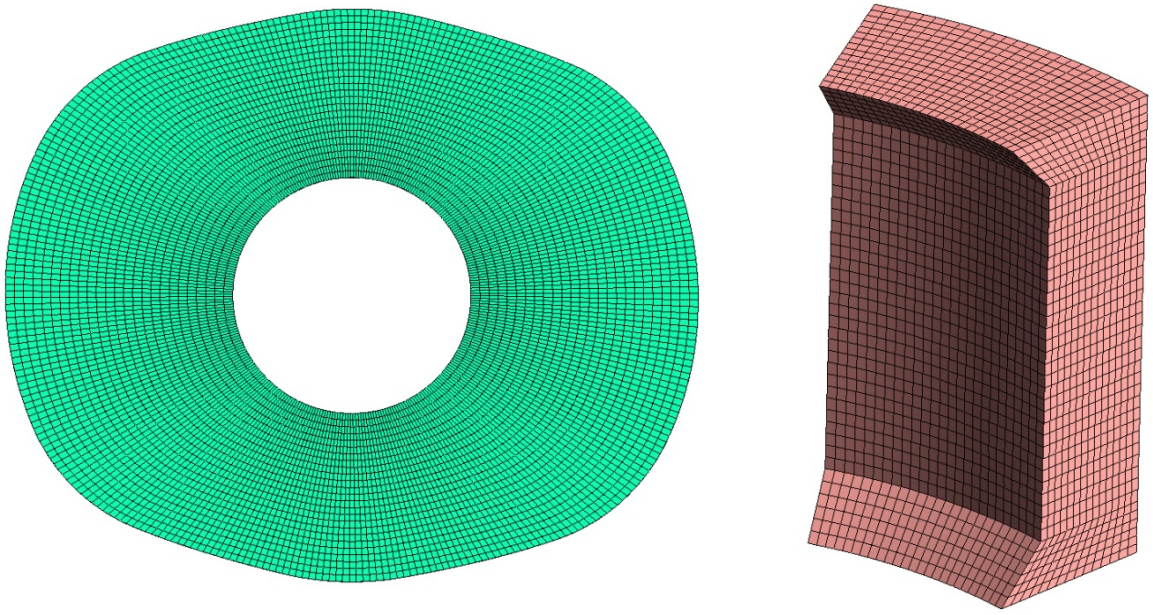


Figure 25: Domain for end-face clearance and one pumping chamber

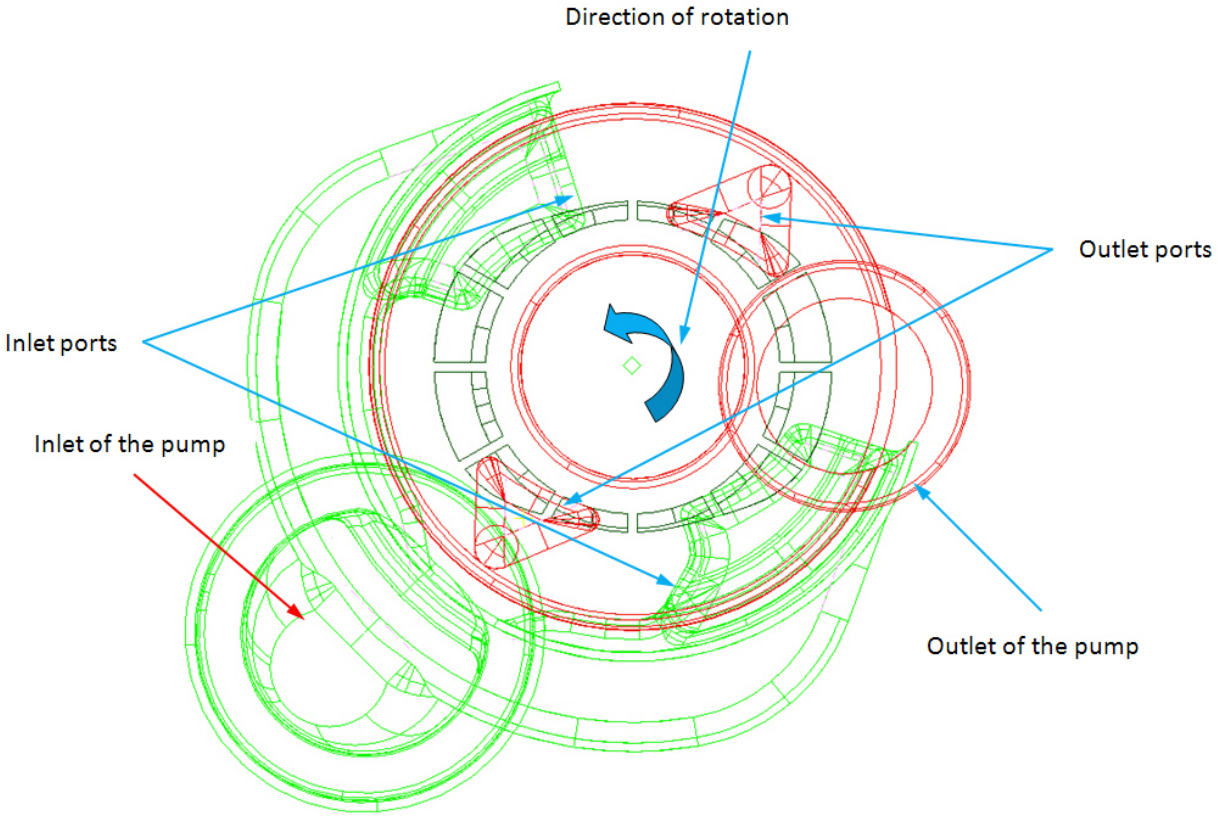


Figure 26: Top view of the pump, with pump chambers, inlet and outlet ports

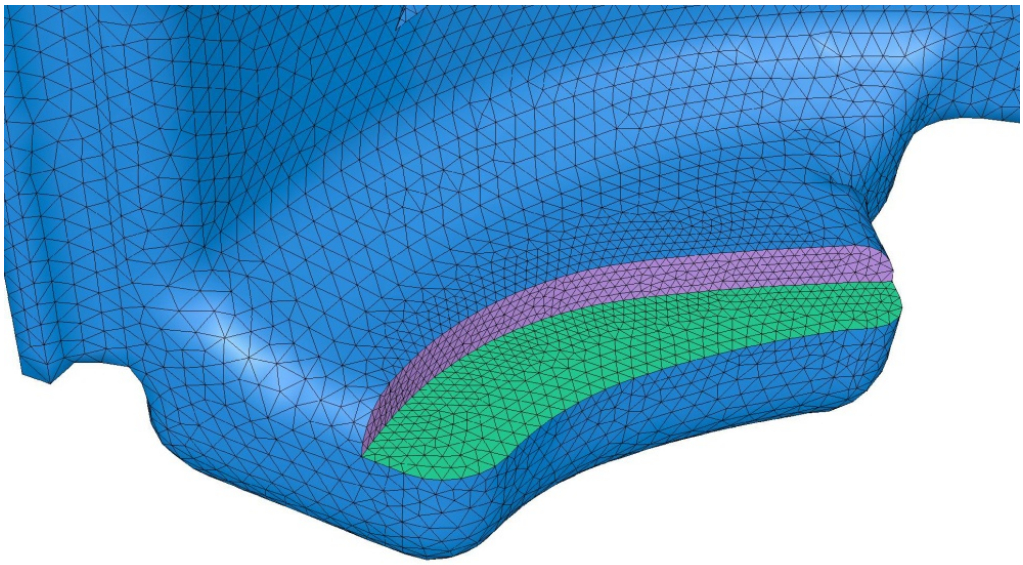


Figure 27: Interface zones at inlet domain of the pump

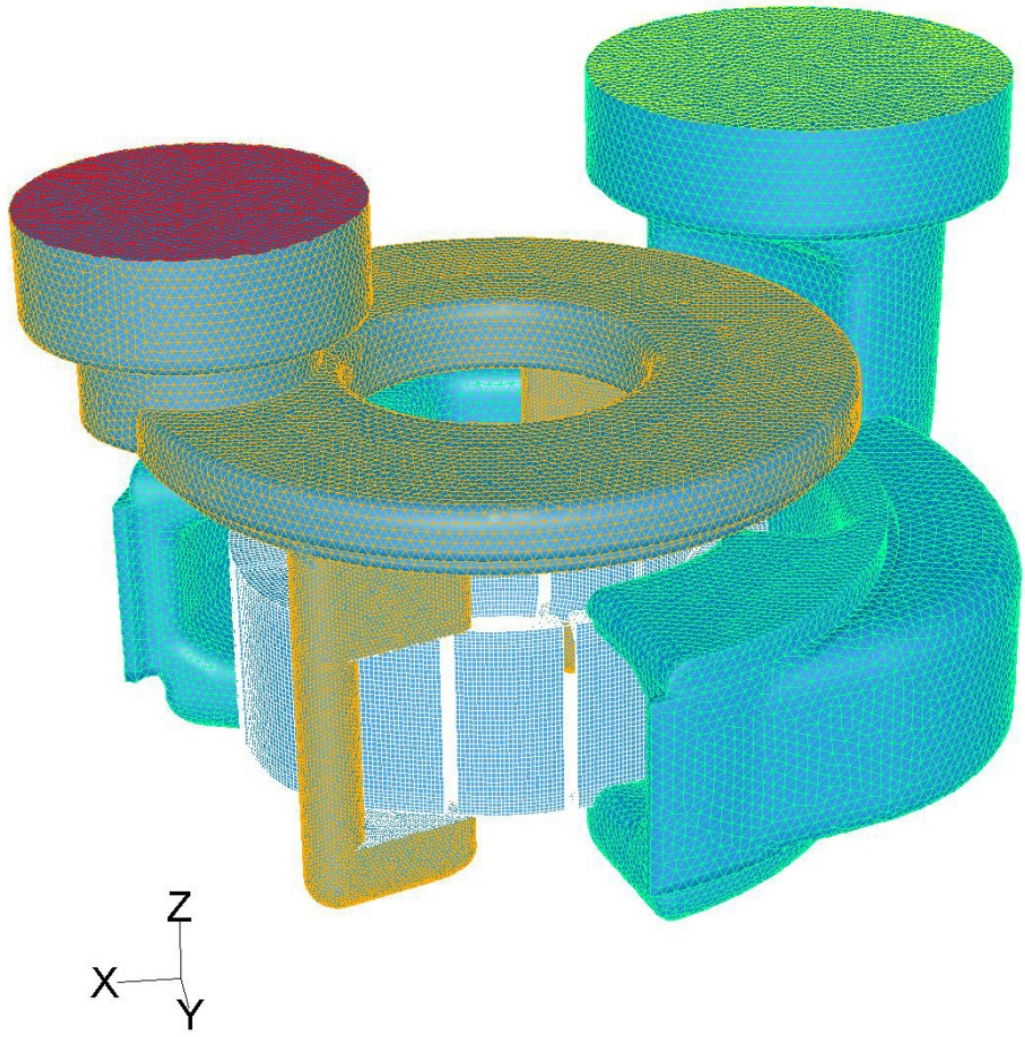


Figure 28: Finite volume mesh of fluid domains of balanced pump

3.4. Mesh generation and moving mesh methodology

In order to allow a realistic numerical reproduction of the rotary vane pump it was necessary to numerically reproduce the change-in-time shape of the rotor chambers: the mesh structure previously presented was chosen also on the basis of this requirement. To simulate the time-varying vane motions, the Fluent v12.1 dynamic mesh capability was used. In particular, the motion was described by the definition of a user defined function (UDF). On the basis of this UDF, the displacement of each mesh node of the fluid chambers was defined as the sum of two motions: a solid body rotation and a radial

translation. Node displacements were a function of the chamber eccentricity, the revolution speed, and the cam shape. Based on these inputs, Fluent automatically updated the rotor mesh at each time step. The use of this UDF to manage the vane motion instead of a classic smoothing/re-meshing approach was very useful to reduce the computational cost and to improve the numerical stability of the simulation. Moreover, using the defined UDF no addition/subtraction of cells was necessary during the mesh motion. The UDF used in simulations is given in

Appendix B.

3.5. Boundary conditions for CFD model, model conditions, turbulence and solver setting

3.5.1. Boundary conditions for CFD model

A total pressure boundary condition was placed on the inlet of the pump and a static pressure boundary condition was placed on the outlet of the pump. The rotational velocity for the pump is 4000 rpm. The temperature of the transmission oil is 90°C and the outlet pressure set to 8 bar.

Fluid used in Simulation is Dexron VI. Properties are given in **Table 1**.

Table 1: Properties for fluid used in simulation

Type of oil used in simulation	Dexron VI transmission oil
Temperature for testing	90°C
Oil Density at 90°C	804 kg/m^3
Dynamic oil viscosity at 90°C	0.00588528 kg/m-s

Additional details on fluid properties are given in **Appendix C**.

3.5.2. Model conditions and time step used

The code treats fluid as a mixture with a constant aeration of 4% of air. The mixture model can model *n* phases (fluid or particulate) by solving the momentum, continuity, and energy equations for the mixture, the volume fraction equations for the secondary phases, and algebraic expressions for the relative velocities. Additional details on mixture model are given in **Appendix D**.

In transient simulation it is important to assess the effect of the time step. Initially, the time step was set to yield a rotation of 2.5 degrees per time step. To assess the effect of time step, the time step was decreased to yield a rotation of 1.09 degrees per time step, which allowed the mesh movement to exactly match the node to node at the interface for each consecutive step. The finer time step predicted pressure pulsations peaks which were not captured with the larger time step. The pressure predictions are shown in **Figure 29** where the red curve represents the result for the time step of 2.5 degrees, and the green curve is for time step of 1.09 degrees. At the initial stage, finer time step of 0.545 degrees is also explored, which gave same results as time step of 1.09 degrees. Thus, all consecutive analyses were completed with a step of 1.09 degrees of mesh movement.

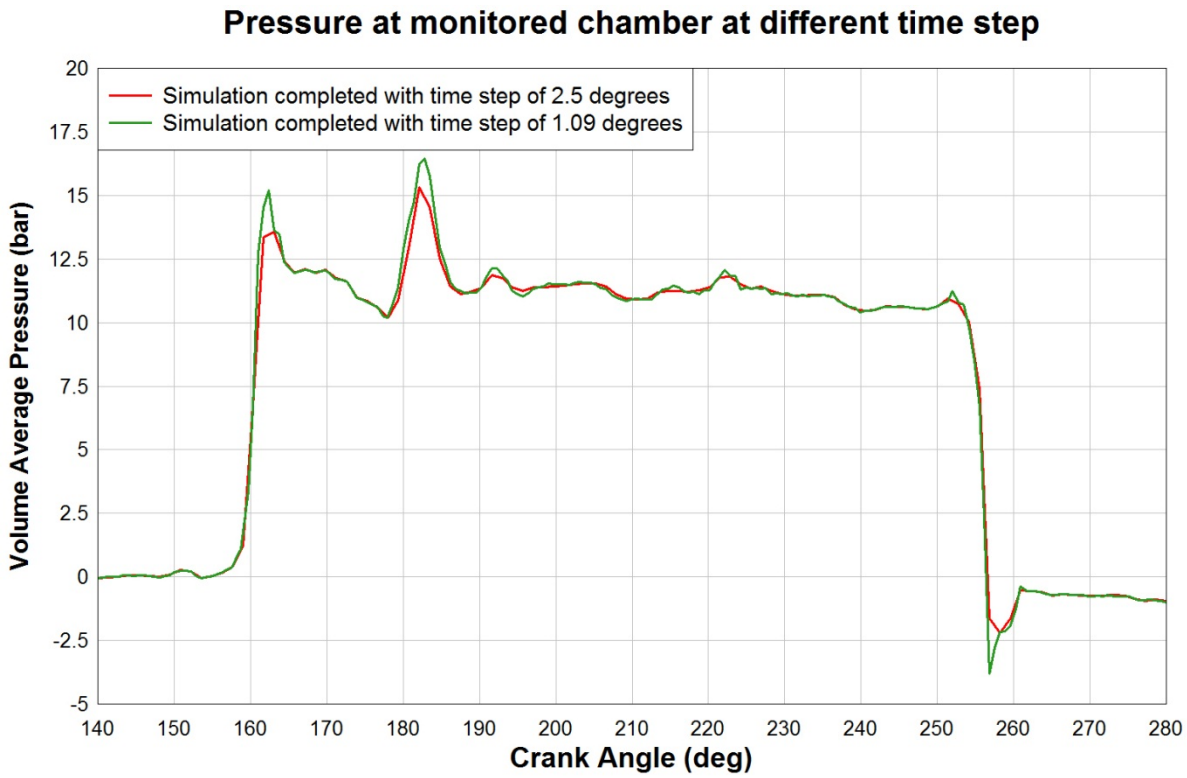


Figure 29: Pressure monitors at chamber for different time steps (green curve is for time step of 1.09 degrees and red curve is for2.5 degree change)

3.5.3. Turbulence Modeling

It is generally recognized that all existing turbulence models, including those mentioned here, are inexact representations of the physical phenomena of turbulence. The degree of approximation in a given model depends on the nature of the flow to which it is being applied, and the characterization of the circumstances which give rise to 'good' and 'bad' performance must unfortunately be based mainly on experience.

The turbulence model used in the simulations is the standard k-ε model. During initial research, the other turbulence model was also considered (the *k- ω m*odel). There were no differences in results between those two turbulence models (see **Figure 30**). Therefore, all other consecutive simulations were conducted using the standard k-ε model.

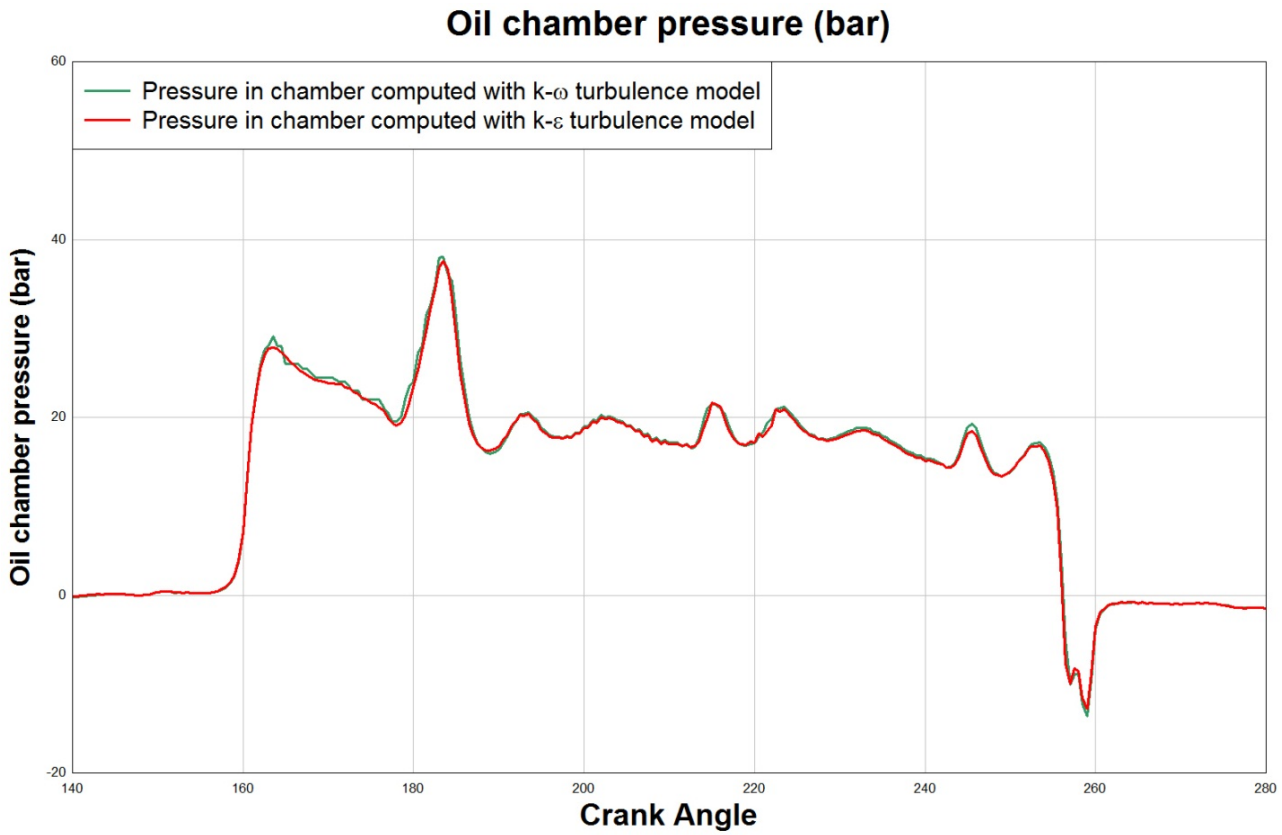


Figure 30: Pressure in one chamber computed with both turbulence models

The exact equation for ε is derived by taking the following moment of the Navier-Stokes equation (Ref: Ansys Fluent 2011, Theory Guide).

$$\overline{2v\frac{\partial u_i'}{\partial x_j}\frac{\partial}{\partial x_j}[\mathcal{N}(u_i)]} = 0$$

where $\mathcal{N}(u_i)$ is the Navier-Stokes operator. After a considerable amount of algebra, the following equation for ε results:

$$\rho\frac{\partial \epsilon}{\partial t} + \rho U_j\frac{\partial \epsilon}{\partial x_j} = -2\mu\left[\overline{u_{i,k}'u_{j,k}'} + \overline{u_{k,i}'u_{k,J}'}\right]\frac{\partial U_i}{\partial x_j} - 2\mu\,\overline{u_k'u_{i,J}'}\frac{\partial^2 U_i}{\partial x_k \partial x_j} - 2\mu\,\overline{u_{i,k}'u_{i,m}'u_{k,m}'} -$$

$$2\mu\, v\,\overline{u_{i,km}'u_{i,km}'} + \frac{\partial}{\partial x_j}\left[\mu\frac{\partial \epsilon}{\partial x_j} - \mu\,\overline{u_j'u_{i,m}'u_{i,m}'} - 2v\,\overline{p_{,m}'u_{j,m}'}\right] \qquad \textbf{(7)}$$

This equation is far more complicated than the turbulence kinetic energy equation and involves several new unknown double and triple correlations of fluctuating velocity, pressure and velocity gradients. The terms on the three lines of the right-hand side of **Eq. (7)** are generally regarded as **Production of Dissipation, Dissipation of Dissipation,** and the sum of **Molecular Diffusion of Dissipation** and **Turbulent Transport of Dissipation,** respectively.

The **Standard k- ε model** is as follows:

Eddy Viscosity:

$$\mu_T = \rho C_\mu k^2/\epsilon$$

Turbulence Kinetic Energy:

$$\rho\frac{\partial k}{\partial t} + \rho U_j\frac{\partial k}{\partial x_j} = \tau_{ij}\frac{\partial U_i}{\partial x_j} - \rho\epsilon + \frac{\partial}{\partial x_j}\left[(\mu + \mu_T/\sigma_k)\frac{\partial k}{\partial x_j}\right]$$

Dissipation Rate:

$$\rho\frac{\partial \epsilon}{\partial t} + \rho U_j\frac{\partial \epsilon}{\partial x_j} = C_{\epsilon 1}\frac{\epsilon}{k}\tau_{ij}\frac{\partial U_i}{\partial x_j} - C_{\epsilon 2}\rho\frac{\epsilon^2}{k} + \frac{\partial}{\partial x_j}\left[(\mu + \mu_T/\sigma_\epsilon)\frac{\partial \epsilon}{\partial x_j}\right]$$

Closure Coefficients:

$$C_{\epsilon 1} = 1.44, \qquad C_{\epsilon 2} = 1.92, \qquad C_\mu = 0.09, \qquad \sigma_k = 1.0, \qquad \sigma_\epsilon = 1.3$$

Auxiliary Relations:

$$\omega = \epsilon/(C_\mu k) \quad \text{and} \quad \ell = C_\mu k^{3/2}/\epsilon$$

3.5.4. Governing Equations

The derivation of the principal equations of fluid dynamics is based on the fact that the dynamical behavior of a fluid is determined by the following *conservation laws,* namely:

1. the conservation of mass,
2. the conservation of momentum, and
3. the conservation of energy.

The conservation of a certain flow quantity means that its total variation inside an arbitrary volume can be expressed as the net effect of the amount of the quantity being transported across the boundary, any internal forces and sources, and external forces acting on the volume (Ref: J. Blazek (2001). Computational Fluid Dynamics - Principles and Applications).

Consider a general flow field as represented by streamlines in **Figure 31**. An arbitrary finite region of the flow, bounded by the closed surface dΩ and fixed in space, defines the control volume Ω. We also introduce a surface element as *dS* and its associated, outward pointing unit normal vector as $\vec{n}$.

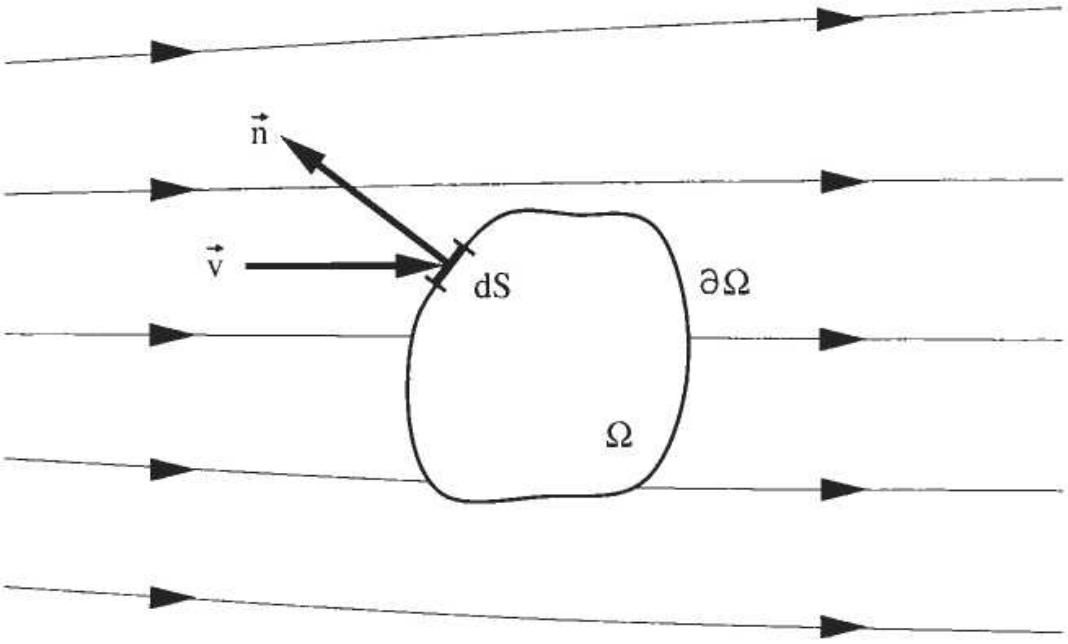


Figure 31: Definition of a finite control volume (fixed in space)

General form of the conservation law for the scalar quantity *U*

$$\frac{\partial}{\partial t}\int_{\Omega} U d\Omega + \oint_{\partial\Omega}[U(\vec{v}\cdot\vec{n}) - \kappa\rho(\nabla U^{*}\cdot\vec{n})]dS \;=\; \int_{\Omega} Q_V d\Omega + \oint_{\partial\Omega}(\vec{Q}_S\cdot\vec{n})\,dS$$

where U^{*} denotes the quantity *U* per unit mass, i.e., *U/ρ.*

3.5.5. Solver settings

Ansys Fluent allows for a choice of one of the two numerical methods for solving the governing equations (Ref: Ansys Fluent, 2011, Theory Guide):

- pressure-based solver
- density-based solver

The pressure-based approach was developed for low-speed incompressible flows, while the density-based approach was mainly used for high-speed compressible flows. In both methods, the velocity field is obtained from the momentum equations. In the density-based approach, the continuity equation is used to obtain the density field while the pressure field is determined from the equation of state. With the pressure-based approach, the pressure field is extracted by solving a pressure or pressure correction equation which is obtained by manipulating the continuity and momentum equations.

Using either method, Ansys Fluent will solve the governing integral equations for the conservation of mass and momentum, and (when appropriate) for energy and other scalars such as turbulence and chemical species. In both cases a control-volume-based technique is used that consists of:

- Division of the domain into discrete control volumes using a computational grid.
- Integration of the governing equations on the individual control volumes to construct algebraic equations for the discrete dependent variables ("unknowns") such as velocities, pressure, temperature, and conserved scalars.
- Linearization of the discretized equations and solution of the resultant linear equation system to yield updated values of the dependent variables.

Since transmission oil has very low compressibility, the pressure based solver is used for all analyses presented in this research.

3.5.6. Pressure-Based Solver

The pressure-based solver employs an algorithm which belongs to a general class of methods called the projection method. In the projection method, wherein the constraint of mass conservation (continuity) of

the velocity field is achieved by solving a pressure (or pressure correction) equation (Ref: Ansys Theory Guide 2011). The pressure equation is derived from the continuity and the momentum equations in such a way that the velocity field, corrected by the pressure, satisfies the continuity. Since the governing equations are nonlinear and coupled to one another, the solution process involves iterations wherein the entire set of governing equations is solved repeatedly until the solution converges.

Two pressure-based solver algorithms are available in Ansys Fluent: a segregated algorithm, and a coupled algorithm. These two approaches are discussed in the sections below.

3.5.7. Pressure-Based Segregated Algorithm

The pressure-based solver uses a solution algorithm where the governing equations are solved sequentially (that is, segregated from one another). Because the governing equations are non-linear and coupled, the solution loop must be carried out iteratively in order to obtain a converged numerical solution.

In the segregated algorithm, the individual governing equations for the solution variables are solved one after another. Each governing equation, while being solved, is "decoupled" or "segregated" from other equations, hence its name. The segregated algorithm is memory-efficient, since the discretized equations need only be stored in the memory one at a time. However, the solution convergence is relatively slow, in as much as the equations are solved in a decoupled manner.

3.5.8. The Pressure-Based Coupled Algorithm

Unlike the segregated algorithm described above, the pressure-based coupled algorithm solves a coupled system of equations comprising the momentum equations and the pressure-based continuity equation. Thus, in the coupled algorithm, Steps 2 and 3 in the segregated solution algorithm are replaced by a single

step in which the coupled system of equations are solved. The remaining equations are solved in a decoupled fashion as in the segregated algorithm.

Since the momentum and continuity equations are solved in a closely coupled manner, the rate of solution convergence significantly improves when compared to the segregated algorithm. However, the memory requirement increases by 1.5 - 2 times that of the segregated algorithm since the discrete system of all momentum and pressure-based continuity equations must be stored in the memory when solving for the velocity and pressure fields (rather than just a single equation, as is the case with the segregated algorithm).

Both the segregated and coupled algorithms were tried during initial evaluation of models. There was no significant difference in processing speed when compared both algorithms. Therefore, the segregated solver settings are used for simulations presented in this research.

Summarized solver settings are shown at **Table 2**:

Table 2: Solver settings used in simulations

Function	Setting
Solver	Segregated Implicit
Time-stepping	1st order implicit
Pressure Discretization	Standard
Momentum Discretization	1st order upwind
Pressure-velocity coupling	SIMPLEC
Time step size	Node to node on chambers

3.6. Leakages taken into account during the simulation

The leakages are also accounted for in the CFD model of the pump. The leakages are a function of the pump clearances and oil temperature. The following leakages exist in the actual pump:

1. Leakage due to the housing, cover and cam deformation at high pressure
2. Leakage due to axial clearance between the rotor and cam along the height
3. Leakage past the vane tips
4. Leakage along the vane slots
5. Leakage through the seals of the pump

Since the CFD model represents a rigid model, leakages due to deformation cannot be captured. Leakage due to axial clearance between rotor and cam along the height are taken into account through the end face clearances on both sides of pump chambers. The leakage past the vane tips are neglected and compensated by the end face clearances in the fluid domains. The vane slots leakage is not investigated, because it is normally very small in comparison to the other leakages. Leakage through the pump seals are taken into account with a circular outlet at the clearances of the fluid domain. The thickness of the fluid domains for the end-face clearances is set to 35 microns, on each side. The effect of presumed thickness was tested and found that it had a small effect on the performance of the oil pump.

3.7. Description of leakages, oil properties and aeration taken into account during the simulation

As previously mentioned, end-face clearances are set to 35 microns, on each side. This clearance is higher than actual clearance measured on the balanced pump. The reason for having higher clearance than actual is to compensate leakages due to deformation, leakages past vane tips, vane slots and through seals. According previous studies for vane oil pumps with similar sizes and performances, end-face clearances is adopted to be 35 microns, on each side. These clearances are assumed to be at 90°C, which is the pump operating temperature. These clearances are expected to vary with different temperature. If the pump needs to be tested at 0°C, the clearances used in CFD simulation would have different value.

Similarly, values for oil density and viscosity are taken at 90°C. Input parameters for CFD simulations are oil density and oil dynamic viscosity. There are noticeable differences in the simulations results if we compare flow and pressure pulsations of the same CFD model, but for different oil properties resulting at different operating temperature. Thus, oil properties are important in simulation process.

Aeration on all models is set to be 4%. This value is adopted using previous studies for pressure pulsations for oil pumps. Aeration in the model has a dampening effect on pressure pulsations.

Pressure pulsations are highest at aeration of 0% (homogenous oil as a single phase). Pressure pulsations have almost the same value if aeration of 4% to 7 % is used. Thus, aeration is set to be 4% for all CFD simulations in this thesis.

Chapter 4 - Test stand set-up and results from the testing

Testing of the balanced pump prototype was conducted at the testing facility located in Magna Powertrain plant. My contributions to the experimental part of this thesis include the design of the test pump, planning the locations of pressure measurements and defining the procedure and conditions of testing. However, the tests were conducted by a test technician and testing engineer. The testing facility has the capability to test engine oil pumps, transmission oil pumps, water pumps and vacuum pumps. Testing of the balanced pump was conducted at room temperature, in a normal environment. Anechoic chamber is not used during this testing. Pressure sensors are installed at the inlet and outlet of the pump and at shaft. The purpose of the shaft pressure sensor is to determine if there is any significant pressure drop due to leakage caused by failure of a mechanical seal. The sound pressure level is also measured using a microphone installed above the balanced pump. Details of the test setup are depicted on **Figure 32**. On the back side of the test-stand, pump speed and torque are monitored and recorded. Details of the setup for torque and speed monitoring of drive shaft is depicted on **Figure 33**. Other details for test stand are depicted on **Figure 35** to **Figure 38**.

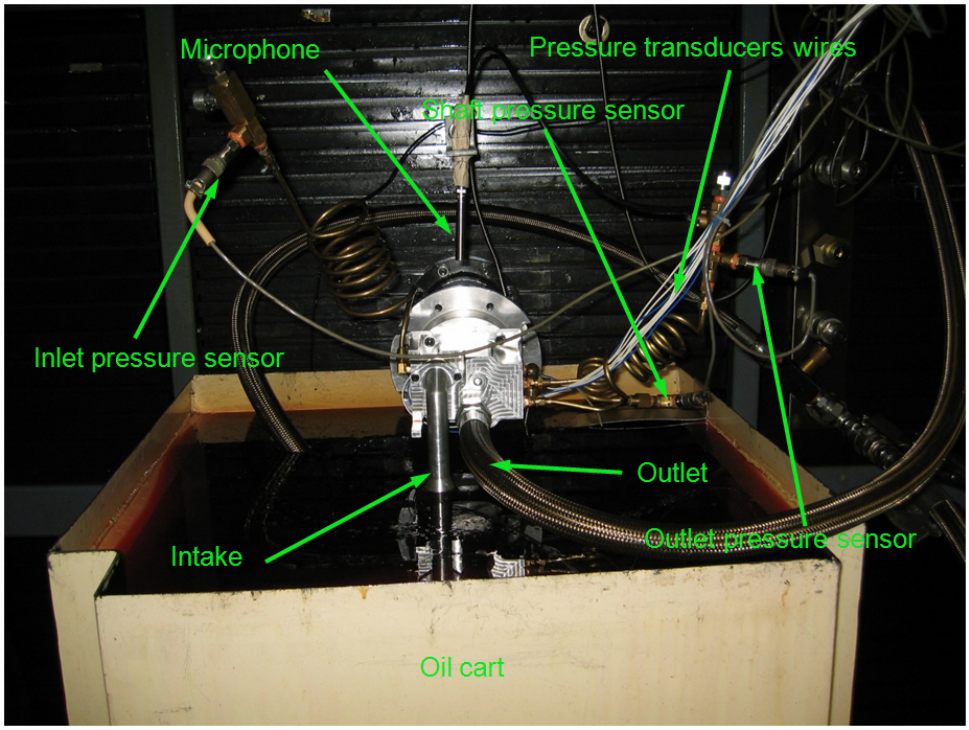


Figure 32: Oil Pressure Pump Measurement, Test setup

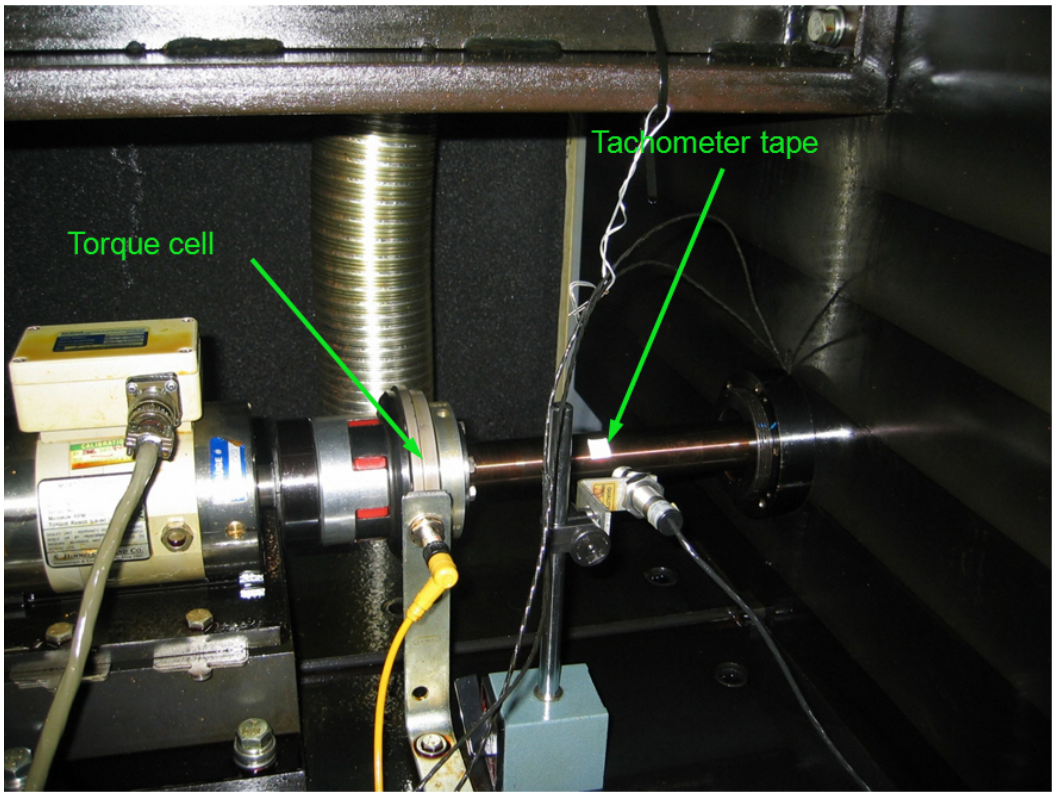


Figure 33: Test setup, speed and torque measurement

4.1. Data Acquisition

The data captured during the test originated in eight PCB (113B26) piezoelectric pressure sensors, single G.R.A.S. (40 AE ½" free field microphone with preamplifier 26 CA) microphone and one optical tachometer (Monarch PLT 200).

During testing, signals from the sensors where fed into an HP E1432A 16-chanel digitizer and sampled at the rate of 51200 Samples/second (S/s). The digitizer was connected via FireWire to a laptop. The laptop-resident Matlab with Data Acquisition Toolbox was used to set-up the data acquisition conditions and the resultant data storage on the laptop.

Equipment is calibrated by manufacturer and errors for data collections are specified on all components and are described below:

- Pressure sensors error is 0.1% of full scale (35 bar), which is 3.5 kPa maximum for the measurement.
- Error on torque cell is specified with non-linearity (±0.15%), hysteresis (±0.15%) and non-repeatability (±0.07%) of full scale, which is up to 226 Nm.
- Flow meter error is 0.05% of measured flow rate. Maximum flow rate which can be measured is 200 liters per minute.
- Encoder error for measuring rpm is ±30 rpm, or 0.75% for tested speed.
- Temperature sensor error is ±0.5 C°.
- Microphone open circuit sensitivity is ±2 dB at 250 Hz.

4.2. 4000 rpm – Oil Pressure Data Processing

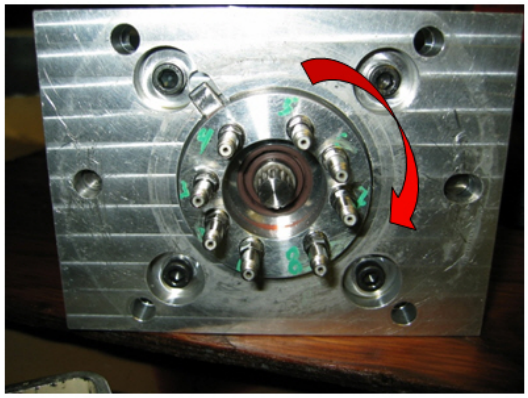
Balanced pump is tested extensively, at different speeds and two different temperatures. The pump is tested at room temperature of 25°C and operating temperature for transmission, which is 90°C. The pump is not tested at extreme temperatures at -40°C and 120°C. For this research purposes, only pump operating temperature is taken into consideration, which is 90°C. Test stand has with oil heaters controlled with thermocouple. Oil temperature does not vary for more than ±1°C at tested operating temperature of 90°C.

