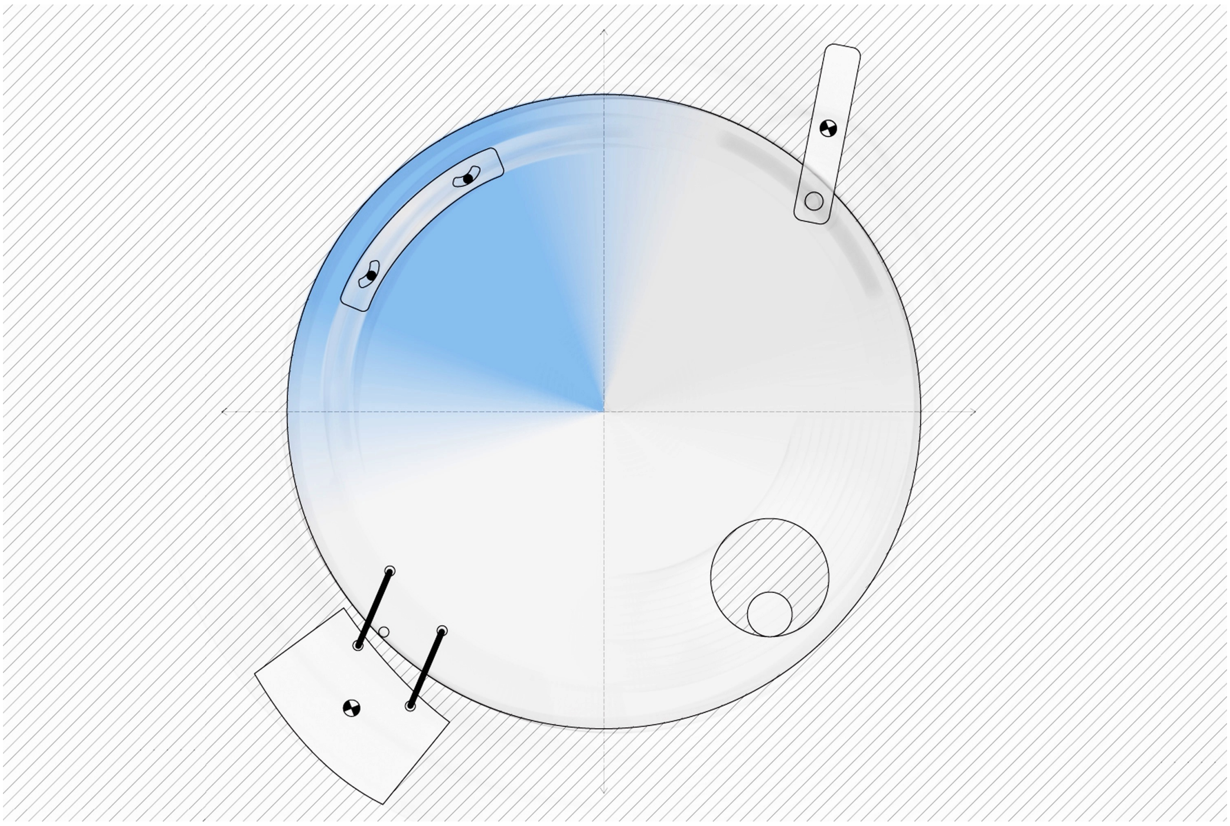


Reduction of Vibrations in Engines using Centrifugal Pendulum Vibration Absorbers

Master's Thesis in the Master's programme Automotive Engineering

ANDERS WEDIN

Department of Product and Production Development
Division of Product Development
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden 2011

Master's Thesis 2011

MASTER'S THESIS 2011

Reduction of Vibrations in Engines using Centrifugal Pendulum Vibration Absorbers

– a Study with Modeling and Simulation of Centrifugal Pendulum Vibration Absorbers in the Dual Mass Flywheel

Master's Thesis in the Master's programme Automotive Engineering

ANDERS WEDIN

Department of Product and Production Development
Division of Product Development
CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2011

Reduction of Vibrations in Engines using Centrifugal Pendulum Vibration Absorbers
– a Study with Modeling and Simulation of Centrifugal Pendulum Vibration Absorbers in the Dual Mass Flywheel
ANDERS WEDIN

Master's Thesis 2011
Department of Product and Production Development
Division of Product Development
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: + 46 (0)31-772 1000

Cover:
Sketch of Simple, Bifilar, Roll Form, and a possible design of a Bifilar CPVA with General Path (in blue). Drawn by Anna Wedin.

Chalmers Reproservice
Göteborg, Sweden 2011

Reduction of Vibrations in Engines using Centrifugal Pendulum Vibration Absorbers
– a Study with Modeling and Simulation of Centrifugal Pendulum Vibration Absorbers in the Dual Mass Flywheel
Master's Thesis in the Master's programme Automotive Engineering
ANDERS WEDIN
Department of Product and Production Development
Division of Product Development
Chalmers University of Technology

Abstract

The development and use of CPVAs (Centrifugal Pendulum Vibration Absorbers) started in the beginning of the 20th century, and one of the first traces is a model by Kutzbach in 1911. Most of the models were developed and patented during the 1930s, and those types are still used today. CPVAs came to play a vital role in the development of radial aircraft engines, which were used extensively in World War II. With the aid of correctly tuned CPVAs, vibrations of different orders could be reduced significantly, ensuring reliable operation. The CPVA is particularly useful for engines since it is not dependent on frequency, but the order of the applied torque. The CPVA can therefore reduce vibrations in the entire engine speed range.

CPVAs are now finding use in powertrains of cars, more specifically in the DMF (Dual Mass Flywheel). The problem automotive manufacturers face today, from a vibration perspective, is that engines generally tend to become smaller in size and have fewer cylinders, while the power output tend to increase. The DMF has been an effective way of reducing vibrations from engine to transmission for some time now, but ever increasing vibration levels from newly developed engines create a need of using CPVAs.

The main goal of this thesis work is to, at Volvo Cars engine development department, model and simulate different types of CPVAs in the software Simdrive 3D by Contecs Eng, where simple vibration models can be made for first investigations.

Different CPVAs have been modeled through a fully customizable element, programmed in C/C++ for Simdrive. Special attention has been given to model CPVAs with cycloidal and epicycloidal paths, since they are important for practical use. The CPVA unit performance is analyzed in one model with an n:th order sine-signal torque, and also in a more realistic powertrain model with cylinders, DMF with CPVAs attached, FEAD (Front Engine Accessory Drives), and realistic input torque from combustions in the engine cylinders.

In simulations with sine-signal torque, all CPVA types considered seem to have capabilities of absorbing vibrations. Some of the types with cycloidal or epicycloidal paths reduce vibrations of the designated order to a level very close to zero, for a major part of the engine speed range. In the advanced powertrain model, where the pendulum mass is restricted to 1 kg, the CPVA units are stretched to their limit. Types with regular circular paths either fail to reduce or amplify vibrations, while types with cycloidal or epicycloidal paths manage to reduce vibrations for parts of the engine speed range. With more simulation time, the CPVA parameters could be optimized further.

Future work suggestions include further development of models, introduction of ”stoppers” in models, and introduction of multiple independent absorbers to account for non-unison motion.

Keywords: Torsional Vibrations, Centrifugal Pendulum Vibration Absorbers, Dual Mass Flywheel, Bifilar, Salomon, General-Path Representation, Cycloid, Epicycloid, Tautochrone, Modeling, Simulation, Car, Powertrain, Crankshaft

Sammanfattning

I början av 1900-talet började utvecklingen och användandet av centrifugal-pendeldämpare (engelska: Centrifugal Pendulum Vibration Absorbers), där ett av de första spåren är en modell från 1911 av Kutzbach. De flesta av modellerna utvecklades och patenterades under 1930-talet, vilka används ännu idag. Pendeldämpare spelade en stor roll i utvecklingen av de motorer av stjärntyp som satt i många flygmaskiner under andra världskriget. Med hjälp av pendeldämpare kunde vibrationer av olika ordningar reduceras och förhindra motorhaveri. Pendeldämpare är särskilt väl lämpade för förbränningsmotorer eftersom de dämpar vibrationer i hela varvtalsregistret, inte bara vid en viss frekvens.

Pendeldämpare börjar nu användas i drivlinan hos bilar, närmare bestämt i tvåmassesvänghjul (TMS). I dagens utveckling tenderar motorer att bli allt mindre, ha färre cylindrar, och samtidigt utveckla mer effekt vilket skapar problem på vibrationssidan. TMS har använts under en tid för att dämpa vibrationer från motor till transmission, men senare motorer med hög effekt skapar ett behov att använda pendeldämpare i TMS:et. Det huvudsakliga målet med detta examensarbete är att, på Volvo personbilars motorutveckling, modellera och simulera olika typer av pendeldämpare i programvaran Simdrive 3D av Contecs Eng, där enklare vibrationsmodeller kan skapas.

Olika typer av pendeldämpare har modellerats manuellt i ett element, som har programmerats i C/C++ för Simdrive. Särskilt fokus har lagts på att modellera pendeldämpare med cykloid- och epicykloidbanor, eftersom dessa typer är viktiga i praktiken. Prestandan hos pendeldämparna undersöks i en modell med ett sinus-vridmoment som insignal, och också i en mer realistisk drivlina-modell med cylindrar, TMS med pendeldämpare, hjälpapparatdrivning, och realistiskt motor-vridmoment.