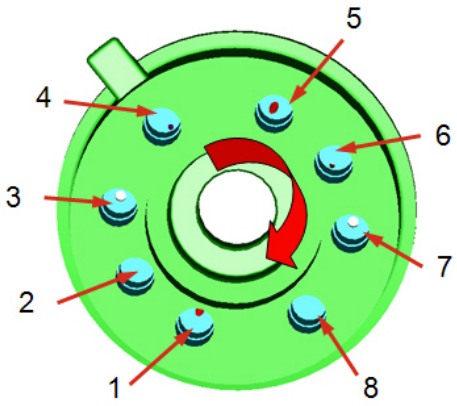
The balanced pump is tested at various rotational speeds, ranging from 500 rpm (which is at idle engine speed), up to 6000 rpm, which is maximum required speed for the balanced pump. The balanced pump is tested with increments of 1000 rpm, with addition of 4500 rpm and 5500 rpm. At room temperature, idle speed is not tested, because pump performance at idle speed on a cold engine is not considered as important compared to operational temperature of 90°C. In order to ensure stability of data acquisition, data during the test is collected after 1 minute of pump operation at certain speed. Data is collected over a time duration of 20 seconds. The first 100,000 samples (~ 2s) of the 20 seconds of the captured raw data were discarded to remove the initial part of the sound signal, since it can be affected by the settling of the antialiasing filter. Although the antialiasing filter did not adversely influence the oil pressure signal, to preserve the consistency of the data, it was decided to apply the same treatment to the complete, simultaneously sampled set. Data is collected 2 times separately, with 2 different data acquisition systems, just to ensure repeatability of test.

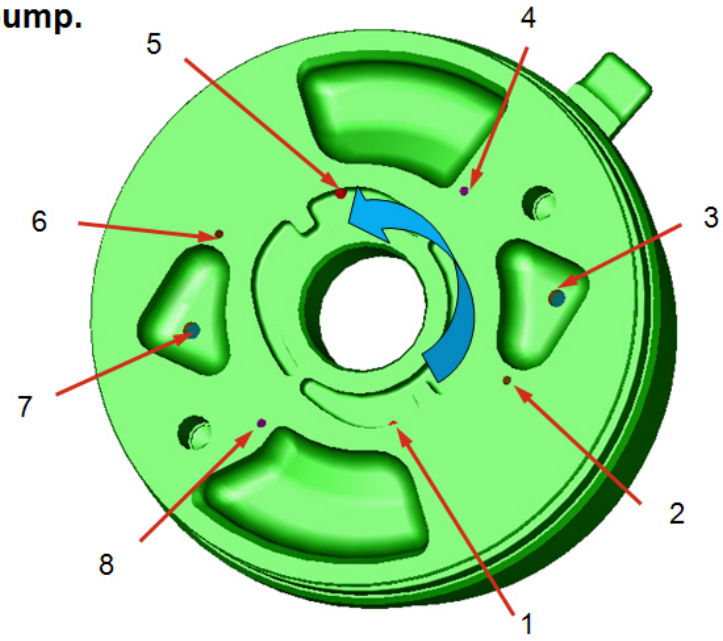
Data is collected for all of the rotational speeds shown in **Table 3**. However, for purpose of this research, only one rotational speed is chosen to be closely monitored, and that is 4000 rpm, conducted at pump operating point at 90°C. Reason for choosing rotational speed of 4000 rpm is that speed is within a range of non-cavitation regime. Also, all CFD simulations are conducted at a speed of 4000 RPM.

The location of the piezoelectric pressure sensors is shown on **Figure 34**. The location of each is marked on test stand in order to keep better track of recorded data and for post processing purposes.

- **Pressure transducers are marked on the pump.**







Pressure transducer	Data to be captured:
1 and 5	Back vane chambers pressure
2 and 6	Maximum pressure at chambers
3 and 7	Pressure at discharge ports
4 and 8	Minimum pressure at chambers

Figure 34: Test setup, pressure transducers locations

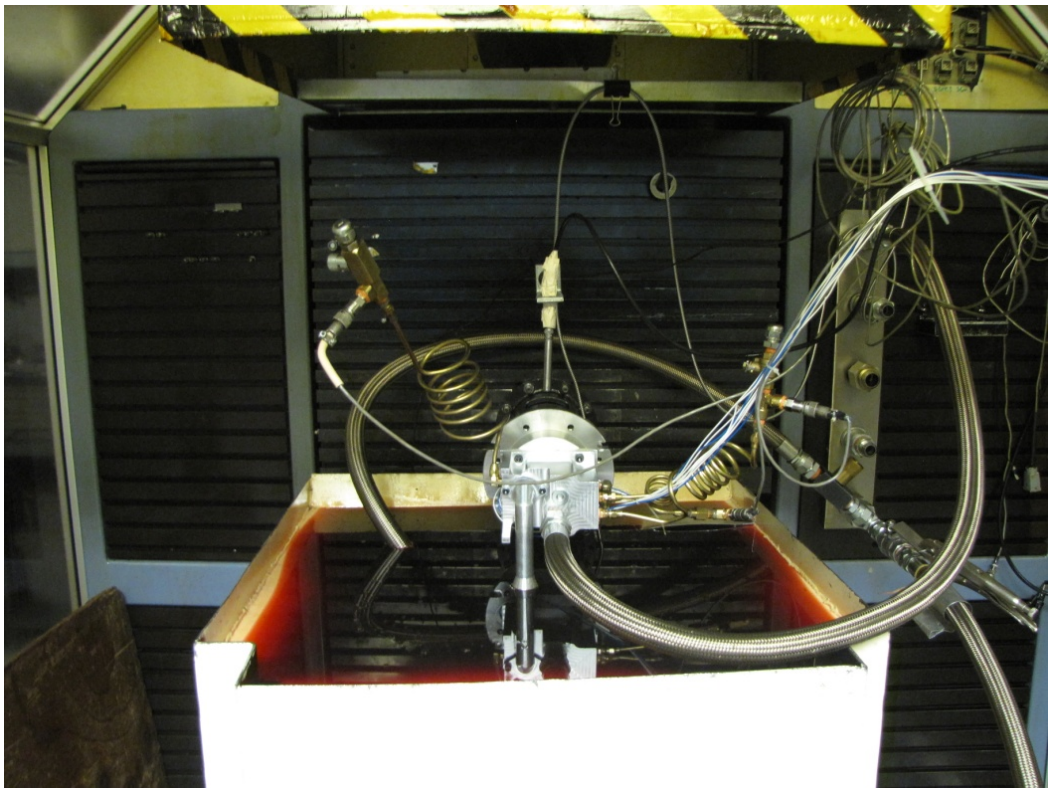


Figure 35: SPL and Oil Pressure Pulsations Measurement, Test setup

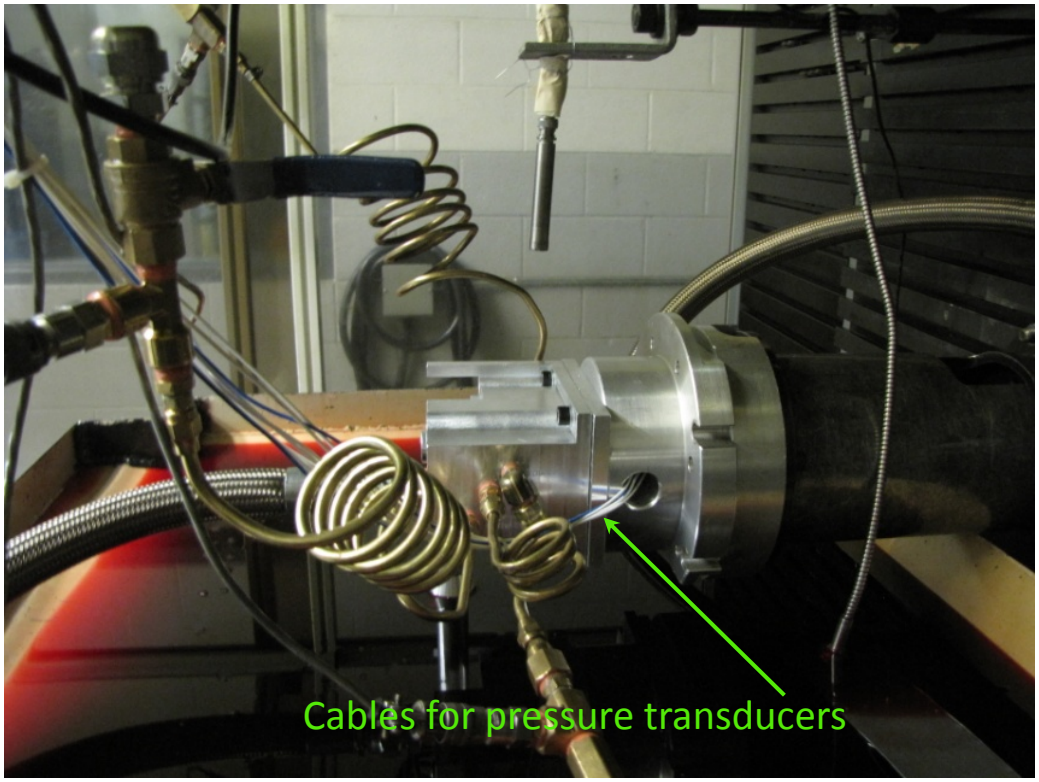


Figure 36: SPL and Oil Pressure Pulsations Measurement, Test setup, detail

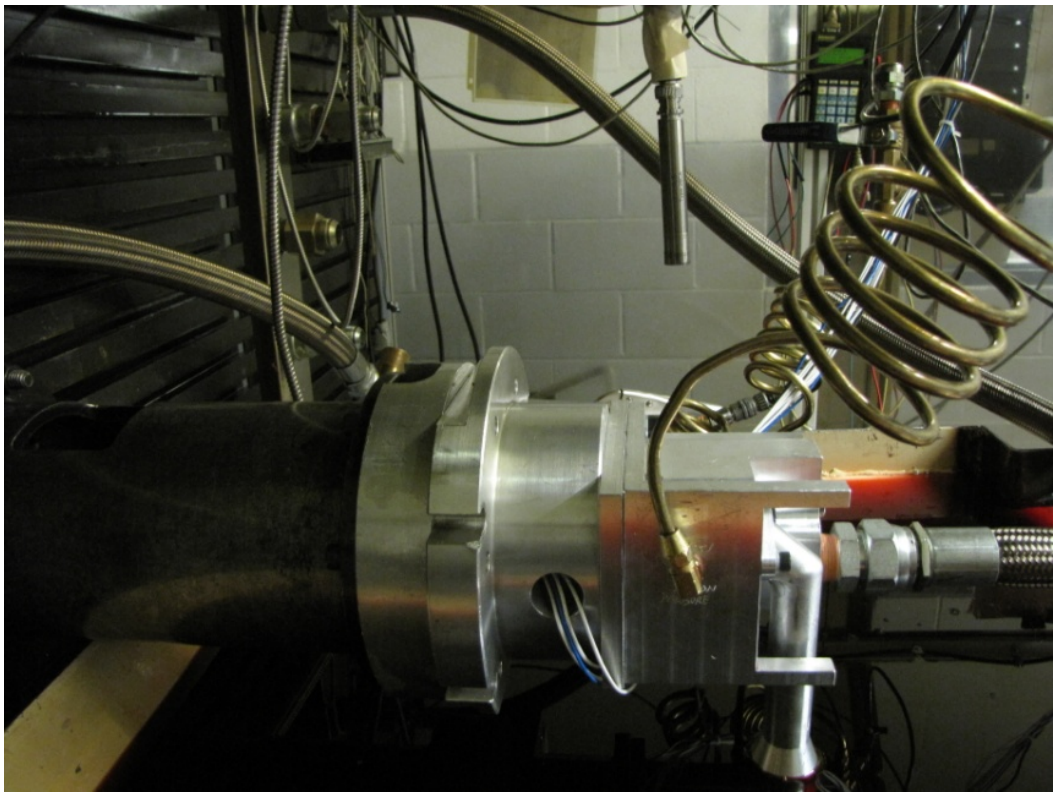


Figure 37: SPL and Oil Pressure Pulsations Measurement, Test setup, detail

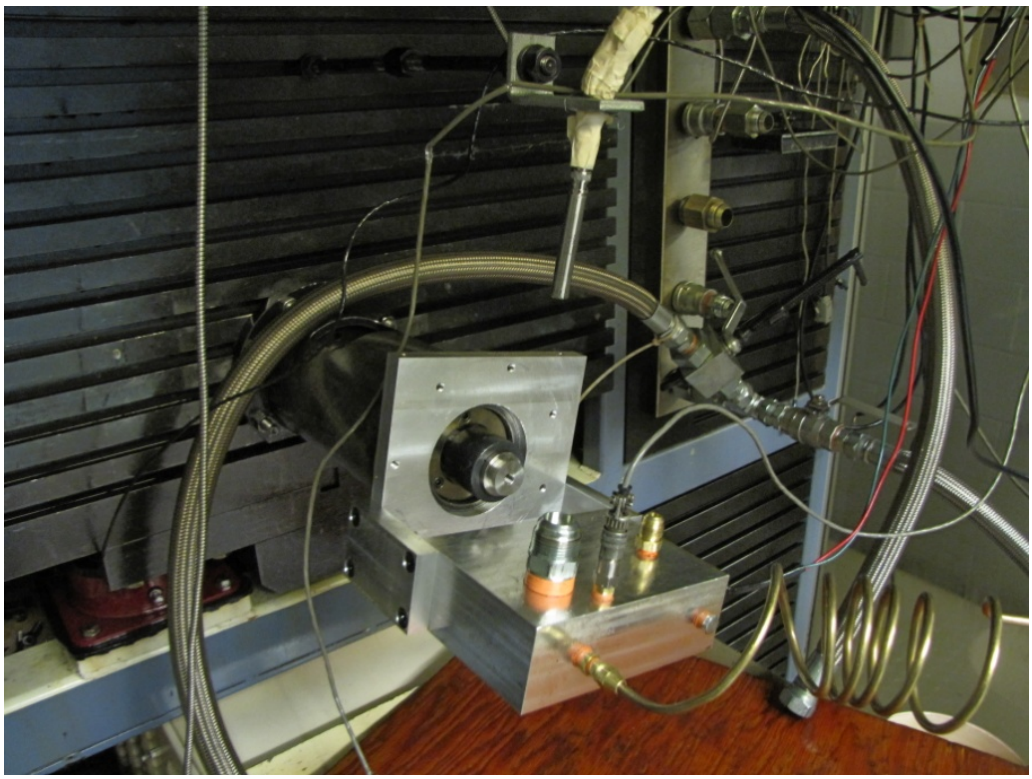


Figure 38: SPL and Oil Pressure Pulsations Measurement, Test setup, background noise measurement

Table with all test results are shown of **Table 3**.

Table 3: Test results

	Speed	Flow	Temp	Pump Torque	Suction	Discharge
	RPM	(lpm)	°C	Nm	(bar)	(bar)
25°C	1000	10.00	23.9	0.96	-0.04	2.02
	2000	20.65	24.1	1.62	-0.10	3.97
	3000	31.26	24.5	2.37	-0.16	6.00
	4000	41.89	26.0	3.13	-0.24	7.99
	4500	47.19	27.0	3.21	-0.27	7.97
	5000	52.36	27.0	3.87	-0.32	9.90
	5500	55.19	27.1	3.97	-0.32	9.99
	6000	54.07	26.6	3.83	-0.32	9.92
90°C	500	1.91	90.00	0.55	0.00	1.06
	1000	7.43		0.91	0.00	2.07
	2000	18.49	87.79	1.60	-0.02	4.07
	3000	28.61	88.28	2.20	-0.05	5.96
	4000	39.13	89.41	2.89	-0.09	7.99
	4500	44.82	88.25	2.88	-0.12	8.03
	5000	49.65	88.72	3.66	-0.15	9.97
	5500	55.32	89.76	3.66	-0.18	10.11
	6000	60.22	89.20	3.69	-0.22	10.06

In order to determine relationship between flow, pressure, torque and speed, another test is conducted, at a constant speed of 4000 rpm. This test is conducted with variable restriction at outlet, adjusted by restriction valve. Restriction varies the pressure from 0 bar, up to 30 bar. This test is conducted to determine pump performance with higher restriction at outlet and to establish relationship between pressure, flow and torque for the balanced pump. Results from this test are depicted on **Figure 39**.

The second test is with constant restriction at outlet which is adjusted with a valve to be 8 bar at 4000 rpm. Speed is increased up to 6500 rpm and data for flow, pressure and torque is collected and plotted. Results from the second sweep test are depicted on **Figure 40**. From this test it can be seen that cavitation starts from about 5000 RPM. At about 5500 RPM, the flow experiences a sudden drop, which

is an indication for severe cavitation. The CFD code which is used in this research has the capability to determine pressure and flow during cavitation regime. However, this falls outside of the scope of this research and can be a subject for a separate study.

Flow characteristics vs. RPM for the balanced pump are in linear relationship (see flow curve on **Figure 40**), up to the cavitation regime. Thus, validation with CFD is conducted only at 4000 RPM and other rotational speeds are not computationally investigated.

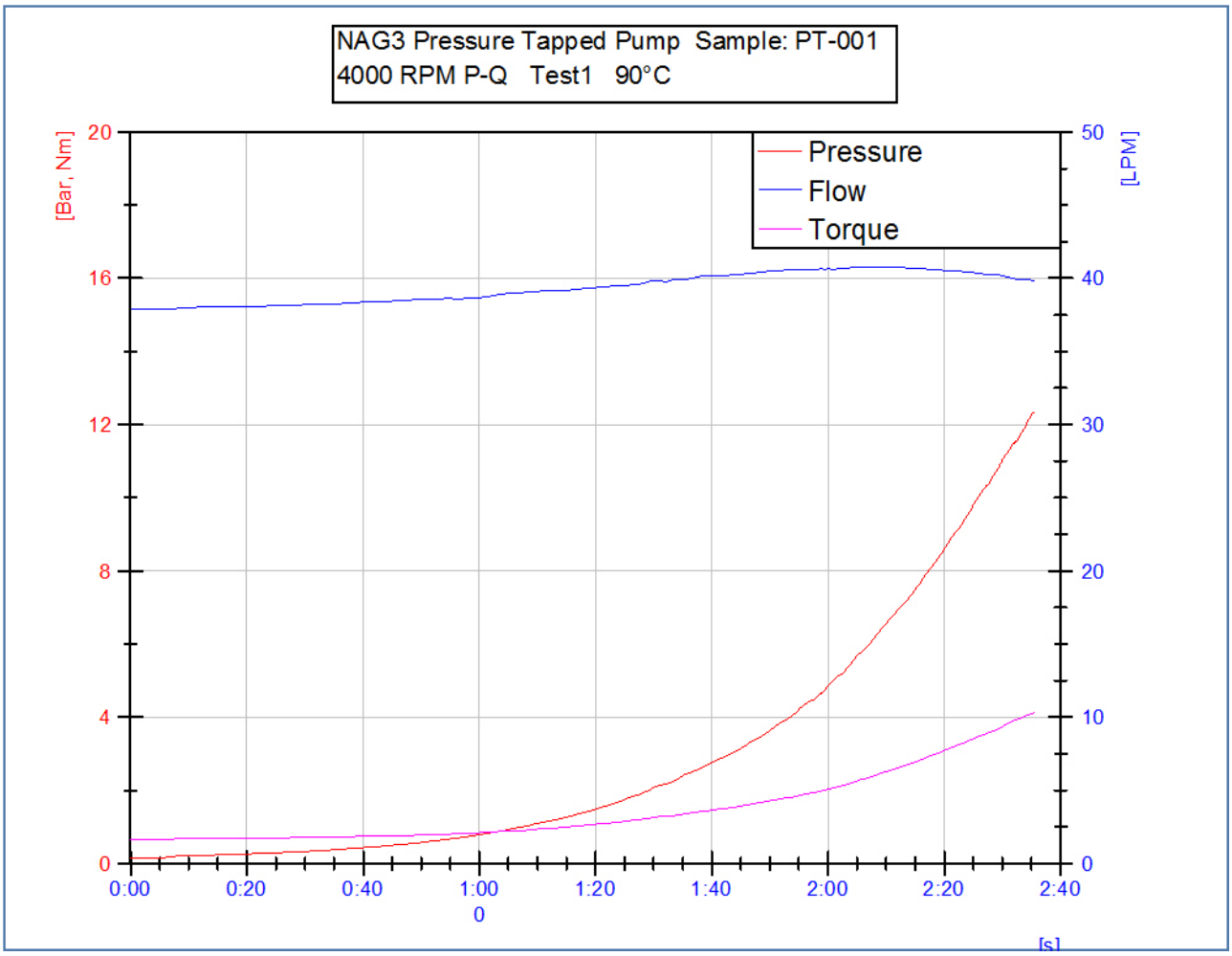


Figure 39: Test results for flow, pressure and torque

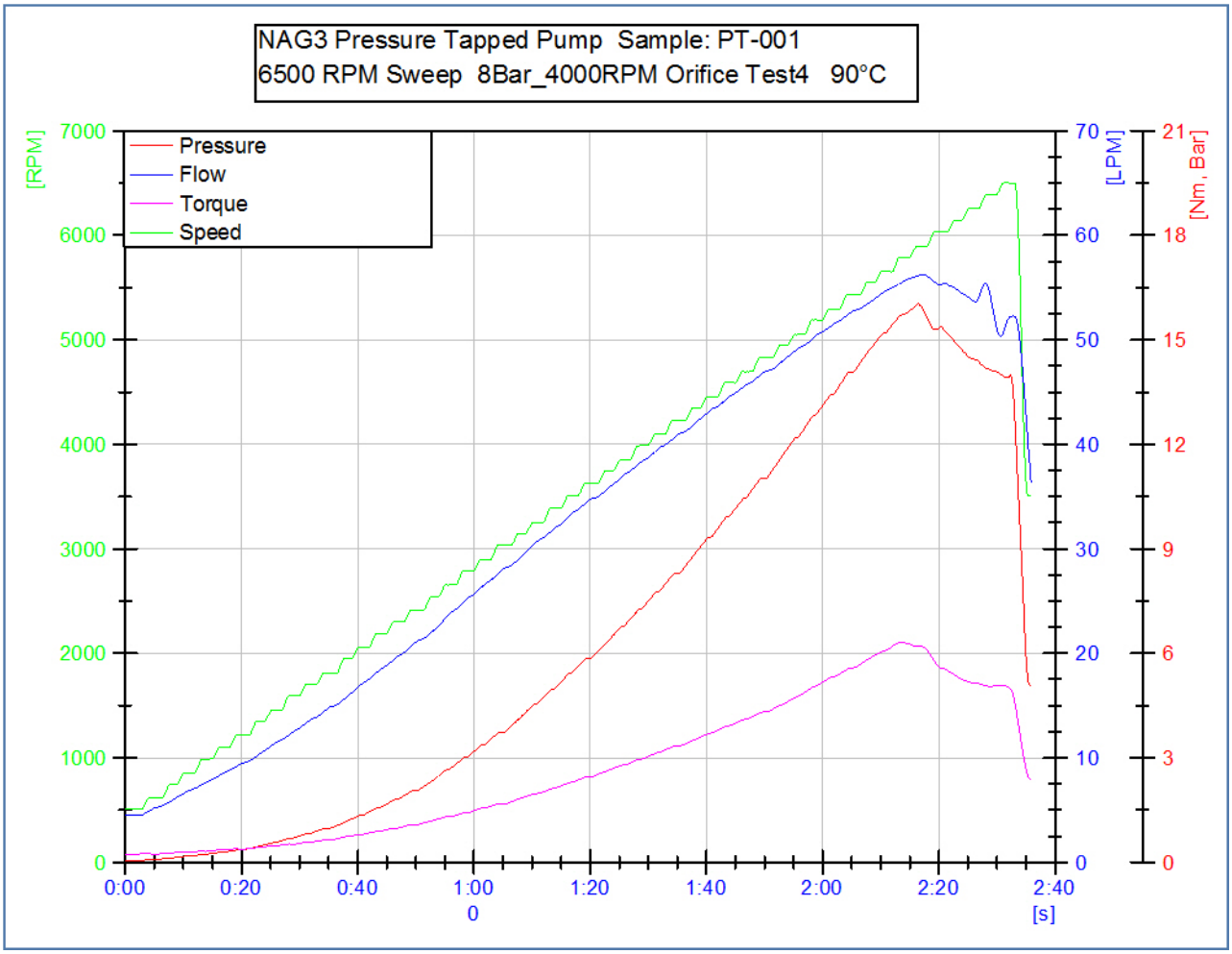


Figure 40: Test results for flow, pressure, torque and speed

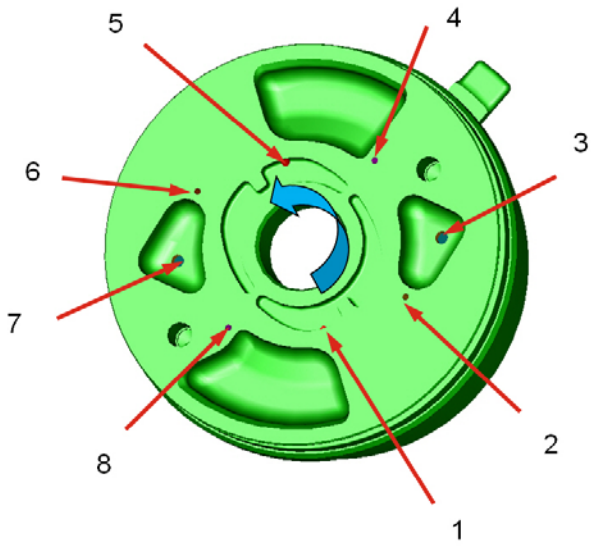
Chapter 5 - Results from baseline CFD simulation of balanced pump

CFD model was run on a dual Xeon processor, on Windows XP operating system. The segregated implicit solver setting was used, with standard pressure discretization. The advection scheme was set to 1st order upwind for the momentum discretization. The use of a 2nd order scheme was tested and found to give the same prediction as the first order scheme. Details for 1st and 2nd order scheme are given in **Appendix E**.

5.1. Comparison between simulation prediction and actual testing results

5.1.1. Pressure pulsation comparison between simulation prediction and actual testing results

In order to capture pressure variations during operation of the pump, its housing is instrumented with dynamic pressure transducers. Pressure transducers are installed in pairs and they are installed in axi-symmetrical locations. All pressure transducers are marked on the pump for easier recognition. The purpose of the pressure transducers is to capture pressure variations where the pump is experiencing the lowest and highest pressures as well as the pressure at the outlet port. Pressure transducer locations on the balanced pump are depicted on **Figure 41** . The highest pressure at the pump is before the outlet port (pressure transducers 2 and 6) and the lowest pressure is when the pumping chamber is close to the intake port (pressure transducers 4 and 8). Pressure at the discharge port is measured in order to establish pressure magnitude in the pump channels (pressure transducers 3 and 7). Pressure transducers 1 and 5 are installed in order to capture pressure variations in the back vane chambers. That particular study is not part of this research project and those results will not be presented in this research project.



Pressure transducer	Data to be captured:
1 and 5	Back vane chambers pressure
2 and 6	Maximum pressure at chambers
3 and 7	Pressure at discharge ports
4 and 8	Minimum pressure at chambers

Figure 41: Pressure transducers locations on the pump

On **Figure 42** and **Figure 43**, static pressure contours are shown for balanced pump.

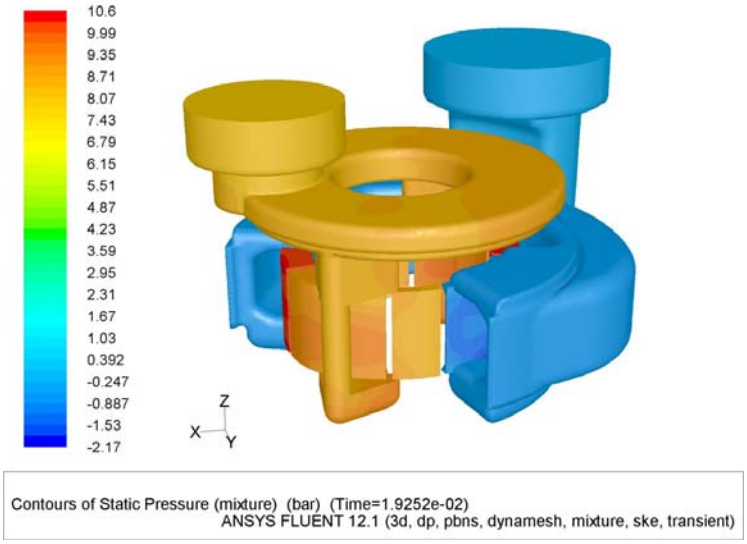


Figure 42: Contours of static pressure of balanced pump, isometric view

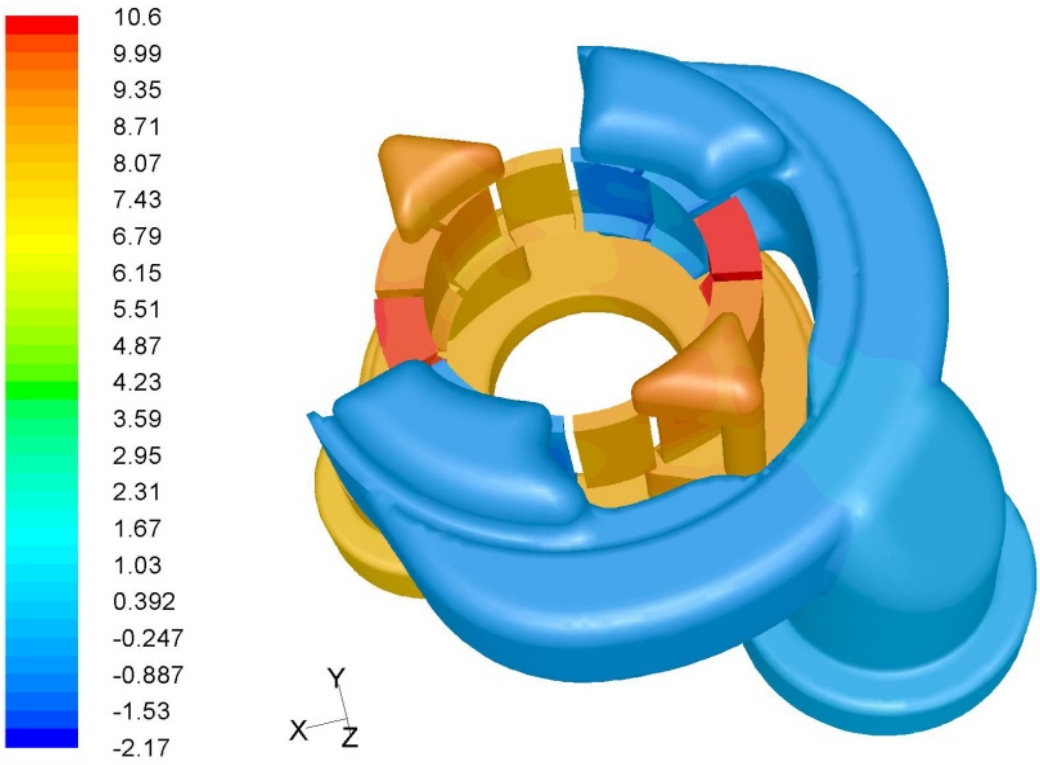


Figure 43: Contours of static pressure of balanced pump, bottom view

Results from the pressure transducer located in discharge port and results from the CFD simulation are shown on **Figure 44**. The blue curve represents the experimental data and the red curve is the prediction from the CFD simulation. As seen from the figure, the curves are in very good agreement especially regarding the maximum and minimum pressure at the pressure pulsations.

Every wiggle on the curve is when the chamber discharges oil into the outlet port. The chamber volume is formed between rotor, cam and 2 consecutive vanes. Small wiggles between higher and lower pressure on the curve obtained from the CFD simulation is created from sudden connection between chambers to discharge port and numerical discretization.

Small wiggles between higher and lower pressure on curve obtained from testing (from 8.35 bar to 8.9 bar) could be created by vanes or vibrations of the pump.

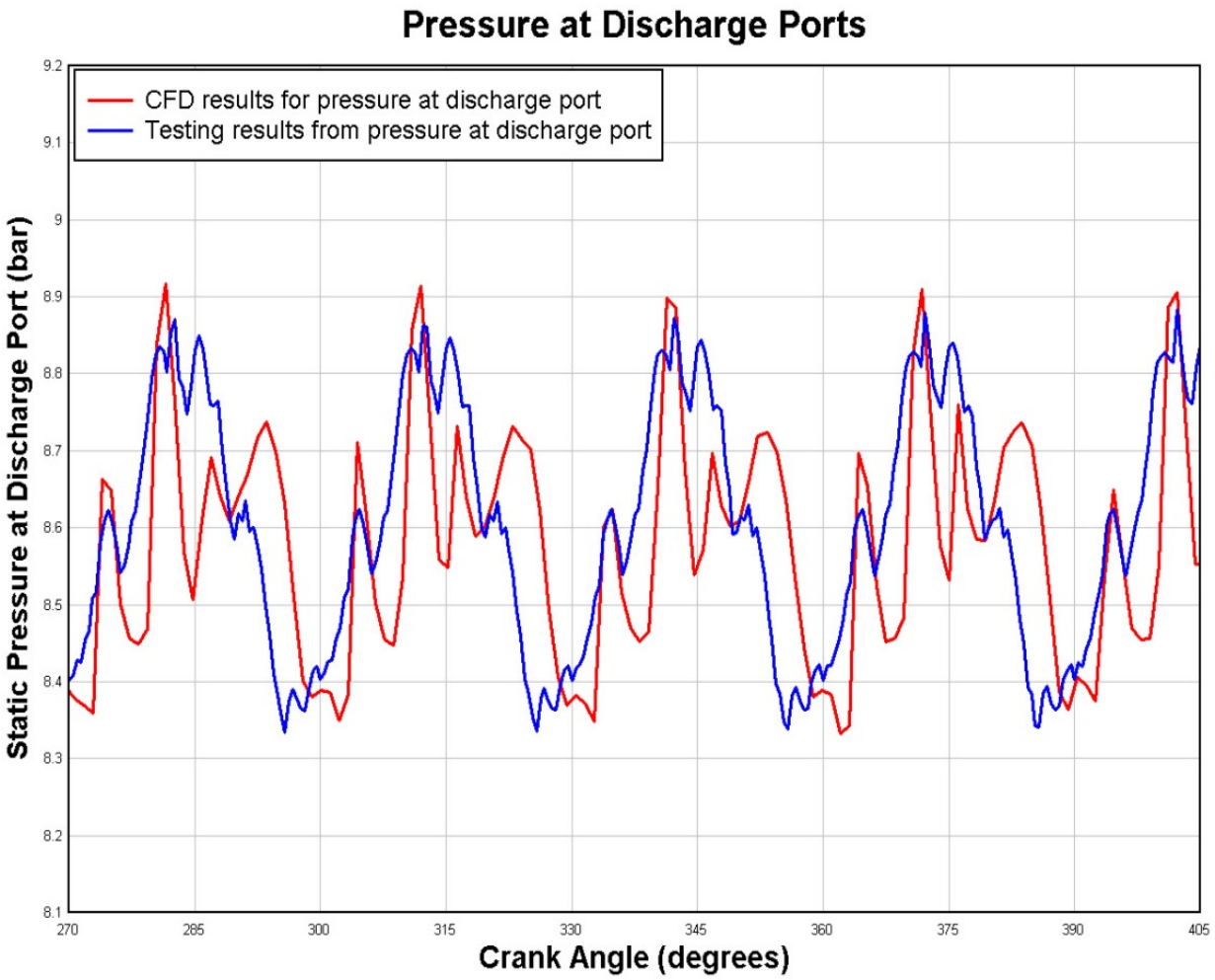


Figure 44: Results from pressure transducer 3 and CFD results at discharge ports

The most important comparison for this research project is the accuracy of the prediction of the pressure at the high pressure location. This is important because reducing pressure pulsation in the pump will reduce noise, vibration and will increase operational life of the balanced pump. Experimental and predicted results from the pressure transducer located before the discharge port are shown on **Figure 45**.

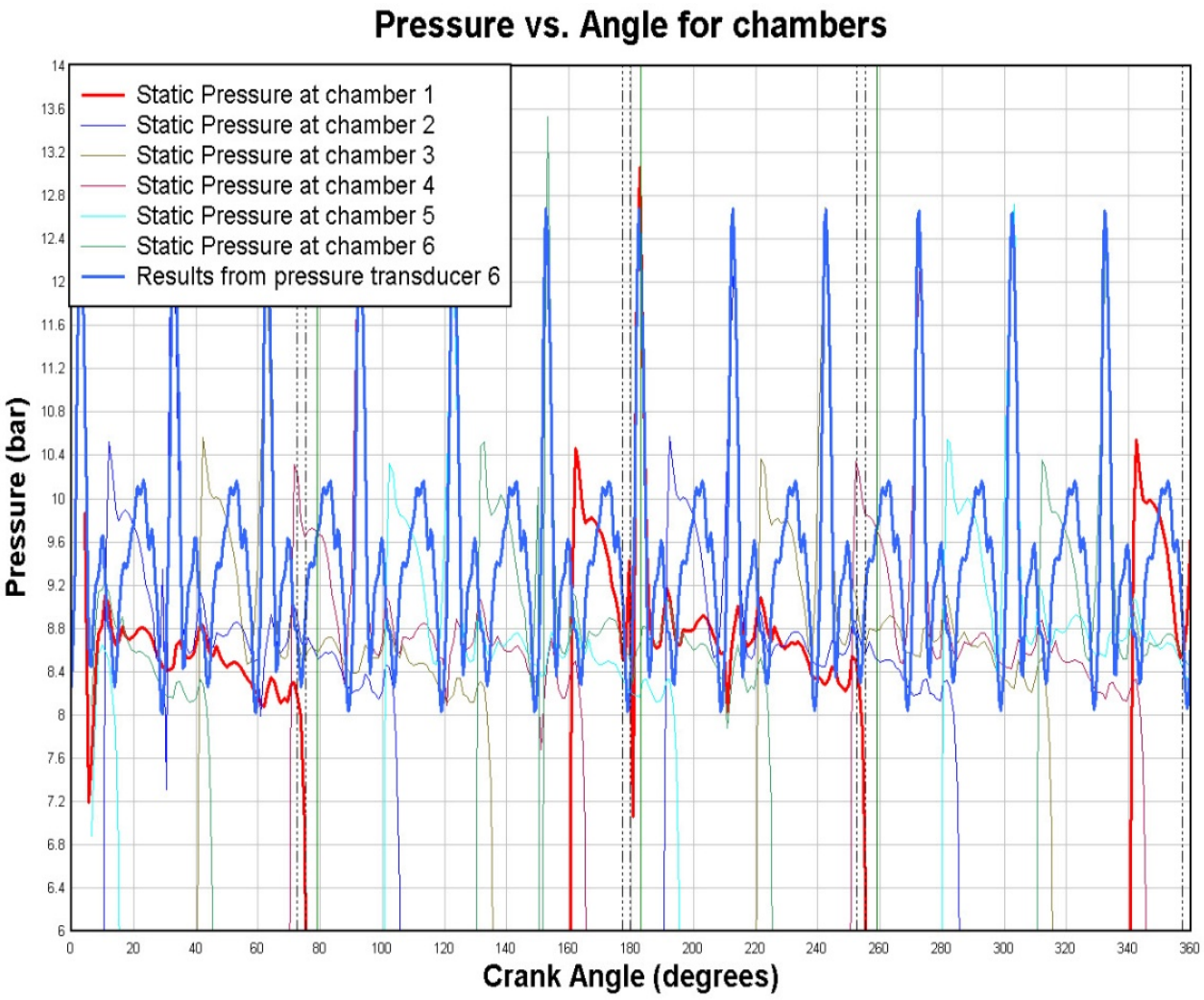


Figure 45: Results from pressure transducer and CFD results at pumping chambers

Figure 45 show 6 curves for calculated pressure in the pump chambers and 1 curve of the experimental data. All 12 chambers are not shown on attached figures, because they are identical due to symmetry. The CFD simulation predictions show higher peaks than the experiments at some points. However, those higher CFD values are expected since the CFD simulation is conducted with rigid bodies, compared to real pump, which has some deformation and different rate of leakages arising from variations in the clearances. **Figure 46** shows an enlarged view of the experimental data and the CFD predictions. Maximum pressure measured during the testing is about 12.5 bar. On **Figure 46,** dashed lines represent small angle when vane passes over the pressure transducer. Therefore, pressure experiences small sudden drop of the pressure. This effect cannot be seen on red curve, which

represents pressure obtained from CFD simulation. This is shown on **Figure 46**, from 177 to 180 degrees.

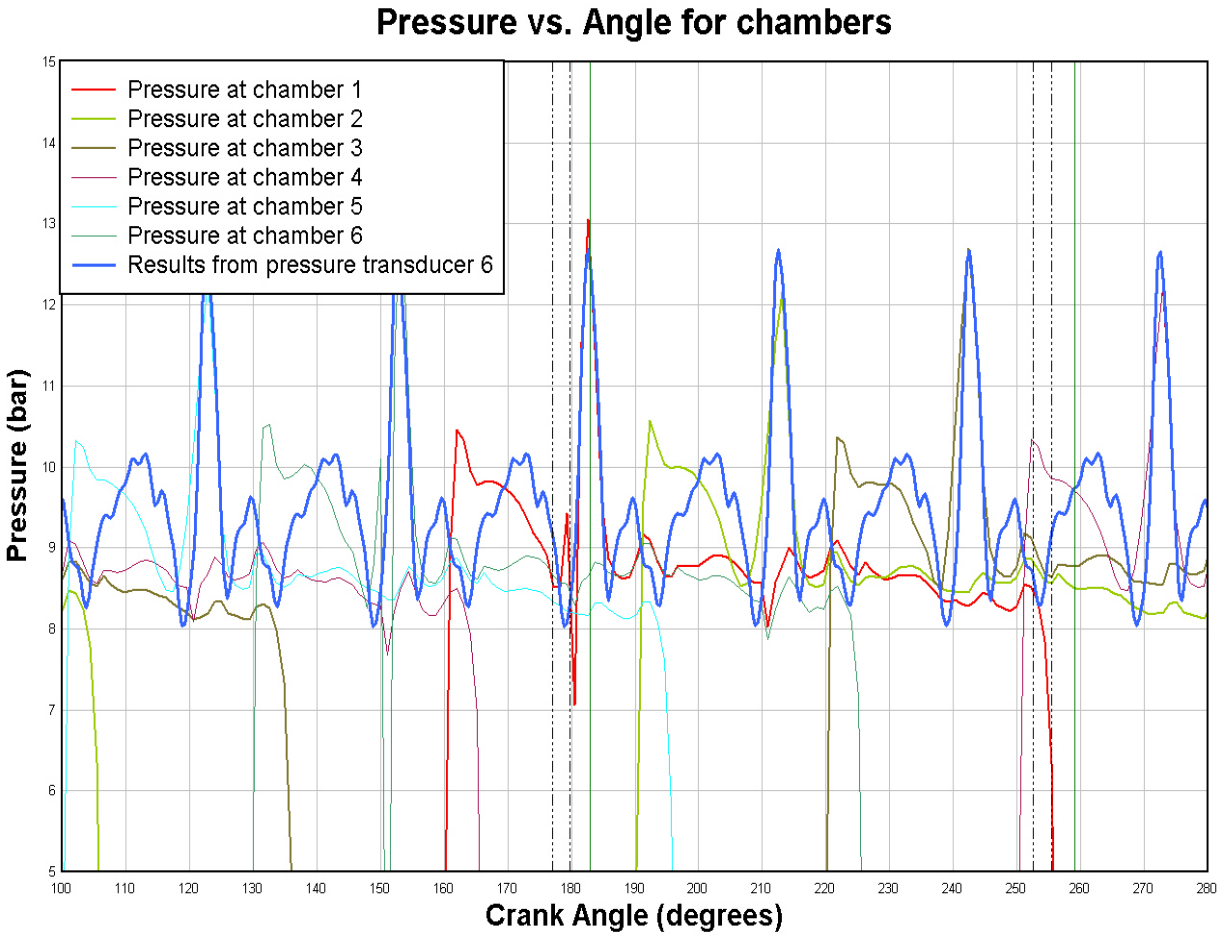


Figure 46: Results from pressure transducer and CFD results at pumping chambers

For clarity, on **Figure 47**, pressure of only one chamber and experimental results from pressure transducers are plotted. Blue curve on **Figure 47** represents data from pressure transducer; green curve represents pressure from single pumping chamber obtained from CFD simulation.

Results are also compared between computational and experimental results at the same location where pressure transducer is installed on the housing. This comparison is depicted on **Figure 48**, where red curve depicts computational data and blue curve represents experimental data.

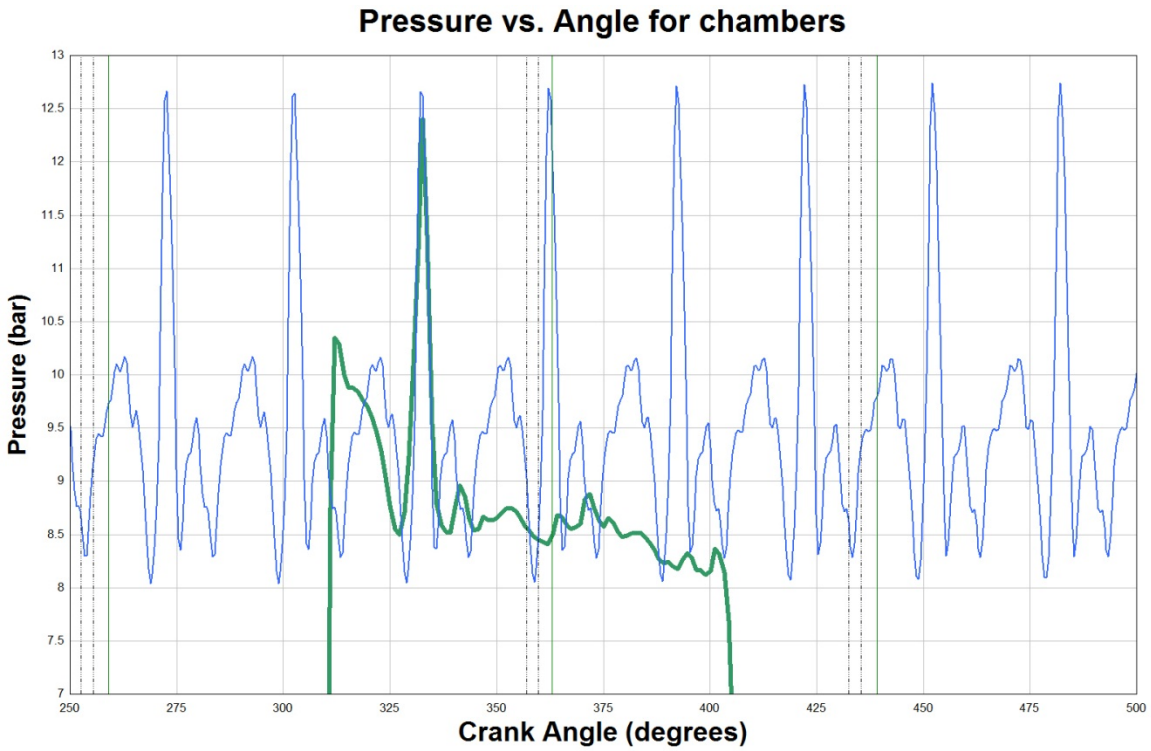


Figure 47: Results from pressure transducer and CFD results at one pumping chamber

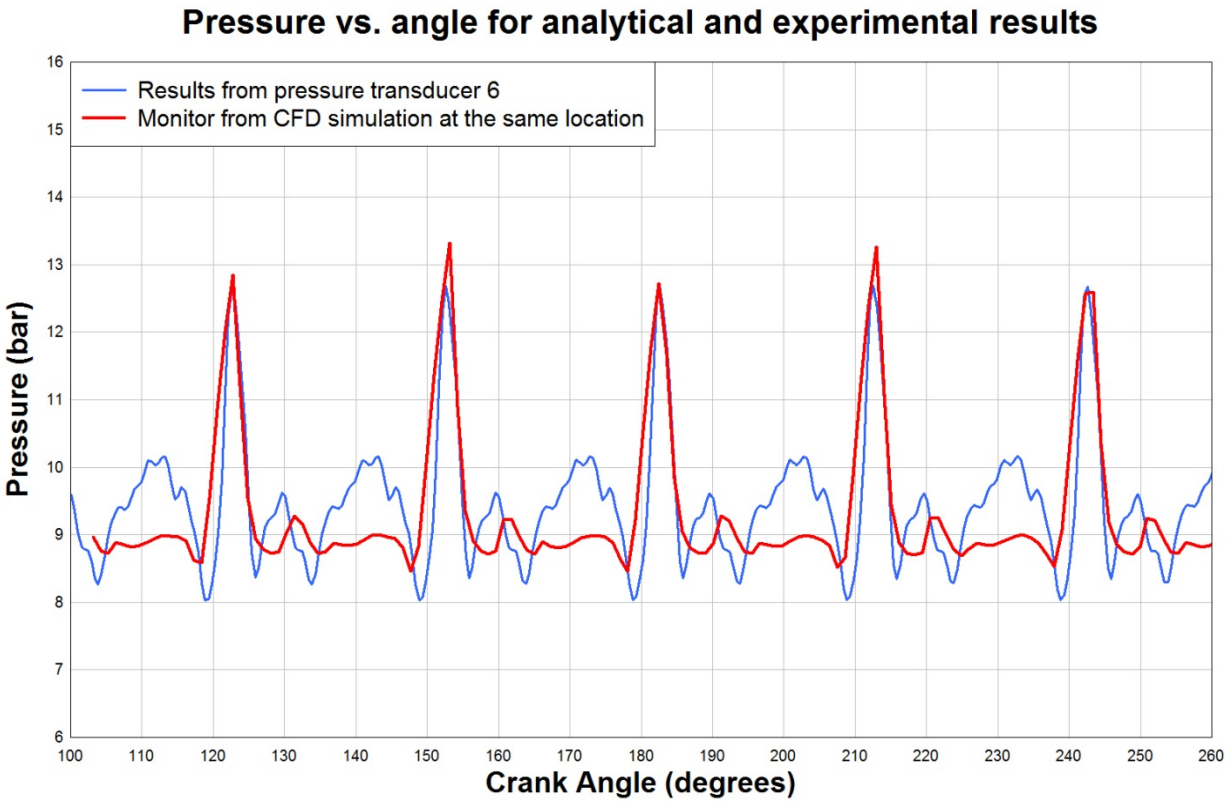


Figure 48: Results from pressure transducer and monitor from CFD analysis at the same location

The curve obtained by experimental data, shown on **Figure 48**, has two smaller humps between maximum peaks of pressure. There are several possible reasons for these two humps which occur during the testing. The two humps could be a result of the tolerances between the vanes and rotor slots which allow for the vanes not to be perfectly rigid in their slots. Another reason could be concentricity of drive shaft, shaft extension and sprocket of the balanced pump, which could have effect on vibration. Also, there are deformations of pump during pump operation which modifies leakage paths which are not captured by CFD simulation. All of this factors combined with manufacturing tolerances, could add unaccounted humps on pressure pulsations.

Similar trends are seen for the minimum pressure region which is measured before the inlet port as shown in **Figure 49**, which depicts pressure measurement from pressure transducer before inlet and results from CFD simulation. Minimum pressure measured has to be taken as reference only, because of the numerical procedure and the way how AC transducers transmit the signal. In order to illustrate the repeatability for one chamber, the first chamber is represented with red curve. Enlarged view of testing data and simulation from same transducer is shown on **Figure 50**. Comparing results from CFD simulation and experimental data, it must be noted that CFD simulation does not capture minimum pressure peak from all chambers, due to the time step used. Dashed lines on **Figure 49** represent angle when vane passes over pressure transducer. It has to be kept in mind that pressure transducer is located at fixed location and readings are from same location, compared to CFD data, which show pressure measured at chambers as they rotate around and is impossible to measure physically.

Curves for pressure pulsations from the CFD simulation shown on **Figure 49** and **Figure 50** indicate low pressure before intake port, which could be a sign for cavitation. Cavitation can be quantified by monitoring vapor volume in the chamber. Vapor volume for analyzed balanced pump at 4000 RPM is negligible, which means there is no significant cavitation for a balanced pump at these operating

conditions. The performance drop due to cavitation starts at 5000 RPM, which is confirmed with testing and depicted on **Figure 40**.

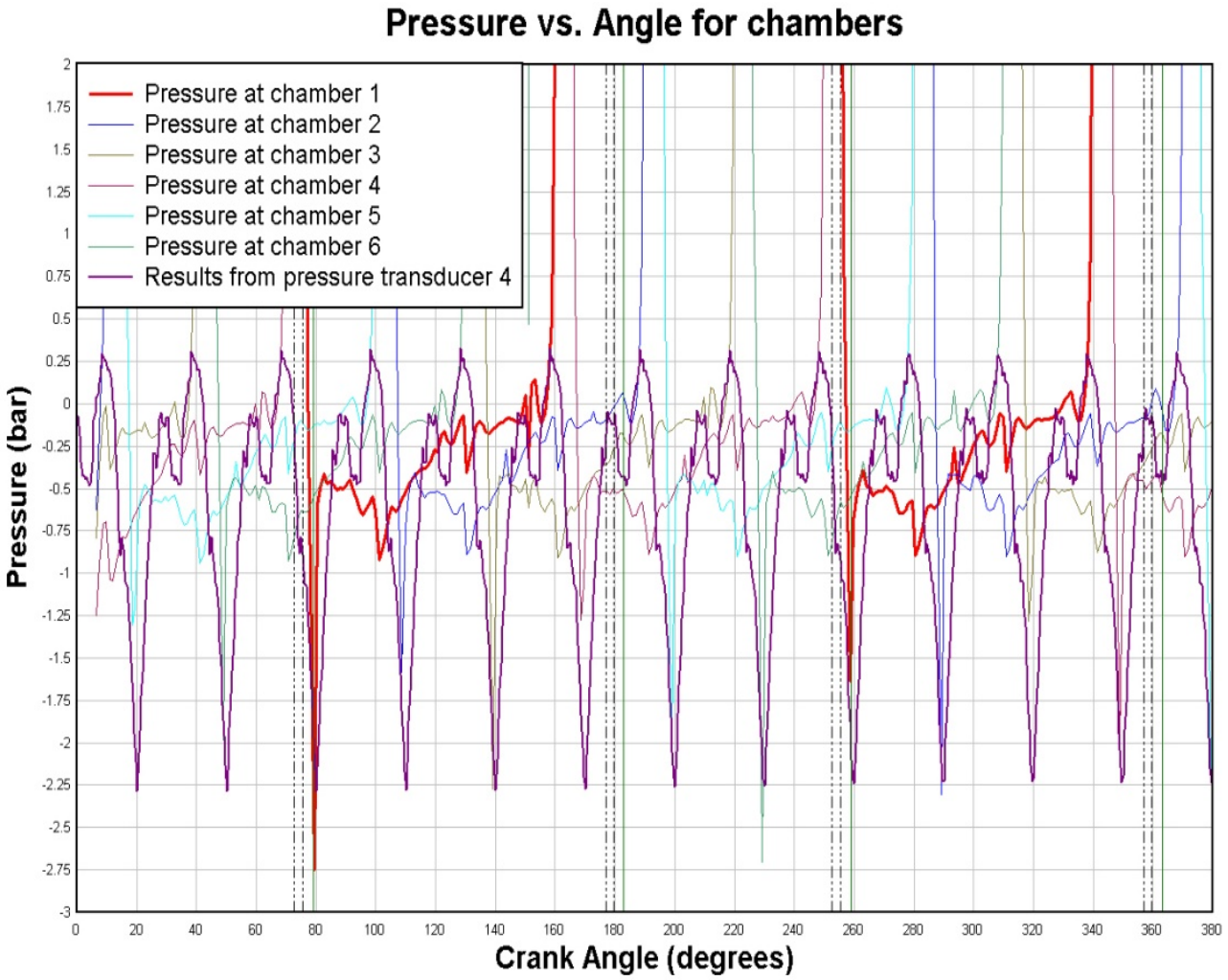


Figure 49: Pressure from CFD results at pumping chambers and signal of pressure transducer positioned just before the inlet port

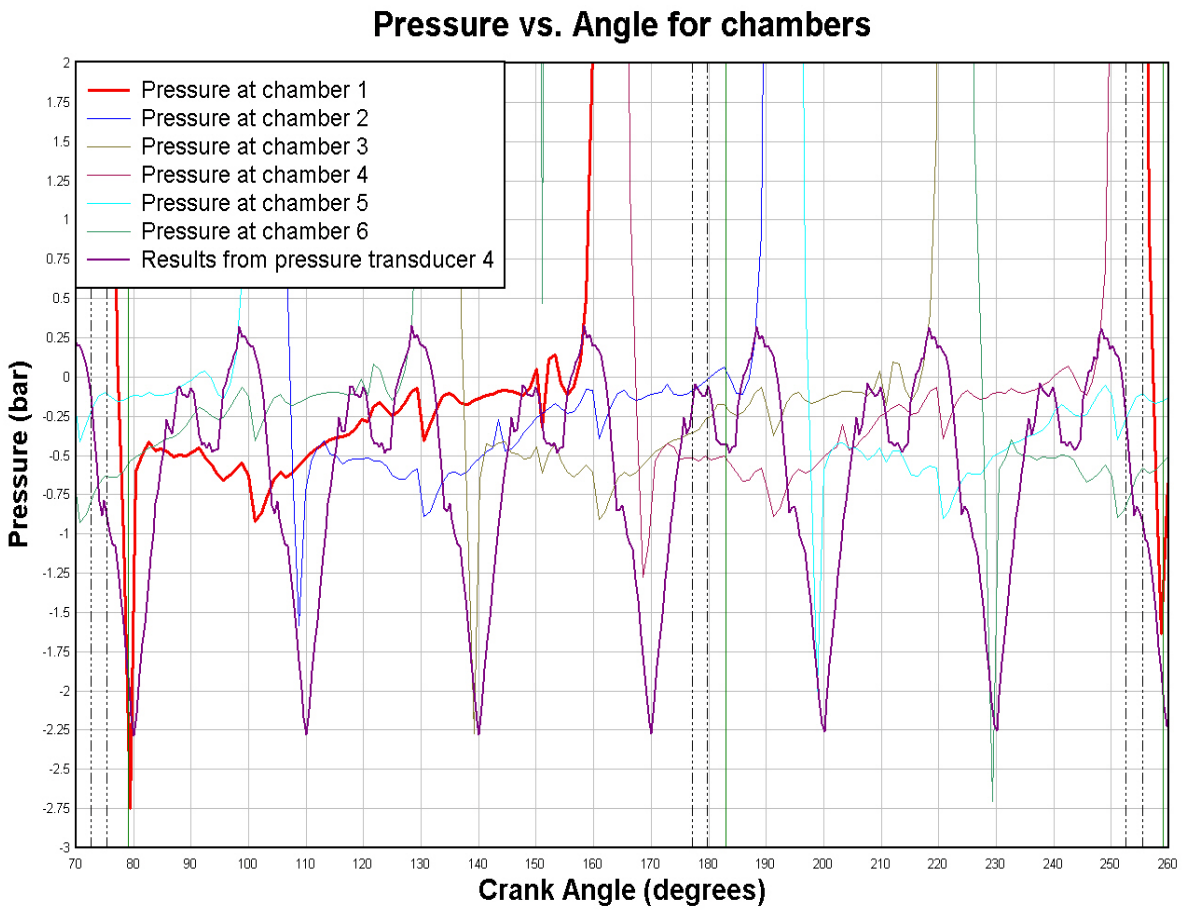


Figure 50: Pressure from CFD results at pumping chambers and signal of pressure transducer positioned just before the inlet port

For clarity, on **Figure 51**, pressure of only one chamber and experimental results from pressure transducer are plotted. Purple curve on **Figure 51** represents data from pressure transducer; green curve represents pressure from single pumping chamber obtained from CFD simulation. Reference lines when chamber is opened to intake port are shown at X=140 and X=220 degrees. It must be noted that X-coordinate (crank angle) for minimum pressure from CFD simulation and data for pressure transducers is not identical, due to the time step used in CFD simulation. This is because pressure curve is created by connection of points obtained from CFD simulation. Due to finite time step used in simulation, point on X axis obtained from simulation does not match peak of experimental curve.

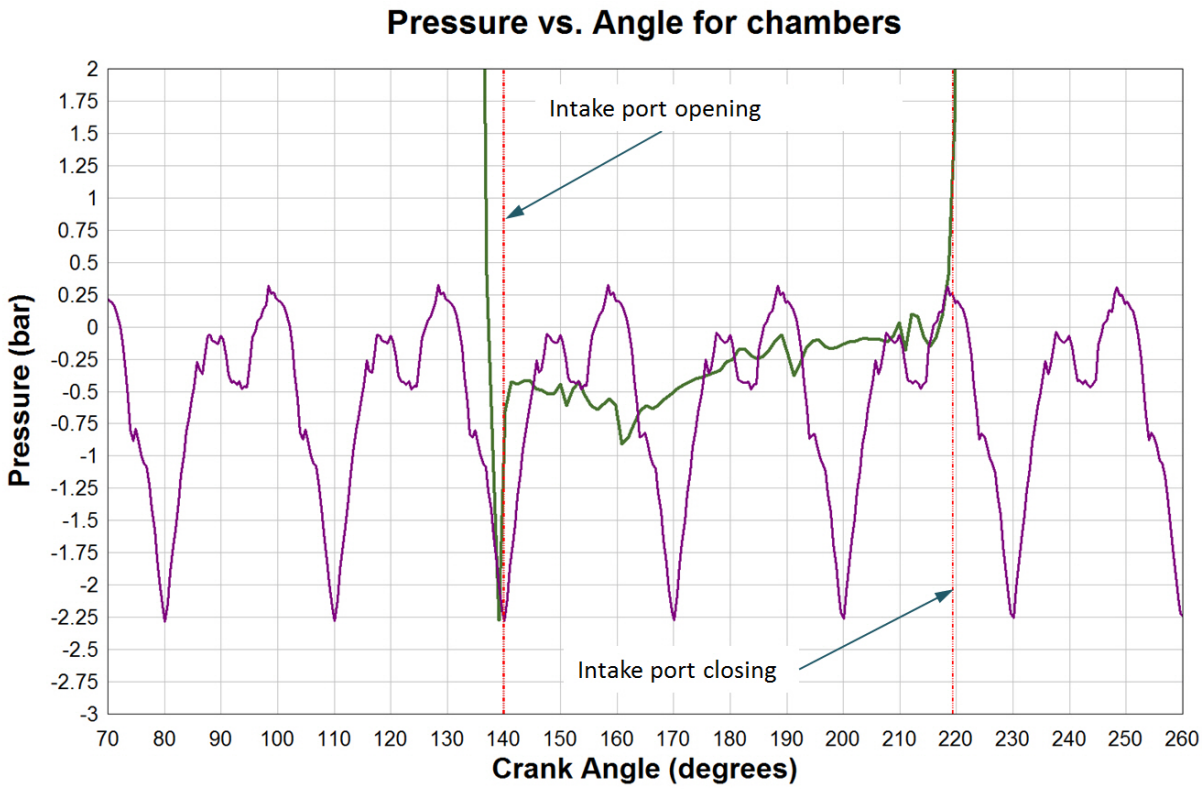


Figure 51: Results from pressure transducer and CFD results at one pumping chamber at inlet port

5.1.2. Flow comparison between simulation prediction and actual experimental results

A comparison between the predicted flow and that measured experimentally is shown in **Figure 52**. As seen in the figure, the average predicted flow is 39.75 liter per minute which compares well with the experimentally measured value of 39.15 liters per minutes yielding an error of roughly 1.5%. It is of interest to note that one dimensional model (see **Figure 53**) and two dimensional models (AmeSim model, see **Figure 54**) predicted values of 43 and 42 liters per minute, respectively. As expected, as the complexity of the models is increased, the accuracy of the prediction improved.

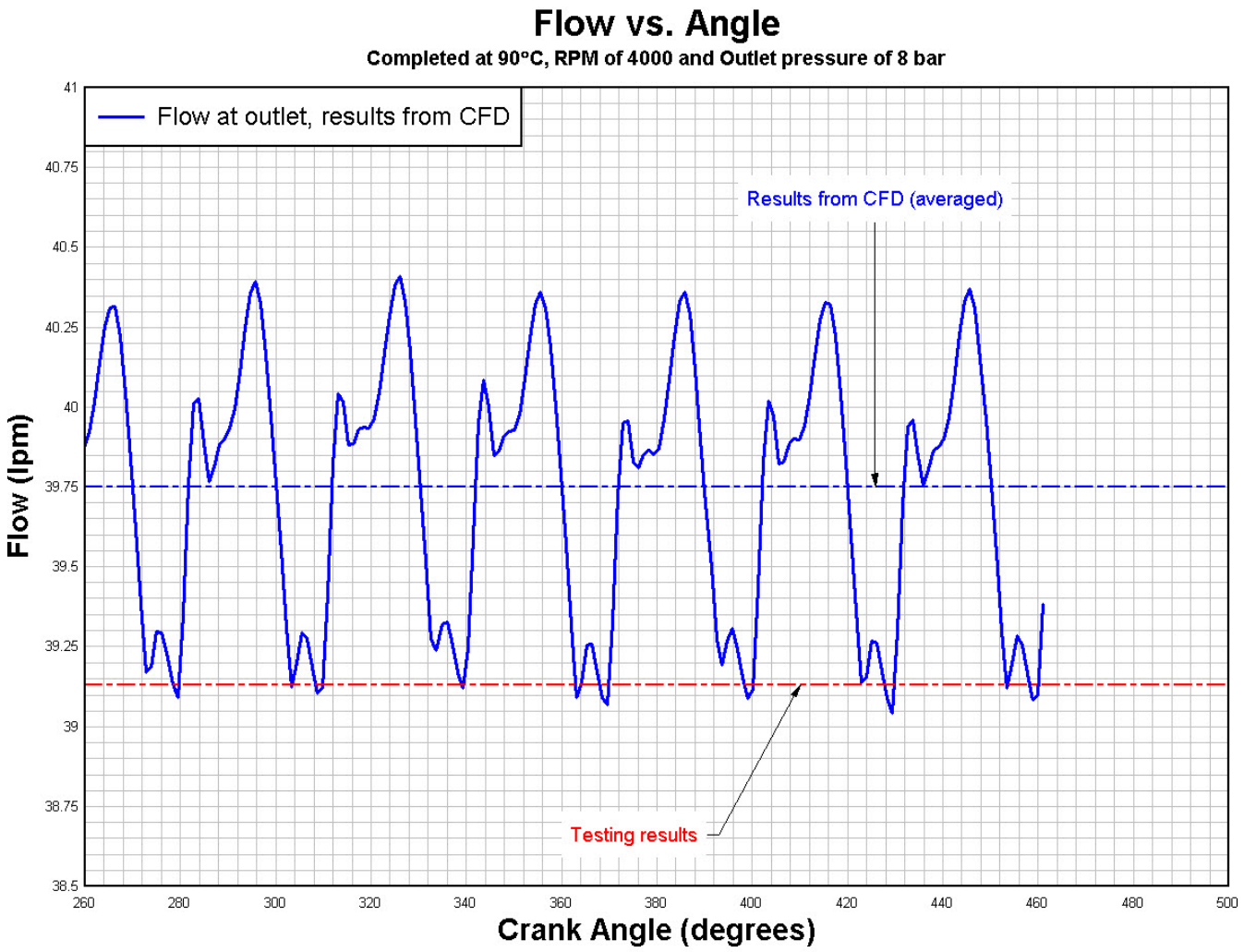


Figure 52: Comparison for flow from test data and CFD analysis

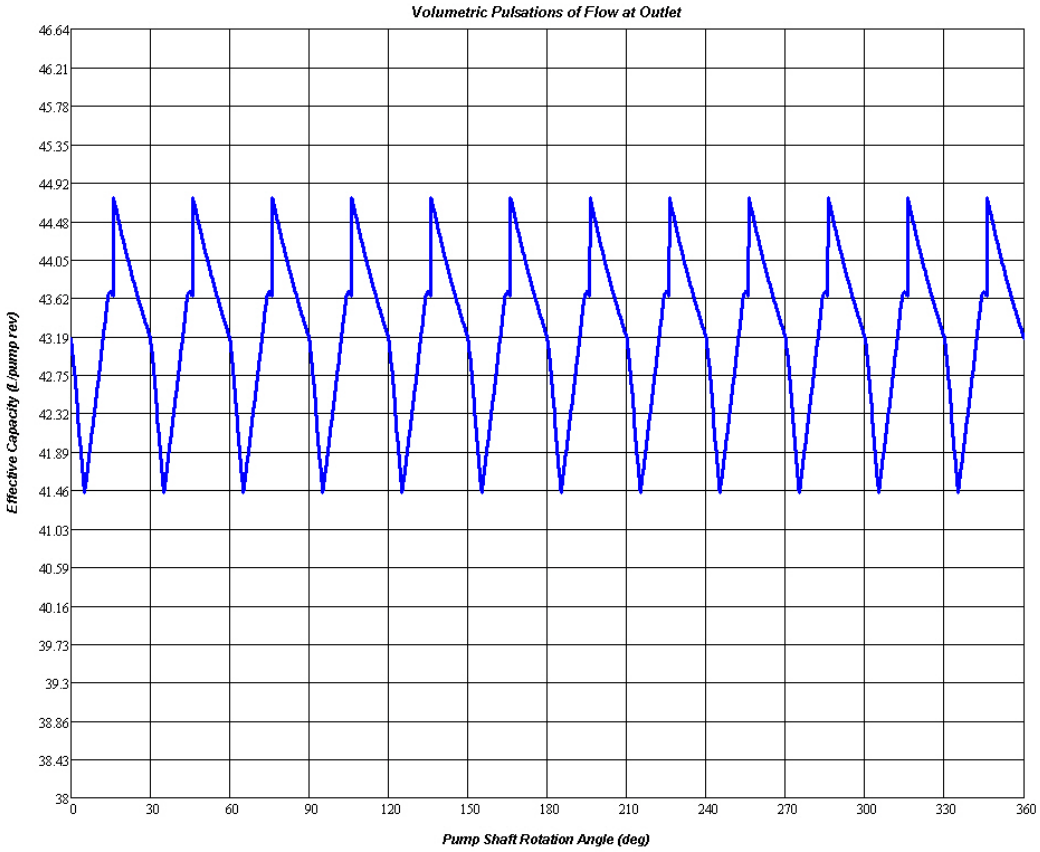


Figure 53: Flow performance of the pump, results from one-dimensional model

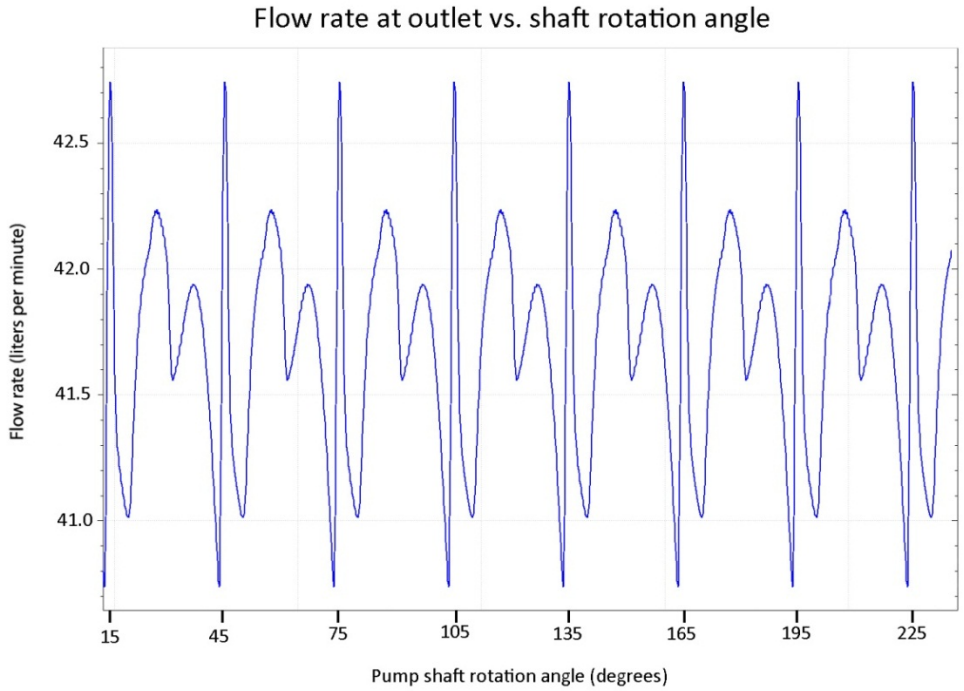


Figure 54: Flow performance from AmeSim model for balanced pump

Chapter 6 - Optimization of ports and groove implementation for balanced pump

6.1. Optimization of intake and discharge ports

Initial design of ports for balanced pump is carried using industrial experience guidelines, which specifies angle between ports with respect to number of vanes and their thickness.

The objective of this research is to develop an improved pump design that will result in reduced pressure pulsations. Different approaches will be used to attain this goal. One approach is to move the intake and outlet port closer to each other. This may decrease pressure peaks before outlet port is reached and also increase minimum pressure reading before the chamber is opened by intake port. For this reason, the angle between the ports is reduced by about 7 degrees (from 80 degrees to 73 degrees measured on sealing areas). The value for the angle change was determined by inspecting the pressure pulsations shown on **Figure 46** and **Figure 50**. Looking at **Figure 46**, it is evident that the measured pressure rises from 8 bar to 12.5 bar with crank rotation of about 3 degrees. Similarly, at inlet side, pressure peaks shown on **Figure 50** decreases rapidly within about 4 degrees. Thus, the total angle required for ports to be moved closer to each other is chosen to be about 7 degrees. In **Figure 55**, both designs of ports are depicted, original and revised design. Discharge ports of initial balanced pump design are depicted with red color. Intake ports of initial balanced pump design are depicted with light green color. Discharge ports of revised balanced pump design are depicted with dark green color. Intake ports of revised balanced pump design are depicted with dark green color.

After implementing the revised porting of the pump, CFD simulations were conducted on a balanced pump with modified porting. Results for pressure monitors in pump chambers are shown on **Figure 56**.

Pressure measured at pumping chambers varies from -1 bar before inlet port to 12.5 bar before discharge port.

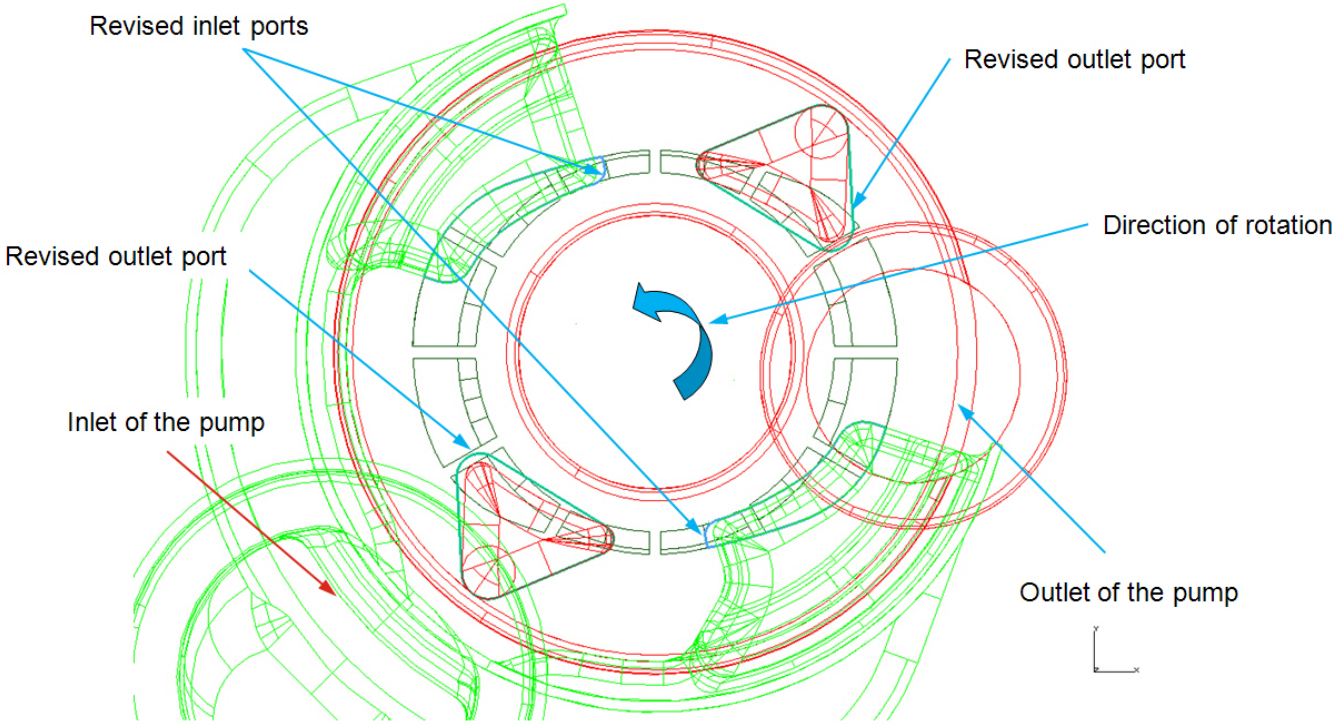


Figure 55: Ports layout, tested and modified

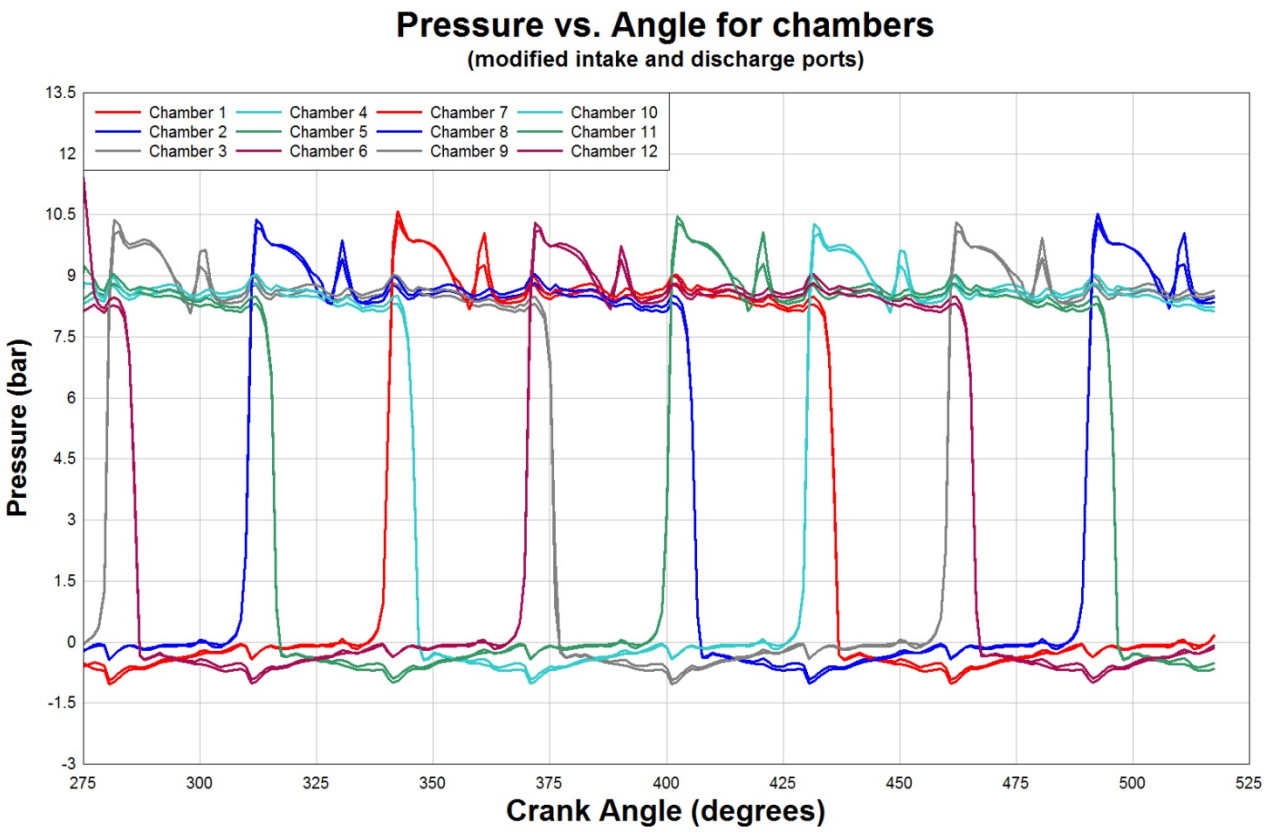


Figure 56: Results from pressure monitors in pumping chambers of pump with modified ports

The predicted pressure monitors for the initial and revised design are plotted on **Figure 57**. It is clearly seen that the pressure pulsation from the initial design is radically reduced from about 12.5 bar (initial design) to 10.5 bar (seen at design with modified porting). In **Figure 57**, curves for pressure pulsations from initial design are shown in red color and curves for pressure pulsations from revised design are shown in green color. It must be noted that maximum values for pressure at initial design are not identical, but very close to 12.3 bar. This discrepancy comes from numerical computation and from mismatching of the position of the chamber with respect to port opening during mesh rotation.