Simuleringar av modellen med sinus-vridmoment visar att alla typer av pendeldämpare verkar ha förmågan att dämpa vibrationer. Vissa av typer med cykloid- och epicykloidbanor dämpar vibrationerna så pass bra att de nära nog elimineras, över hela varvtalsregistret. I drivlina-modellen med realistiskt motor-vridmoment, där den totala tillåtna massan för pendlarna är 1 kg, verkar pendeldämparna befinna sig på stabilitetsgränsen. De typer som har vanliga cirkulära banor lyckas inte dämpa vibrationer, vissa av modellerna förstärker till och med vibrationerna. Typerna med cykloid- eller epicykloidbanor lyckas att dämpa vibrationerna över vissa delar av varvtalsregistret. Om fler simuleringar körs kan parametrarna för de olika typerna optimeras ytterligare.

Några rekommendationer för framtida arbete är fortsatt arbete med utveckling av modeller, införande av ”stopp” för pendelrörelsen och införande av modeller med flera oberoende pendlar på samma pendeldämparenhet eftersom de i verkligheten inte rör sig unisont.

Table of Contents

Preface

In this study, several types of Centrifugal Pendulum Vibration Absorbers (CPVAs) have been modeled and simulated in the software Simdrive 3D, as a possible method to reduce vibrations of specific orders, originating from engine combustion cycles. Simple, Bifilar, Salomon, and General-Path type CPVAs have been considered. The work has been carried out from May to November 2010 at Volvo Cars – Engine Transmissions in Göteborg, Sweden, with Anders Wedin as student. Supervisors have been Ph.D. Tomas Johannesson at Volvo Cars, and Senior Lecturer Göran Brännare at the Department of Product and Production Development, Chalmers University of Technology in Göteborg, Sweden.

Acknowledgements

First I would like to thank my supervisor Tomas Johannesson, for initiating and giving me this very varied and rewarding thesis, and also for being very supportive and helpful with whatever issues during the work. I would also like to thank Peter Norin at Volvo, for starting this thesis in collaboration with Tomas, and for letting me work on his group in Engine Transmissions. Thanks also to my supervisor at Chalmers, Göran Brännare, who has been very helpful during this work.

Further, I would like to thank Tobias Hansson at Volvo for his help with Simdrive among other things, and hope that he will find this work useful in the future. Thanks also to Niclas Andersson at Volvo for his comprehensive guidance on the function of Dual Mass Flywheels, and how they can be modeled.

Special thanks to the guys at Contecs Engineering for their invaluable support, especially for their help with programming issues. It would not have been possible to model the CPVA units without their help. Thanks also to Andreas Käll for getting me started with C programming, and Victor Sandgren for being a brilliant student reviewer of this thesis.

A very special thanks to family friend Elisabet Leppänen, for her help with finding this work by spreading the word at Volvo Cars. Thanks also to Professor S. W. Shaw for sending me previous work in the field, which proved to be most helpful. I would also like to thank the people at Volvo that showed interest in my work by listening to my presentations.

Finally, a warm thanks to family and friends for their support. Thanks to my sister for drawing the nice cover picture for this report.

Göteborg, February 2011
Anders Wedin

List of Abbreviations and Notations

Abbreviations

CPVA(s): Centrifugal Pendulum Vibration Absorber(s)

DMF: Dual Mass Flywheel

FEAD: Front Engine Accessory Drives

ODE(s): Ordinary Differential Equation(s)

Notations

t: time $[s]$

q_i: i:th generalized coordinate

$\dot{q}_i$: i:th generalized velocity

θ: CPVA unit rotor angle, first generalized coordinate $[rad]$

ϕ: CPVA unit pendulum angle, second generalized coordinate $[rad]$

T: kinetic energy $[kg \cdot m^2/s^2]$

V: potential energy $[kg \cdot m^2/s^2]$

L: Lagrangian, the difference in kinetic and potential energy $[kg \cdot m^2/s^2]$

$Yn2$: array in C/C++ storing generalized coordinates, velocities, and accelerations

Ω: steady rotary speed of CPVA rotor

ω_n: natural angular frequency of CPVA $[rad/s]$

m: total pendulum mass $[kg]$

m_p: mass of each pendulum $[kg]$

J: moment of inertia for rotor $[kg \cdot m^2]$

I: total moment of inertia pendulum for pendulums $[kg \cdot m^2]$

I_p/I_{pend}: moment of inertia for each pendulum $[kg \cdot m^2]$

x: coordinate in inertial system of the CPVA $[m]$

y: coordinate in inertial system of the CPVA $[m]$

s: arc length variable, second generalized coordinate

$\mathbf{v}$: velocity vector for pendulum center of mass

R: distance from rotor centre to pendulum attachment point on rotor $[m]$

r: distance from pendulum attachment point on rotor, to pendulum center of mass $[m]$

k_λ: radius of gyration for pendulum $[m]$

c_s: viscous damping coefficient for rotor $[Nm \cdot s/rad]$

c_a: viscous damping coefficient for pendulum absorber $[Nm \cdot s/rad]$

T_a: externally applied torque on CPVA unit $[kg \cdot m^2/s^2]$

n: harmonic order

e: distance between bored hole center and roller center (Salomon type) $[m]$

I: roller moment of inertia (Salomon type) $[kg \cdot m^2]$

D: diameter of bored hole (Salomon type) $[m]$

d: diameter of roller (Salomon type) $[m]$

ϕ_c: second generalized coordinate for Cycloid path type $[rad]$

ϕ_{epi}: second generalized coordinate for Epicycloid (cardoid) path type $[rad]$

λ: parameter to adjust the curvature of the general path (Bifilar type with rollers)

ρ_0: curvature at point $(0, -c)$ (Bifilar type with rollers)

ρ: curvature along the general path (Bifilar type with rollers)

i_R: moment of inertia for roller (Bifilar type with rollers) $[kg \cdot m^2]$

a: roller radius (Bifilar type with rollers) $[m]$

m_R: mass of roller (Bifilar type with rollers) $[kg]$

y_P: y position of pendulum center of mass on the general path (Bifilar type with rollers) $[m]$

List of Figures

List of Tables

1 INTRODUCTION

1.1 Company Background – Volvo Cars

Volvo Cars is an automobile manufacturer that was founded in 1927, in Göteborg, Sweden, and recorded global sales of 334,808 cars in 2009. Volvo Cars was owned by AB Volvo until 1999 when it was acquired by Ford Motor Company, in 2010 it was acquired by Zhejiang Geely Holding Group. The cars are assembled on three main locations – Torslanda in Göteborg, Sweden, Ghent in Belgium, and Uddevalla in Sweden. Torslanda is also the location of the headquarters. The largest markets for Volvo cars are in the US, Sweden, Great Britain, China and Germany.

Safety aspects have always been very important at Volvo Cars, e.g. the modern version of the 3-Point safety belt was invented and patented by Volvo employee Nils Bohlin in 1958. In 1959 it became standard equipment on every produced Volvo car. Volvo has also been in the front end when developing safety systems such as; rearward-facing child safety seat, side impact protection systems (including airbags), whiplash protection systems, and most recently, pedestrian detection with auto-brake.

Volvo cars are also known for high mileage and long lifetime, the average lifetime is about 20 years. A Guinness World Record has been set by a 1966 Volvo P1800 for highest mileage, with over 2.8 million miles driven. This car model became known to many when it was driven by Roger Moore in the TV-series ”The Saint” in the 60s, [1].

1.2 Historical Use and Development of CPVAs

The history of CPVAs leads back to the beginning of the 20th century, and one of the first traces is from 1911, when Kutzbach made a proposition to create U-shaped channels with fluid masses in a flywheel. Irregularities in the motion of the flywheel cause the masses to move about their equilibrium, creating a counteracting torque, this is the basic principle of CPVAs. This solution did however not raise much attention at the time, there was no urgent need for the device. It was not until the 1930s that most pendulum assemblies were developed and patented, at this time the fluid masses were replaced by solids, [2].

The Roll Form CPVA, in damped form, was invented by Carter in 1929, and the undamped form was invented by Salomon in 1932. Sarazin invented a link form of the Bifilar type CPVA in 1930, and the more useful Bifilar suspension type in 1935. Salomon also developed the Ring Form and Duplex Suspension type a few years later, [2]. All these models each have their benefits and drawbacks. The types developed in the 1930s are still used today, and some of them are subject to investigation in this work.

It was found that the major advantage of these vibration absorbers was that they, instead of just damping vibrations at a specific frequency, could dampen vibrations over a range of frequencies. It made them ideal for piston engine applications. In the 1930s and 1940s, CPVAs played a substantial role in the development of aircraft, since the aircraft were still propelled by piston engines only.

A context in which CPVAs came to play a vital role was the development of the Pratt & Whitney R-2800 ”Double Wasp” aircraft engine, a race against the clock to develop the first air-cooled engine with a rating of more than 2,000 hp. This was at a time when test engineers did not have access to computer simulation tools or other modern aids. The only methods available were trial-and-error, i.e. build something with the best knowledge available, test it until it breaks, and then understand why it broke and rebuild it. This method was used until the part was up to specifications.

At Pratt & Whitney, vibrations problems became apparent when they switched to controllable-pitch propellers, since they were heavier than previous simpler propeller types. This brought the resonant frequency of the crankshaft into the operating range of the engine. This problem was experienced also at Curtiss-Wright, rival of Pratt & Whitney. They responded to this issue by first using a puck-type damper, developed by Taylor of MIT, that is very similar to the Roll Form type by Salomon. Later another employee, Chilton, designed a new superior type that came to be named after its inventor. This type though, was very similar to the Bifilar design by Sarazin. Interesting to note is that MIT Professor J. P. Den Hartog became a consultant to Pratt & Whitney at the same time, who would later write his classic book on mechanical vibrations.