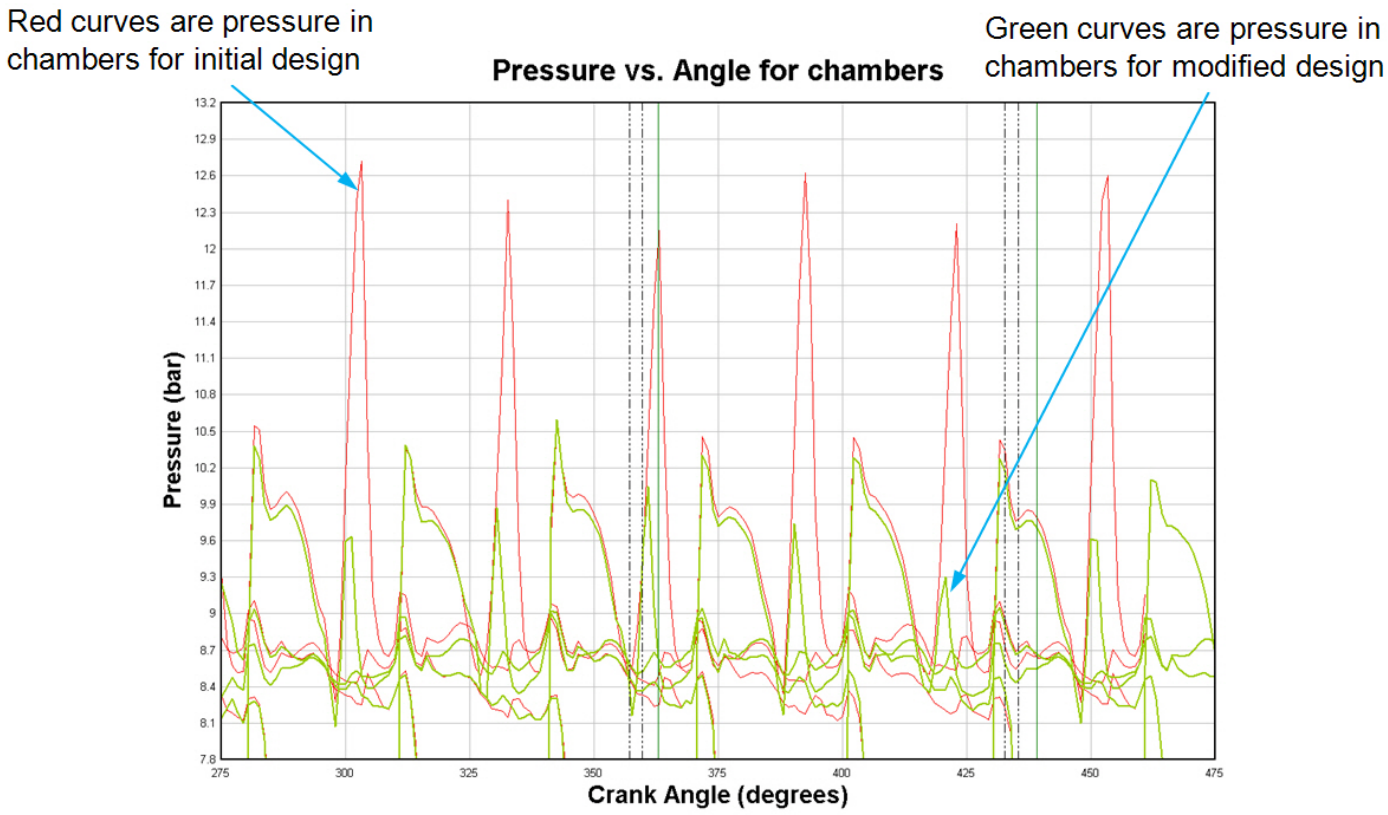


Figure 57: Comparison of results for pressure in pumping chambers at discharge for the initial and modified port design

A similar trend is observed at the low pressure monitors. Pressure measured before inlet has spikes which go down to -2.3 bar. For the revised design, however, the pressure spikes go only down to -1 bar, which is considered a significant decrease in the amplitude of pressure pulsation. Pressure monitors for initial and revised design for measurements before inlet are plotted on **Figure 58**, in which curves for pressure pulsations from initial design are shown in red color and curves for pressure pulsations from revised design are shown in green color.

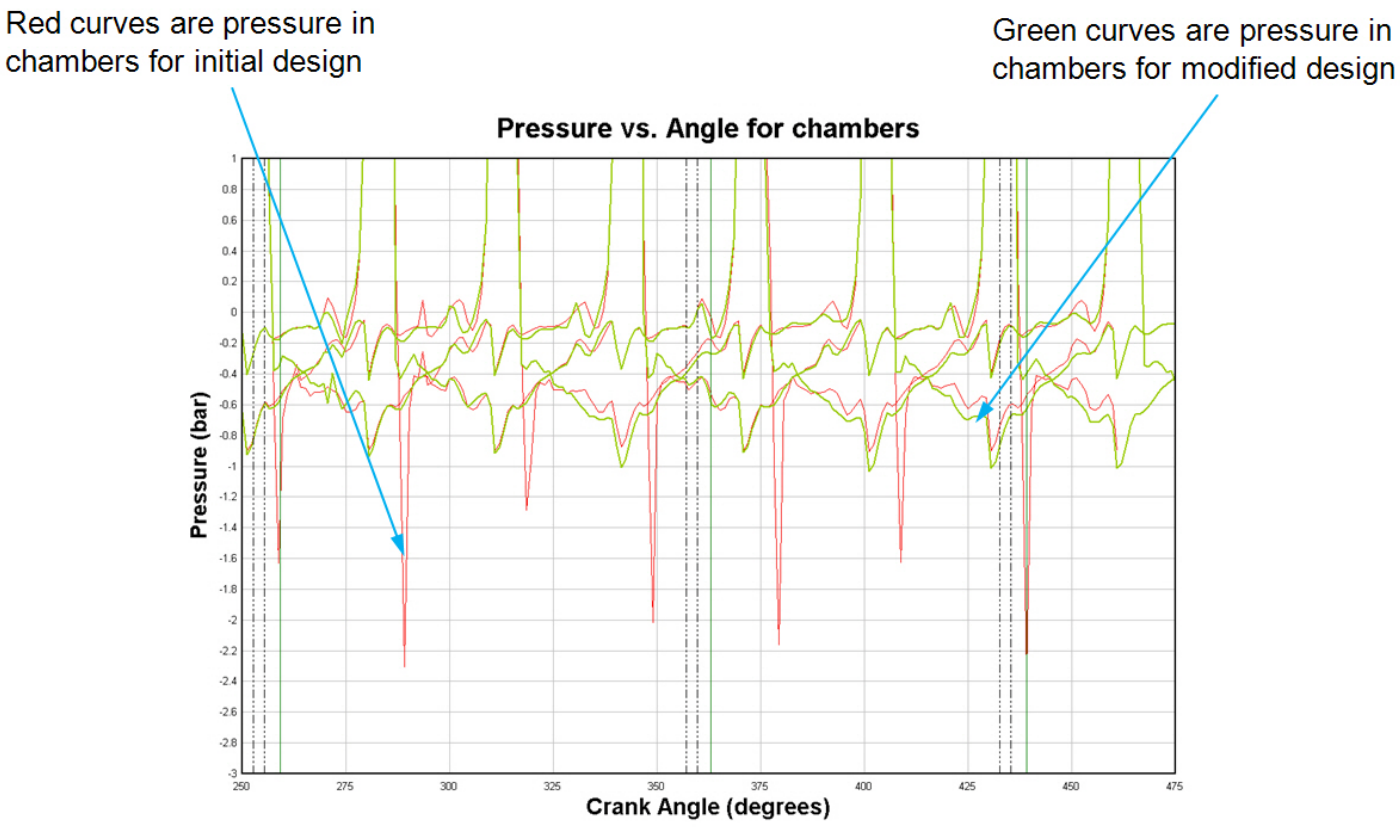


Figure 58: Comparison of results for pressure in pumping chambers before inlet for the initial and modified ports design

The simulation shows that the new design maintains the same flows as the original design. This is important since the reduction in pressure fluctuations must not come at a cost of reduced flow performance. The flow performance of the initial and revised porting design (overlaid) is shown on **Figure 59**. On this figure, curves for flow performance from initial design are shown in blue color and curves for pressure pulsations from revised design are shown in red color. Modification of ports also improves slightly flow oscillation, which is shown on lower side of blue curve, close to crank angle of about 302 degree mark.

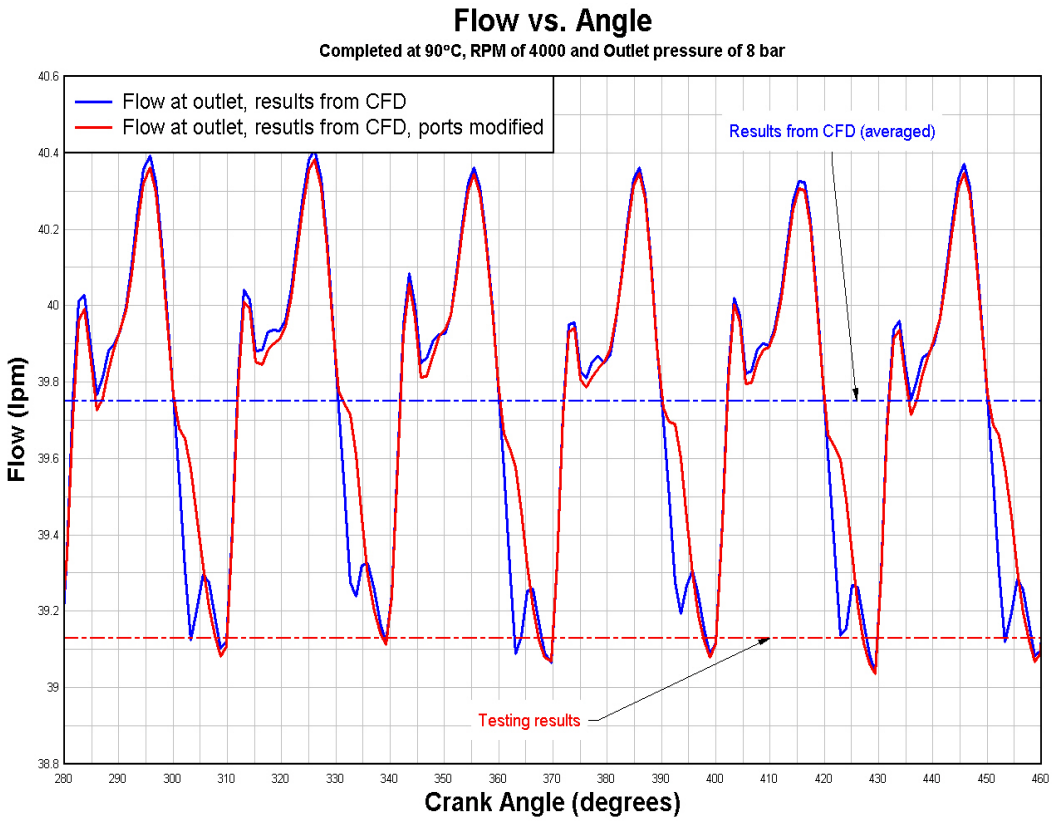


Figure 59: Flow performance for original and modified design

6.2. Groove implementation

From a study of different designs of balanced pumps, it was observed that different designs have different metering grooves which are implemented in pumps. Since it is not clear which groove design is better, it was decided to investigate the effect of grooves on pressure pulsations using two different designs, trapezoidal and triangular grooves, implemented in the *original* design of a balanced pump. Details for these different design approaches are depicted below.

6.2.1. Trapezoidal groove approach at discharge ports

A trapezoidal groove was considered first. The trapezoidal groove is implemented at the leading edge of discharge port, with the intention of lowering the pressure pulsation by creating a more gradual port opening for pumping chambers. The height of this groove is constant throughout its length. Details for this groove are depicted on **Figure 60** and **Figure 61**.

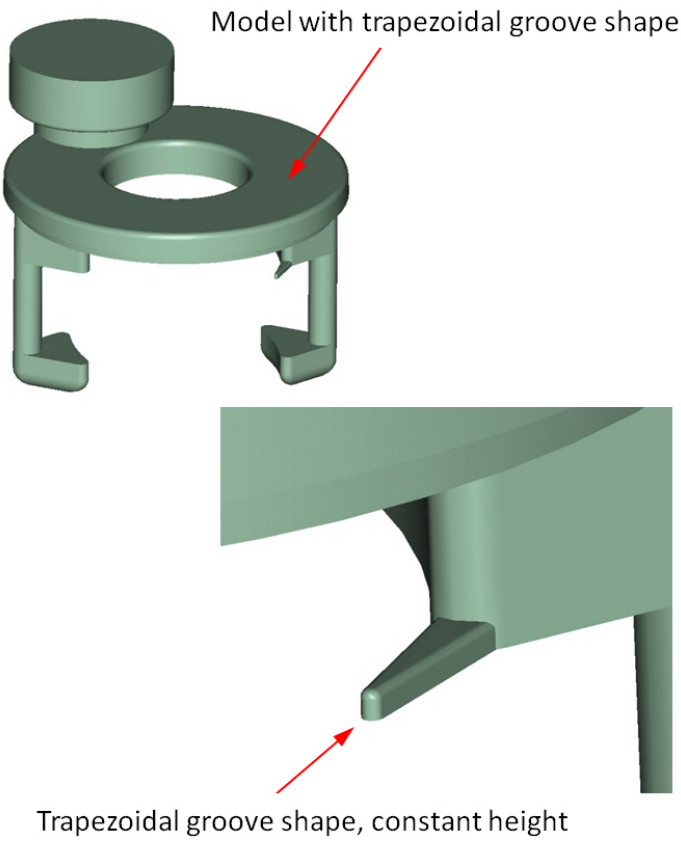


Figure 60: Trapezoidal groove implementation at pump discharge fluid domain

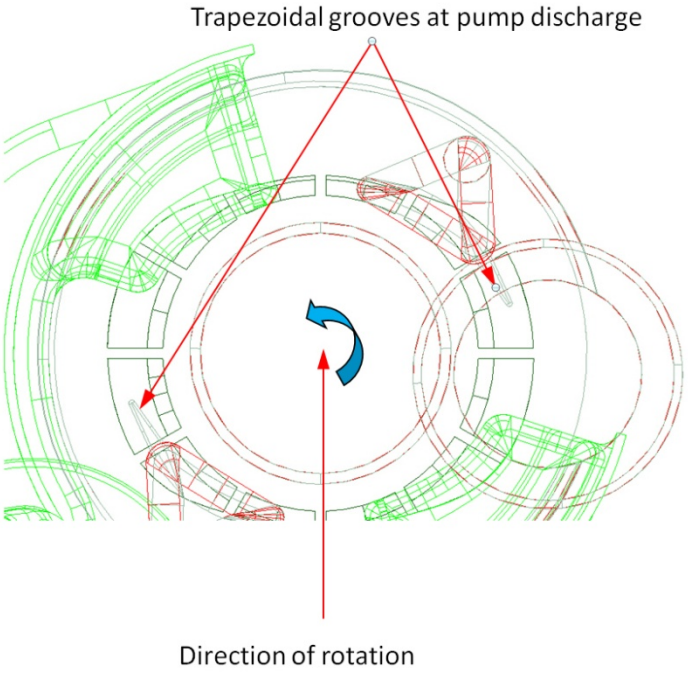


Figure 61: Trapezoidal groove implementation at pump discharge, top view

6.2.2. Triangular groove approach at discharge ports

As a second step, triangular groove shape is taken in consideration. This groove is implemented also at leading edge of the discharge port (at the same location as the trapezoidal groove), with the same intention of lowering the pressure pulsation by having more gradual port opening for the pumping chambers. The height of this groove is variable throughout its length. Details for this groove are depicted on **Figure 62** and **Figure 63**.

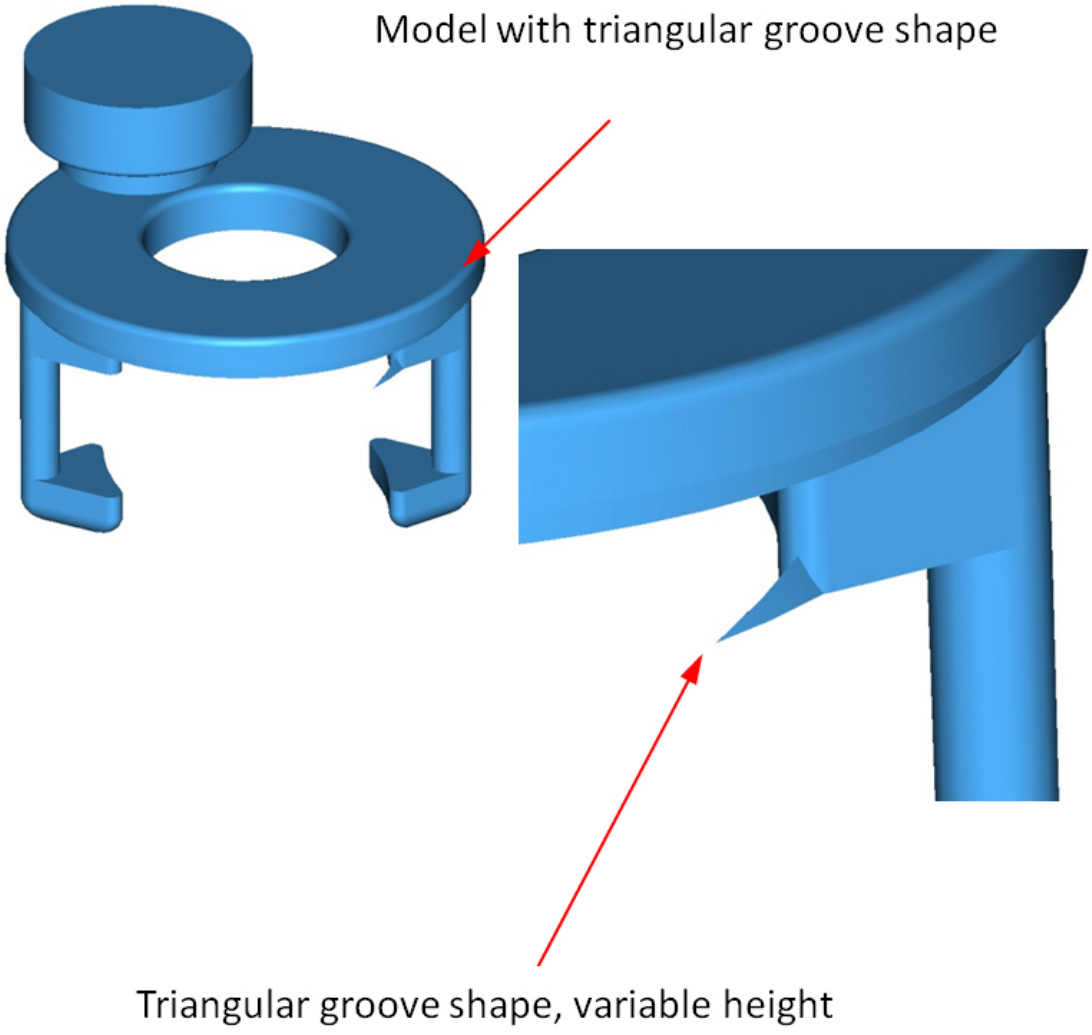


Figure 62: Triangular groove implementation at pump discharge fluid domain

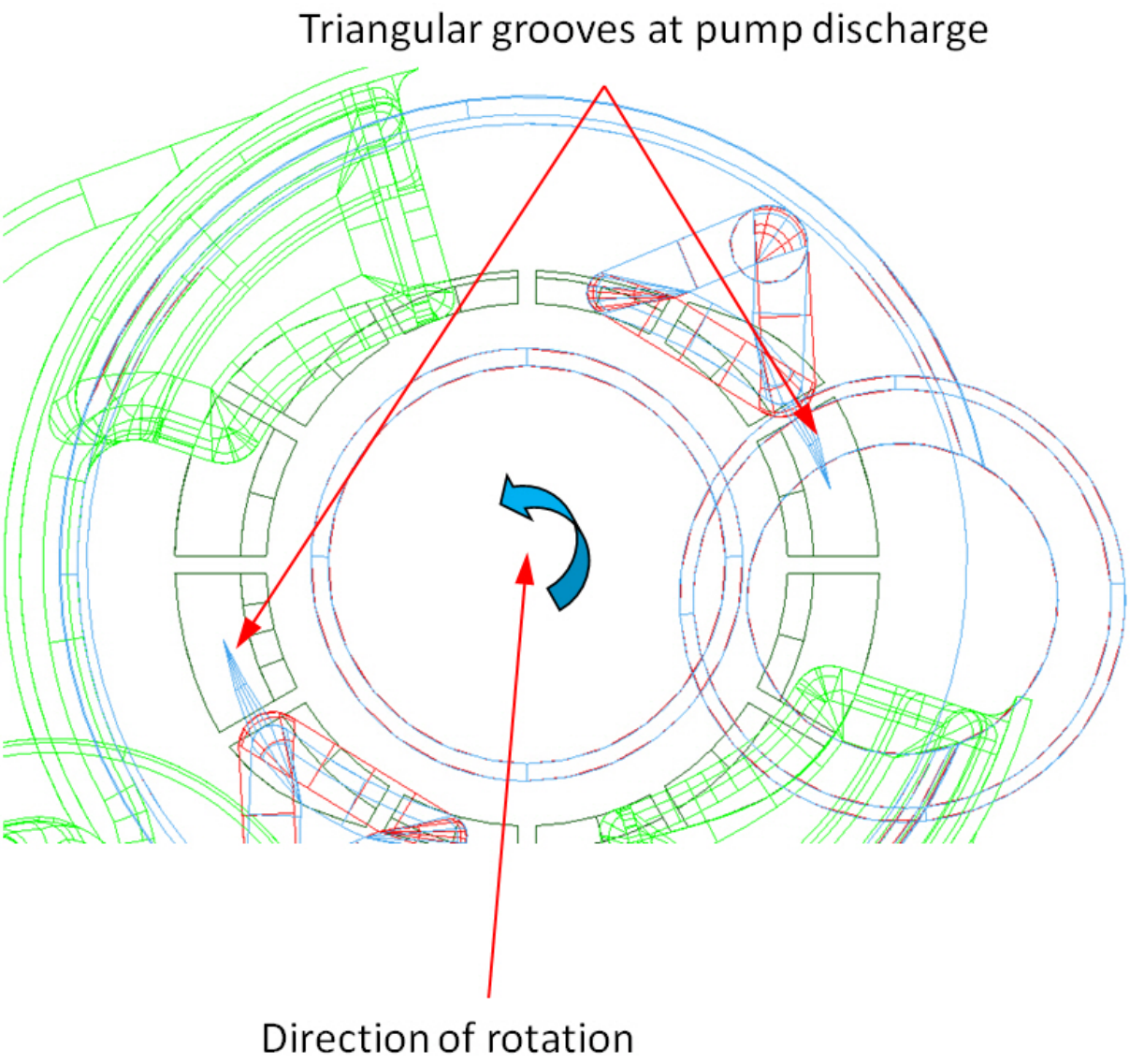


Figure 63: Triangular groove implementation at pump discharge, top view

6.2.3. Triangular grooves at inlet domain

As a third step, triangular groove shape is taken in consideration, implemented at the trailing edge of inlet port of the pump. Intention for implementation of this groove is to lower the pressure pulsations for pump while exiting inlet fluid domain. The height of this groove is variable throughout its length. Details for this groove are depicted on **Figure 64** and **Figure 65**.

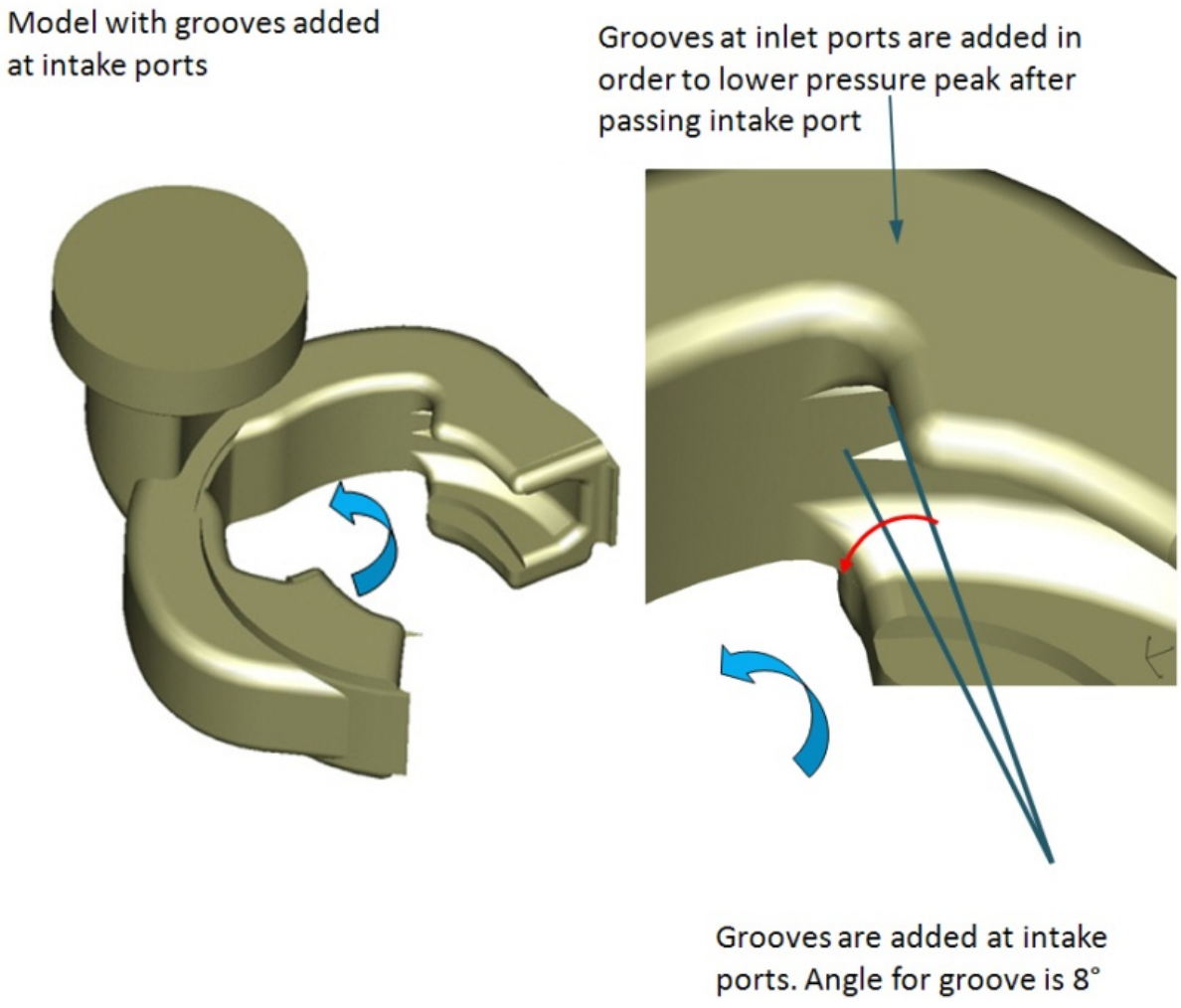


Figure 64: Triangular groove implementation at pump discharge fluid domain

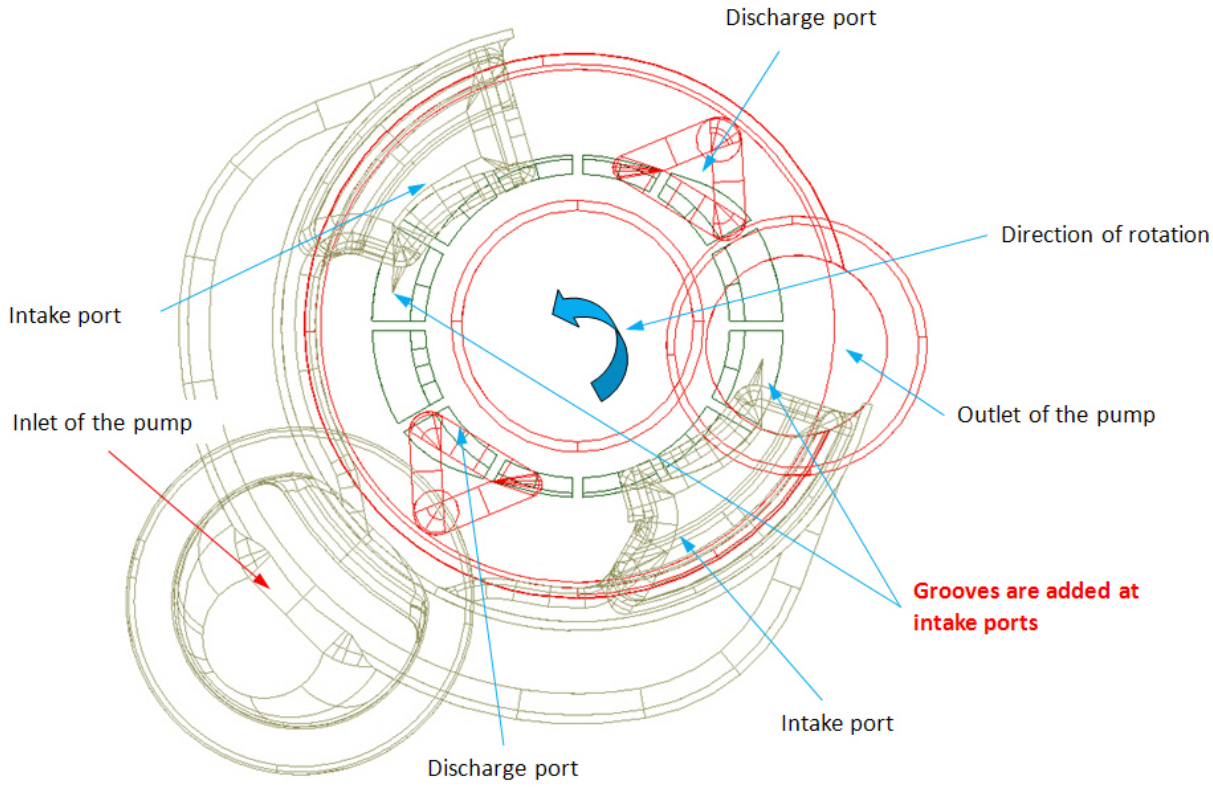


Figure 65: Triangular groove implementation at pump discharge, top view

6.3. Effects of different groove designs on simulation results

Results from the CFD simulations from models with grooves have been analyzed and overlaid with the *modified* port design pump.

Pressure pulsation curves can be compared from model with modified ports and models with grooves. In **Figure 66**, the pressure pulsation comparison between the model with modified ports (where the angle between ports has been reduced by 7 degrees) and trapezoidal grooves is shown. **Figure 67** depicts pressure pulsation comparison between model with trapezoidal and triangular grooves. It is clear that the magnitude of pressure pulsations is almost identical. However, shape of pressure pulsation curves is slightly different when comparing all three models together.

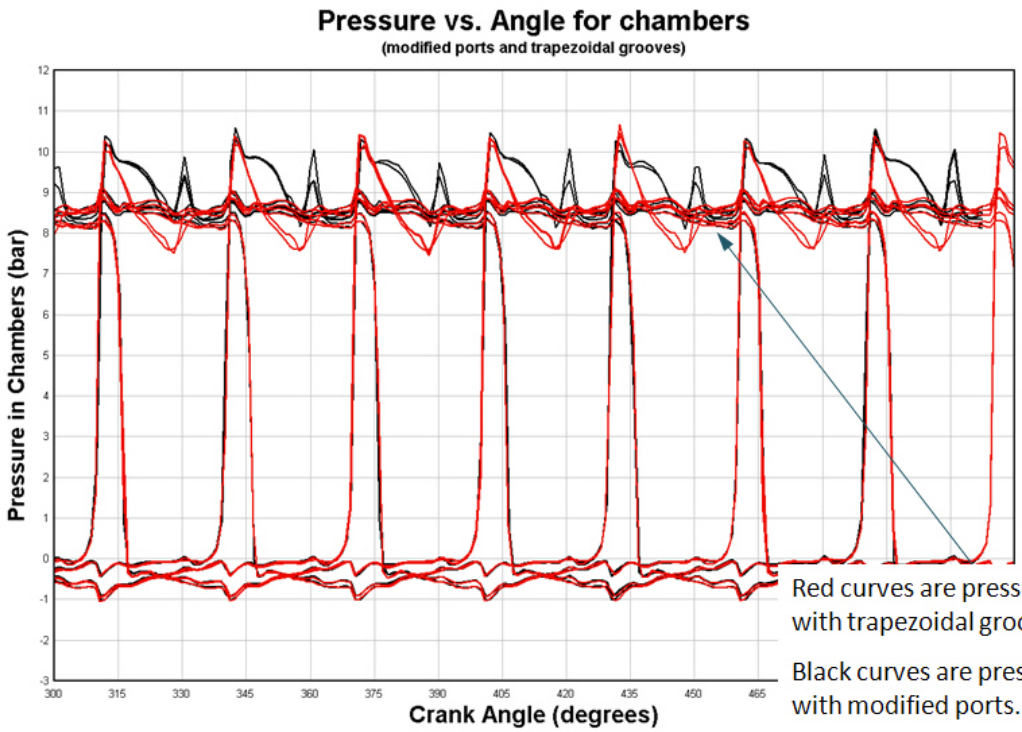


Figure 66: Pressure pulsation comparison between model with modified ports and trapezoidal grooves

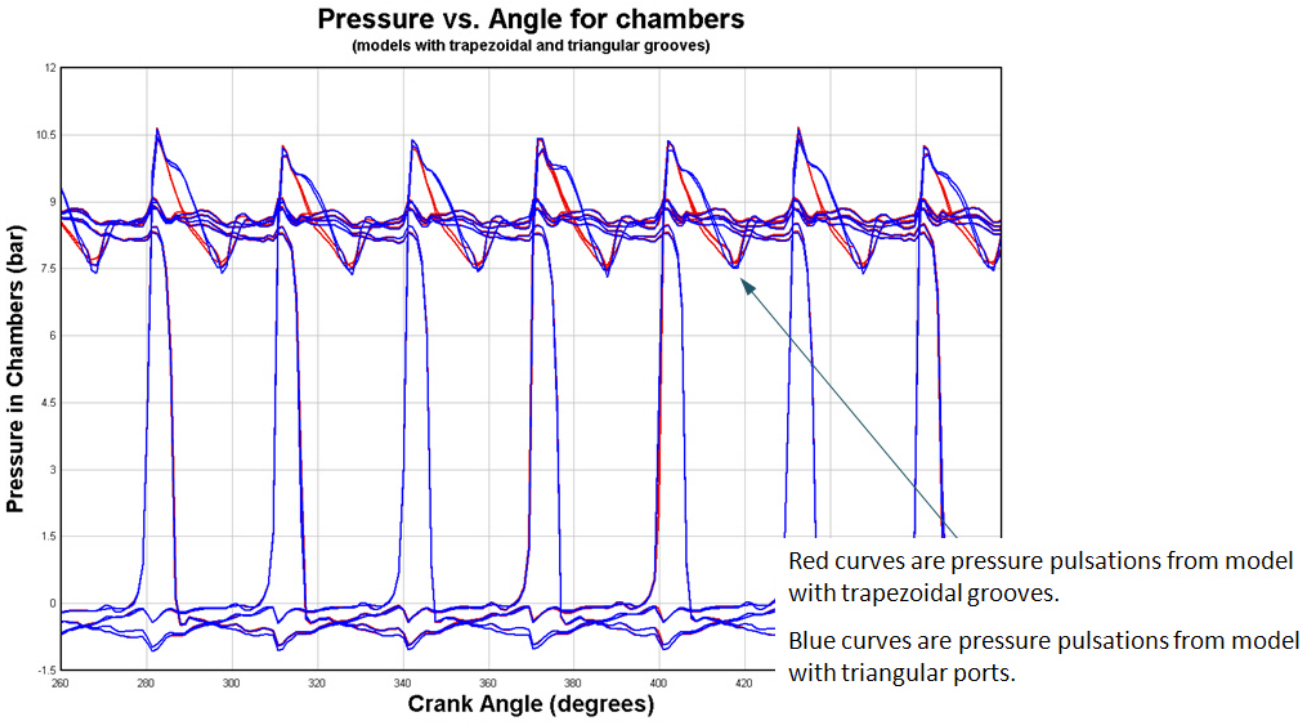


Figure 67: Pressure pulsation comparison between model with trapezoidal and triangular grooves

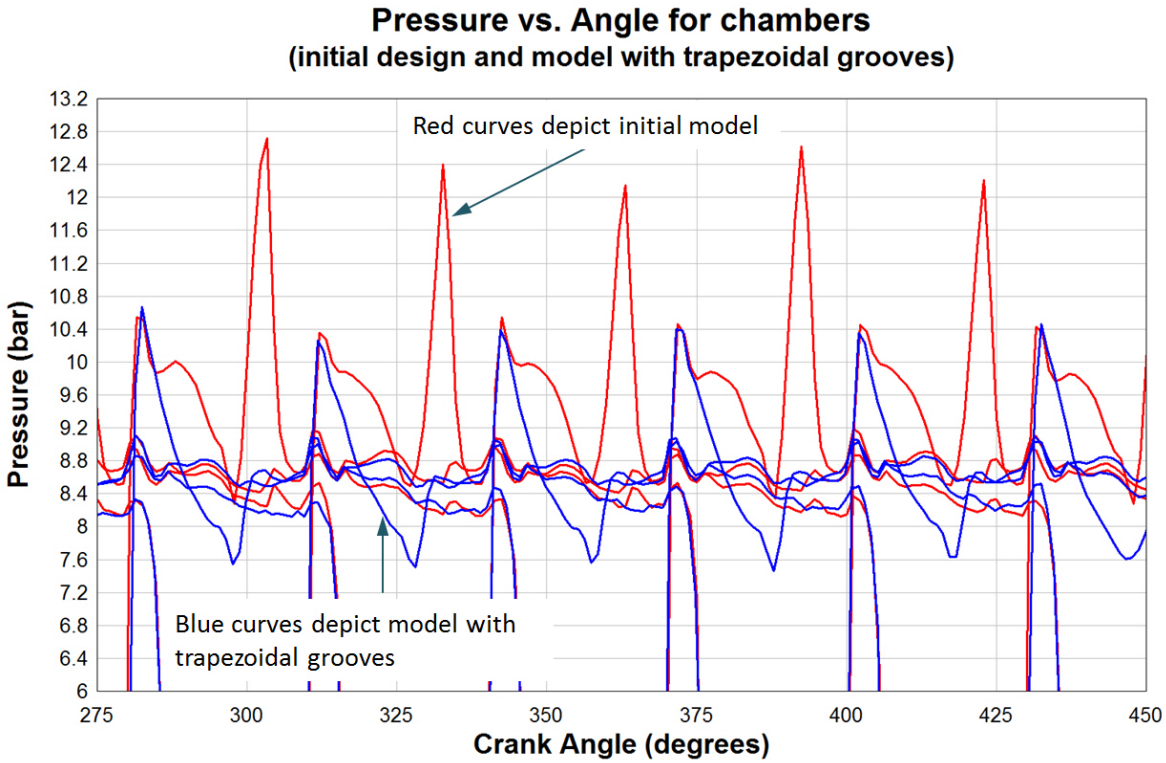


Figure 68: Pressure pulsation comparison between initial model and model with trapezoidal grooves

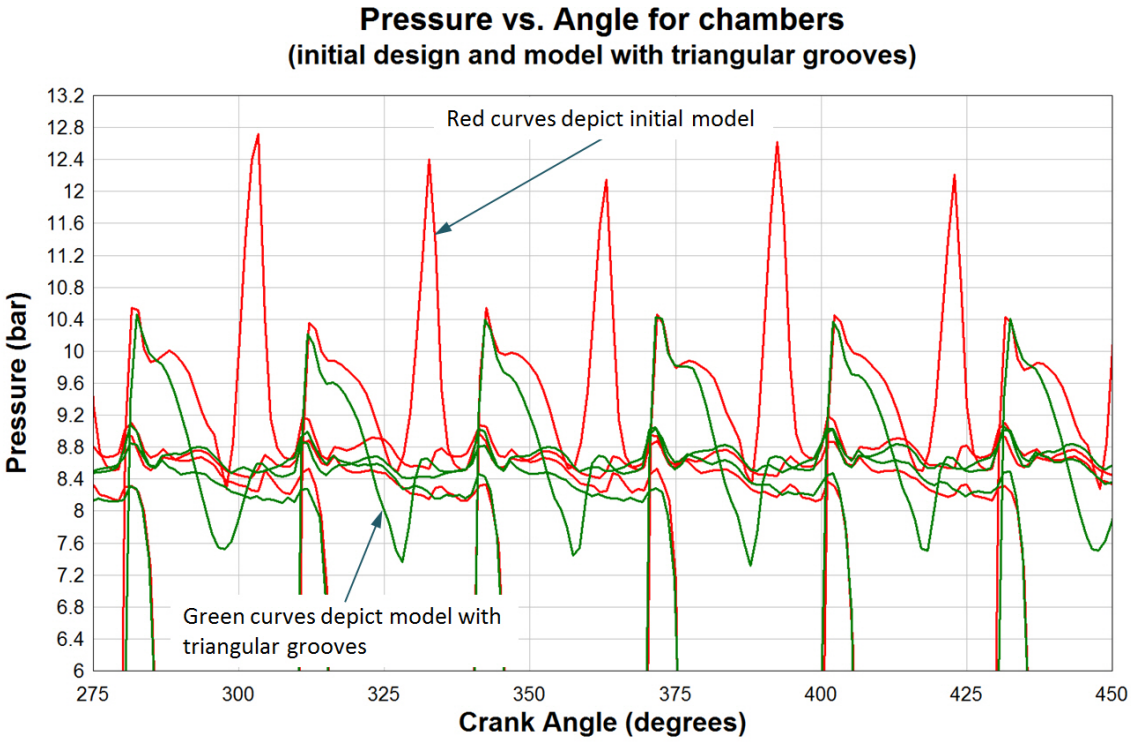


Figure 69: Pressure pulsation comparison between initial model and model with triangular grooves

In **Figure 68**, the pressure pulsation comparison between the initial and model with trapezoidal grooves is shown. **Figure 69** depicts pressure pulsation comparison between initial and model with triangular grooves. These comparisons show that grooves do lower pressure pulsations significantly. The precise groove geometry, however, does not have a great impact on the pressure pulsations since both geometries gave similar results.

Comparison of the pressure pulsations at inlet between model with modified ports and model with grooves implemented at inlet domain, gives lowered pressure pulsations for model with grooves. Details for the pressure pulsations of the pump are shown on **Figure 70**. It is clear that the slope of curves for the pressure pulsations is different when grooves are added. Change of slope of the curve which depicts closing of the chamber after passing the intake port is shown about 310° mark on **Figure 70**.