Due to patent issues as the most probable cause, Pratt & Whitney had to use the Puck-type damper for the first versions of the R-2800 engine, and it was not until engine version C that the Bifilar or Chilton design absorber was implemented. These absorbers solved some of the vibration issues of these engines, and made the R-2800 a very good and robust engine that powered many fighters and medium bombers in World War II. It was also used to a great extent after the war in civilian aircraft, and became known for its reliability, [2, 3].

1.3 Problem Background and Description

The public awareness that the Earth's resources of crude oil are limited, cause many type of businesses to operate and create products or services that consume a minimum amount of oil. In the car industry, it is of great importance that emissions from the cars are kept to a minimum. This is due to legislation with the objectives of minimizing global warming (CO_2 emissions) and pollution. A goal for the car manufacturers is therefore to produce cars with low fuel consumption, which is also interesting for car customers because of increasing oil prices. These demands have led to what in the car industry is known as "downsizing", a principle stating that maximum possible power should be extracted from engines with minimized swept volume. This ensures that the engine operates with a better combination of fuel consumption and power output.

Downsizing trends cause engines to have higher specific power, which from the vibration perspective is unfavorable. A small mass associated with high-level exciting torques cause high levels of vibration. In the car engine, torsional vibrations in the crankshaft pose a problem since the torque from the cylinders are delivered in pulses to the crankshaft. This causes the crankshaft to rotate with irregular motion. In a four-stroke engine, a torque pulse (also called power-stroke) is delivered once every two revolutions of the engine, from every cylinder. In a four-cylinder engine, the cylinders excite the crankshaft twice every revolution, and therefore the torque has a dominant second order harmonic component. These torque pulses are not desirable in the engine because of vibrations that cause noise or even failure of engine components.

For some years now a special type of flywheel, called the Dual Mass Flywheel (DMF), has been used in powertrains of cars to protect the transmission and the rest of the driveline from engine vibrations, see Figure 1.1, and the work of Albers [4]. This flywheel isolates the engine (primary side) from the transmission (secondary side) by coupling them with a weak arc spring, which causes about 80 - 90% vibration reduction. The gearbox may then operate without rattle and noise, ensuring a good level of driving comfort. The motivation to use CPVAs on the DMF is recent development of the most powerful engines, where the construction of the DMF is simply not enough to reduce vibrations to an acceptable level. The implementation of CPVAs is an addition to the DMF that will account for about 10% of the total vibration reduction. CPVAs are suitable because of their order dependence – they operate in the entire engine speed range.

A DMF from German manufacturer LuK is shown in Figure 1.2. The leftmost part is the primary flywheel, attached to the crankshaft. The arc spring (in yellow) connects the primary flywheel to the secondary flywheel, rightmost in the picture. The CPVAs (arc-shaped on the middle carrier in purple) are attached to a separate carrier, which is attached to the secondary flywheel. In this model, the number of CPVAs is eight (four on each side of the carrier). The secondary flywheel connects to the clutch and transmission.

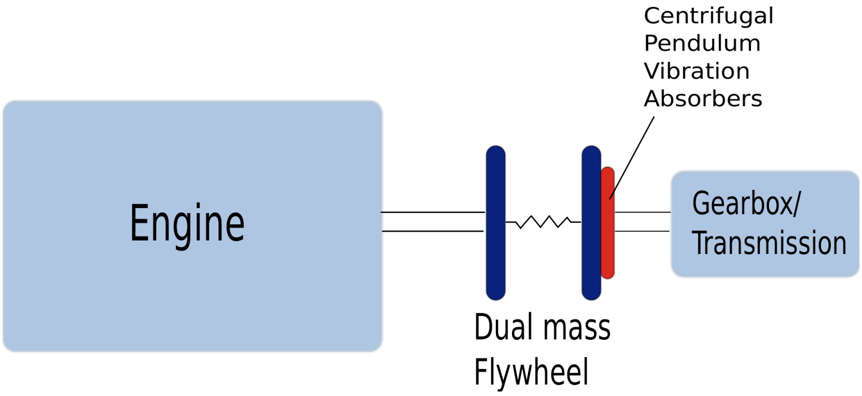


Figure 1.1: *Simple sketch of a modern Powertrain with Engine, DMF, and Gearbox. CPVAs are attached to the DMF on the gearbox side.*

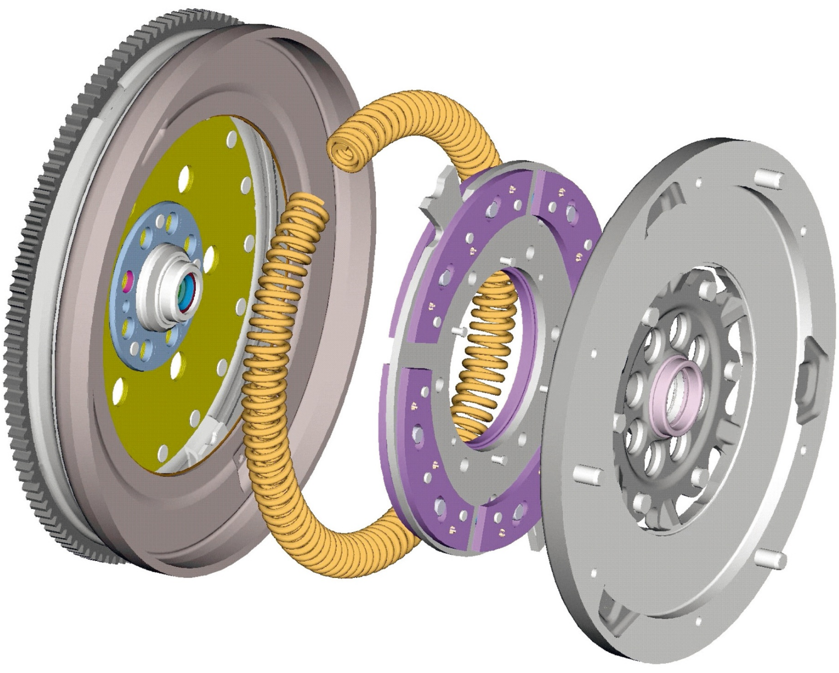


Figure 1.2: *Dual Mass Flywheel (DMF) from German manufacturer LuK (Printed with permission from LuK GmbH & Co. KG).*

1.4 Thesis Purpose and Goals

The purpose of this thesis work is to increase the understanding of Centrifugal Pendulum Vibration Absorbers (CPVAs) at Volvo Cars, regarding basic operation and function, stability, performance, types, application, and modeling capabilities. Of special importance is to create simulation models of CPVAs to understand their behavior, and create models of realistic powertrains to understand the vibration reduction capabilities of CPVAs in the car engine.

The goals of this thesis are to:

- Create models, in Simdrive, of several different types of CPVAs, including cycloidal and epicycloidal path types
- Create a simulation environment for verification, in Simdrive, where created CPVA models can be compared with already existing Simdrive CPVA models
- Create a simulation environment, in Simdrive, where CPVAs are subjected to a sine-signal torque
- Create a simulation environment, in Simdrive, with realistic engine torque, cylinders, FEAD (Front Engine Accessory Drives), and DMF (Dual Mass Flywheel) with CPVAs
- Produce simulation results that reveal the potential of CPVAs
- Investigate previous work in the field of CPVAs and gather the most important theory

1.5 Delimitations

The most important delimitations, in this thesis work, are the following:

- Only torsional vibrations will be considered, not linear – CPVAs are implemented to reduce torsional vibrations, only the rotational vibrations will be considered
- The analysis will be restricted to 1-D
- Gravitational/potential energy and the Coriolis effect is assumed to be small compared to kinetic energy, and are neglected in the models
- Unison motion – usually, several absorbers are used on a single CPVA unit. In this study, they are assumed to move in unison, which is not always the case in reality
- Only the primary harmonic order will be considered – in reality CPVAs tuned to several different orders might have to be used to suppress irregularities
- Rigid body assumption – only rigid body dynamics is used in this work, no elements have structural flexibility in the models
- Only four-cylinder engines will be considered – the work will be limited to four-cylinder engines with a dominant second order harmonic component
- An overall vibration analysis of the powertrain will not be performed in detail, the focus will be on how CPVAs can reduce vibrations at their position

1.6 Methodology

The method used in this work to model the CPVA unit is based on Lagrangian mechanics, where two coupled equations of motion are found for the rotor and pendulum separately. To be able to solve the equations of motion numerically, the equations are decoupled and written as a system of first-order ordinary differential equations. These equations may be solved in Matlab, but for Simdrive simulation this system is solved for and formulated in C/C++ code. The CPVA unit in Simdrive is programmed manually in C/C++, to allow for more types and adjustable parameters.

The simulation environments created consists of a *Verification model*, a *Sine-signal torque model*, and a more realistic *Advanced powertrain model*. The verification model is created to compare the manually created CPVA units with already existing CPVA models in Simdrive. The sine-signal torque model is a simplified powertrain model, where only the dominant order harmonic torque is included. This model is fast to simulate and good for estimation of CPVA parameters. The realistic model is created to capture the behavior of a real powertrain, and give results that are useful to roughly dimension CPVAs for car engines.

1.7 Simdrive 3D and other used Software

The Java-based Simdrive 3D software is provided by German company *CONTECS Engineering Services GmbH*[1] and is used for simulations in this work. Simdrive has a simple-to-use graphical interface where simple models can be built quickly, for first predictions. Simulations also run fast, and allow for quick design changes and improvements. The results are presented against time and rotary speed, as well as order analysis, which is convenient. Simdrive has several expansion modules, such as structural analysis, transmission and engine components etc., so the possibilities to expand models to higher levels of complexity are good.

Other software that has been used for this thesis includes Matlab by MathWorks, Mathematica by Wolfram Research, and Visual Studio 2010 by Microsoft. The report has been typeset in LaTeXwith BibDesk to handle references.

1.8 Thesis Overview

The reminding chapters of this thesis have the following contents. Chapter 2 contains the theory that is necessary to understand the behavior of CPVAs, and how to derive equations of motion for them. The basic function, tuning, stability, performance and effects of general paths, such as cycloids and epicycloids, are described. In section 2.7, the CPVAs used in this work are described, and their equations of motion.

Chapter 3 comprises a literature review, where previous work is presented and analyzed briefly. In Chapter 4, the methodology used is described in detail. The modeling of the CPVA unit is treated, as well as the modeling in Simdrive. In Chapter 5, results of the Sine-signal torque model and Advanced model are presented. In Chapter 6, the results are discussed, and in Chapter 7 conclusions are drawn about the findings in this study, some recommendations for the future are also given.

[1] See http://www.contecs-engineering.de for more information.

2 THEORETICAL FOUNDATION

This chapter contains the most important theory, necessary for modeling and simulation of CPVAs. In section 2.1, the sources of vibrations in engines are described briefly. Section 2.2 describe how equations of motion, through Lagrangian formulation, may be formed from expressions for kinetic and potential energy. In section 2.3, the reformulation of equations of motion to a system of first-order system of ordinary differential equations is described, which is necessary for numerical simulation in Matlab or Simdrive. The following sections, 2.4, 2.5, and 2.6 describe the basic theory and operation of the CPVA, and how its behavior can be altered through different methods. These sections are much intertwined; therefore it is good to understand all three of them. Section 2.7 contains detailed descriptions of CPVAs modeled in this work, and derivations of their equations of motion.

2.1 Vibrations in Multicylinder Engines

There are a number of sources that cause vibration and parts in the engine that affect the vibration characteristics throughout the engine. In this work though, consideration will be given only to the torsional vibrations in the crankshaft, see Figure 2.1. These vibrations are induced by:

- Inertia in moving parts such as pistons and rods
- Combustion induced gas pressure in cylinders

In this work, only the gas pressure effects will be considered since the CPVA unit aims at reducing those irregularities. The combustion gas pressure torque of the first cylinder in a typical four-cylinder engine is shown in Figure 2.2. The figure shows that a combustion pulse occurs once every two revolutions of the crankshaft. If all four cylinders are included, there will be four combustion pulses every two revolutions of the crankshaft, or two pulses in every revolution. This implies that, for a four-stroke engine, the engine will produce vibrations with a dominant excitation harmonic of half the number of cylinders.

In this work, only four-cylinder engines will be considered why the dominant harmonic is of *second order*. If six-cylinder engines are considered, the dominant harmonic will be of third order, eight-cylinder implies fourth order etc. The CPVA units will therefore be tuned close to second order harmonic. CPVAs tuned to other order torques can of course be used with the four-cylinder engines, but the second order component is considered to be the most important to reduce. Therefore, only CPVAs tuned to second order will be used in this work, [5].

2.2 Lagrangian Formulation

For a system such as the Centrifugal Pendulum Vibration Absorber (CPVA), it is suitable to use the Lagrangian formulation in generalized coordinates, developed by Joseph Louis Lagrange in 1788. First, the Cartesian coordinates are expressed in a set of generalized coordinates. In the case of the Simple type CPVA (described in section 2.7.1, Figure 2.6) as an example, they are chosen as

$$x_1 = x_1(q_1, q_2, t) \tag{2.1}$$

$$x_2 = x_2(q_1, q_2, t) \tag{2.2}$$

where $q_1 = \theta$ is the rotor angle, $q_2 = \phi$ is the pendulum angle relative to the pendulum attachment point on the rotor. A detailed derivation on how to formulate the equations of motion will not be included here, but it may be shown that the equations of motion can be expressed as

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_j}\right) - \frac{\partial T}{\partial q_j} + \frac{\partial V}{\partial q_j} = 0, \tag{2.3}$$

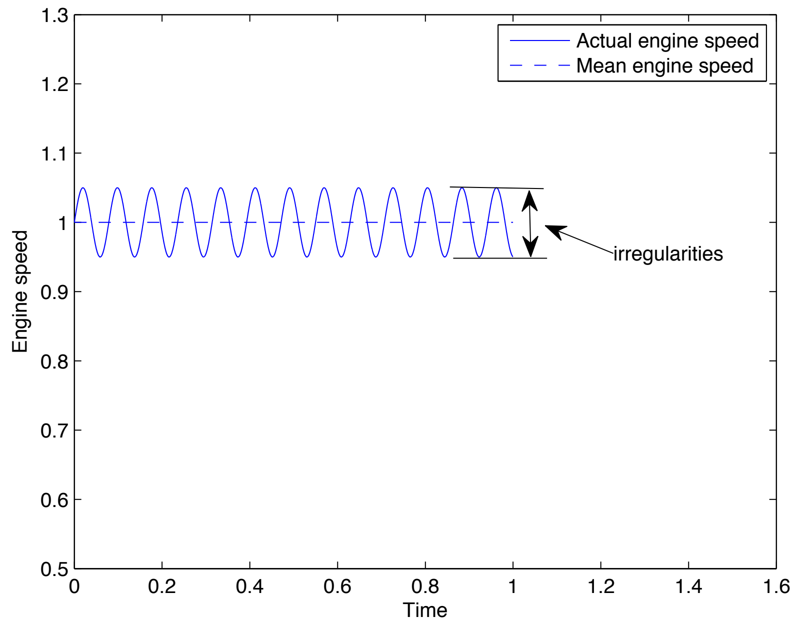


Figure 2.1: *A simple illustration of torsional vibrations in the engine. It is these irregularities that the CPVAs are supposed to reduce.*

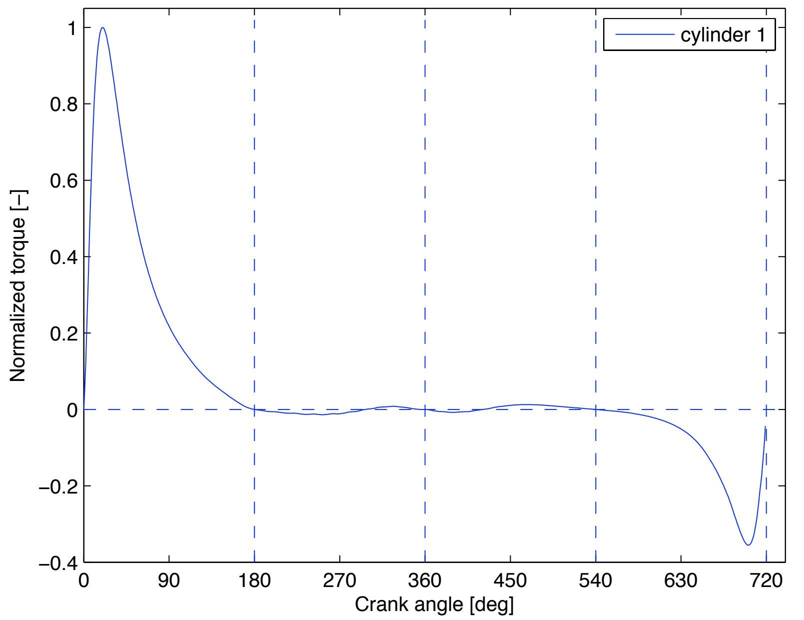


Figure 2.2: *An example of the combustion gas pressure torque in the first cylinder in a four-cylinder engine.*

or

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_j}\right)-\frac{\partial L}{\partial q_j}=0 \tag{2.4}$$

where

$$L=T-V \tag{2.5}$$

is named the Lagrangian. T is the kinetic energy of the system, V is the potential energy of the system, and q_j and $\dot{q}_j$ are the generalized coordinates and generalized velocities. These equations form a system of coupled second order linear differential equations, called the Lagrangian equations of motion of the second kind. For the case of the Simple CPVA system, they are:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right)-\frac{\partial L}{\partial \theta} = 0 \tag{2.6}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\phi}}\right)-\frac{\partial L}{\partial \phi} = 0. \tag{2.7}$$

These equations are used to describe the dynamics of all the types of CPVAs treated in this work, [6, 7].