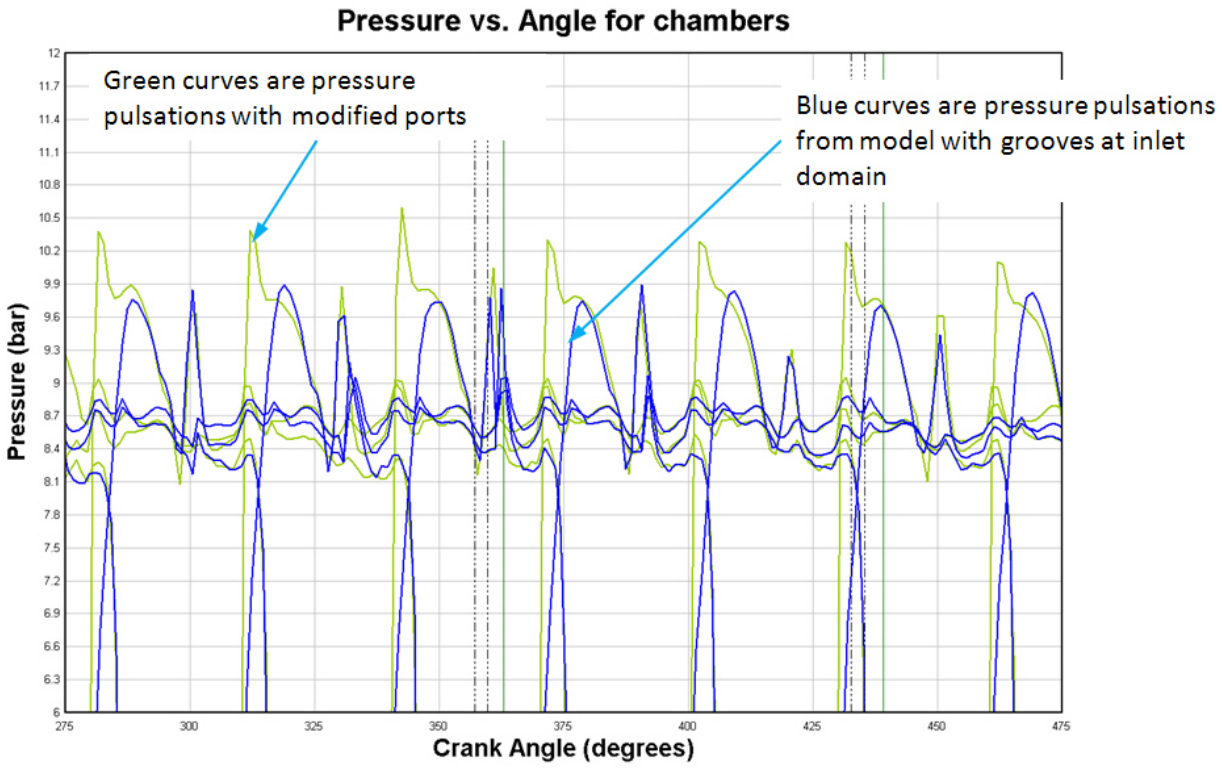


Figure 70: Pressure pulsation comparison between model with modified ports and model with grooves at intake port

Results for flow performance from the CFD simulations from models with grooves have been analyzed and overlaid with the modified design.

It is observed that there are negligible differences in terms of flow between model with modified ports and models with grooves. In **Figure 71**, the red curve depicts model with modified ports and the blue curve depicts model with trapezoidal grooves. Flow performance for model of balanced pump with trapezoidal grooves is slightly higher, but this difference is negligible. Also, **Figure 72** shows the flow performance comparison between model with trapezoidal and triangular grooves. The curve for model with triangular grooves is shown in red color, while the curve from model with trapezoidal grooves is shown in blue color. Comparison clearly shows that there is almost no difference in flow using different shape of grooves. The only difference between flow performance curves is slightly different slope of the flow curve, obtained with model with triangular grooves.

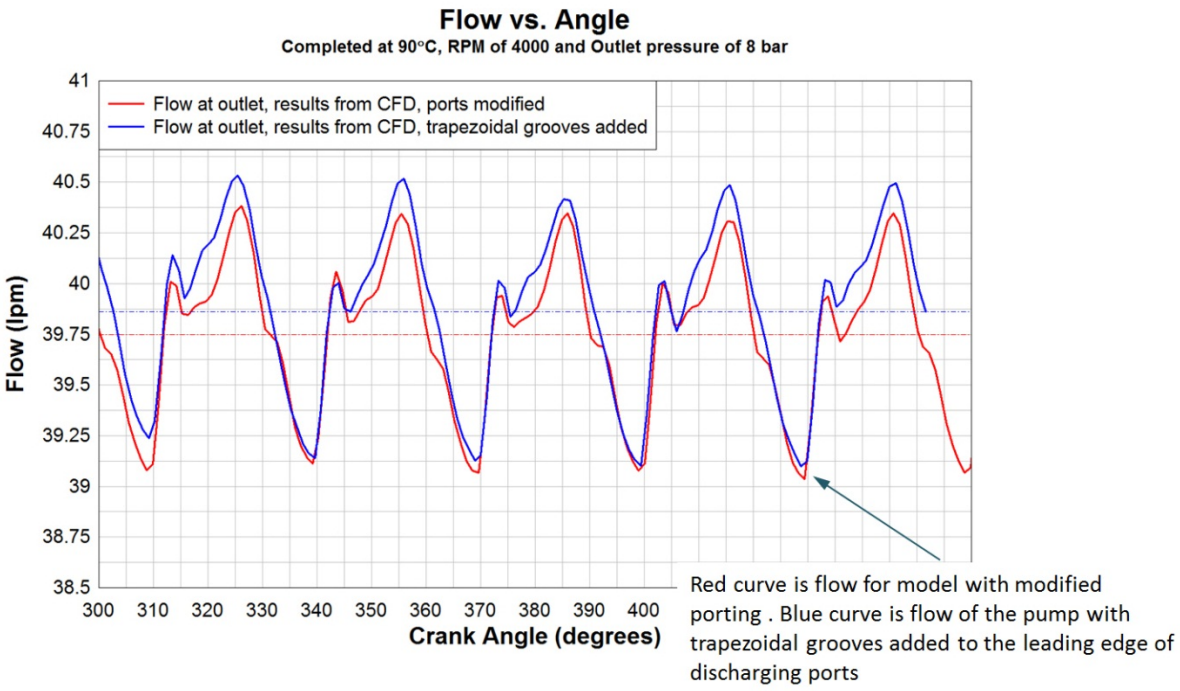


Figure 71: Flow performance comparison between model with modified ports and trapezoidal grooves at the leading edge of discharge ports

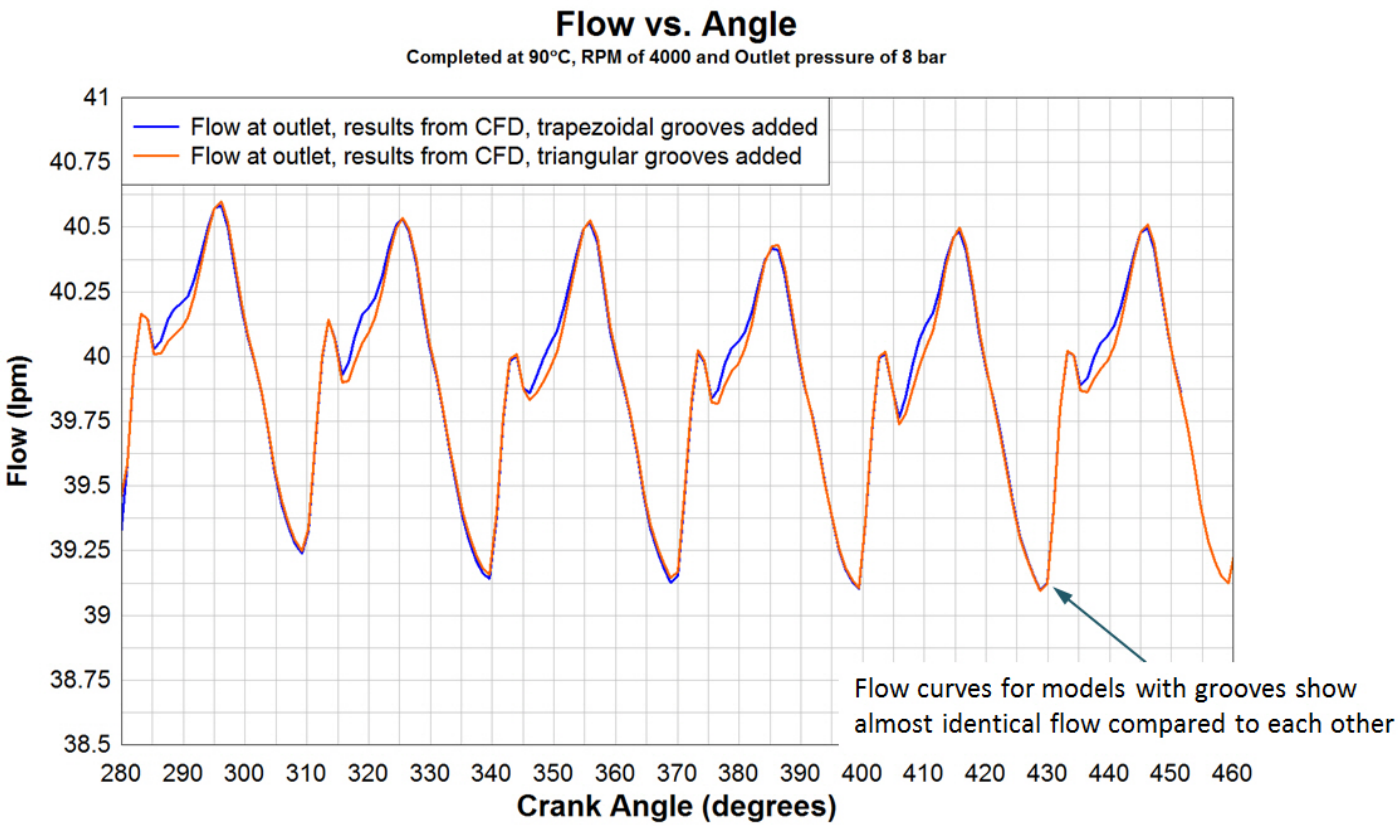


Figure 72: Flow performance comparison between model with trapezoidal and triangular grooves at the leading edge of discharge ports

Chapter 7 – Spectral analysis of simulated pressure signals

A useful method to evaluate the pressure pulsations is spectral analysis of the CFD simulation. Several different design iterations have been presented and results have been explained. However, spectral analysis of results is still remaining. Therefore, Fast Fourier Transformation (FFT) is conducted for the simulated pressure signals described in previous section. Pressure pulsations are taken from *one chamber* only (since pressure signals from other chambers are identical) and from *pressure pulsations at discharge port*. All of presented signals are created using pressure pulsations curves obtained from CFD simulation at 4000 rpm. The time step used for all simulations is the same and matches the node to node at the interface for each consecutive step.

Following design iterations have been analyzed and the results are presented in this section:

1. *Initial* design of balanced pump with corresponding FFT from rotating chamber. Pressure pulsation is depicted on **Figure 73**; corresponding FFT is depicted on **Figure 74**. Pressure pulsation from CFD simulation, at the location of pressure transducer is depicted on **Figure 75**; corresponding FFT is depicted on **Figure 76**. Pressure pulsations at discharge port are depicted on **Figure 77**; corresponding FFT is depicted on **Figure 78**.
2. *Revised* design of modified port pump with corresponding FFT from rotating chamber. Pressure pulsation is depicted on **Figure 79**; corresponding FFT is depicted on **Figure 80**. Pressure pulsations at discharge port are depicted on **Figure 81**; corresponding FFT is depicted on **Figure 82**.
3. Pressure pulsations from initial balanced pump model with *trapezoidal* grooves with corresponding FFT from rotating chamber. Pressure pulsation is depicted on **Figure 83**; corresponding FFT is depicted on **Figure 84**. Pressure pulsations at discharge port are depicted on **Figure 85**; corresponding FFT is depicted on **Figure 86.**

4. Pressure pulsations from initial balanced pump model with *triangular* grooves with corresponding FFT from rotating chamber. Pressure pulsation is depicted on **Figure 87**; corresponding FFT is depicted on **Figure 88**. Pressure pulsations at discharge port are depicted on **Figure 89**; corresponding FFT is depicted on **Figure 90.**
5. Pressure pulsations from balanced pump model with *revised* porting design and triangular grooves implemented at *trailing edge at intake ports* with corresponding FFT from rotating chamber. Pressure pulsation is depicted on **Figure 91**; corresponding FFT is depicted on **Figure 92**. Pressure pulsations at discharge port are depicted on **Figure 93**; corresponding FFT is depicted on **Figure 94.**

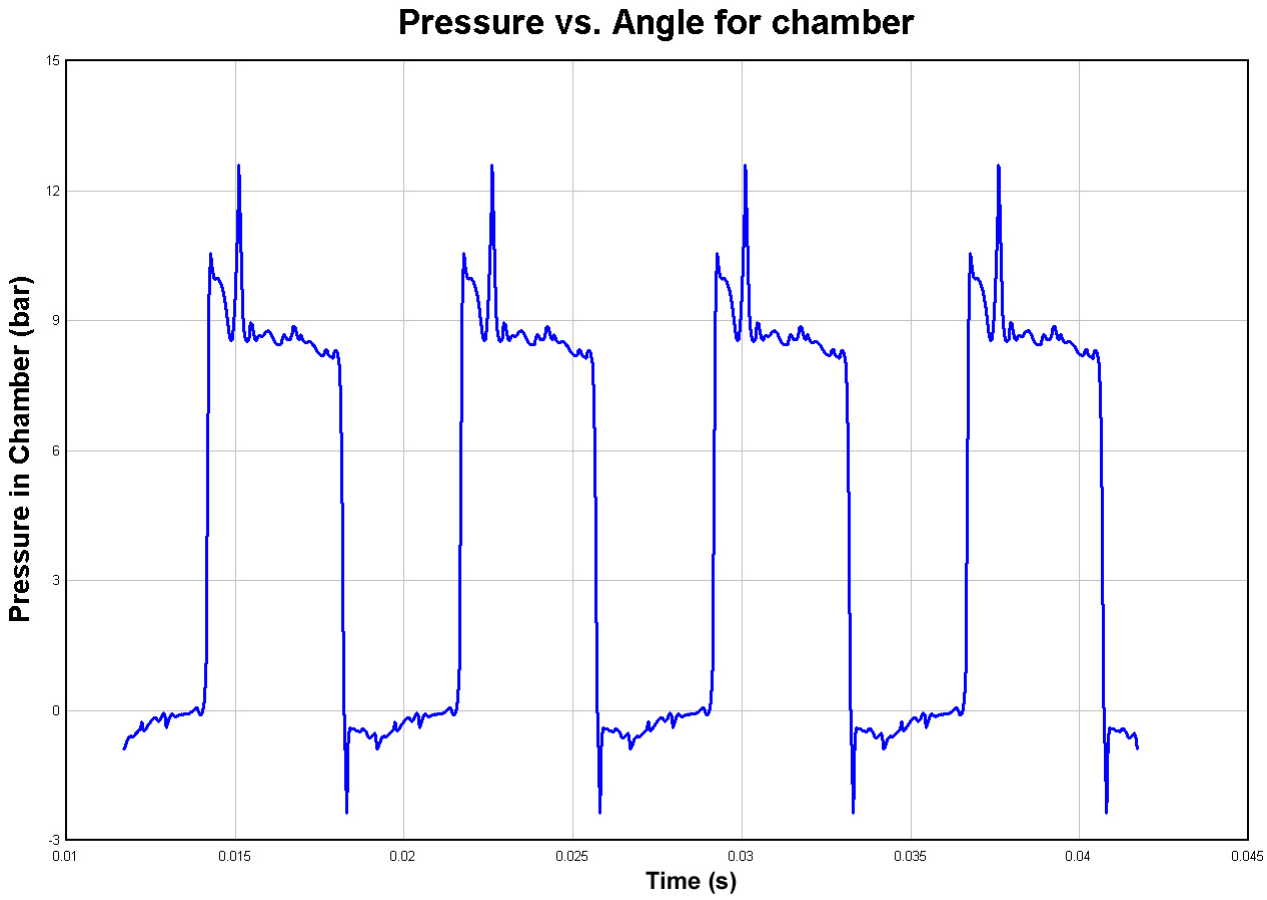


Figure 73: Pressure pulsation of one chamber from initial design of balanced pump

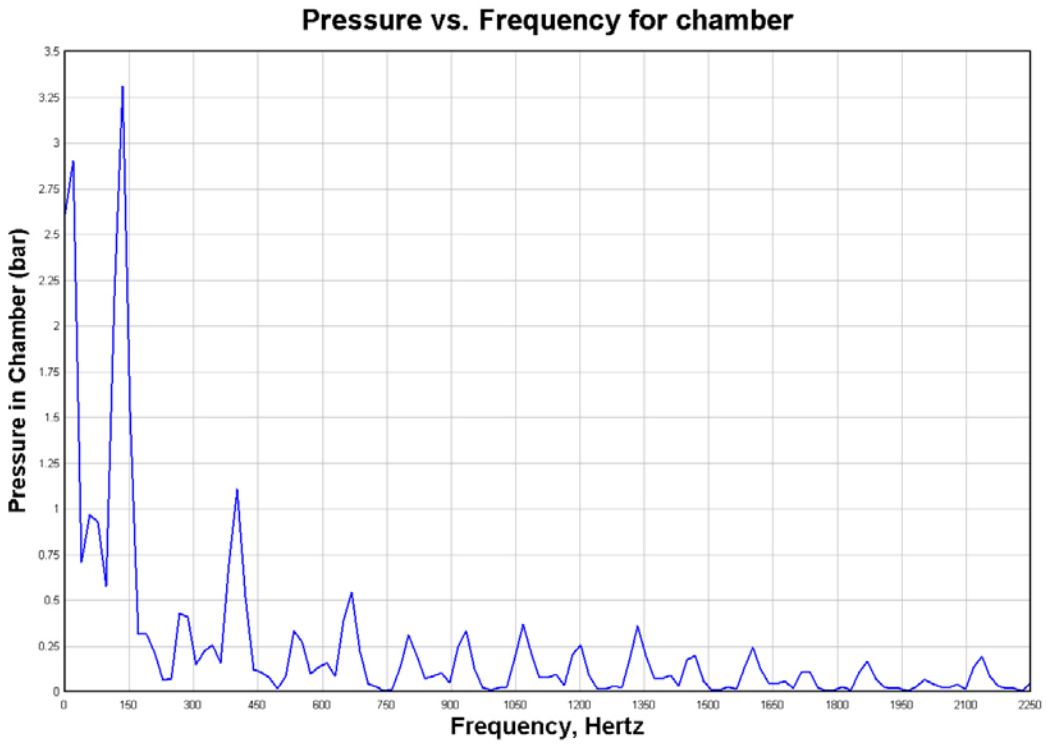


Figure 74: FFT from pressure pulsation of one chamber from initial design of balanced pump

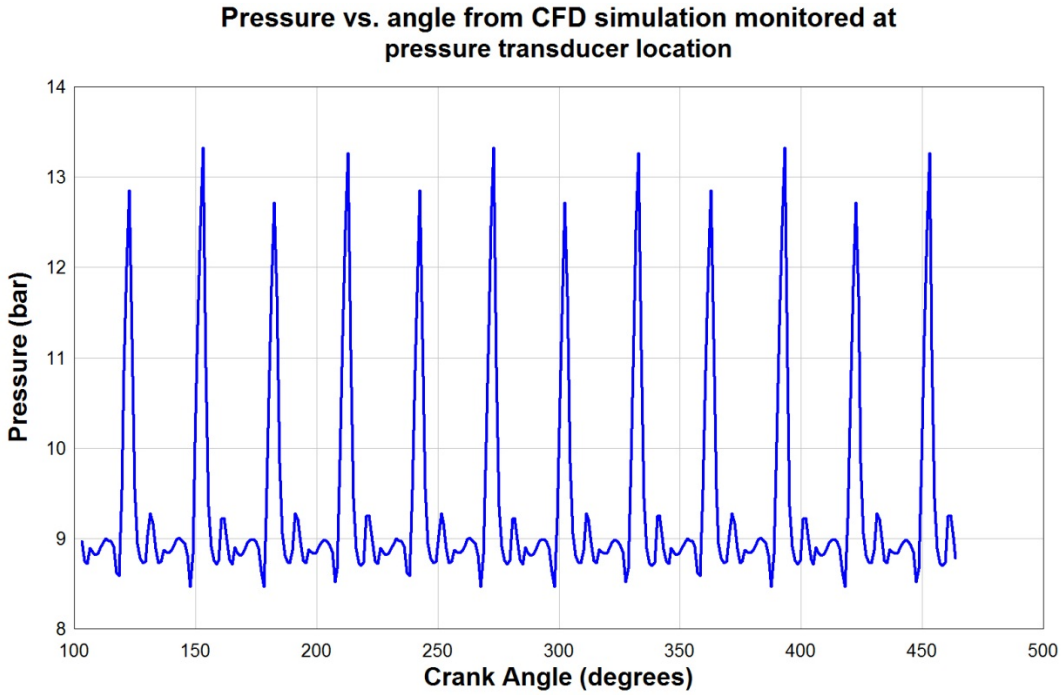


Figure 75: Pressure pulsation from CFD simulation from initial design of balanced pump at the pressure transducer location

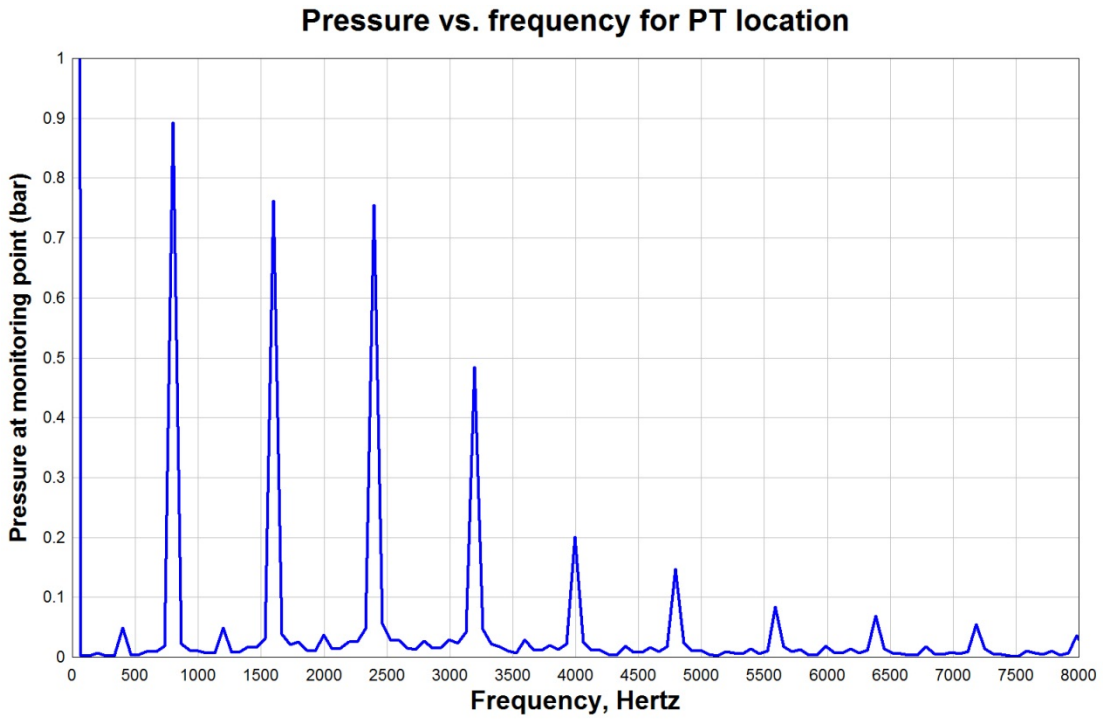


Figure 76: FFT from pressure pulsation from CFD simulation from initial design of balanced pump at the pressure transducer location

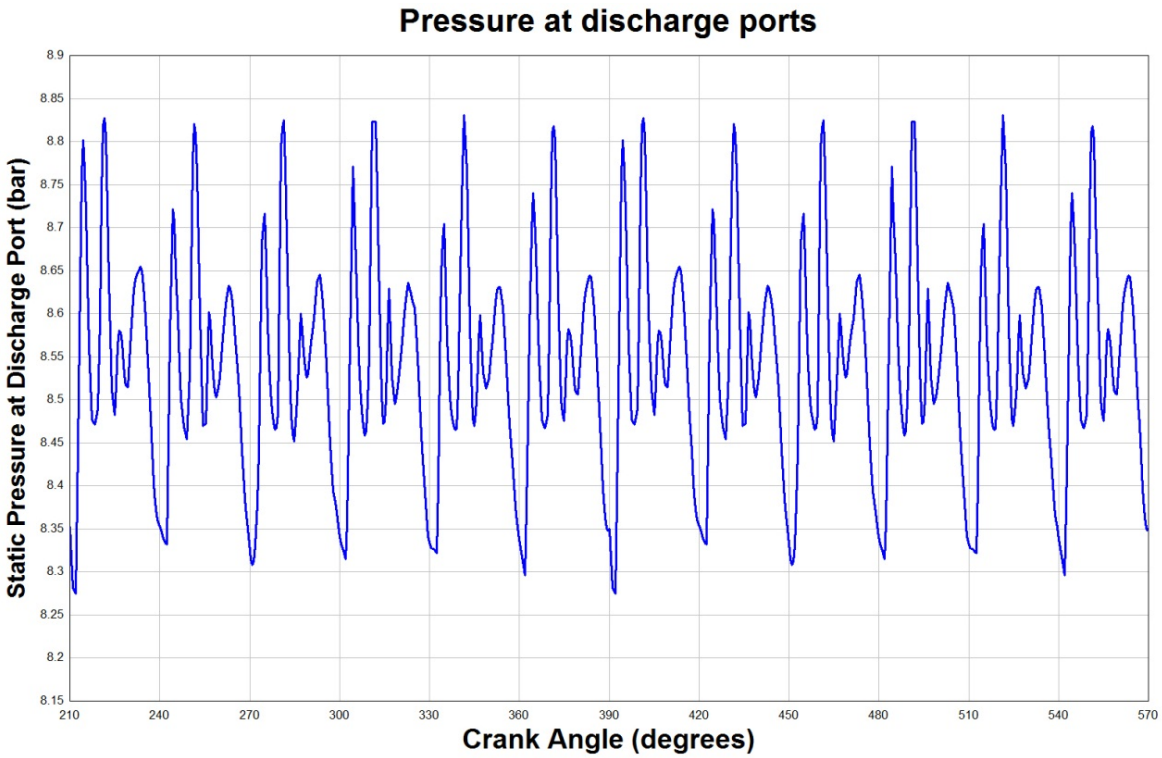


Figure 77: Pressure pulsation at discharge ports from initial design of balanced pump

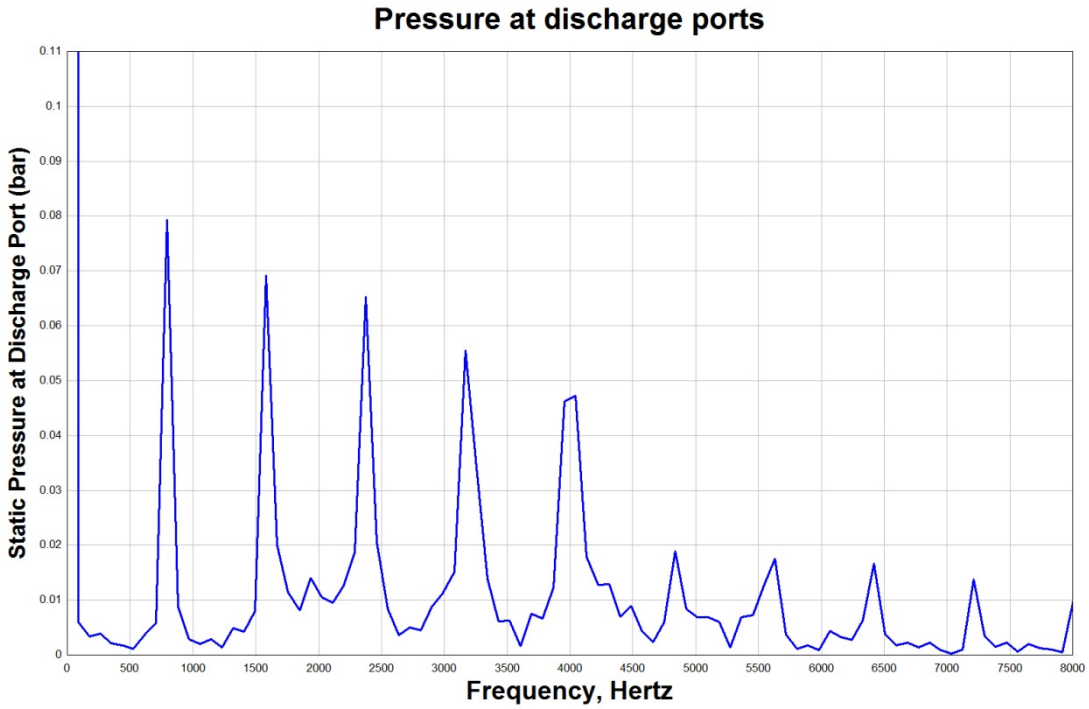


Figure 78: FFT from pressure pulsation of discharge ports from initial design of balanced pump

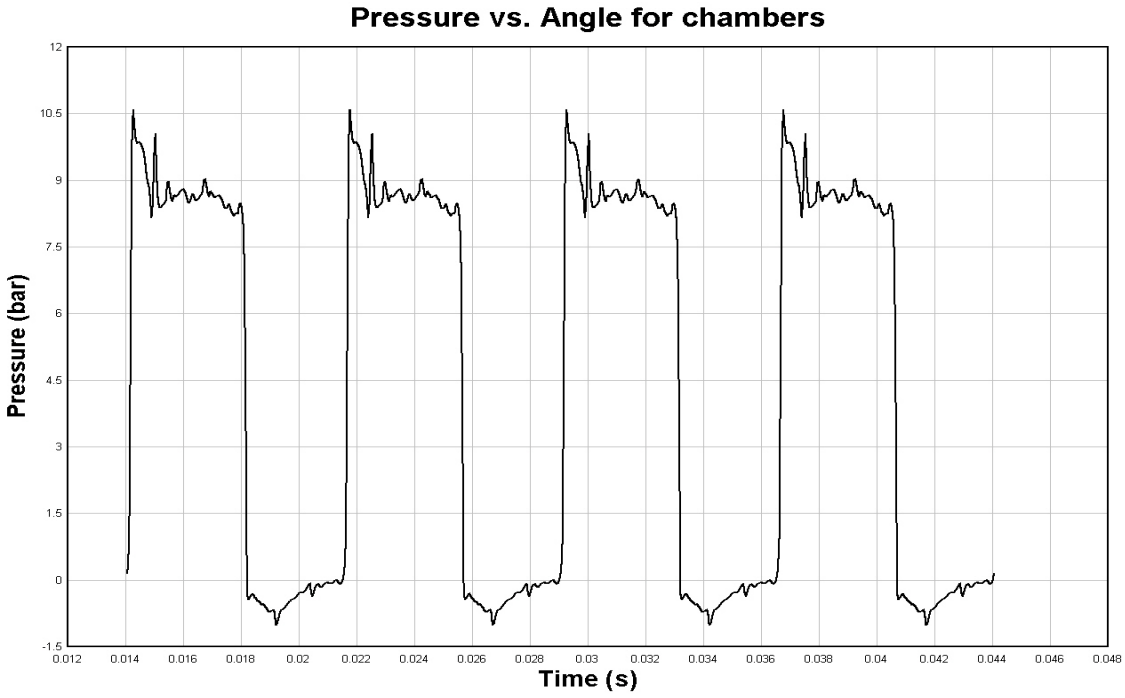


Figure 79: Pressure pulsation of one chamber from revised port design of balanced pump

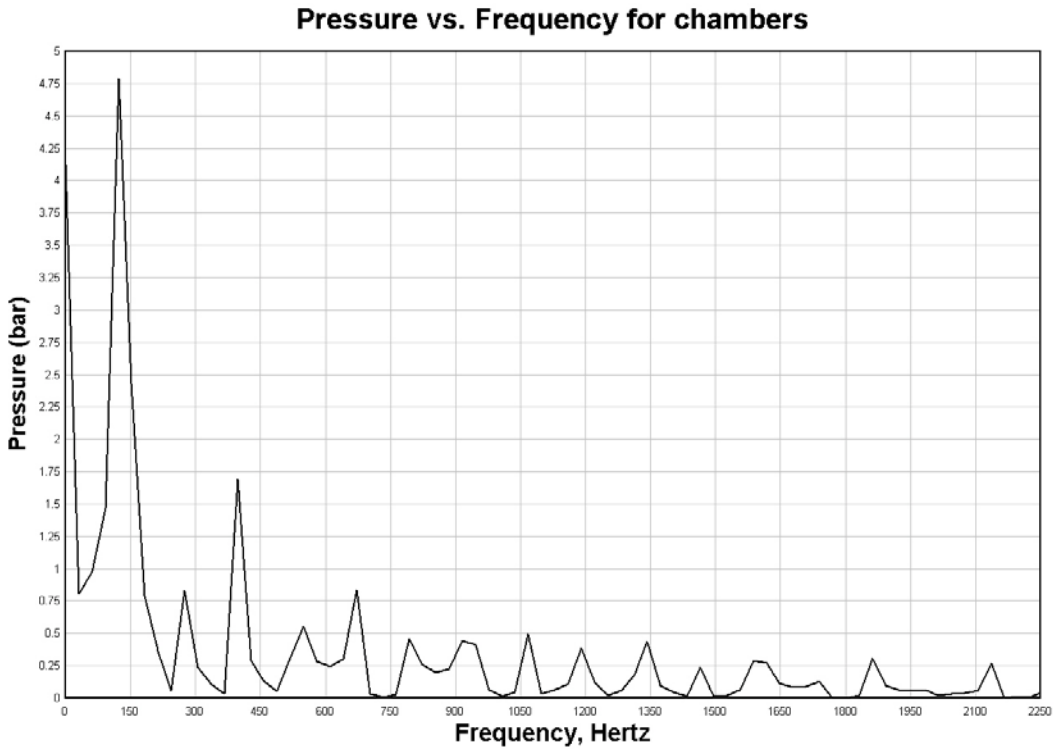


Figure 80: FFT from pressure pulsation of one chamber from revised port design of balanced pump

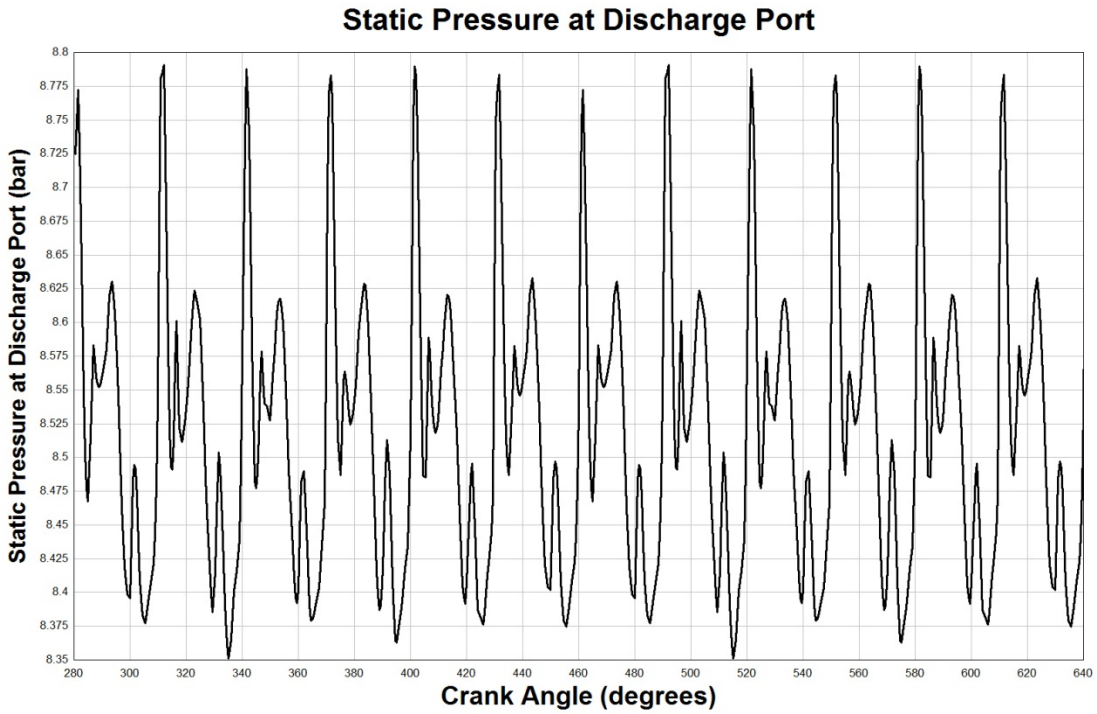


Figure 81: Pressure pulsation at discharge ports from revised port design of balanced pump

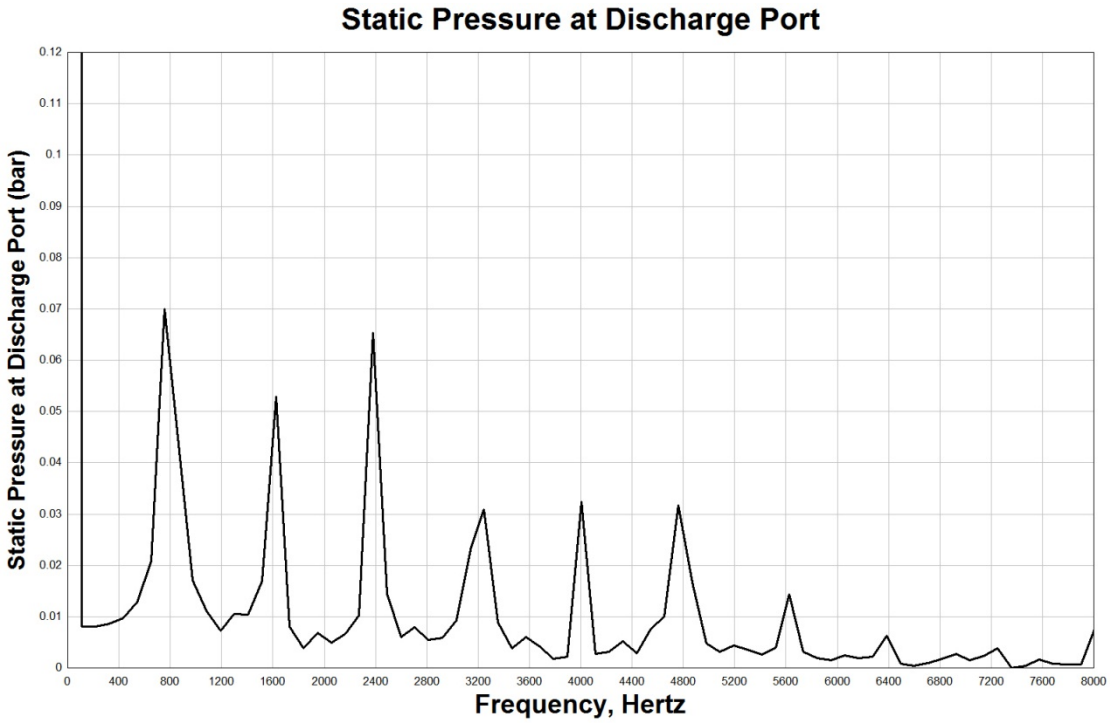


Figure 82: FFT from pressure pulsation of discharge ports from revised port design of balanced pump

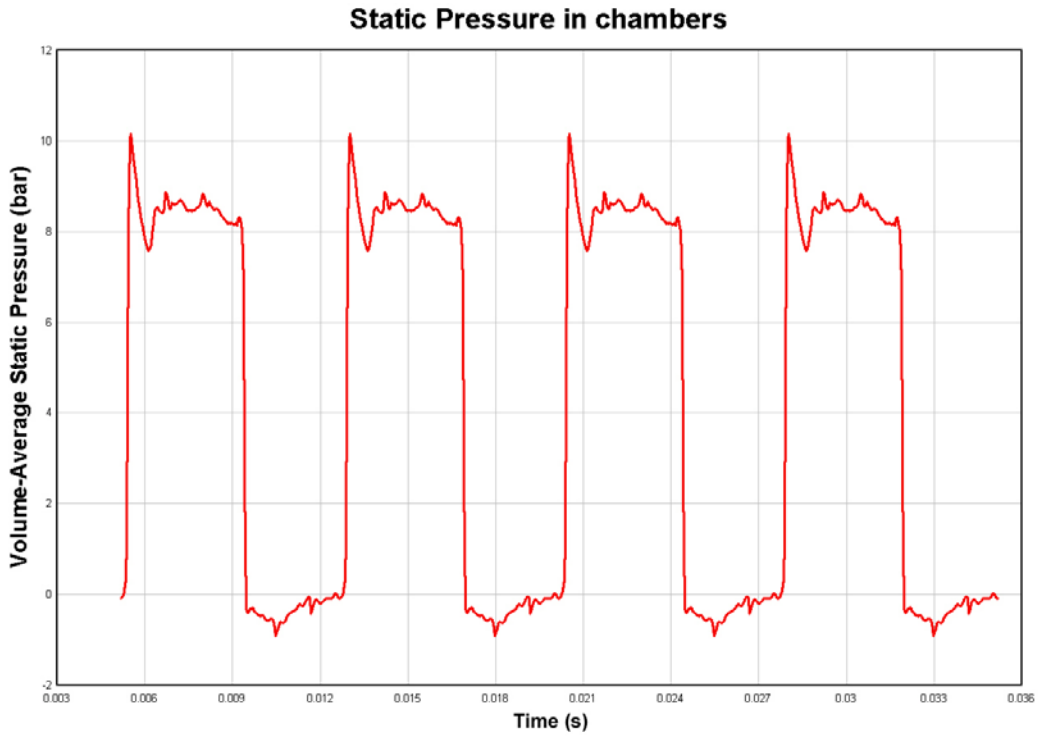


Figure 83: Pressure pulsation of one chamber from model with trapezoidal grooves

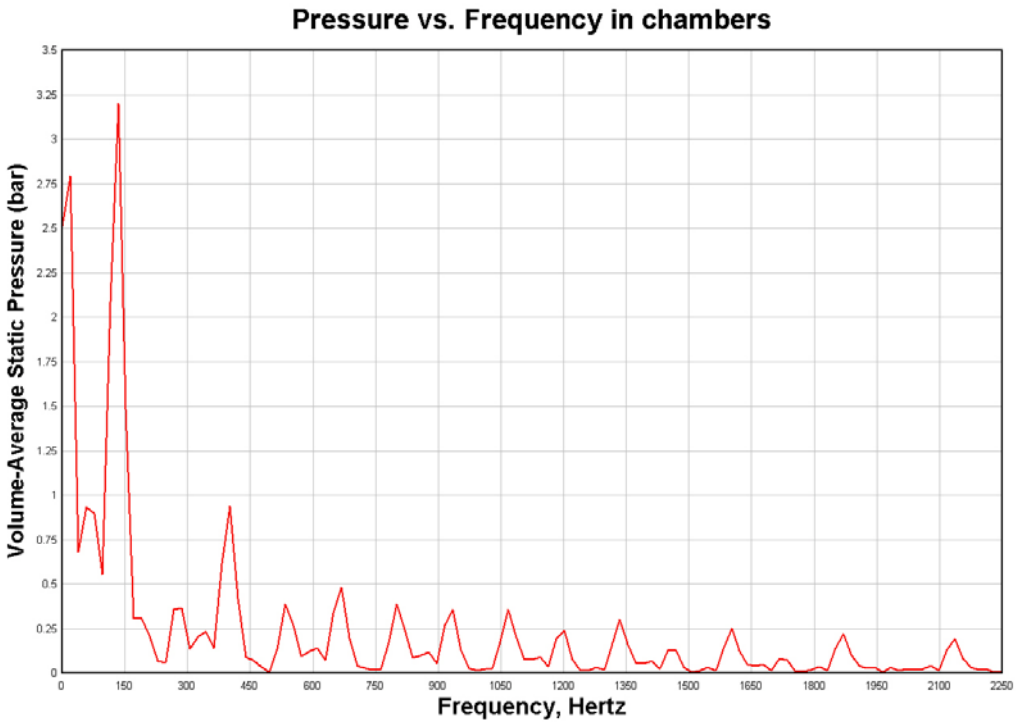


Figure 84: FFT from pressure pulsation of one chamber from model with trapezoidal grooves