2.3 Equations of Motion as a System of First-Order Ordinary Differential Equations

The equations of motion, derived in section 2.2, that describe the dynamics of the CPVA are of second order and can not be inserted directly into a differential equation solver, such as the Runge-Kutta method solver *ode45* in *Matlab*. It is therefore necessary to introduce new variables for the time derivatives and formulate the equations of motion as a system of first-order ODEs (Ordinary Differential Equations). The Lagrangian equations of motion, as in Equations (2.6), (2.7), are usually on this form:

$$k_1(\phi)\,\ddot{\theta}+k_2(\phi)\,\ddot{\phi}+k_3(\phi)\,\dot{\theta}^2+k_4(\phi)\,\dot{\phi}^2+k_5(\phi)\,\dot{\theta}\dot{\phi} = 0 \tag{2.8}$$

$$k_6(\phi)\,\ddot{\theta}+k_7(\phi)\,\ddot{\phi}+k_8(\phi)\,\dot{\theta}^2+k_9(\phi)\,\dot{\phi}^2+k_{10}(\phi)\,\dot{\theta}\dot{\phi} = 0. \tag{2.9}$$

To reduce this equation system to first order, new variables are introduced as

$$\mathbf{y}(t)=\begin{bmatrix}\theta(t)\\ \dot{\theta}(t)\\ \phi(t)\\ \dot{\phi}(t)\end{bmatrix} \tag{2.10}$$

and the time derivative of $\mathbf{y}(t)$ is consequently:

$$\frac{d}{dt}(\mathbf{y}(t))=\dot{\mathbf{y}}(t)=\begin{bmatrix}\dot{\theta}(t)\\ \ddot{\theta}(t)\\ \dot{\phi}(t)\\ \ddot{\phi}(t)\end{bmatrix}. \tag{2.11}$$

Now, the equation system is formulated as

$$\mathbf{A}\,\dot{\mathbf{y}}(t)=\mathbf{f}. \tag{2.12}$$

With Equations (2.8), (2.9) and notations in Equation (2.10), the matrices become:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & k_1\left(y\left(3\right)\right) & 0 & k_2\left(y\left(3\right)\right) \\ 0 & 0 & 1 & 0 \\ 0 & k_6\left(y\left(3\right)\right) & 0 & k_7\left(y\left(3\right)\right) \end{bmatrix}$$

$$\dot{\mathbf{y}}\left(t\right) = \begin{bmatrix} \dot{\theta}\left(t\right) \\ \ddot{\theta}\left(t\right) \\ \dot{\phi}\left(t\right) \\ \ddot{\phi}\left(t\right) \end{bmatrix} \tag{2.13}$$

$$\mathbf{f} = \begin{bmatrix} y\left(2\right) \\ -\left(k_3\left(y\left(3\right)\right)y\left(2\right)^2 + k_4\left(y\left(3\right)\right)y\left(4\right)^2 + k_5\left(y\left(3\right)\right)y\left(2\right)y\left(4\right)\right) \\ y\left(2\right) \\ -\left(k_3\left(y\left(3\right)\right)y\left(2\right)^2 + k_4\left(y\left(3\right)\right)y\left(4\right)^2 + k_5\left(y\left(3\right)\right)y\left(2\right)y\left(4\right)\right) \end{bmatrix}. \tag{2.14}$$

The formulation has now been transformed into a first-order system of differential equations and may be inserted into *Matlab* and solved with e.g. *ode45*. For purposes of simulation in *Simdrive* the second time derivatives should be given explicitly. Vector $\dot{\mathbf{y}}\left(\mathbf{t}\right)$ then needs to solved for as:

$$\dot{\mathbf{y}}\left(t\right) = \mathbf{A}^{-1}\mathbf{f}. \tag{2.15}$$

The accelerations, $\ddot{\theta}$ and $\ddot{\phi}$, can then be extracted in explicit form, from $\dot{\mathbf{y}}\left(t\right)$.

The differential equations are then decoupled and formulated as a first-order system, i.e. functions of $\theta, \dot{\theta}, \phi, \dot{\phi}$ only. Vector $\dot{\mathbf{y}}\left(t\right)$ is thus on this form:

$$\dot{\mathbf{y}}\left(t\right) = \begin{bmatrix} \dot{\theta}\left(t\right) \\ \ddot{\theta}\left(t\right) \\ \dot{\phi}\left(t\right) \\ \ddot{\phi}\left(t\right) \end{bmatrix} = \begin{bmatrix} \dot{\theta}\left(t\right) \\ f\left(\theta, \dot{\theta}, \phi, \dot{\phi}, t\right) \\ \dot{\phi}\left(t\right) \\ g\left(\theta, \dot{\theta}, \phi, \dot{\phi}, t\right) \end{bmatrix}. \tag{2.16}$$

For coding in C/C++ which is the format for *Simdrive*, the generalized coordinates and their time derivatives are stored in two-dimensional arrays. First-order ODE variables are stored in $Yn1$ and second-order ODE variables are stored in $Yn2$. The dynamics of the CPVA requires two second-order differential equations. Consequently the variables, in C/C++ coding, are stored as:

$$\begin{aligned} \theta &= Yn2[0][0] \\ \dot{\theta} &= Yn2[0][1] \\ \ddot{\theta} &= Yn2[0][2] \\ \phi &= Yn2[1][0] \\ \dot{\phi} &= Yn2[1][1] \\ \ddot{\phi} &= Yn2[1][2]. \end{aligned} \tag{2.17}$$

Thus,

$$\ddot{\theta} = Yn2[0][2] = f\left(Yn2[0][0], Yn2[0][1], Yn2[1][0], Yn2[1][1]\right) \tag{2.18}$$

$$\ddot{\phi} = Yn2[1][2] = g\left(Yn2[0][0], Yn2[0][1], Yn2[1][0], Yn2[1][1]\right). \tag{2.19}$$

This method is used for all types of CPVAs modeled in this work, described in section 2.7.

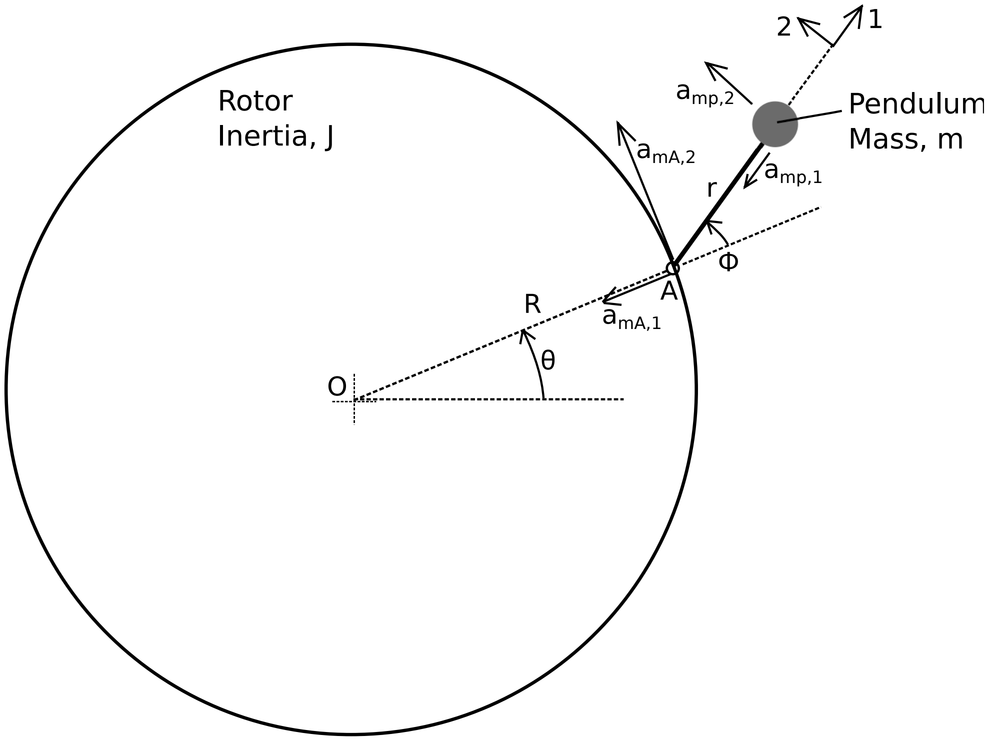


Figure 2.3: *CPVA with point-mass pendulum. The pendulum is fixed to the rotor at point A. θ is the rotor angle and ϕ is the pendulum angle. Accelerations of point A and the pendulum are shown.* [8]

2.4 Basic Function and Tuning of CPVAs

To describe the basic functionality of the CPVA, it is convenient to study the simplest type with a point-mass pendulum, of mass m, that is attached to a rotor of rotational inertia J. The pendulum is allowed to rotate freely about its attachment point. This is illustrated in Figure 2.3, where the components of attachment point (A) and pendulum mass accelerations also are shown. The assumption is made, as in section 1.5, that *gravitational* and *Coriolis* forces can be neglected. That assumption has been made in this entire work. No investigation is performed in this work to analyze the validity of this assumption. It is however common in the field to make this assumption. The acceleration components are, from Figure 2.3:

$$\begin{aligned}
a_{mA,1} &= R\dot{\theta}^2 \\
a_{mA,2} &= R\ddot{\theta} \\
a_{mp,1} &= r\left(\dot{\theta}+\dot{\phi}\right)^2 \\
a_{mp,2} &= r\left(\ddot{\theta}+\ddot{\phi}\right).
\end{aligned} \tag{2.20}$$

The acceleration of the pendulum mass is the vector sum of the acceleration of A and the acceleration of the pendulum mass relative to A. From geometry in Figure 2.3 the acceleration vector is found to be:

$$\begin{aligned}
\mathbf{a}_m = &[-a_{mp,1} - a_{mA,1}\cos(\phi) + a_{mA,2}\sin(\phi), \\
&a_{mp,2} + a_{mA,2}\cos(\phi) + a_{mA,1}\sin(\phi)]
\end{aligned} \tag{2.21}$$

and with expressions in Equation (2.20):

$$\mathbf{a}_m = [-r\left(\dot{\theta}+\dot{\phi}\right)^2 - R\dot{\theta}^2\cos(\phi) + R\ddot{\theta}\sin(\phi), \quad r\left(\ddot{\theta}+\ddot{\phi}\right) + R\ddot{\theta}\cos(\phi) + R\dot{\theta}^2\sin(\phi)]. \tag{2.22}$$