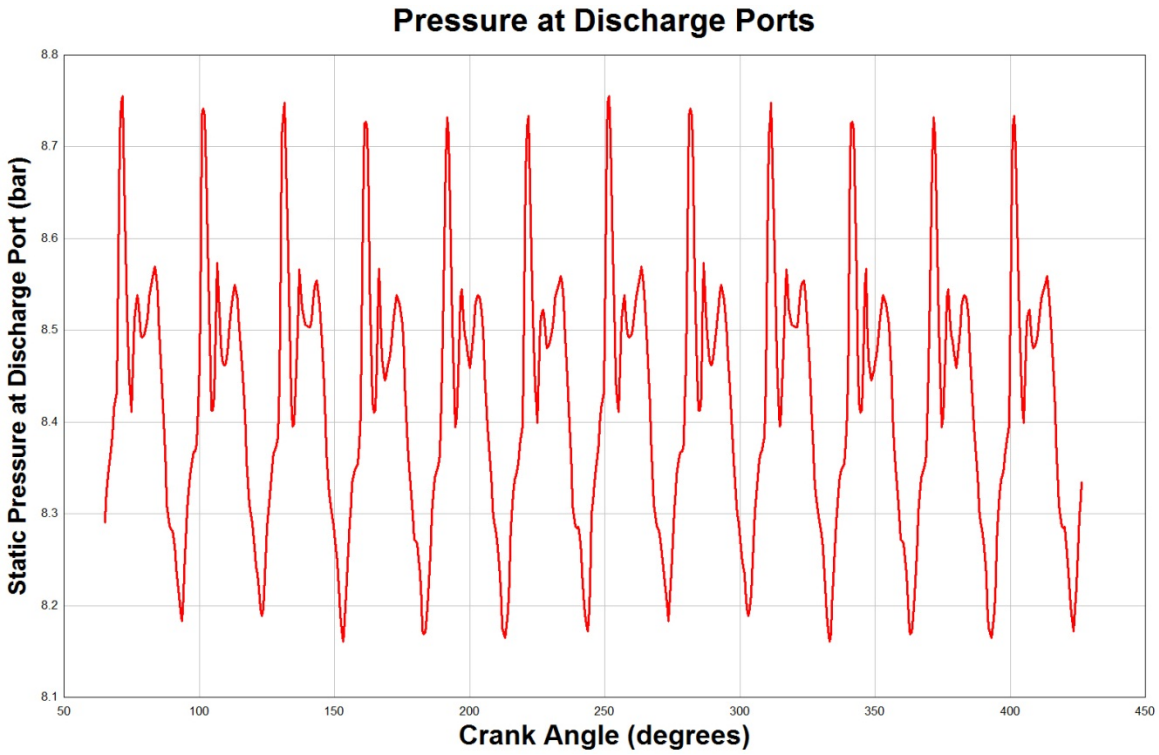


Figure 85: Pressure pulsation at discharge ports from model with trapezoidal grooves

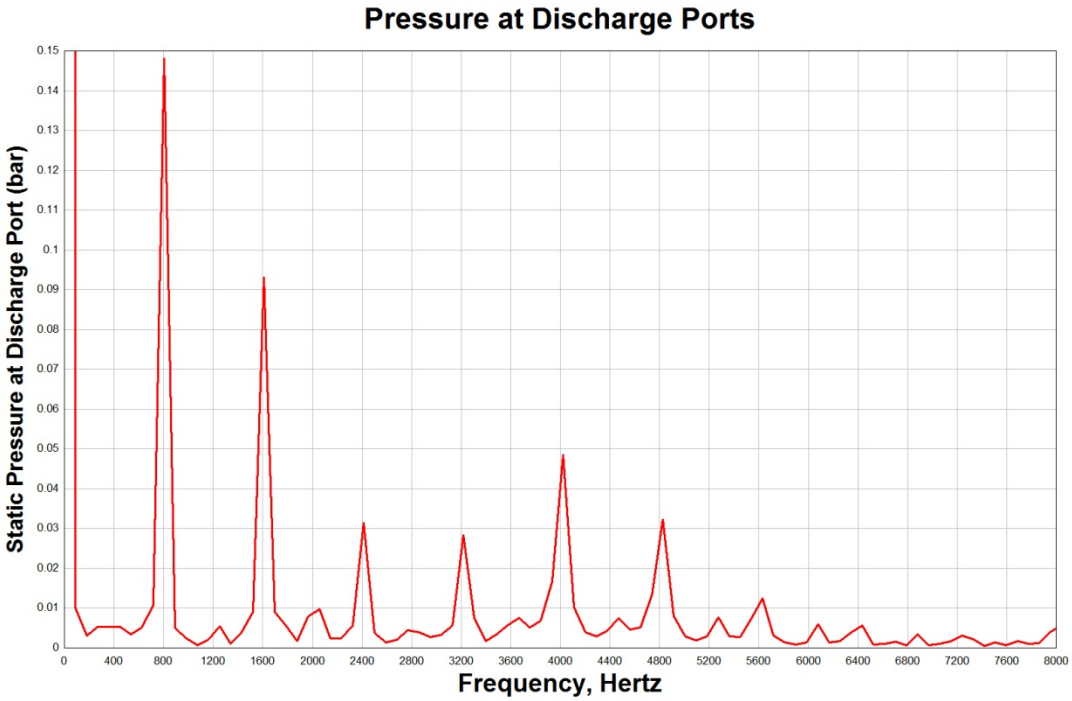


Figure 86: FFT from pressure pulsation at discharge ports from model with trapezoidal grooves

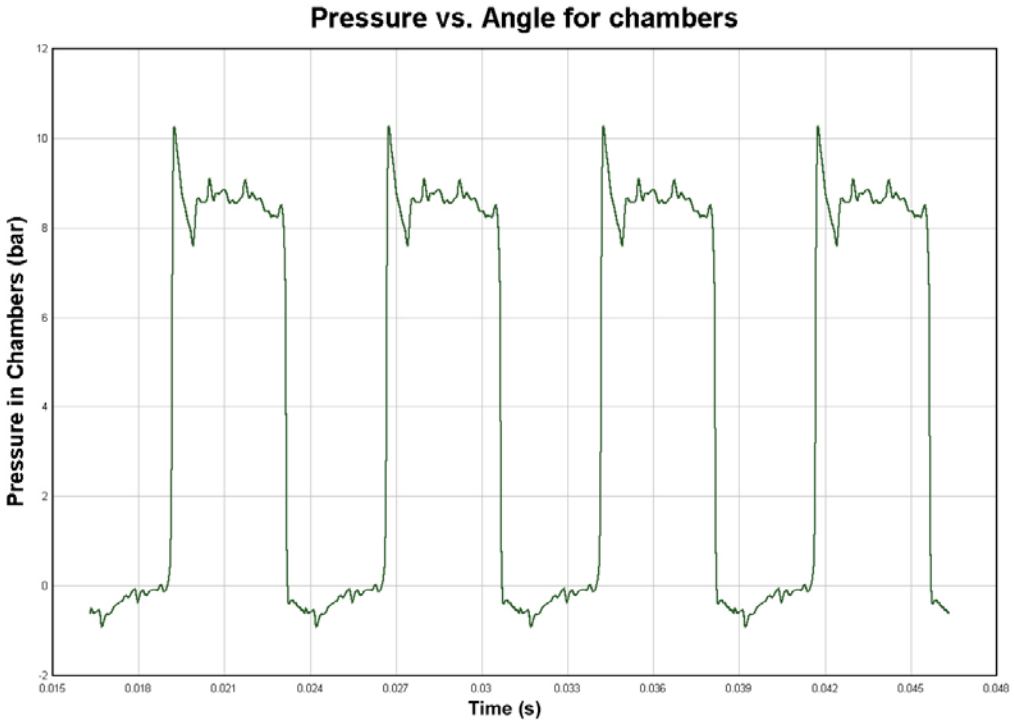


Figure 87: Pressure pulsation of one chamber from model with triangular grooves

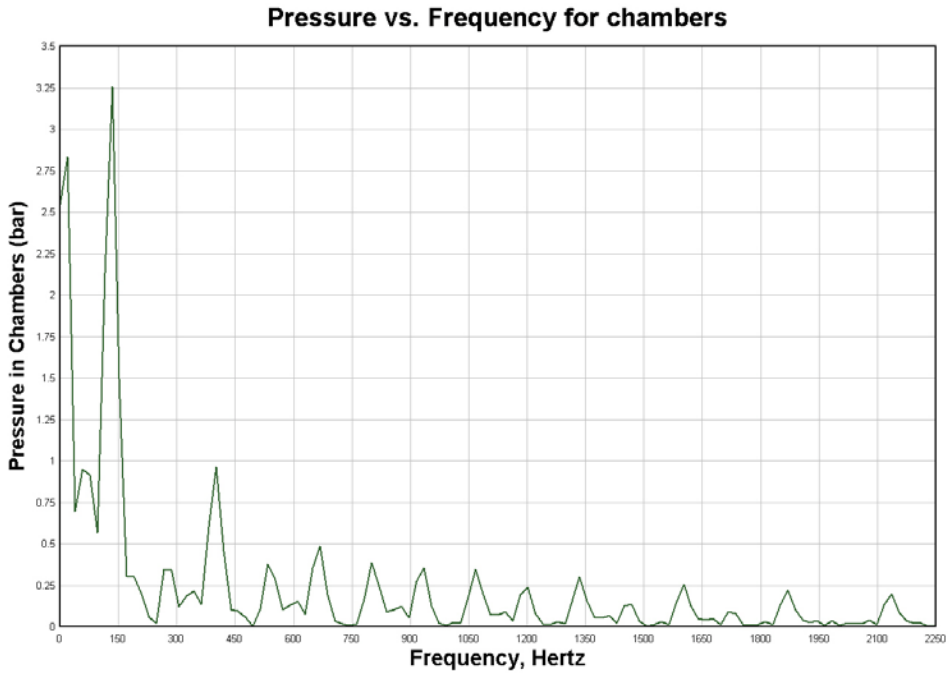


Figure 88: FFT from pressure pulsation of one chamber from model with triangular grooves

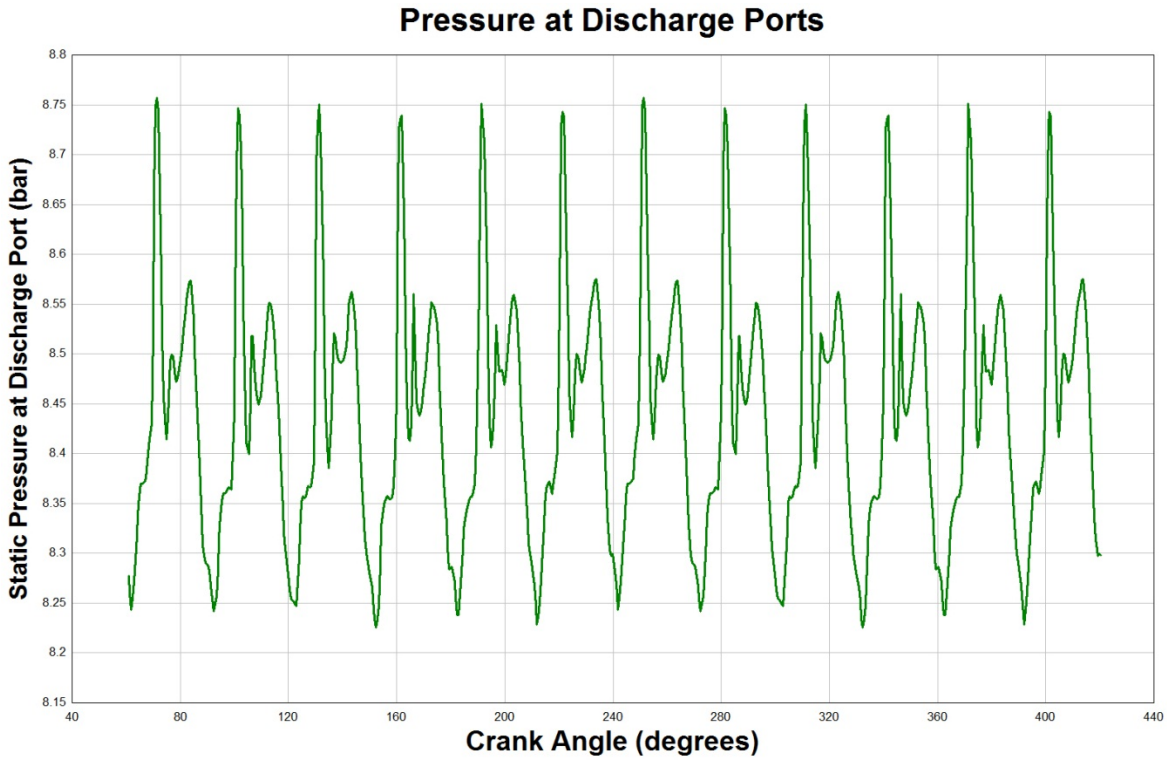


Figure 89: Pressure pulsation at discharge ports from model with triangular grooves

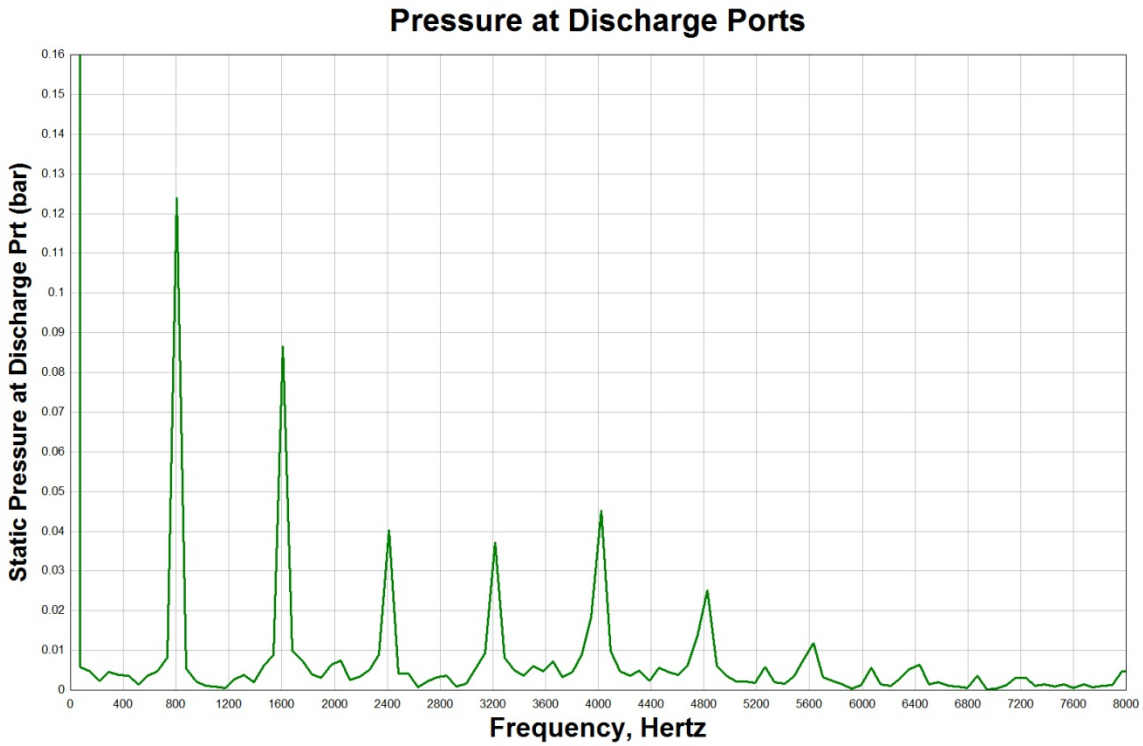


Figure 90: FFT from pressure pulsation at discharge ports from model with triangular grooves

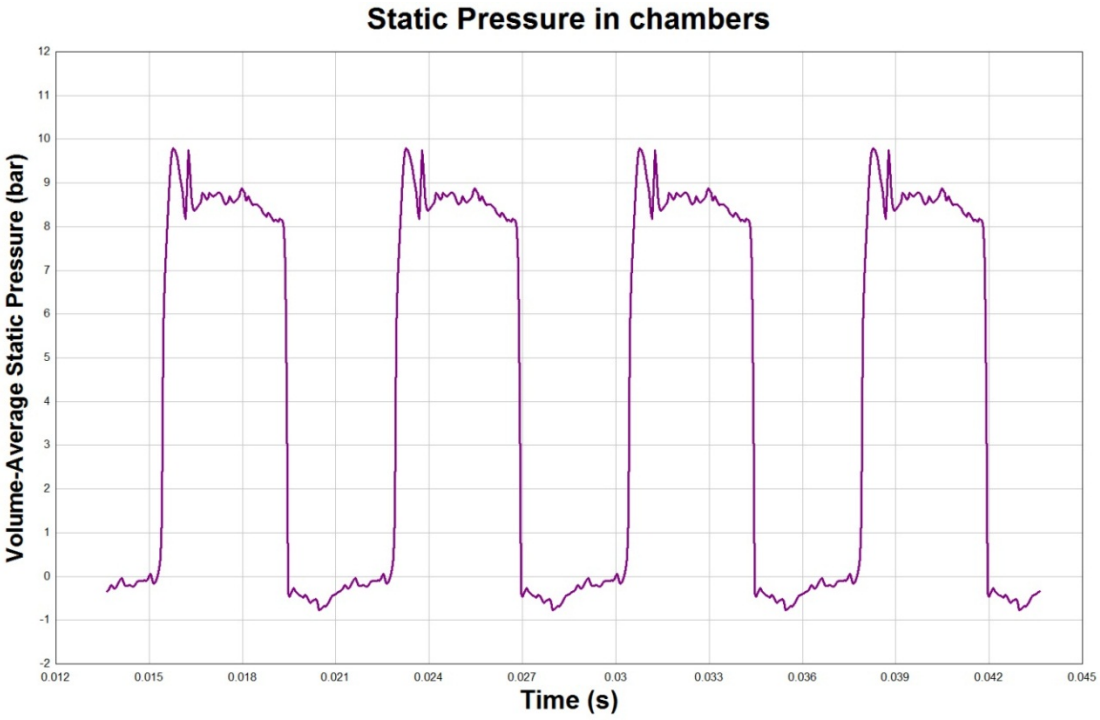


Figure 91: Pressure pulsation of one chamber from model with revised port design and intake grooves

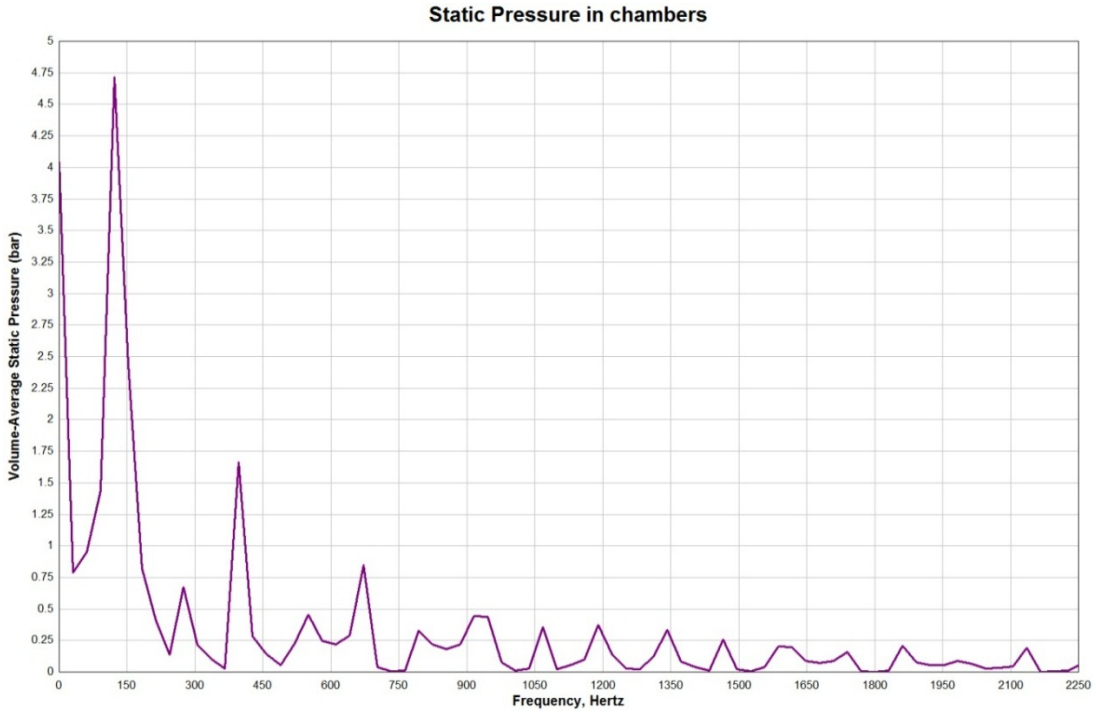


Figure 92: FFT from pressure pulsation of one chamber from model with revised port design and intake grooves

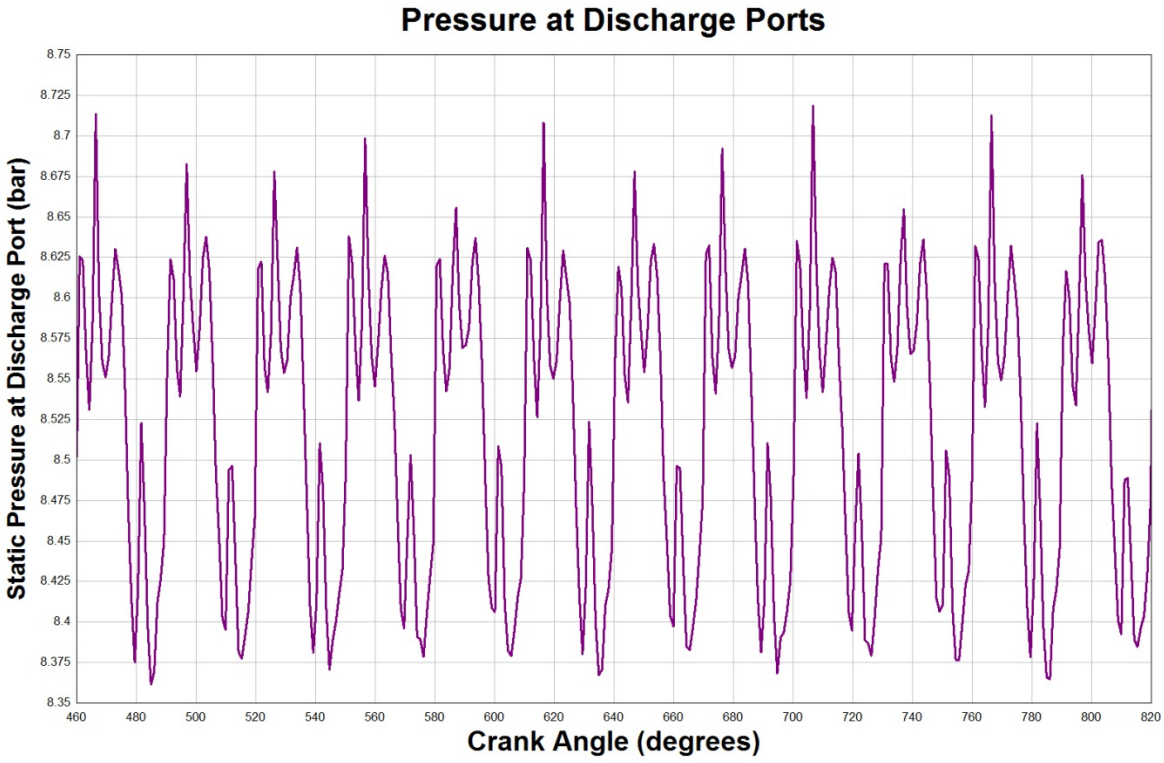


Figure 93: Pressure pulsation at discharge ports from model with revised port design and intake grooves

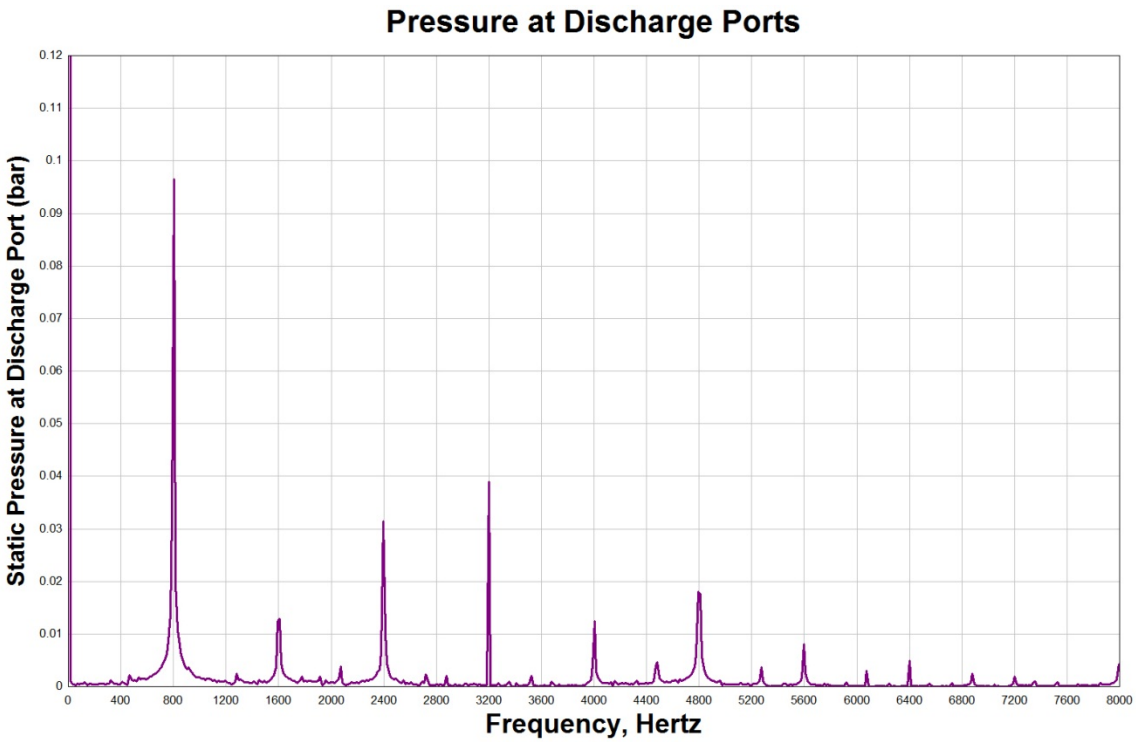


Figure 94: FFT from pressure pulsation at discharge ports from model with revised port design and intake grooves

It is evident that pressure pulsations are reduced significantly in comparison to the initial design. However, the FFT analysis does not quantify which of the other cases has more acceptable pressure pulsations. Therefore, another method is used in order to quantify pressure pulsations for balanced pump in order to compare different design iterations.

The CFD simulation predicts pressure variations at the pump chambers. The pressure rises rapidly when the chamber is closed after passing inlet zone and fluctuates significantly when the enclosed volume reaches discharge port or metering groove. The pressure fluctuations associated with one volume, are immediately followed by (in principle) the same pressure fluctuation associated with the next volume and so forth. The leftmost blue trace of **Figure 95** displays the pressure variation calculated for the initial design of balanced pump. For clarity, only the top of the pressure excursion is shown here.

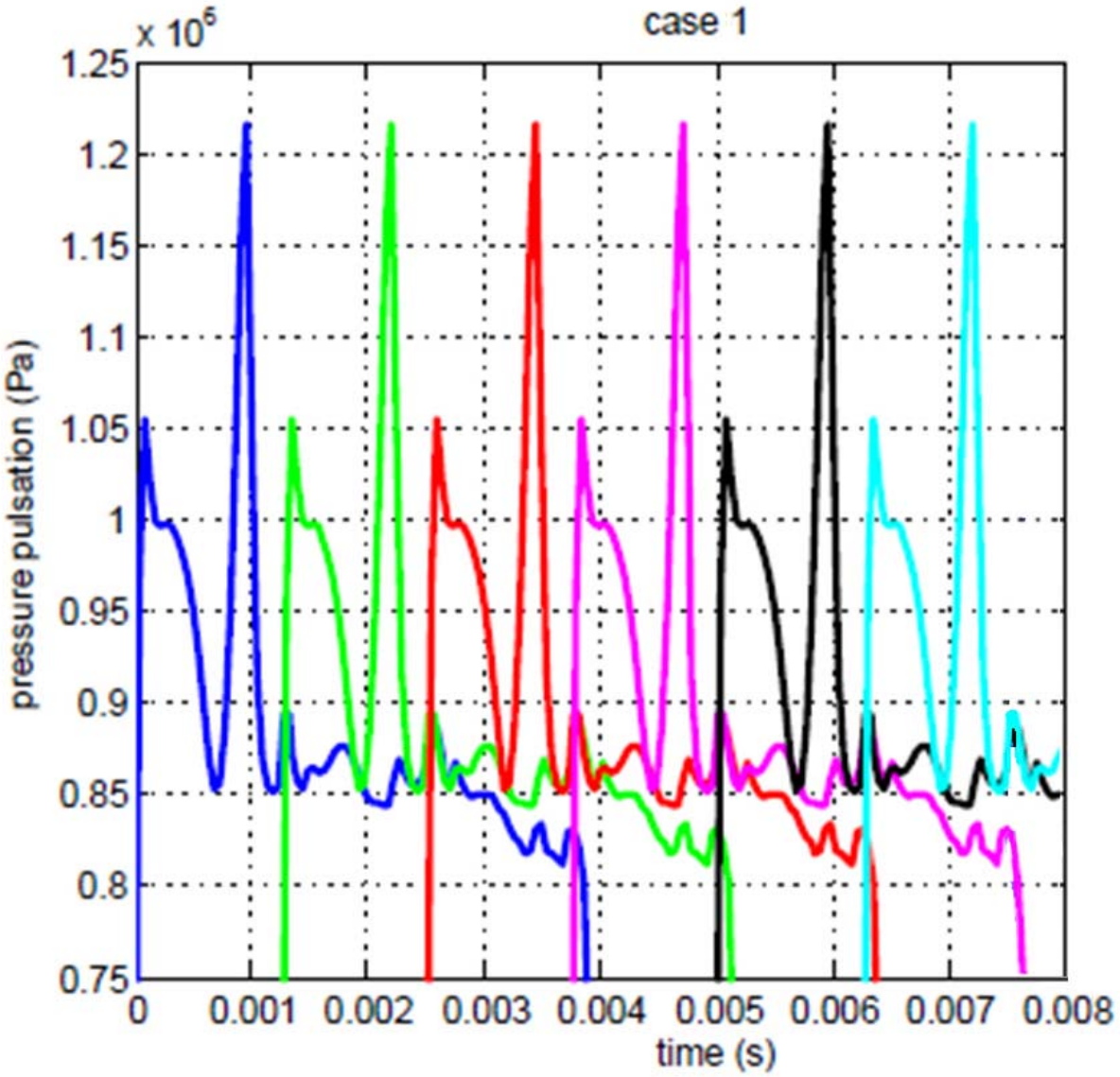


Figure 95: Pressure pulsation of chambers from initial design of balanced pump

The following green trace was obtained by time-shifting the blue trace by the amount corresponding to the vane-passing period, which was calculated as 60 (sec/min)/4000 (rpm)/12(vanes) = 0.00125 s. The figure shows a number of traces, each shifted with respect to the previous one by the same period. It was assumed that the upper envelope of the collection of the pressure traces would closely approximate the pressure pulsations in the vicinity of the discharge port.

The same calculations were repeated for cases 2, 3, 4 and 5. **Figure 96** displays all the upper pressure envelopes superimposed. The range limits are annotated in **Figure 96**.

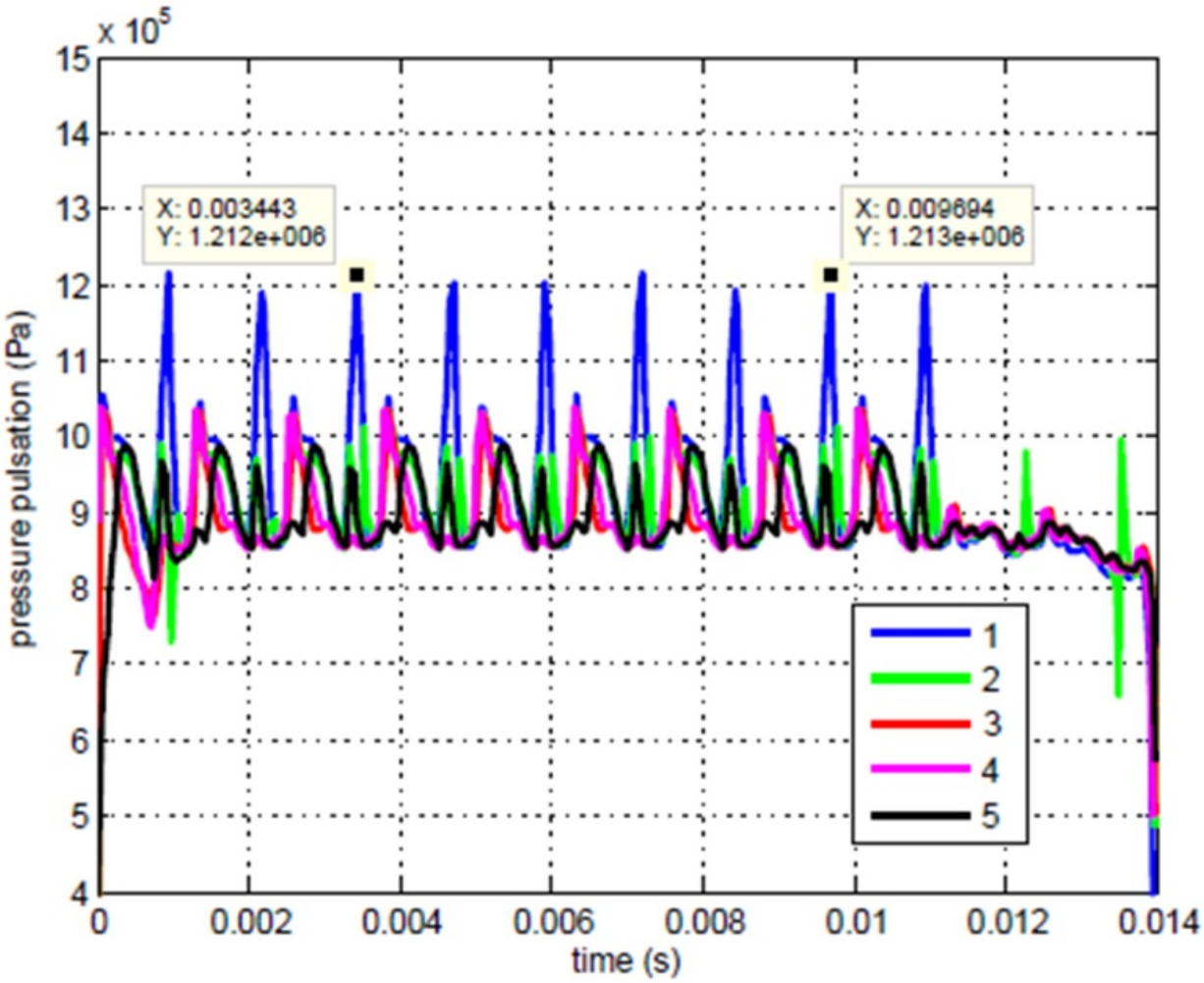


Figure 96: Pressure pulsations of chambers superimposed, valid part is between marks

To facilitate a review of the presented data, an expanded part of **Figure 96** is displayed in **Figure 97**.

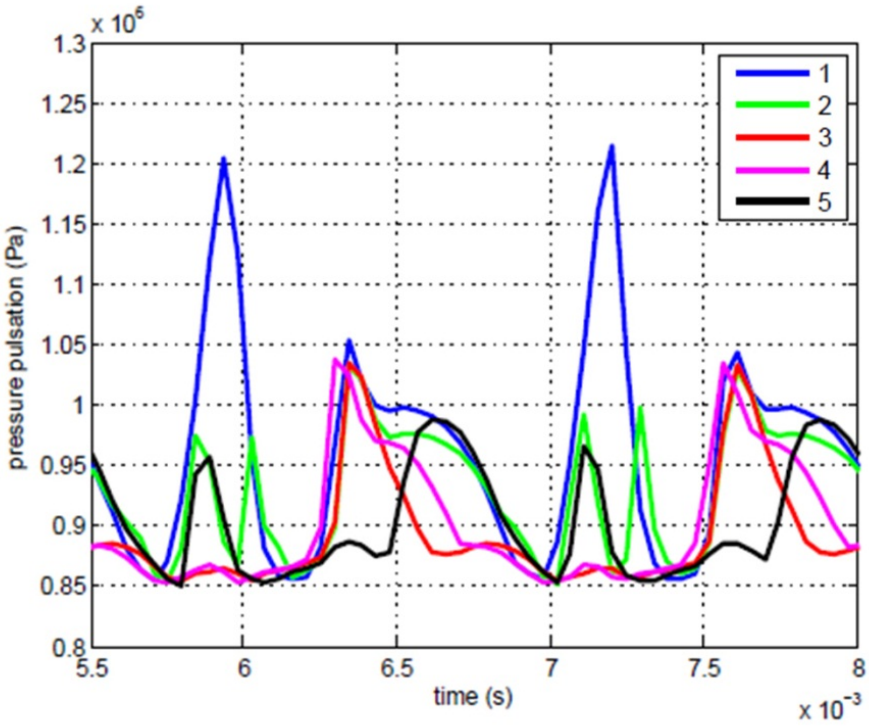


Figure 97: Pressure variations at pumping chamber used for RMS calculations: Curve 1 – initial design; Curve 2 – revised porting design; Curve 3 – trapezoidal grooves; Curve 4 – triangular grooves; Curve 5 – revised ports with inlet grooves

Root mean square (rms) values of the pressure variations from **Figure 96** were found to be:

Curve 1 - Initial design of balanced pump - 90114 Pa.

Curve 2 - Revised porting design of balanced pump - 52953 Pa.

Curve 3 - Pressure pulsations from initial balanced pump model with trapezoidal grooves - 47764 Pa

Curve 4 - Pressure pulsations from initial balanced pump model with triangular grooves - 53483 Pa.

Curve 5 -Pressure pulsations from balanced pump model with revised porting design and inlet grooves – 45565 Pa.

All above results are obtained by using the standard deviation function in Matlab.

Similar results are obtained with analysis of pressure variations curve from CFD analysis at the same location as pressure transducer, which is depicted on **Figure 98**.

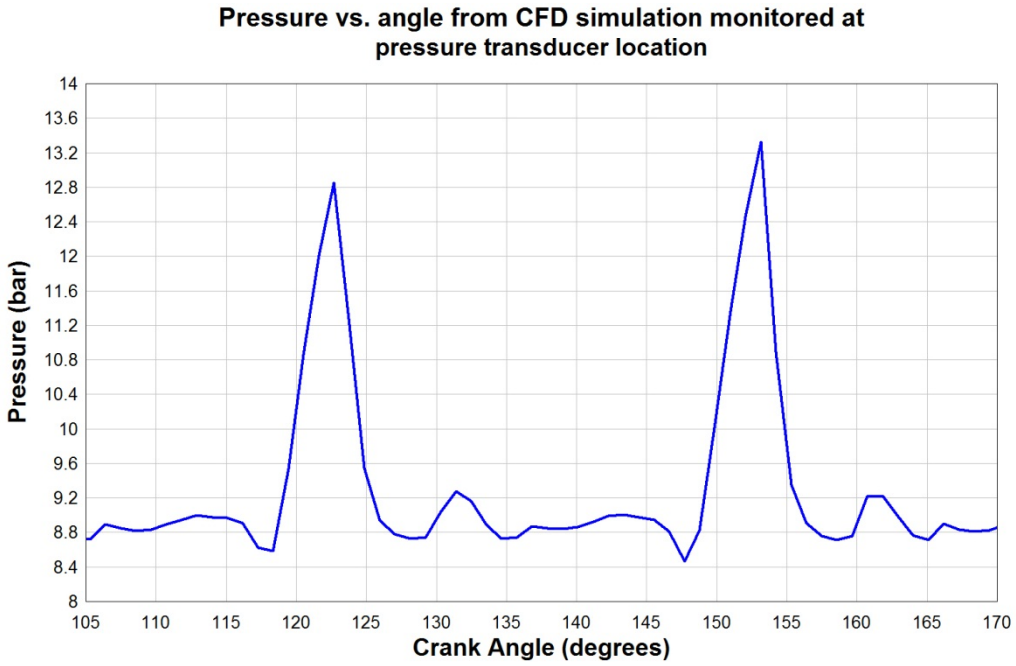


Figure 98: Pressure variations from CFD simulation, at the pressure transducer location for initial model of balanced pump

Summarized results for RMS values of the pressure variations for pumping chambers and CFD results measured at pressure transducer location are given in **Table 4**.