Using the fact that the moment about A is zero, the following expression can be formed, using only the second component (in direction 2) of $\mathbf{a}_m$:

$$M_A = m\left[r\left(\ddot{\theta}+\ddot{\phi}\right) + R\ddot{\theta}\cos(\phi) + R\dot{\theta}^2\sin(\phi)\right]r = 0. \tag{2.23}$$

Making the assumption that pendulum angles are small ($\cos(\phi) \approx 1$, $\sin(\phi) \approx \phi$), and that the rotor has a steady rotational speed Ω with small sinusoidal fluctuations of amplitude θ_0 and frequency ω, leads to the following differential equation for the pendulum:

$$\ddot{\phi} + \left(\frac{R}{r}\Omega^2\right)\phi = \left(\frac{R+r}{r}\right)\omega^2\theta_0\sin(\omega t). \tag{2.24}$$

With these assumptions, it is now possible to analyze the CPVA behavior. The natural frequency of the pendulum is easily found to be:

$$\omega_n = \Omega\sqrt{\frac{R}{r}} \tag{2.25}$$

and this shows perhaps the most important property of the CPVA – the natural frequency of the pendulum changes proportionally with the rotational speed of the rotor. This property is very useful in engine applications, where the frequency of the exerted torque is proportional to the rotational speed of the engine. The CPVA can then be thought of as a pendulum suspended by a spring that changes its stiffness with rotational speed. During operation, the *centrifugal field*, $R\Omega^2$, dominates over the *gravity field*, g, why the potential energy may be neglected in the Lagrangian equations of motion. Whether this assumption can be made or not depends on the operating speed, for engine operating speeds it is assumed to be valid.

Further, by neglecting minor terms of $\mathbf{a}_m$ the torque from the pendulum on the rotor may be formulated as

$$T = -m(R+r)\Omega^2 R\phi \tag{2.26}$$

and with the steady-state solution of Equation (2.24) the torque may be written as

$$T = -\left(\frac{m(R+r)^2}{1-\frac{r\omega^2}{R\Omega^2}}\right)\ddot{\theta}. \tag{2.27}$$

The pendulum can be compared to a flywheel with inertia J_e. In that case:

$$J_e = -\frac{m(R+r)^2}{1-\frac{r\omega^2}{R\Omega^2}}. \tag{2.28}$$

Recognizing that $\omega^2/\Omega^2 = n^2$ where n is the harmonic order of the applied fluctuating torque, the denominator becomes $1 - n^2\frac{r}{R}$. When the denominator is zero, the effective inertia of the pendulum will be infinite, and it is found that this condition is met when $n = \sqrt{R/r}$. This is then the resonance tuning condition for the simple point-mass CPVA, when the pendulum oscillation angles are small and the rotor is rotating at a steady speed with small sinusoidal fluctuations.

In order to tune the CPVAs correctly, it might be good to be aware of and understand these tuning concepts:

- *Resonance tuning* – Resonance occurs when the order of the applied torque is exactly the same as the order of the tuned pendulum assembly. The effective inertia of the pendulum is infinite with respect to a certain order of vibration.

- *Overtuning (positive mistuning)* – Overtuning occurs when the pendulum assembly is tuned to a *higher* order number than the order of the applied torque. The effective inertia of the pendulum assembly in this case is positive. When the order tuning number of the pendulum assembly is close to the resonance condition, the inertia is large, then inertia decreases with increased overtuning. This condition is used in reality, since both good performance and stability can be achieved.

- *Undertuning (negative mistuning)* – Undertuning occurs when the pendulum assembly is tuned to a *lower* order number than the order of the applied torque. The effective inertia of the pendulum is in this case negative, and therefore not suitable as a vibration absorber. This is usually not a desirable condition.

It is suitable to mention here that positive mistuning is used in practice also to improve stability of CPVAs. It may seem that resonance tuning should be the best alternative, but in practice a little mistuning must be used to ensure a stable and reliable operation of the CPVA. Stability of CPVAs will be covered in more detail in section 2.6, [2, 8].

2.5 Cycloids, Epicycloids and Tautochrones

The type of CPVA described in the previous section is constructed in such a way that the path the pendulum travels, about the attachment point, will be circular. This has proven to work quite well, but there are some issues with a circular path, both in a constant gravity field and in a centrifugal field. The main problem is that the natural frequency of the pendulum with circular path will change with pendulum amplitude. It has a softening nonlinear behavior, which essentially means that the natural frequency of the pendulum will decrease with increased pendulum amplitude. This will reduce the performance of the CPVA. The reason for this is that it might become undertuned for high amplitudes.

The method from previous work, such as Denman [9], is therefore to change the circular path into a generalized path that can allow for better performance and stability of the CPVA. Of special interest for these path representations are the cycloid and epicycloid curve families.

A cycloid curve is obtained by rolling a circle on a straight line, see Figure 2.4. An epicycloid is similar, but the curve is obtained by rolling a circle around another circle, see Figure 2.5.

The equations for the parameterized cycloid coordinates are:

$$\begin{aligned} x &= a\,(t - \sin t) \\ y &= a\,(1 - \cos t) \end{aligned} \tag{2.29}$$

where a is the radius of the rolling circle, the equations will complete a cycloid every $t = 2\pi$.

The epicycloid is parameterized with equations:

$$\begin{aligned} x &= (a+b)\cos\phi - b\cos\left(\frac{a+b}{b}\phi\right) \\ y &= (a+b)\sin\phi - b\sin\left(\frac{a+b}{b}\phi\right) \end{aligned} \tag{2.30}$$

where a is the radius of the fixed inner circle, and b is the radius of the rolling circle. The ratio of the inner and rolling circle is $k = a/b$, defines the number of epicycloids drawn in one revolution

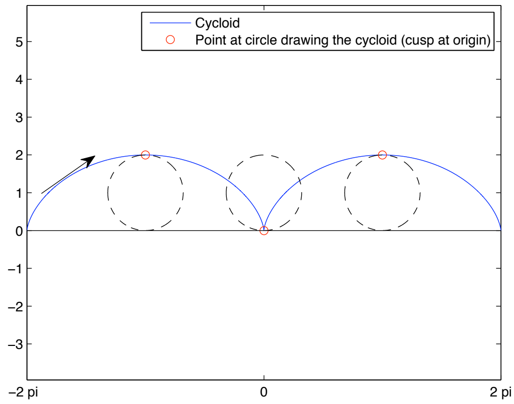
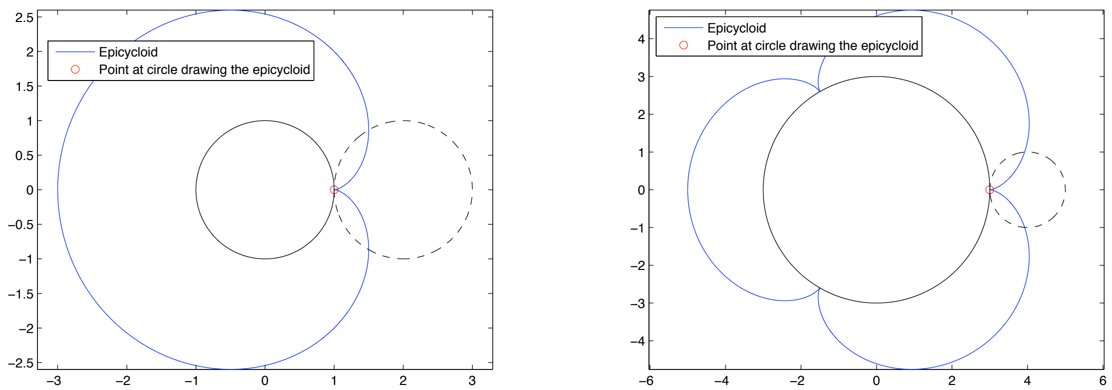


Figure 2.4: *Description of how a cycloid is created. A cycloid is constructed by rolling a circle of radius a on a straight line. A new cycloid curve is created every $t = 2\pi$, with cusps in the limit. Here $a = 1$.*



(a) (b)

Figure 2.5: *Description of how an epicycloid is created. An epicycloid is constructed by rolling a circle around a fixed circle. (a) Epicycloid with $k = a/b = 1$. (b) Epicycloid with $k = a/b = 3$.*

around the fixed circle, as in Figure 2.5. If k is an integer, the curves drawn will be identical for every revolution.

These type of curves may be adjusted so that the path has less softening non-linear behavior, than the circular path. As mentioned earlier, a circular path of the pendulum will cause the pendulum to behave differently when the pendulum amplitude is large. Its natural frequency will change with the amplitude, which is unfavorable due to stability issues, as will be explained further in section 2.6. The ideal path for the pendulum should be one where the frequency is independent of the amplitude. Such a path is called *isochronous* or more commonly *tautochronous* and such a path allows for a pendulum that behaves entirely linear, i.e. has the same frequency regardless of amplitude. The path will have neither softening nor hardening behavior. The path may also be adjusted to have hardening non-linear behavior, i.e. increased natural frequency for increased amplitude. The following statements hold for these type of paths:

- A tautochronous motion in a constant *gravitational field* is a certain *cycloid*
- A tautochronous motion in a constant *centrifugal field* is a certain *epicycloid*

The conclusion can then be made that epicycloidal paths are particularly interesting for CPVA applications since the centrifugal field is dominant there.