Table 4: RMS values for pressure variations for different design iterations

	RMS for pressure variations (Pa)
Initial design of balanced pump, CFD results measured at pressure transducer location	91885
Initial design of balanced pump for pumping chamber	90114
Revised porting design for pumping chamber	52953
Initial design with trapezoidal grooves for pumping chamber	47764
Initial design with triangular grooves for pumping chamber	53483
Revised porting design and inlet grooves for pumping chamber	45565

From above table, it can be concluded that there is negligible difference between RMS values for pressure variations measured at rotating chamber and pressure variations taken from CFD simulation at the pressure transducer location.

This analysis shows that all of above mentioned design iterations have lower pressure pulsations than the initial model for the balanced pump. All of revised models have similar characteristics in terms of mean pressure for their pressure pulsations. However, the best result is obtained with the last model, which is the balanced pump model with redesigned ports and grooves added at the intake port.

Chapter 8 - Conclusion and Suggestions for Future Work

8.1. Summary and Conclusions

Several approaches are used to design a balanced pump and to understand the advantages and disadvantage of each approach and adopt the best approach to meet the objective of decreasing pressure pulsations for a balanced pump, which could increase the expected pump lifetime and could decrease noise and vibration.

As a development start, one-dimensional approach was used in order to determine the basic dimensions of a balanced pump. This approach is purely theoretical and takes into account the basic displacement of a balanced pump (rotor diameter, eccentricity and cam height) for a flow calculation. This approach does not take into account vane thickness or fluid properties, such as viscosity or density. Thus, this approach is considered adequate for the initial phase of balanced pump design. One-dimensional approach does not give information about pressure pulsations of a balanced pump, nor takes into account any of the possible leakage paths. Advantage of this approach is that the basic dimensions of a balanced pump can be determined fast and flow performance can be determined within reasonable accuracy. Difference in flow performance between one-dimensional approach and experimental data is about 10%. Determining of the basic dimensions of a balanced pump is important for establishing overall packaging requirement for a balanced pump into transmission assembly.

Another approach which is used in this research is the two-dimensional approach, which is built into AmeSim program. In this approach, the basic geometry of a balanced pump, including port shapes is accounted for. Port shape generated in CAD system is input into AmeSim as look up tables. AmeSim simulation gives closer results for flow compared to experimental data than one-dimensional approach. Difference in flow performance between two-dimensional approach and experimental data is about 7.3%. Difference in flow performance between two-dimensional approach and experimental data could be further reduced with implementing leakages into AmeSim model. The simulation can be completed

in a time frame of several minutes providing predictions for flow and pressure pulsations for any angular speed. However, there are few limitations to this approach: Aeration and cavitation are not modeled either and pressure contours throughout 3D geometry cannot be visualized or thoroughly studied.

Last approach which is used in this study is CFD analysis, which takes into account full geometry and gives best description of pump behavior. In contrast to the 1D and 2D models described above, CFD analysis provides a detailed description of the velocity and pressure field throughout the domain. CFD modeling takes into account mixture properties of fluid, which includes viscosity, density, aeration and cavitation. Thus, CFD modeling can allow for the prediction of the effects of aeration of the fluid and leakages between chambers and is also capable of predicting cavitation.

Many CFD analyses were conducted in order to investigate the in depth behavior and performance of a transmission balanced oil pump. In order to check the effect of grid density to results of CFD simulations, several CFD models with different grid densities were analyzed, until results become identical with further grid refinement. Similar study was carried to examine the effect of using different simulation time step. Initially, the time step was set to yield a rotation of 2.5 degrees per time step. To assess its effect, the time step was decreased to yield a rotation of 1.09 degrees per time step, which allowed the mesh movement to exactly match the node to node at the interface for each consecutive step. Further time step reduction did not give different results for pressure or flow. Thus, the time step which allowed the mesh movement to exactly match the node to node was adopted for all future analyses. In terms of turbulence model for CFD simulations, the standard k-ε model was used. During the initial research, another turbulence model was also considered (the *k*- *ω m*odel). There were no differences in results between those two turbulence models. Therefore, all other consecutive simulations were conducted using the standard k-ε model.

Numerous CFD analyses were conducted in order to investigate in depth the behavior and performance of a transmission balanced oil pump. Initial design of model for a balanced pump was developed and validated by experiments. Although the pressure curves (experimental and CFD) were not identical (some variations exist), the CFD would predict relatively well the maximum peaks in the individual chambers and discharge channel. The difference between experimental data and CFD analysis for pressure pulsation peaks at pumping chambers is about 1.6 % of absolute value, which is considered to be excellent experimental vs. computational correlation. Difference between experimental data and CFD analysis for pressure pulsation peaks at discharge channel is about 0.5% of absolute value. The difference between experimental data and CFD analysis for flow and pressure pulsations could be further reduced with adjusting clearances and aeration in the model.

To meet the objective of this research, which is reduction of the pressure pulsations, many different design iterations were analyzed, from which several are presented in this research. The first design iteration was modification of ports; to move them close to each order in order to reduce pre-compression of oil which led to a pressure pulsation reduction of about 24%. There were experiments with enlarging shadow ports and connection channels which did not give satisfactory results and are not presented in this research. The next design iteration was implementation of metering grooves (trapezoidal and triangular), which decreased pressure pulsations compared to initial design by about 24%. The precise shape of the grooves, whether they are triangular or trapezoidal, did not affect the overall flow produced by the pump. The length of metering grooves does affect pressure pulsations of the pump. As a final step, metering grooves were implemented at intake ports of a balanced pump, in order to reduce sudden closing of a pumping chamber. This modification reduced pressure peaks at intake port by 7.1%. During CFD analyses, pressure was monitored at the pumping chamber, while rotating. This monitor, along with monitor of pressure pulsations at discharge port were plotted and presented in this research. For quantification of pressure pulsations, two different paths are taken: FFT

of a pressure pulsation curve and RMS representation of its value. Comparing results from both approaches (from FFT method and RMS method); it is evident that design with modified ports and metering grooves at inlet ports gave best results for the pressure pulsations reduction for above presented design iterations.

The presented design iterations are only small part of possible combination of intake and discharge port design for a balanced pump. Additional CFD simulations could be conducted to further optimize the balanced pump and reduce pressure pulsations.

8.2. Suggestions for Future Work

The operating point for the balanced pump was chosen to be outside of the cavitation regime. However, cavitation is present in every pump when operating at high rotation rates. A large percentage of the damage and failures of oil pumps is because of cavitation. Thus, a thorough study of cavitation is suggested for future work. This would require including additional parameters in the CFD model such as oil vapor flow, cavitation model and energy balance. The cavitation study should explore a range of cavitation models to determine which one is the most suitable for this particular application. The effect of the cavitation for different metering grooves can be investigated and erosion can be quantified. Ports, rotor and channels could be optimized to have better cavitation performance of the pump. During operation in cavitation regime, the pump generates high noise level and excessive vibrations. The pump should be instrumented with pressure transducers, flow meter, strain gauges, microphone and accelerometer, to have full picture of pump performance during cavitation regime. Relationship between operating speed, noise and vibrations could be established. Noise and vibrations can be quantified for full spectrum of pump operation regime. Durability of a balanced pump could be thoroughly studied during the cavitation regime, with erosion quantification and location during the operation.

In a balanced pump, vanes are being forced radially to the cam ring by a hydraulic pressure obtained from the outlet of the transmission system. Fluid domains of the back side of vane chambers are connected with grooves with geometries which varies at different designs of a transmission pump. The tested pump for purposes of this research had pressure measurements at above mentioned grooves and they were in excellent agreement with experimental data. However, effect of groove shape modification for the back side of vane chambers is not investigated. This effect could be thoroughly studied in the future in order to obtain complete understanding of vane forces during pump operation.

References

A. Guiffrida, R. Lanzafame (2004). Cam shape and theoretical flow rate in balanced vane pumps, Department of Industrial and Mechanical Engineering, Catania, Italy, Elsevier

A. M. Karmel (1986). A Study of the Internal Forces in Variable Displacement Vane Pump, General Motors Laboratories. Journal of Fluids Engineering

A. M. Karmel (1987). Modeling and Analysis of the Dynamics of a Variable-Displacement Vane-Pump with a Pivoting Cam, General Motors Research Laboratories, Warren, Michigan, American Control Conference

Amesim Reference Manual V10 (2009). LMS International

Ansys Fluent (2011). Theory Guide

Automatic Transmission Hydraulic Pumps Design Application Manual (2001). Tesma International Inc.

B. L. Jones, D. N. Johnston and D. K. Longmore (1998). Simulation of Suction Flow Ripple in Power Steering Pumps, University of Bath, SAE 982023

C. Hirsch (2007). Numerical Computation of Internal and External Flows, Elsevier

C. Takemori, E. Paladino (2006). Numerical Simulation of Oil Flow in a Power Steering Pump, DHB Componentes Automotivos, Ansys Conference

D. Manolakis, Vinay K. Ingle (2011). Applied Digital Signal Processing, Cambridge

D. Staley, B. Pryor and K. Gilgenbach (2007). Adaption of a Variable Displacement Vane Pump to Engine Lube Oil Applications, GM Corporation, SAE

David C. Wilcox (1994). Turbulence Modeling for CFD, DCW Industries

F. Brusiani, G. M. Bianchi and M. Costa (2009). Analysis of Air Cavitation Interaction Inside a Rotary Vane Pump for Application on Heavy Duty Engine, University of Bologna, SAE

F. van der Sluis (2003). A New Pump for CVT Applications, Bosch Group, SAE 2003-01-3207

G.K. Batchelor (2000). An introduction to fluid dynamics, Cambridge University Press

H. Li, F. Kelecy, A. Egelja-Maruszewski, S. Vasquez (2008). Advanced Computational Modeling of Steady and Unsteady Cavitating Flows, International Mechanical Engineering Congress and Exposition IMECE2008-67450, ASME

H. Oh (2010). Computational Fluid Dynamics, Intech

H. Sasaki, N. Inui, Y. Shimada and D. Ogata (2008). Development of High Efficiency Powder Metal Internal Gear Pump Rotor (Megafloid Rotor), SEI Technical Review Number 66

J. Anderson (1995). Computational Fluid Dynamics - The Basics with Applications, McGraw-Hill

J. Blazek (2001). Computational Fluid Dynamics - Principles and Applications, Elsevier

J. Ferziger, M. Peric (2002). Computational Methods for Fluid Dynamics, 3rd ed., Springer

J-S Jang, K-H Kim, M-R Cho and D-C Han (2002). The characteristics of pressure ripple in variable displacement vane pumps: comparison between theory and experiment, School of Mechanical and Industrial Engineering, KyungHee University, Kyonggi-do, Korea, Proc Instn Mech Engrs Vol 216 Part A: Journal of Power and Energy

K. Hattori, H. Suzuki and J. Hasegawa (1987). Design Method of Small-Ripple Vane Pump, Toyota Central Res. & Development Labs. Inc. SAE 871681

K. Manssouri (2003). Modeling & Validation of Electrically Controlled Hydraulic Power Assist Pump – FORD, Report number: DCT.2003.44

L. Lessa, R. Ataides, C. Siqueira, M. P. Kessler, L. Foley, A. Johri, A. Khondge (2006). Numerical Analysis of Oil Flow and Noise in a Power Steering Pump, DHB Automotive Parts – Brazil, Engineering Simulation and Scientific Software (ESSS) - Brazil, Ansys INC, International Ansys Conference

M. A. Kluger, D. R. Fussner (1996). A Performance Comparison Various Automatic Transmission Pumping Systems SAE 960424, SAE

M. Cho and D. Han (1998). Vane Tip Detachment in Positive Displacement Vane Pump, KSME International Journal Volume 2

M. Rundo, N. Nervegna (2007). Geometry Assessment of Variable Displacement Vane Pumps, The Fluid Power Research Laboratory, Politecnico di Torino, Italy, ASME

Methodology (2011). Star-CD, CD-Adapco

P. Gamez-Montero, E. Macià (2000). Fluid Dynamic Behavior of Gerotor Pump, Technical University of Catalonia, Barcelona. Spain, Proc. of 1st FPNI-PhD Symp., Hamburg

P. Kundu, I. Cohen (2002). Fluid Mechanics, Academic Press

P. Wesseling (1991). Principles of Computational Fluid Dynamic, Springer

S. Hain-Wurtenberger (2007). Simulation of Cavitating Flow in Balanced Vane Pump, 3rd European Automotive CFD Conference, EACC

S. Manco, N. Nervegna and M. Rundo (2004). Displacement vs. Flow Control in IC Engine Lubricating Pumps, SAE 2004-01-1602, SAE Congress, Detroit

S. Manco, N. Nervegna and M. Rundo (2004). Modeling and Simulation of Variable Displacement Vane Pumps for IC Engine Lubrication, Politecnico di Torino, 2004-01-1601, SAE

S. Mancò, N. Nervegna, M. Rundo Gerotor (1998). lubricating oil pump for IC engines, Politecnico di Torino – Italy, SAE International, San Francisco

S. Patankar (1980). Numerical Heat Transfer and Fluid Flow, McGraw-Hill

Sliding vane pump (2002). United States Patent US 6,497,557 B2

T. Cebeci, J.RShao, F. Kafyeke E. Laurendeau (2005). Computational Fluid Dynamics for Engineers, Springer

T. Chung (2010). Computational Fluid Dynamics 2nd Edition, Cambridge

T. Singh (1991). Design of Vane Pump Suction Porting to Reduce Cavitation at High Operation Speeds, General Motors Powertrain Division, SAE

Y. Inaguma, A. Hibi (2005). Vane Pump Theory For Mechanical Efficiency, Department of Steering Engineering, Toyoda Machine Works Ltd, Okazaki, Japan, SAE

Zanetti Rocha, L., Johnston, D. N. Gerges (2010). Flow Ripple Reduction in Power Steering Hydraulic Pumps, S. N. Y., University of Bath

Appendix A

Analysis of volumes variation

The volumes derivatives of the chambers between two consecutive vanes are difficult to evaluate, essentially because of the vane thickness. Vane thickness for transmission pump presented in this research is 1 mm. Therefore, an approximation regarding the shape of the vane is introduced (Ref: A. Guiffrida & R. Lanzafame, 2004). Considering that the vane touches the cam contour almost always with only one edge, the little nail of oil at the tip of the vane (represented by the triangle shaped area $P_1P_2P_3$ on **Figure 99**) may be neglected.

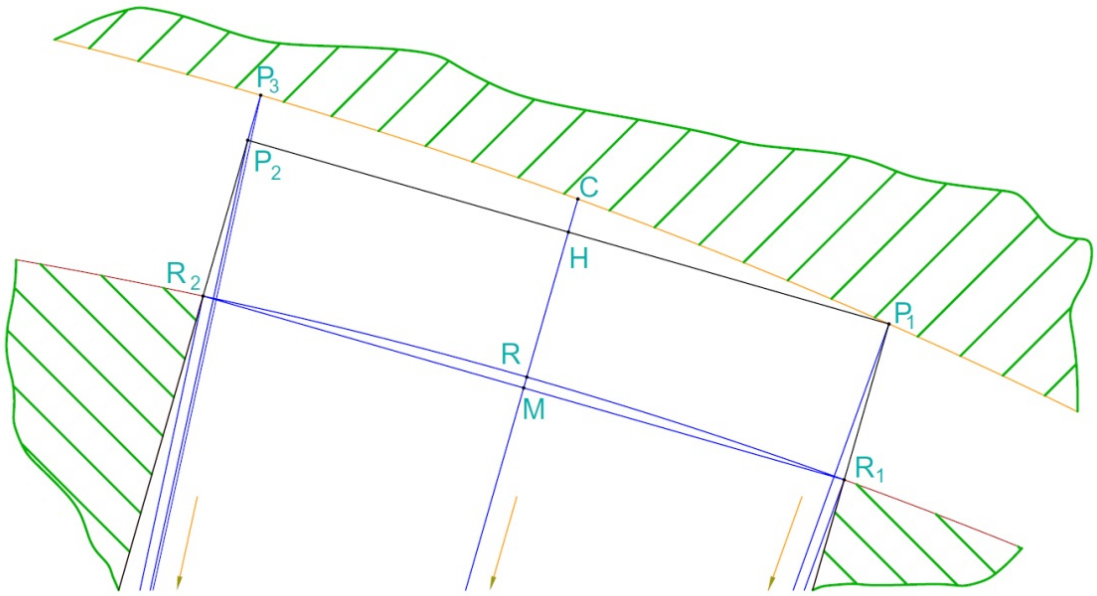


Figure 99: Vane touching the inner contour of the cam ring in P1

Thus, the tip of the vane is considered to be in contact with cam. With reference to the configuration shown on the left of **Figure 100**, it is sufficient to consider the segments P_1R_1 and P_3R_2, portions of vanes sides emerging out from the rotor. Both these segments depends on ϑ. The chamber between these two consecutive vanes occupies the angular position $\vartheta - \frac{\pi}{N}$.

The segments P_1R_1 and P_3R_2 at the left of **Figure 100** have moved to $P_{1n}R_{1n}$ and $P_{3n}R_{2n}$ respectively, at the right of the same figure (the subscript n indicates the new current position). The prolongations of these segments are always tangent to a circumference whose diameter is equal to t_v, vane thickness. This circumference is concentric to the rotor.

Figure 101 shows the foregoing configurations, opportunely superimposed in order to evaluate the variations in area, then in volume if the axial depth of the cam ring is taken into account.

One can easily evaluate the volume of fluid entering (zone with hexagonal net) a chamber between two consecutive vanes. Considering the segment P_1J_1, tangent to a circumference whose diameter is equal to t_v, the area uncovered during its movement is equal to (Ref: A. Guiffrida & R. Lanzafame, 2004)

$$\mathrm{d}A_{in\_P}\left(\vartheta-\frac{\pi}{\mathrm{N}}\right)=\overline{OJ_1}\cdot\overline{J_1T_1}+\frac{1}{2}\cdot\overline{T_1P_1}^2\cdot\mathrm{d}\vartheta$$

$$=\frac{t_\mathrm{v}}{2}\left(\frac{t_\mathrm{v}}{2}\cdot\frac{\mathrm{d}\vartheta}{2}\right)+\frac{1}{2}\cdot\left[\left(\overline{P_1J_1}(\vartheta)\right)-\left(\frac{t_\mathrm{v}}{2}\cdot\frac{\mathrm{d}\vartheta}{2}\right)\right]^2\cdot\mathrm{d}\vartheta$$

and represents the area of the closed mistiline $OJ_1P_1P_{1n}J_{1n}O$ as reported in **Figure 100**. Of course, the segment P_1J_1 is always perpendicular to the segment OJ_1. If the following equation, representing the area of the closed mistiline $OJ_1R_1R_{1n}J_{1n}O$

$$\mathrm{dA}_{\mathrm{in\_R}}\left(\vartheta-\frac{\pi}{\mathrm{N}}\right)=\overline{OJ_1}\cdot\overline{J_1T_1}+\frac{1}{2}\cdot\overline{T_1R_1}^2\cdot\mathrm{d}\vartheta$$

$$=\frac{t_\mathrm{v}}{2}\left(\frac{t_\mathrm{v}}{2}\cdot\frac{\mathrm{d}\vartheta}{2}\right)+\frac{1}{2}\cdot\left[\left(\overline{R_1J_1}(\vartheta)\right)-\left(\frac{t_\mathrm{v}}{2}\cdot\frac{\mathrm{d}\vartheta}{2}\right)\right]^2\cdot\mathrm{d}\vartheta \qquad \textbf{(8)}$$

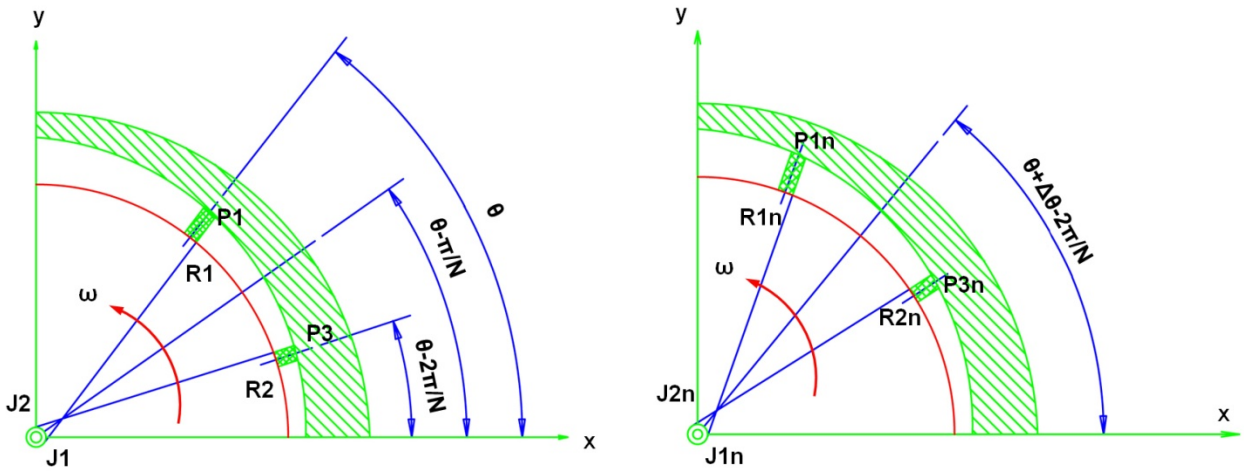


Figure 100: Positions of two consecutive vanes and next configuration after a Δθ rotor rotation

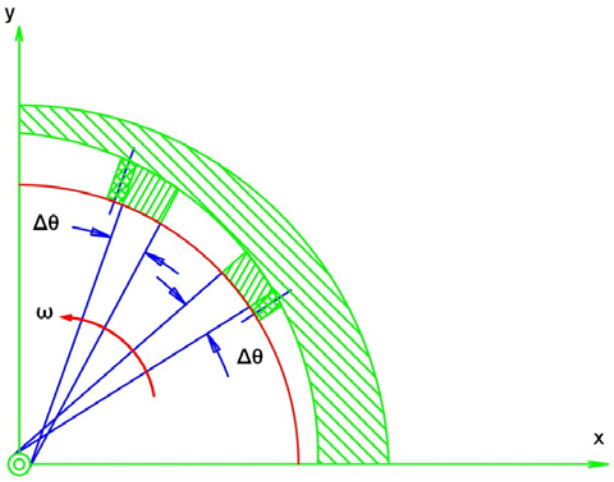


Figure 101: Volumes of fluid entering and exiting a chamber between two consecutive vanes

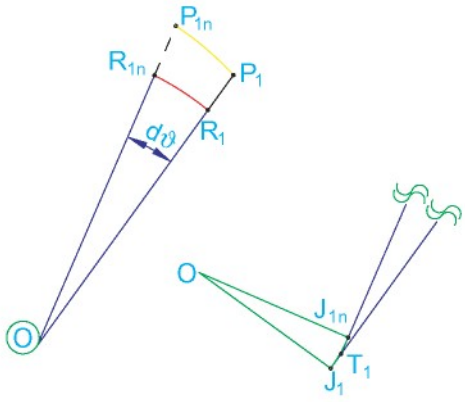


Figure 102: Area uncovered during the vane movement and its enlargement close to the center of the rotor

is subtracted to **Eq. (16)**, it is possible to come to following equation, taking the vane width into account.

$$\mathrm{d}V_{\mathrm{in}}\left(\vartheta-\frac{\pi}{N}\right)=\frac{1}{2}\cdot b\cdot\left[\left(\overline{P_1J_1}(\vartheta)\right)^2-\left(\overline{R_1J_1}(\vartheta)\right)^2\right]\cdot\mathrm{d}\vartheta \qquad \textbf{(9)}$$

Such formulations are significant if infinitesimals of order greater than the first are neglected. Analogously, the volume of fluid exiting (zone with quadrate net in **Figure 100**) the same chamber is equal to

$$\mathrm{d}V_{\mathrm{out}}\left(\vartheta-\frac{\pi}{N}\right)=\frac{1}{2}\cdot b\cdot\left[\left(\overline{P_3J_2}\left(\vartheta-\frac{2\cdot\pi}{N}\right)\right)^2-\left(\overline{R_2J_2}\left(\vartheta-\frac{2\cdot\pi}{N}\right)\right)^2\right]\cdot\mathrm{d}\vartheta \qquad \textbf{(10)}$$

Thus, the volume variation for a chamber between two consecutive vanes occupying the angular positions ϑ and $\vartheta - 2\pi/N$ respectively is

$$\mathrm{d}V_{\mathrm{bv}}\left(\vartheta-\frac{\pi}{N}\right)=\mathrm{dV}_{\mathrm{in}}\left(\vartheta-\frac{\pi}{N}\right)-\mathrm{dV}_{\mathrm{out}}\left(\vartheta-\frac{\pi}{N}\right)$$

which, assembling **Eqs. (9)** and **(10)**, (Ref: A. Guiffrida & R. Lanzafame, 2004) leads to

$$\frac{\mathrm{d}V_{\mathrm{bv}}\left(\vartheta-\frac{\pi}{N}\right)}{\mathrm{d}\vartheta}=\frac{b}{2}\cdot\left[(\overline{P_1J_1}(\vartheta))^2-\left(\overline{P_3J_2}\left(\vartheta-\frac{2\cdot\pi}{N}\right)\right)^2\right]$$

since

$$\overline{R_1J_1}(\vartheta)=\overline{R_2J_2}\left(\vartheta-\frac{2\cdot\pi}{N}\right)=\mathrm{R_r}\cdot\cos\left(\arcsin\frac{\mathrm{t_v}}{2\cdot R_{\mathrm{r}}}\right)=R_{\mathrm{r}}\cdot\cos\frac{\chi}{2}$$

These segments do not depend on the current angular position ϑ. Thus, two segments *(P_1J_1* and *P_3J_2)* are sufficient for the calculation of the variation of the volume of a chamber between two consecutive vanes.