A derivation of the tautochronous cycloidal motion results in this equation of motion:

$$\ddot{s} = -\frac{g}{4l}s \tag{2.31}$$

where s is the arc length variable of the path, g is the gravitation constant and l is the pendulum length. It can be seen that it is independent of amplitude and linear.

From the book of Rana and Joag [10], the tautochronous motion in a centrifugal field is described by the following equation of motion:

$$\ddot{s} + \frac{\omega^2 s}{\frac{4b}{a}\left(1+\frac{b}{a}\right)} = 0 \tag{2.32}$$

with

$$s = \frac{\sqrt{\left((a+2b)^2 - r^2\right)\left((a+2b)^2 - a^2\right)}}{a}$$

where ω is the angular speed of the disc, r is the radial distance from the centre of the fixed circle to path, and a, b are defined as earlier. This motion will follow the path of a certain epicycloid, [9, 10, 11].

2.6 Stability and Performance of CPVAs

An understanding of the behavior of the CPVA with regard to stability is essential to ensure reliable operation and good performance. It is good to be aware of two types of instabilities when designing CPVAs with either just a single absorber or several absorbers. These are the

- *jump* bifurcation (affects every individual absorber)
- bifurcation to *non-unison* (several absorbers lose their synchronous character)

The first type, jump bifurcation, normally occurs when the amplitude of the absorber reaches a certain moderate level caused by the applied fluctuating torque. The absorber jumps to another solution through a saddle-node bifurcation, where the absorber has a behavior that is highly undesirable for practical applications – it enforces vibrations. This behavior is hysteretic which means that the applied fluctuating torque has to be reduced to a significantly lower level for the absorber to jump back, than was originally required to cause the jump. It is therefore very important to prevent the jump condition.

The classic approach to avoid the jump condition is to intentionally mistune the CPVA in relation to the applied torque. The use of positive mistuning or overtuning, as explained in section 2.4, will increase stability since the jump condition will occur at a higher level of applied torque and consequently at higher absorber amplitude. An increase in mistuning will gradually decrease the performance of the CPVA, why a suitable compromise should be found between performance and torque range.

The second type, bifurcation to non-unison, is an instability experienced in CPVA arrangements with several absorbers. Usually, the desired condition is that all absorbers move in a synchronous manner. This should ensure maximum performance and stability. The bifurcation to non-unison can lead to several types of operation modes for the CPVA, where certain absorbers may be active and others inactive. These differences in behavior depend on the levels of mistuning and damping of the individual absorbers.

Another way to affect the stability of the CPVA unit is to force the pendulum to travel on a cycloidal or epicycloidal path, as explained in section 2.5. These paths are preferred compared to the regular circular path, since they experience neither the jump nor the non-unison bifurcation. Consequently, their stability is better and also their operating torque range. To summarize, positive mistuning and cycloidal or epicycloidal paths are powerful tools for improving the stability of CPVAs, [12, 13, 14, 15, 16, 17].

2.7 Description of CPVAs Modeled in This Work

This section contains descriptions of all CPVA types investigated in this work. For the first five types, the pendulum mass m_p, pendulum moment of inertia I_{pend}, and the number of pendulums on the CPVA unit are given as inputs in simulations. These quantities are added together to form one CPVA unit with total pendulum mass m and total moment of inertia I. These values are then used in the equations of motion. This method can be used since it is assumed that all pendulums move in unison. In the last type (section 2.7.6), the CPVA unit is considered to have only one pendulum, so m_p and I_{pend} correspond to m and I.

2.7.1 Simple Type with Circular Path

The Simple type of CPVA, with circular path, is suitable for first investigations. It is not common for use in practice though since there are other types with better properties. The equations of motion will be derived according to the Lagrangian formulation described in section 2.2. The kinetic and potential energy of the system is needed for the equations of motion.

From Figure 2.6, the kinetic energy T for the CPVA unit is:

$$T = \frac{1}{2}m|\mathbf{v}|^2 + \frac{1}{2}J\dot{\theta}^2 + \frac{1}{2}I\left(\dot{\theta} + \dot{\phi}\right)^2 \tag{2.33}$$

where $\mathbf{v}$ is the velocity vector of the centre of mass of the pendulum, J is the moment of inertia of the rotor, and I is the moment of inertia of the pendulum.

Further, the potential energy of the CPVA is needed. It will be shown later that the kinetic energy will be dominant, and the potential energy can be neglected during CPVA operation, but it will be included in the Simple type CPVA derivation to show the general method. In simulations, the potential energy will not be included. The potential energy V, also derived from Figure 2.6, is:

$$V = mg\left(R\sin\left(\theta\right) + r\sin\left(\theta + \phi\right)\right). \tag{2.34}$$

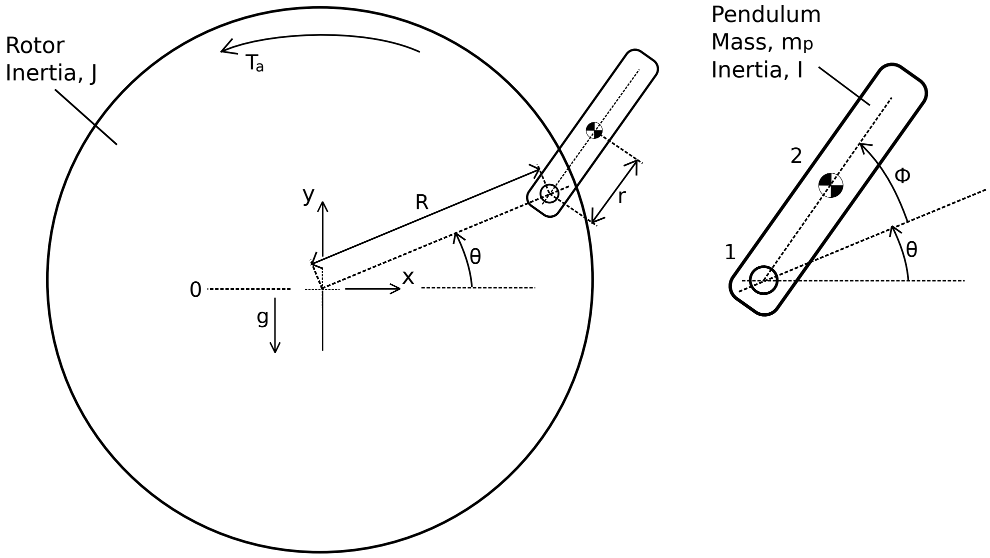


Figure 2.6: *Free body diagram of the Simple Type CPVA unit with circular path. The rotor has inertia J and the pendulum inertia I. The rotor and pendulum angles are θ and ϕ respectively.* [18]

The coordinates for the CPVA, from Figure 2.6, are:

$$x_1 = R\cos(\theta) \tag{2.35}$$

$$y_1 = R\sin(\theta) \tag{2.36}$$

$$x_2 = R\cos(\theta) + r\cos(\theta+\phi) \tag{2.37}$$

$$y_2 = R\sin(\theta) + r\sin(\theta+\phi)\,. \tag{2.38}$$

Taking the time derivative of Equations (2.37), (2.38), the pendulum centre of mass position, gives:

$$\dot{x}_2 = -R\dot{\theta}\sin(\theta) - r\left(\dot{\theta}+\dot{\phi}\right)\sin(\theta+\phi) \tag{2.39}$$

$$\dot{y}_2 = R\dot{\theta}\cos(\theta) + r\left(\dot{\theta}+\dot{\phi}\right)\cos(\theta+\phi)\,. \tag{2.40}$$

The velocity of the pendulum center of mass is then

$$\mathbf{v} = (\dot{x}_2, \dot{y}_2) \tag{2.41}$$

and the speed is consequently

$$|\mathbf{v}| = \sqrt{\dot{x}_2^2 + \dot{y}_2^2}. \tag{2.42}$$

Substitution of (2.39), (2.40) in (2.42), and expressing the square of the speed, yields:

$$|\mathbf{v}|^2 = 2r\left(R\cos(\phi) + r\right)\dot{\theta}\dot{\phi} + \left(R^2 + r^2 + 2Rr\cos(\phi)\right)\dot{\theta}^2 + r^2\dot{\phi}^2. \tag{2.43}$$

The moment of inertia is expressed as

$$I = mk_\lambda^2 \tag{2.44}$$

where k_λ is the radius of gyration. Substitution of (2.43), (2.44) in (2.33) yields:

$$\begin{aligned} T &= \frac{1}{2}(J\dot{\theta}^2 + mk_\lambda^2\left(\dot{\theta}+\dot{\phi}\right)^2 + m[(R\dot{\theta}\cos(\theta) + r\left(\dot{\theta}+\dot{\phi}\right)\cos(\theta+\phi))^2 \\ &\quad +(R\dot{\theta}\sin(\theta) + r\left(\dot{\theta}+\dot{\phi}\right)\sin(\theta+\phi))^2]). \end{aligned} \tag{2.45}$$

The Lagrangian, previously defined in section 2.2 as $L = T - V$, is for this case:

$$\begin{aligned} L &= \frac{1}{2}(J\dot{\theta}^2 + mk_\lambda^2\left(\dot{\theta}+\dot{\phi}\right)^2 + m[(R\dot{\theta}\cos(\theta) + r\left(\dot{\theta}+\dot{\phi}\right)\cos(\theta+\phi))^2 \\ &\quad +(R\dot{\theta}\sin(\theta) + r\left(\dot{\theta}+\dot{\phi}\right)\sin(\theta+\phi))^2]) \\ &\quad -mg\left(R\sin(\theta) + r\sin(\theta+\phi)\right). \end{aligned} \tag{2.46}$$