Appendix B

User defined function for mesh deformation

```
# include "udf.h"
# include "dynamesh_tools.h"
# define noDEBUG
#if RP_DOUBLE
#  define REAL_FMT "%le"
#else
#  define REAL_FMT "%e"
#endif

extern real special_inner_radius(real z);

/* Outer contour shape */
enum define_shape
{
  circle, user_defined, special_circle, special_user_defined
}OUTER_SHAPE, INNER_SHAPE;

/* Number of vanes, core ID and gap ID */
static int N_VANE;

/* single core and single gap flag*/
static int MULTIPLE_CORE, MULTIPLE_GAP;

/* Core ID and gap ID */
static int * CORE_ID, * GAP_ID;

/* Rotating inner profile flag */
static int INNER_TIME_DEP_PROFILE;

/* Inner gear rpm */
static real RPM;

/* Outer circle radius, inner circle radius, half vane width, gap size and offset */
static real R_OUTER, R_INNER, VANE_WIDTH, GAP_SIZE, DELTA;

/* initial vane angle */
static real INIT_VANE_ANGLE;

/* Flag for vane spacing */
static int NON_EQUALLY_SPACED_VANE;

/* angle for sectors */
static real * ANGLE;

/* Number of layers */
int Num_Layers=20;      /* hard coded number for vane */

/************************************ User inputs end ***********************************************/
extern void find_intersection(real (*)[2], int , real, real, real, real, real *, real *);
```

```
extern void find_appp(real *, real *, real *, real, real, real, real (*)(real *, real *, real), real (*)(real *, real *,
real));
extern void find_app_rigid(real *, real *, real *, real);

enum shrink
{
  w_shrink, wo_shrink
};

struct shape
{
 enum define_shape shape_type;

 enum shrink shrink_y_or_n;

 real center[2];
 real radius;

 real (*data)[2];
 int number_of_data;

 int flag_time_dep_profile;

}pump_core_outer_shape, pump_core_inner_shape, pump_gap_outer_shape, pump_gap_inner_shape;

enum UDM
{
  UDM_sector
};

/* Allocate memory for 1d int array with dimention of n_x */
int * alloc_1d_int_array(int n_x)
{
                int *x;

                if(!(x = (int *) calloc(n_x, sizeof(int))))
                  {
                          Message0("Memory y allocation failure-aboring\n\n");
                          exit(0);
                         }

                       return x;
}

/* Allocate memory for 1d real array with dimention of n_x */
real * alloc_1d_real_array(int n_x)
{
                real *x;

                if(!(x = (real *) calloc(n_x, sizeof(real))))
                  {
                          Message0("Memory y allocation failure-aboring\n\n");
                          exit(0);
                         }

                       return x;
}
```

```
static real my_atan2(real y, real x)
{
 real result;

 result=atan2(y,x);
 if(atan2(y,x)<0)
  result = atan2(y,x) + 2*M_PI;

 return result;
}


/* Function used to calcuate op for the gap.  Please refer to note page 4 for the definition. */

static void f_Op_gap(real * xop, real * yop, real xa, real ya, real time, real sector)
{
 real alfa, h;

 alfa = time*RPM*M_PI/30+sector*2*M_PI/N_VANE;

 h=cos(alfa)*ya-sin(alfa)*xa;

 if(fabs(cos(alfa))>0.1)
  {
            *xop = 0.5*xa;
            *yop = (sin(alfa)*0.5*xa+h)/cos(alfa);
  }
 else
  {
    *xop = (cos(alfa)*0.5*ya-h)/sin(alfa);
            *yop = 0.5*ya;
  }

 return;
}

/* Function used to calcuate op.  Please refer to note page 3 for the definition. */

static void f_Op_core(real * x, real * y, real time, real sector)
{
 real alfa;
 int i;
 real total_angle=0.0;


 if(NON_EQUALLY_SPACED_VANE)
  {

   for(i=0; i<=sector; i++)
    {
                    if (i>0) total_angle +=ANGLE[i-1];
            }

   alfa = time*RPM*M_PI/30+total_angle+INIT_VANE_ANGLE*M_PI/180;
```

```
    *x = 0.5*VANE_WIDTH*(cos(alfa)+cos(alfa+ANGLE[(int) (sector+0.1)]))/sin(ANGLE[(int) (sector+0.1)]);
    *y = 0.5*VANE_WIDTH*(sin(alfa)+sin(alfa+ANGLE[(int) (sector+0.1)]))/sin(ANGLE[(int) (sector+0.1)]);
   }
  else
   {
    alfa = time*RPM*M_PI/30+sector*2*M_PI/N_VANE+INIT_VANE_ANGLE*M_PI/180;

    *x = 0.5*VANE_WIDTH*(cos(alfa)+cos(alfa+2*M_PI/N_VANE))/sin(2*M_PI/N_VANE);
    *y = 0.5*VANE_WIDTH*(sin(alfa)+sin(alfa+2*M_PI/N_VANE))/sin(2*M_PI/N_VANE);
   }


 return;
}

/* Rotate a profile around the origin */
static void rotate_profile(real (* orig_profile)[2], real (* final_profile)[2],int no_data, real theta)
{
 int i;
 real origin[2];

 origin[0]=origin[1]=0;
 for(i=0; i<=no_data-1; i++)
   {
    find_app_rigid(orig_profile[i], final_profile[i], origin, theta);
   }

}


/* Generic function to calcualte the distance between point a and c.  Please refer to note page 2.
  The inputs are points a, b, center of the circle (x0, y0), and the circle radius.  The h and H
  for this application are all from this function.
*/
static real f_h(real xa, real ya, real xb, real yb, real z_coord, struct shape * contour_shape, real time)
{
 if (contour_shape->shape_type==circle)
   {
    real t1, t2, a, b, c, t, x0, y0, radius;

    x0 = contour_shape->center[0];
    y0 = contour_shape->center[1];
    radius = contour_shape->radius;

    a=pow(xb-xa,2)+pow(yb-ya,2);
    b=2*(xa-x0)*(xb-xa)+2*(ya-y0)*(yb-ya);
    c=pow(xa-x0,2)+pow(ya-y0,2)-radius*radius;

    t1 = (-b+sqrt(b*b-4*a*c))/(2*a);
    t2 = (-b-sqrt(b*b-4*a*c))/(2*a);

    if (t1>0&&t2<0)
                t = t1;
    else if (t2>0&&t1<0)
                t = t2;
    else
    {
```

```
    Message("\nSomething wrong with f_h. t1=%7.2f t2=%7.2f", t1, t2);
    Message("\nxa=%11.3e ya=%11.3e xb=%11.3e yb=%11.3e x0=%10.3e y0=%10.3e radius=%10.3e",
xa, ya, xb, yb, x0, y0, radius);

    t = 1;
   }

   return t*sqrt((pow(yb-ya,2)+pow(xb-xa,2)));
  }
 else if (contour_shape->shape_type==user_defined)
  {
          real x, y;

    if(contour_shape->flag_time_dep_profile)
     {
       real (*final_profile)[2];

       if(!(final_profile=(real (*)[2]) calloc(contour_shape->number_of_data, 2*sizeof(real))))
        {
                    Message0("\nMemory allocatoin failure-aborting!!\n");
                    exit(0);
             }
       rotate_profile(contour_shape->data, final_profile, contour_shape->number_of_data,
time*RPM*M_PI/30);

       find_intersection(final_profile, contour_shape->number_of_data, xa, ya, xb, yb, &x, &y);

       free(final_profile);
           }
    else
     {
                    find_intersection(contour_shape->data, contour_shape->number_of_data, xa, ya, xb,
yb, &x, &y);
      }


    if(contour_shape->shrink_y_or_n==w_shrink)
     {
        return sqrt(pow(y-ya,2)+pow(x-xa,2))-GAP_SIZE;
           }
    else
     {
        return sqrt(pow(y-ya,2)+pow(x-xa,2));
           }
         }
 else if (contour_shape->shape_type==special_circle)
  {
   real t1, t2, a, b, c, t, x0, y0, radius;

   x0 = 0;
   y0 = 0;

   radius = special_inner_radius(z_coord);
   /*radius = 1;  Hardwire option to get rid of the above function */

   a=pow(xb-xa,2)+pow(yb-ya,2);
   b=2*(xa-x0)*(xb-xa)+2*(ya-y0)*(yb-ya);
```

```
    c=pow(xa-x0,2)+pow(ya-y0,2)-radius*radius;

    t1 = (-b+sqrt(b*b-4*a*c))/(2*a);
    t2 = (-b-sqrt(b*b-4*a*c))/(2*a);

    if (t1>0&&t2<0)
            t = t1;
    else if (t2>0&&t1<0)
            t = t2;
    else
    {
     Message("\nSomething wrong with f_h. t1=%7.2f t2=%7.2f", t1, t2);
     Message("\nxa=%11.3e ya=%11.3e xb=%11.3e yb=%11.3e x0=%10.3e y0=%10.3e radius=%10.3e",
xa, ya, xb, yb, x0, y0, radius);

     t = 1;
    }

    return t*sqrt((pow(yb-ya,2)+pow(xb-xa,2)));
          }
  else if (contour_shape->shape_type==special_user_defined)
   {
          real x, y, radius;

    if(contour_shape->flag_time_dep_profile)
     {
       real (*final_profile)[2];

       if(!(final_profile=(real (*)[2]) calloc(contour_shape->number_of_data, 2*sizeof(real))))
        {
                    Message0("\nMemory allocatoin failure-aborting!!\n");
                    exit(0);
             }
       rotate_profile(contour_shape->data, final_profile, contour_shape->number_of_data,
time*RPM*M_PI/30);

       find_intersection(final_profile, contour_shape->number_of_data, xa, ya, xb, yb, &x, &y);

       free(final_profile);
          }
    else
     {
                    find_intersection(contour_shape->data, contour_shape->number_of_data, xa, ya, xb,
yb, &x, &y);
     }


    if(contour_shape->shrink_y_or_n==w_shrink)
     {
        radius = special_inner_radius(z_coord);
        return sqrt(pow(y-ya,2)+pow(x-xa,2))-GAP_SIZE+radius;
          }
    else
     {
        radius = special_inner_radius(z_coord);
        return sqrt(pow(y-ya,2)+pow(x-xa,2))+radius;
          }
```

```
        }
  else
    {
                Message0("\nwrong type-aborting!!!\n");
                exit(0);
        }
}

static real f_h_core(real * op, real *ap, real time)
{
 return f_h(op[0], op[1], ap[0], ap[1], ap[2], &pump_core_inner_shape, time);
}

static real f_H_core(real * op, real *ap, real time)
{
 real not_used_var=0;
 return f_h(op[0], op[1], ap[0], ap[1], not_used_var, &pump_core_outer_shape, time);
}

static real f_h_gap(real * op, real *ap, real time)
{
 real not_used_var=0;
 return f_h(op[0], op[1], ap[0], ap[1], not_used_var, &pump_gap_inner_shape, time);
}

static real f_H_gap(real * op, real *ap, real time)
{
 real not_used_var=0;
 return f_h(op[0], op[1], ap[0], ap[1], not_used_var, &pump_gap_outer_shape, time);
}

/* Functions used to calcuate a new position for vane pump core (for F6.2 only) */

DEFINE_GRID_MOTION(vane_pump_core, domain, dt, time, dtime)
{
  cell_t c;
  Thread *tc = DT_THREAD ((Dynamic_Thread *)dt);
  int n;
  Node *v;
  real ap[3], op[2], appp[2];

  /* set deforming flags */
  SET_DEFORMING_THREAD_FLAG (tc);

  begin_c_loop(c, tc)
    {
      c_node_loop(c, tc, n)
        {
          v = C_NODE(c, tc, n);
          if (NODE_POS_NEED_UPDATE(v))
            {
              NODE_POS_UPDATED(v);

                        ap[0]=NODE_X(v);
                        ap[1]=NODE_Y(v);
                        ap[2]=NODE_Z(v);
```

```
            f_Op_core(op, op+1, time-dtime, C_UDMI(c, tc, UDM_sector));
            find_appp(appp, ap, op, time-dtime, dtime, RPM*M_PI/30*dtime, f_h_core, f_H_core);

                  NODE_X(v)=appp[0];
                  NODE_Y(v)=appp[1];
          }
        }
      Update_Cell_Metrics (c, tc);
    }
  end_c_loop (c, tc);

}

/* Functions used to calcuate a new position for vane pump gap (for F6.2 only) */

DEFINE_GRID_MOTION(vane_pump_gap, domain, dt, time, dtime)
{
  cell_t c;
  Thread *tc = DT_THREAD ((Dynamic_Thread *)dt);
  int n;
  Node *v;
  real ap[3], op[2], appp[2];

  /* set deforming flags */
  SET_DEFORMING_THREAD_FLAG (tc);

  begin_c_loop(c, tc)
    {
      c_node_loop(c, tc, n)
        {
          v = C_NODE(c, tc, n);
          if (NODE_POS_NEED_UPDATE(v))
            {
              NODE_POS_UPDATED(v);

                  ap[0]=NODE_X(v);
                  ap[1]=NODE_Y(v);
                  ap[2]=NODE_Z(v);

            f_Op_gap(op, op+1, ap[0], ap[1], time-dtime, C_UDMI(c, tc, UDM_sector));
            find_appp(appp, ap, op, time-dtime, dtime, RPM*M_PI/30*dtime, f_h_gap, f_H_gap);

                  NODE_X(v)=appp[0];
                  NODE_Y(v)=appp[1];
          }
        }
      Update_Cell_Metrics (c, tc);
    }
  end_c_loop (c, tc);

}

/* Functions used to calcuate a new position for vane pump core top face (for F6.3 only) */

DEFINE_GRID_MOTION(vane_pump_core_top, domain, dt, time, dtime)
{
  face_t f;
```

```
  Thread *tf = DT_THREAD ((Dynamic_Thread *)dt);
  int n;
  Node *v;
  real ap[3], op[2], appp[2];

  /* set deforming flags */
  SET_DEFORMING_THREAD_FLAG (THREAD_T0(tf));

  begin_f_loop(f, tf)
   {
     f_node_loop(f, tf, n)
      {
        v = F_NODE(f, tf, n);
        if (NODE_POS_NEED_UPDATE(v))
          {
            NODE_POS_UPDATED(v);

                  ap[0]=NODE_X(v);
                  ap[1]=NODE_Y(v);
                  ap[2]=NODE_Z(v);

            f_Op_core(op, op+1, time-dtime, C_UDMI(F_C0(f,tf), THREAD_T0(tf), UDM_sector));
            find_appp(appp, ap, op, time-dtime, dtime, RPM*M_PI/30*dtime, f_h_core, f_H_core);

                  NODE_X(v)=appp[0];
                  NODE_Y(v)=appp[1];
          }
      }
     Update_Face_Metrics (f, tf);
   }
  end_f_loop (f, tf);

}

/* Functions used to calcuate a new position for vane pump gap top face (for F6.3 only) */

DEFINE_GRID_MOTION(vane_pump_gap_top, domain, dt, time, dtime)
{
  face_t f;
  Thread *tf = DT_THREAD ((Dynamic_Thread *)dt);
  int n;
  Node *v;
  real ap[3], op[2], appp[2];

  /* set deforming flags */
  SET_DEFORMING_THREAD_FLAG (THREAD_T0(tf));

  begin_f_loop(f, tf)
   {
     f_node_loop(f, tf, n)
      {
        v = F_NODE(f, tf, n);
        if (NODE_POS_NEED_UPDATE(v))
          {
            NODE_POS_UPDATED(v);

                  ap[0]=NODE_X(v);
```

```
                ap[1]=NODE_Y(v);
                ap[2]=NODE_Z(v);

          f_Op_gap(op, op+1, ap[0], ap[1], time-dtime, C_UDMI(F_C0(f,tf), THREAD_T0(tf),
UDM_sector));
          find_appp(appp, ap, op, time-dtime, dtime, RPM*M_PI/30*dtime, f_h_gap, f_H_gap);

                NODE_X(v)=appp[0];
                NODE_Y(v)=appp[1];
            }
        }
     Update_Face_Metrics (f, tf);
   }
  end_f_loop (f, tf);

}

DEFINE_GRID_MOTION(walls, domain, dt, time, dtime)
{

}

static void intialize_one_shape(struct shape * one_shape,  enum define_shape shape_type, enum shrink
shrink_y_or_n, real * center, real radius,
                        char * file_name, int flag_time_dep_profile)
{
 one_shape->shape_type=shape_type;
 one_shape->flag_time_dep_profile=flag_time_dep_profile;
 if (one_shape->shape_type==circle)
   {
                one_shape->center[0]=center[0];
                one_shape->center[1]=center[1];
                one_shape->radius=radius;
          }
 else if (one_shape->shape_type==user_defined)
   {
    FILE *fp_data;
    int i, j;

    one_shape->shrink_y_or_n=shrink_y_or_n;

    #if PARALLEL
      if(I_AM_NODE_HOST_P)
    #endif
      {
       if(!(fp_data=fopen(file_name,"r")))
        {
          Message0("\nCan not open file %s -aborting!!", file_name);
          exit(0);
        }

       fscanf(fp_data, "%d", &(one_shape->number_of_data));
      }

    host_to_node_int_1(one_shape->number_of_data);

    if(!(one_shape->data=(real (*)[2]) calloc(one_shape->number_of_data, 2*sizeof(real))))
```

```
    {
                    Message0("\nMemory allocatoin failure-aborting!!\n");
                    exit(0);
            }

  for(i=0; i<one_shape->number_of_data; i++)
   for(j=0; j<2; j++)
   {
     #if PARALLEL
       if(I_AM_NODE_HOST_P)
     #endif
       {
         #if RP_DOUBLE
                    fscanf(fp_data, "%le", &(one_shape->data[i][j]));
         #else
                    fscanf(fp_data, "%e", &(one_shape->data[i][j]));
         #endif
       }
       }

  host_to_node_real(&(one_shape->data[0][0]), 2*one_shape->number_of_data);

  #if PARALLEL
    if(I_AM_NODE_HOST_P)
  #endif
     {
      fclose(fp_data);
     }
        }
else if (one_shape->shape_type==special_circle)
  {

  }
else if (one_shape->shape_type==special_user_defined)
  {
   FILE *fp_data;
   int i, j;

  one_shape->shrink_y_or_n=shrink_y_or_n;

  #if PARALLEL
    if(I_AM_NODE_HOST_P)
  #endif
     {
      if(!(fp_data=fopen(file_name,"r")))
       {
         Message0("\nCan not open file %s -aborting!!", file_name);
         exit(0);
       }

      fscanf(fp_data, "%d", &(one_shape->number_of_data));
     }

  host_to_node_int_1(one_shape->number_of_data);

  if(!(one_shape->data=(real (*)[2]) calloc(one_shape->number_of_data, 2*sizeof(real))))
    {
```

```
                    Message0("\nMemory allocatoin failure-aborting!!\n");
                    exit(0);
            }

    for(i=0; i<one_shape->number_of_data; i++)
     for(j=0; j<2; j++)
      {
        #if PARALLEL
          if(I_AM_NODE_HOST_P)
        #endif
          {
            #if RP_DOUBLE
                    fscanf(fp_data, "%le", &(one_shape->data[i][j]));
            #else
                    fscanf(fp_data, "%e", &(one_shape->data[i][j]));
            #endif
          }
            }

    host_to_node_real(&(one_shape->data[0][0]), 2*one_shape->number_of_data);

    #if PARALLEL
      if(I_AM_NODE_HOST_P)
    #endif
       {
        fclose(fp_data);
       }
   }
 else
   {
                Message0("\nWrong shape-aborting!!!\n");
                exit(0);
          }

}

static void initialize_shape(void)
{
 real center[2], radius;

 center[0]=-DELTA;
 center[1]=0;
 radius=R_OUTER;
 intialize_one_shape(&pump_core_outer_shape, OUTER_SHAPE, wo_shrink, center, radius,
"data_outer.txt", 0); /*hard coded no time dep prof*/

 center[0]=0;
 center[1]=0;
 radius=R_INNER;
 intialize_one_shape(&pump_core_inner_shape, INNER_SHAPE, wo_shrink, center, radius,
"data_inner.txt", INNER_TIME_DEP_PROFILE);

 center[0]=-DELTA;
 center[1]=0;
 radius=R_OUTER;
 intialize_one_shape(&pump_gap_outer_shape, OUTER_SHAPE, wo_shrink, center, radius,
"data_outer.txt", 0);/*hard coded no time dep prof*/
```

```
 center[0]=-DELTA;
 center[1]=0;
 radius=R_OUTER-GAP_SIZE;
 intialize_one_shape(&pump_gap_inner_shape, OUTER_SHAPE, w_shrink, center, radius,
"data_outer.txt", 0);/*hard coded no time dep prof*/

 return;
}

static int sector_number(real angle)
{
 int i;
 real total_angle;

 i = 0;
 total_angle = ANGLE[0];

 while(angle>total_angle)
  {
            i++;
            total_angle += ANGLE[i];
  }

 return i;
}

static void init_one_core_cell_zone(Thread * tc)
{
 cell_t c;
 Node *v;
 int j;
 real angle, init_vane_angle_rad;

 begin_c_loop_int(c, tc)
  {
    v = C_NODE(c, tc, 0);

    angle = my_atan2(NODE_Y(v), NODE_X(v));
    init_vane_angle_rad = INIT_VANE_ANGLE*M_PI/180;

    angle -= init_vane_angle_rad;

    if (angle <0)
       angle += 2*M_PI;

    if(NON_EQUALLY_SPACED_VANE)
     {
       C_UDMI(c, tc, UDM_sector) = sector_number(angle);
            }
    else
     {
              j = angle/(2*M_PI/N_VANE);

       C_UDMI(c, tc, UDM_sector) = j;
            }
  }
```

```
 end_c_loop_int(c, tc)
}

static void init_one_gap_cell_zone(Thread * tc)
{
 cell_t c;
 Node *v;
 int j;
 real angle, init_vane_angle_rad;

 begin_c_loop_int(c, tc)
  {
          v = C_NODE(c, tc, 0);

   angle = my_atan2(NODE_Y(v),NODE_X(v));
   init_vane_angle_rad = INIT_VANE_ANGLE*M_PI/180;

   angle -= init_vane_angle_rad;

   if (angle <0)
     angle += 2*M_PI;

          if (fabs(angle-2*M_PI)<2*M_PI/N_VANE*0.3)
                angle = 0;

          j = (angle+2*M_PI/N_VANE*0.5)/(2*M_PI/N_VANE);

          C_UDMI(c, tc, UDM_sector) = j;
  }
 end_c_loop_int(c, tc)
}

/* Initialization function is used to calculate sector number */

DEFINE_INIT(init_sector, domain)
{
 Thread *tc;
 int i;

 /* Initialization can only be done at the beginning */
 Message0("\n\n*********************************************************************************");
 Message0("\n\n Warning: you are initializing the flow.  For the vane pump application,");
 Message0("\n you can only do this when the mesh is at the initial position.  Otherwise,");
 Message0("\n initialization may cause mesh motion failure!!!\n");
 Message0("\n After initialization, please display the UDM-0 using cell value to verify");
 Message0("\n that each chamber has distinct value (DIFFERENT SINGLE COLOR FOR EACH
CHAMBER!\n");
 Message0("\n*********************************************************************************\n");


 /* Initialize the Core sector number*/

 if(MULTIPLE_CORE)
  {
   for(i=0; i<N_VANE; i++)
    {
     tc=Lookup_Thread(domain, CORE_ID[i]);
```

```
      init_one_core_cell_zone(tc);
     }
  }
 else
  {
    tc=Lookup_Thread(domain, CORE_ID[0]);
    init_one_core_cell_zone(tc);
  }


 /* Initialize the Gap sector number*/

 if (GAP_ID[0] > 0)
  {
   if(MULTIPLE_GAP)
    {
     for(i=0; i<N_VANE; i++)
      {
              tc=Lookup_Thread(domain, GAP_ID[i]);
        init_one_gap_cell_zone(tc);
      }
    }
   else
    {
     tc=Lookup_Thread(domain, GAP_ID[0]);
     init_one_gap_cell_zone(tc);
    }
  }
}

/* print the outer contour for user_defined shape_type*/

static void print_one_shape(struct shape * one_shape)
{

 int i, j;

 Message0("shape type: %d\n", one_shape->shape_type);

 if((one_shape->shape_type))
   Message0("shrink: %d\n", one_shape->shrink_y_or_n);

 if (one_shape->shape_type==circle)
  {
    Message0("center:%10.2e %10.2e radius:%10.2e\n", one_shape->center[0], one_shape->center[1],
one_shape->radius);
  }
 else if(one_shape->shape_type==user_defined || one_shape->shape_type==special_user_defined)
  {

   Message0("number of data: %d\n", one_shape->number_of_data);

#ifdef DEBUG

   for(i=0; i<one_shape->number_of_data; i++)
    {
     for(j=0; j<2; j++)
```

```
        {
                        Message0("%12.4e", one_shape->data[i][j]);
                }
     Message0("\n");
    }
#endif
  }
 else if(one_shape->shape_type==special_circle)
  {
            Message0("\nSpecial Inner Circle Shape\n");
  }
 else if(one_shape->shape_type==special_user_defined)
  {
            Message0("\nSpecial Inner User-Defined Shape\n");
  }
 else
  {
            Message0("\nWrong type-aborting!!\n"); exit(0);
  }
}

/* Only node 0 will execute ON_DEMAND udf */

DEFINE_ON_DEMAND(print_info)
{
  int i;
  real total_angle;

  Message0("\n************************* General info *************************\n");

  Message0("\nOUTER_SHAPE: %d\n", OUTER_SHAPE);
  Message0("INNER_SHAPE: %d\n", INNER_SHAPE);
  Message0("N_VANE: %d\n", N_VANE);

  Message0("MULTIPLE_CORE: %d\n", MULTIPLE_CORE);
  if(MULTIPLE_CORE)
          {
           Message0("CORE_ID: ");
           for(i=0; i<N_VANE; i++)
            {
             Message0("%d  ", CORE_ID[i]);
                  }
           Message0("\n");
   }
  else
   {
    Message0("CORE_ID: %d\n", CORE_ID[0]);
          }

  Message0("MULTIPLE_GAP: %d\n", MULTIPLE_GAP);
  if(MULTIPLE_GAP)
          {
            Message0("GAP_ID: ");
            for(i=0; i<N_VANE; i++)
             {
              Message0("%d  ", GAP_ID[i]);
                    }
```

```
            Message0("\n");
    }
  else
    {
      Message0("GAP_ID: %d\n", GAP_ID[0]);
          }

  Message0("INNER_TIME_DEP_PROFILE: %d\n", INNER_TIME_DEP_PROFILE);

  Message0("RPM: %-10.2e VANE_WIDTH: %-10.2e GAP_SIZE: %-10.2e\n", RPM, VANE_WIDTH,
GAP_SIZE);

  Message0("INIT_VANE_ANGLE: %-6.2f\n", INIT_VANE_ANGLE);

  Message0("NON_EQUALLY_SPACED_VANE: %d\n", NON_EQUALLY_SPACED_VANE);
  if(NON_EQUALLY_SPACED_VANE)
   {
                    total_angle = 0;
                    for(i=0; i<N_VANE; i++)
                     {
        Message0("ANGLE[%d]=%-5.1f ", i, ANGLE[i]*180/M_PI);
        total_angle += ANGLE[i]*180/M_PI;
                     }
      Message0("\n");
      Message0("total angle: %-6.1f\n", total_angle);
         }

  Message0("\n********** Below is the pump core outer infor **********\n\n");
  print_one_shape(&pump_core_outer_shape);

  Message0("\n********** Below is the pump core inner infor **********\n\n");
  print_one_shape(&pump_core_inner_shape);

  if(GAP_ID[0]>0)
   {
    Message0("\n********** Below is the pump gap outer infor **********\n\n");
    print_one_shape(&pump_gap_outer_shape);

    Message0("\n********** Below is the pump gap inner infor **********\n\n");
    print_one_shape(&pump_gap_inner_shape);
         }

}

/* The input file will be read each time the UDF is loaded */

DEFINE_EXECUTE_ON_LOADING(on_loading, libudf)
{
  int i;
  FILE* fpin;

  Message0("\nON_LOADING udf is executing ...\n");

 #if PARALLEL
  if(I_AM_NODE_HOST_P)
 #endif
```

```
{
  fpin = fopen("input.txt","r");
  if(fpin == NULL)
     Message0("Input file does not exist!");

}

 #if PARALLEL
   if(I_AM_NODE_HOST_P)
 #endif
    {
            fscanf(fpin, "%d %d %d", &OUTER_SHAPE, &INNER_SHAPE, &N_VANE);

            fscanf(fpin, "%d", &MULTIPLE_CORE);
            if(MULTIPLE_CORE)
             {
               CORE_ID = alloc_1d_int_array(N_VANE);
               for(i=0; i<N_VANE; i++)
                {
                 fscanf(fpin, "%d", CORE_ID+i);
                       }
             }
            else
             {
        CORE_ID = alloc_1d_int_array(1);
        fscanf(fpin, "%d", CORE_ID+0);
             }

            fscanf(fpin, "%d", &MULTIPLE_GAP);
            if(MULTIPLE_GAP)
             {
              GAP_ID = alloc_1d_int_array(N_VANE);
              for(i=0; i<N_VANE; i++)
               {
                fscanf(fpin, "%d", GAP_ID+i);
                       }
      }
     else
      {
       GAP_ID = alloc_1d_int_array(1);
       fscanf(fpin, "%d", GAP_ID+0);
             }

            fscanf(fpin, "%d", &INNER_TIME_DEP_PROFILE);

            fscanf(fpin, REAL_FMT REAL_FMT REAL_FMT REAL_FMT, &RPM, &R_INNER,
&VANE_WIDTH, &GAP_SIZE);
            fscanf(fpin, REAL_FMT REAL_FMT REAL_FMT, &R_OUTER, &DELTA,
&INIT_VANE_ANGLE);

            fscanf(fpin, "%d",&NON_EQUALLY_SPACED_VANE);

            if(NON_EQUALLY_SPACED_VANE)
             {
              ANGLE = alloc_1d_real_array(N_VANE);
              for(i=0; i<N_VANE; i++)
               {
```

```
                fscanf(fpin, REAL_FMT, ANGLE+i);
                ANGLE[i] *=M_PI/180.0;
                    }
     }

          fclose(fpin);
  }

        host_to_node_int_7(OUTER_SHAPE, INNER_SHAPE, N_VANE, MULTIPLE_CORE,
MULTIPLE_GAP, INNER_TIME_DEP_PROFILE, NON_EQUALLY_SPACED_VANE);

#if RP_NODE
        if(MULTIPLE_CORE)
          CORE_ID = alloc_1d_int_array(N_VANE);
        else
     CORE_ID = alloc_1d_int_array(1);

        if(MULTIPLE_GAP)
          GAP_ID = alloc_1d_int_array(N_VANE);
        else
     GAP_ID = alloc_1d_int_array(1);

        if(NON_EQUALLY_SPACED_VANE)
    ANGLE = alloc_1d_real_array(N_VANE);
#endif

        if(NON_EQUALLY_SPACED_VANE)
     host_to_node_real(ANGLE, N_VANE);

        if(MULTIPLE_CORE)
          host_to_node_int(CORE_ID, N_VANE);
        else
          host_to_node_int(CORE_ID, 1);

        if(MULTIPLE_GAP)
          host_to_node_int(GAP_ID, N_VANE);
        else
          host_to_node_int(GAP_ID, 1);

        host_to_node_real_7(RPM, R_INNER, VANE_WIDTH, GAP_SIZE, R_OUTER, DELTA,
INIT_VANE_ANGLE);

        /* Read in any profile data files */
        initialize_shape();
}
```

Appendix C

Properties of transmission oil, Dexron VI

Study for balanced pump was conducted with experimental and computational methods using transmission oil Dexron VI. Input parameters for CFD are density and dynamic viscosity for operating temperature. Graphs which show relationship between density and viscosity vs. temperature are depicted on **Figure 103**, **Figure 104** and **Figure 105**.

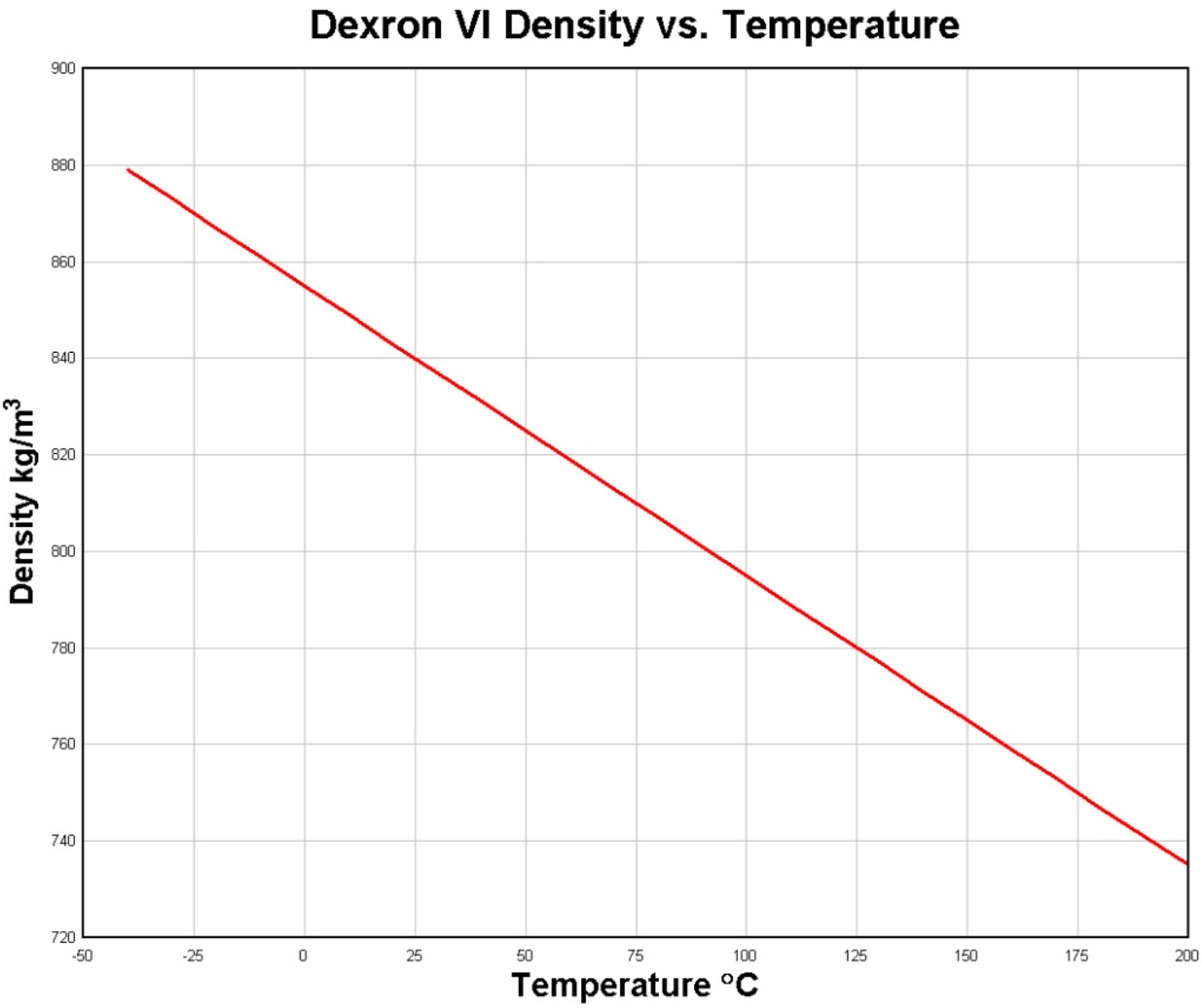


Figure 103: Dexron VI, density vs. temperature

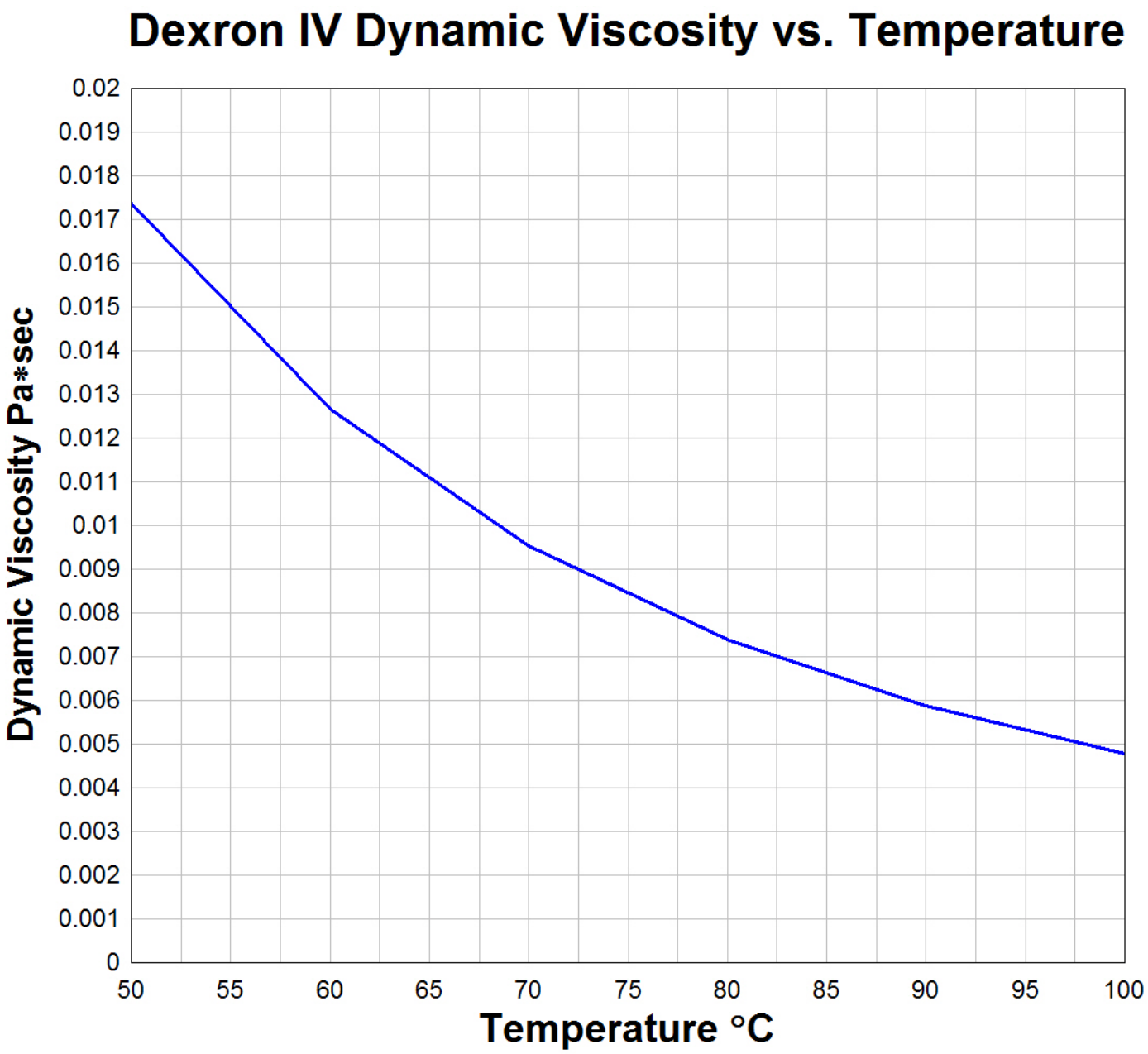


Figure 104: Dexron VI, dynamic viscosity vs. temperature

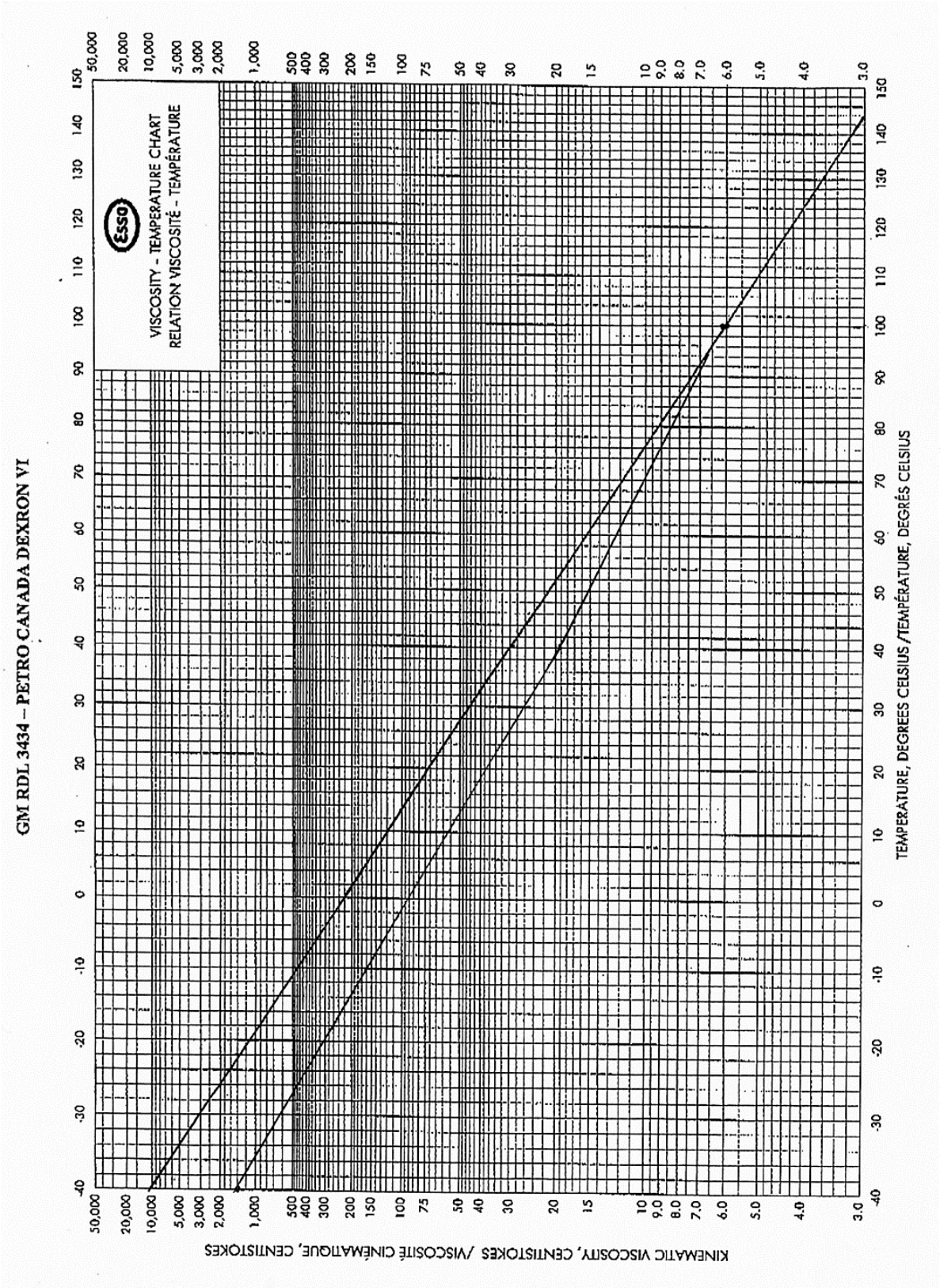


Figure 105: Dexron VI, kinematic viscosity vs. temperature

Appendix D

D.1. Mixture Model Theory

In all CFD simulations presented in this research, transmission oil with 4% of air is used in. The CFD code treats fluid as a mixture with a constant aeration. The mixture model can model *n* phases (fluid or particulate) by solving the momentum, continuity, and energy equations for the mixture, the volume fraction equations for the secondary phases, and algebraic expressions for the relative velocities.

D.1.2. Overview

The mixture model is a simplified multiphase model that can be used in different ways (Ref: Ansys Fluent, 2011, Theory Guide). It can be used to model multiphase flows where the phases move at different velocities, but assume local equilibrium over short spatial length scales. It can be used to model homogeneous multiphase flows with very strong coupling and phases moving at the same velocity and lastly, the mixture models are used to calculate non-Newtonian viscosity.

The mixture model can model *n* phases (fluid or particulate) by solving the momentum, continuity, and energy equations for the mixture, the volume fraction equations for the secondary phases, and algebraic expressions for the relative velocities. Typical applications include sedimentation, cyclone separators, particle-laden flows with low loading, and bubbly flows where the gas volume fraction remains low.

The mixture model is a good substitute for the full Eulerian multiphase model in several cases. A full multiphase model may not be feasible when there is a wide distribution of the particulate phase or when the interphase laws are unknown or their reliability can be questioned. A simpler model like the mixture model can perform as well as a full multiphase model while solving a smaller number of variables than the full multiphase model.

D.1.3. Limitations

The following limitations apply to the mixture model in Ansys Fluent:

- Pressure-based solver must be used. The mixture model is not available with the density-based solver.

- Only one of the phases can be defined as a compressible ideal gas. There is no limitation on using compressible liquids using user-defined functions.

- When the mixture model is used, do not model streamwise periodic flow with specified mass flow rate.

- Do not model solidification and melting in conjunction with the mixture model.

- The Singhal et al. cavitation model (available with the mixture model) is not compatible with the LES turbulence model.

- Do not use the relative formulation in combination with the MRF and mixture model.

- The mixture model does not allow for inviscid flows.

- The shell conduction model for walls is not allowed with the mixture model.

- When tracking particles in parallel, do not use the DPM model with the mixture model if the shared memory option is enabled (Note that using the message passing option, when running in parallel, enables the compatibility of all multiphase flow models with the DPM model.) The mixture model, like the VOF model, uses a *single-fluid* approach. It differs from the VOF model in two respects:

- The mixture model allows the phases to be interpenetrating. The volume fractions α_q and α_p for a control volume can therefore be equal to any value between 0 and 1, depending on the space occupied by phase *q* and phase p.

- The mixture model allows the phases to move at different velocities, using the concept of slip velocities.

(Note that the phases can also be assumed to move at the same velocity, and the mixture model is then reduced to a homogeneous multiphase model.)

The mixture model solves the continuity equation for the mixture, the momentum equation for the mixture, the energy equation for the mixture, and the volume fraction equation for the secondary phases, as well as algebraic expressions for the relative velocities (if the phases are moving at different velocities).

D.2. Continuity Equation

The continuity equation for the mixture is

$$\frac{\partial}{\partial t}(\rho_m) + \nabla \cdot (\rho_m \vec{v}_m) = 0$$

where $\vec{v}_m$ is the mass-averaged velocity:

$$\vec{v}_m = \frac{\sum_{k=1}^{n} \alpha_k \rho_k \vec{v}_k}{\rho_m}$$

and ρ_m is the mixture density:

$$\rho_m = \sum_{k=1}^{n} \alpha_k \rho_k$$

α_k is the volume fraction of phase.

D.3. Momentum Equation

The momentum equation for the mixture can be obtained by summing the individual momentum equations for all phases. It can be expressed as

$$\frac{\partial}{\partial t}(\rho_m \vec{v}_m) + \nabla \cdot (\rho_m \vec{v}_m \vec{v}_m) = -\nabla_p + \nabla \cdot [\mu_m (\nabla \vec{v}_m + \nabla \vec{v}_m^T)] + \rho_m \vec{g} + \vec{F} + \nabla . \left(\sum_{k=1}^{n} \alpha_k \rho_k \vec{v}_{dr,k} \vec{v}_{dr,k} \right)$$

where *n* is the number of phases, $\vec{F}$ is a body force, and μ_m is the viscosity of the mixture:

$$\mu_m = \sum_{k=1}^{n} \alpha_k \mu_k$$

$\vec{v}_{dr,k}$ is the drift velocity for secondary phase k *:*

$$\vec{v}_{dr,k} = \vec{v}_k - \vec{v}_m$$

D.4. Energy Equation

The energy equation for the mixture takes the following form:

$$\frac{\partial}{\partial t} \sum_{k=1} (\alpha_k \rho_k E_k) + \nabla \cdot \sum_{k=1} \left(\alpha_k \vec{v}_k (\rho_k E_k + p) \right) = \nabla \cdot \left(k_{eff} \nabla T \right) + S_E$$

where k_{eff} is the effective conductivity ($\Sigma\ \alpha_k(k_k + k_t)$) where k_t is the turbulent thermal conductivity, defined according to the turbulence model being used. The first term on the right-hand side of above equation represents energy transfer due to conduction. S_E includes any other volumetric heat sources.

In above stated equation,

$$E_k = h_k - \frac{p}{\rho_k} + \frac{v_k^2}{2}$$

for a compressible phase, and $E_k = h_k$ for an incompressible phase, where h_k is the sensible enthalpy for phase k.

Appendix E

E.1. First and second order scheme results comparison

E.1.1. First-Order Upwind Scheme

When first-order accuracy is desired, quantities at cell faces are determined by assuming that the cell-center values of any field variable represent a cell-average value and hold throughout the entire cell; the face quantities are identical to the cell quantities (Ref: Ansys Fluent, 2011, Theory Guide). Thus when first-order upwind is selected, the face value is set equal to the cell-center value of in the upstream cell.

First-order upwind is available in the pressure-based and density-based solvers.

E.1.2. Second-Order Upwind Scheme

When second-order accuracy is desired, quantities at cell faces are computed using a multidimensional linear reconstruction approach. In this approach, higher-order accuracy is achieved at cell faces through a Taylor series expansion of the cell-centered solution about the cell centroid .

E.1.3. Differences in computational time using first and second upwind scheme

Second order upwind is usually used when higher precision is needed near walls or flow separations. For transient simulation, number of sub-steps for step convergence can be radically increased, since much more sub-steps are needed to obtain the same level or residual error for second upwind scheme. In simulations for positive displacement pumps, second order does not give different results for pressure and flow compared to first order upwind. Thus, first order upwind is used in balanced pumps simulations.

Sample graphs for first and second order upwind are depicted on **Figure 106** and **Figure 107**. On **Figure 106**, blue curve represents pressure in chamber from CFD model solved with 1st order upwind and green curve represents pressure in chamber from CFD model solved with 2nd order upwind. Differences between these two curves are negligible.

Similarly, on **Figure 107**, blue curve represents pump flow performance from CFD model solved with 1st order upwind and red curve represents pump flow performance from CFD model solved with 2nd order upwind. Differences between these two curves are negligible. However, computational time to obtain convergence in each sub-step for model solving at 2nd order upwind is much longer.

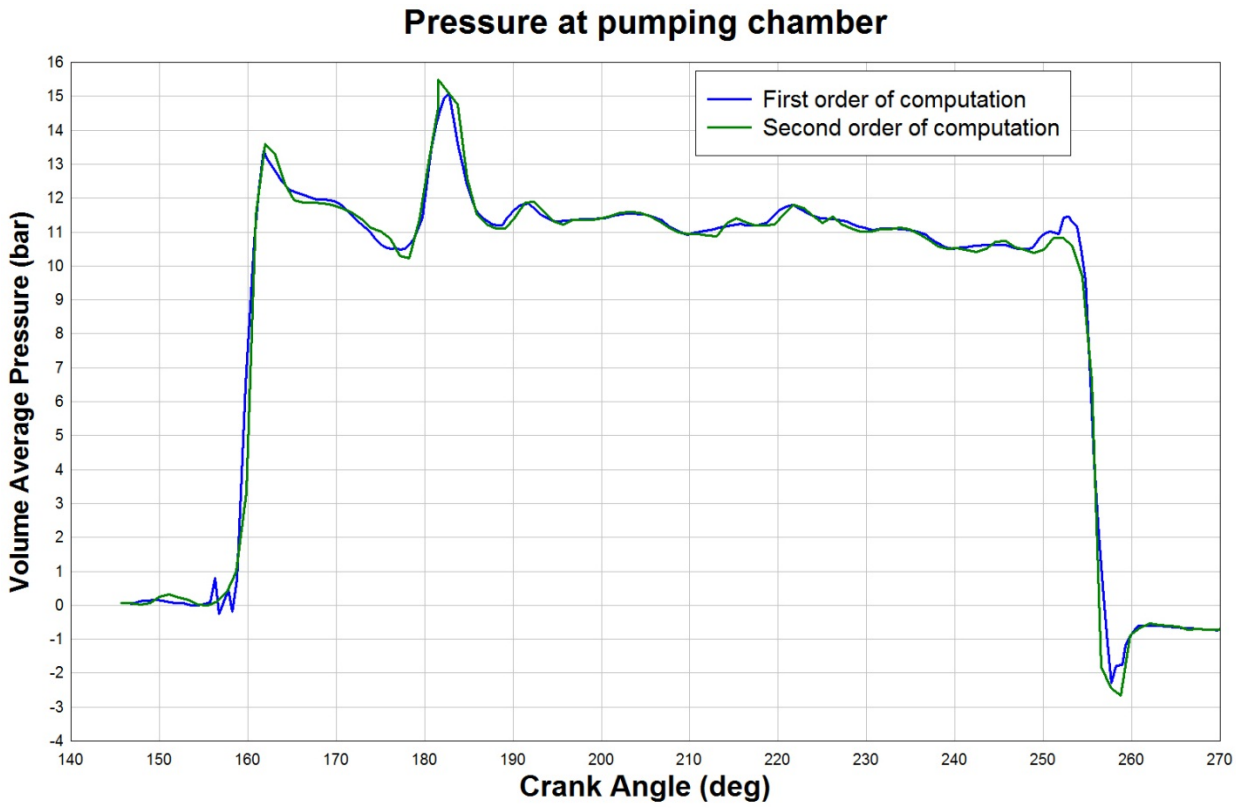


Figure 106: Pressure in chamber computed with 1st (blue curve) and 2nd order (green curve) upwind

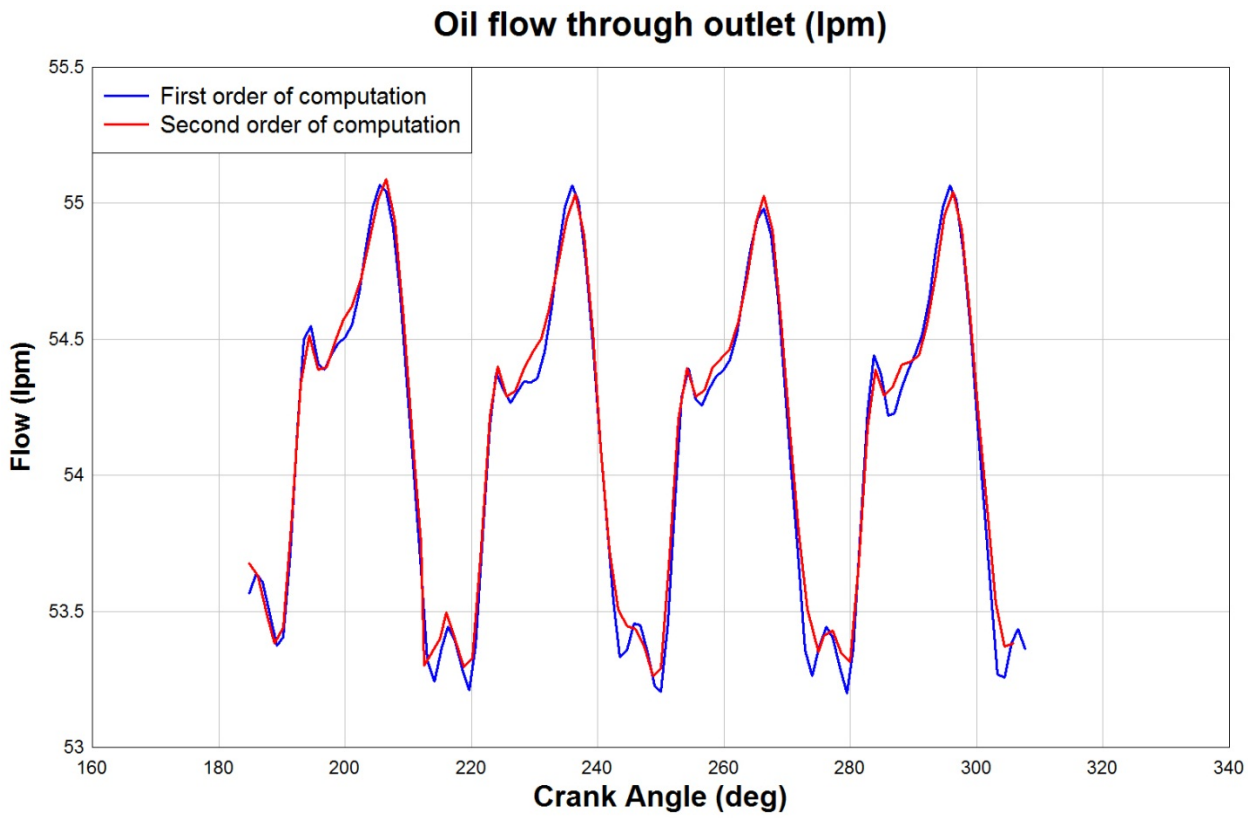


Figure 107: Pump flow performance computed with 1st and 2nd order upwind (blue and red curve respectively)