The Lagrangian equations of motion, as described by Equations (2.6), (2.7), for this case are

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\theta}}\right) - \frac{\partial L}{\partial\theta} = T_a - c_s\dot{\theta} \tag{2.47}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\phi}}\right) - \frac{\partial L}{\partial\phi} = -c_a\dot{\phi} \tag{2.48}$$

where T_a is the externally applied torque, and c_s, c_a are viscous damping coefficients for rotor and absorber. Using the Lagrangian in (2.46),

$$\begin{aligned} \frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\theta}}\right) &= -mRr\sin(\phi)\dot{\phi}^2 - 2mRr\sin(\phi)\dot{\theta}\dot{\phi} \\ &\quad + \left(mk_\lambda^2 + mr^2 + mRr\cos(\phi)\right)\ddot{\phi} \\ &\quad + \left(J + mk_\lambda^2 + mr^2 + mR^2 + 2mRr\cos(\phi)\right)\ddot{\theta} \end{aligned} \tag{2.49}$$

$$\frac{\partial L}{\partial\theta} = -mg\left(R\cos(\theta) + r\cos(\theta+\phi)\right) \tag{2.50}$$

$$\begin{aligned} \frac{d}{dt}\left(\frac{\partial L}{\partial\dot{\phi}}\right) &= -mRr\sin(\phi)\dot{\theta}\dot{\phi} + \left(mk_\lambda^2 + mr^2\right)\ddot{\phi} \\ &\quad + \left(mk_\lambda^2 + mr^2 + mRr\cos(\phi)\right)\ddot{\theta} \end{aligned} \tag{2.51}$$

$$\frac{\partial L}{\partial\phi} = -mr\left(g\cos(\theta+\phi) + R\sin(\phi)\dot{\theta}\dot{\phi} + R\sin(\phi)\dot{\theta}^2\right). \tag{2.52}$$

Substitution of (2.49) – (2.52) in (2.47) and (2.48) results in the final equations of motion:

$$\begin{aligned} & - mRr\sin(\phi)\dot{\phi}^2 - 2mRr\sin(\phi)\dot{\theta}\dot{\phi} \\ & + \left(mk_\lambda^2 + mr^2 + mRr\cos(\phi)\right)\ddot{\phi} \\ & + \left(J + mk_\lambda^2 + mr^2 + mR^2 + 2mRr\cos(\phi)\right)\ddot{\theta} \\ & + mg\left(R\cos(\theta) + r\cos(\theta+\phi)\right) = T_a - c_s\dot{\theta} \end{aligned} \tag{2.53}$$

$$\begin{aligned} & - mRr\sin(\phi)\dot{\theta}\dot{\phi} + \left(mk_\lambda^2 + mr^2\right)\ddot{\phi} \\ & + \left(mk_\lambda^2 + mr^2 + mRr\cos(\phi)\right)\ddot{\theta} \\ & + mr\left(g\cos(\theta+\phi) + R\sin(\phi)\dot{\theta}\dot{\phi} + R\sin(\phi)\dot{\theta}^2\right) = -c_a\dot{\phi}. \end{aligned} \tag{2.54}$$

To reduce these equations to a first-order system of ODEs, methods of section 2.3 are used. With these expressions,

$$\begin{aligned} k_1 &= m\left(R^2 + r^2 + k_\lambda^2\right) + J \\ k_2 &= mRr \\ k_3 &= m\left(r^2 + k_\lambda^2\right) \end{aligned}$$

the equation becomes

$$\mathbf{A}\dot{\mathbf{y}}(t) = \mathbf{f} \tag{2.55}$$

with,

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & k_1 + k_2\cos\left(y\left(3\right)\right) & 0 & k_3 + k_2\cos\left(y\left(3\right)\right) \\ 0 & 0 & 1 & 0 \\ 0 & k_3 + k_2\cos\left(y\left(3\right)\right) & 0 & k_3 \end{bmatrix} \tag{2.56}$$

$$\mathbf{f} = \begin{bmatrix} y\left(2\right) \\ T_a - c_s y\left(2\right) + k_2 y\left(4\right)\left(2y\left(2\right)\sin\left(y\left(3\right)\right) + y\left(4\right)\sin y\left(3\right)\right) \\ y\left(4\right) \\ -c_a y\left(4\right) - \left(k_2 y\left(2\right)^2 \sin\left(y\left(3\right)\right)\right) \end{bmatrix}. \tag{2.57}$$

This matrix equation is suitable for simulations of the CPVA unit in Matlab, as in Appendix A.1. For use in the C/C++ file, the equation has to be solved as $\dot{\mathbf{y}}\left(t\right) = \mathbf{A}^{-1}\mathbf{f}$, which gives expressions for the rotor and pendulum angular accelerations. These expressions are inserted in the C/C++ code, as in Appendix B.

Using the assumptions in section 2.4, small pendulum angles, steady rotation of the rotor etc., the natural frequency of the Simple type CPVA is:

$$\omega_n = \Omega\sqrt{\frac{Rr}{\left(r^2 + k_\lambda^2\right)}}. \tag{2.58}$$

The harmonic tuning order of the CPVA is, consequently:

$$\frac{\omega_n}{\Omega} = n = \sqrt{\frac{Rr}{\left(r^2 + k_\lambda^2\right)}} \tag{2.59}$$

and, recognizing that for the point-mass pendulum, $k_\lambda^2 = 0$, which gives $n = \sqrt{\frac{R}{r}}$.

With further analysis of the Simple type CPVA unit it is possible to relate the tuning order n to pendulum mass m and rotor inertia J, with the aim of finding an optimal relationship. The pendulum mass and rotor inertia must be related to each other so that the irregularities in motion may be cancelled out. For example, irregularities in a very heavy rotor may not be cancelled out by very small pendulums. To find the optimal pendulum mass, this relationship may be used:

$$\left(\frac{mR^2}{J + I_{pend}}\right)_{opt} = \frac{3n^4}{2\left(n^2+1\right)^2}. \tag{2.60}$$

For a derivation of this relationship, see the reference below. The tuning and optimal pendulum mass equations are very useful for design purposes, [18].

2.7.2 Bifilar Type

The Bifilar type CPVA unit is quite similar to the Simple type CPVA, but is suspended in two points, see Figure 2.7. This type is commonly used in practice, and has some advantages. For small pendulum angles, the pendulum rotates about the rotor centre. Therefore, the shape and size of the pendulum can be virtually arbitrary. This CPVA can then have large pendulum mass relative to the total weight of the unit, and be effective in terms of both weight and space.

The coordinates are found in the same manner as for the Simple type CPVA, and leads to the following expression for the kinetic energy, T:

$$T = \left(J + I\right)\frac{\dot{\theta}^2}{2} + m\frac{\left|\mathbf{v}\right|^2}{2}. \tag{2.61}$$

The square of the speed of the pendulum centre of mass is

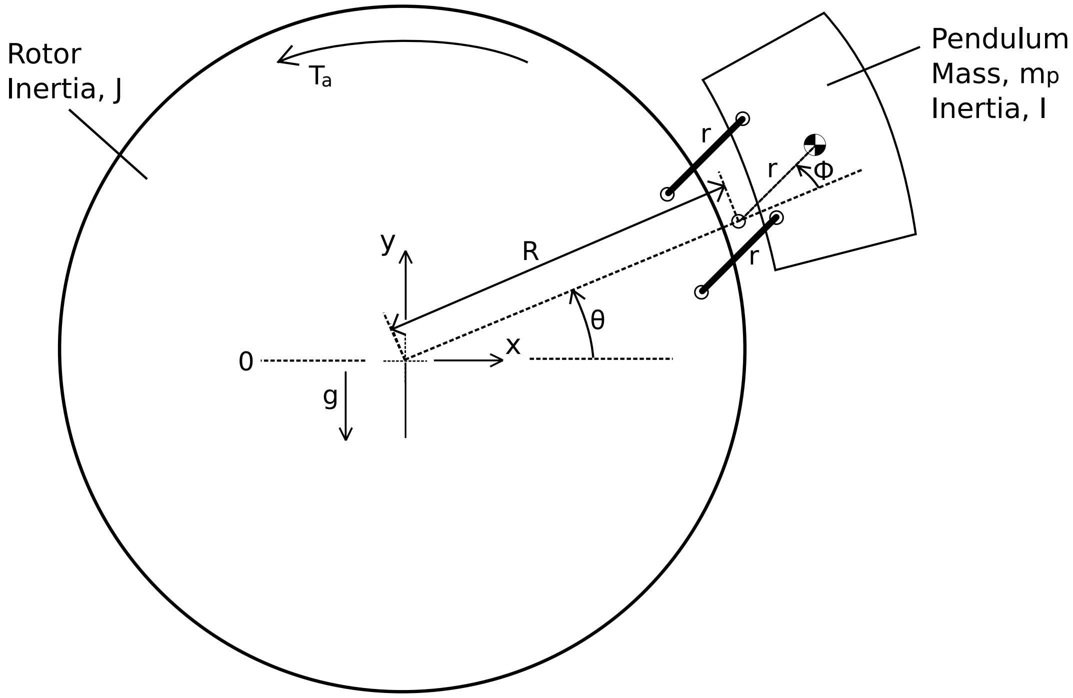


Figure 2.7: *Free body diagram of the Bifilar Type CPVA unit. The rotor has inertia J and the pendulum inertia I. The rotor and pendulum angles are θ and ϕ respectively.* [18]

$$|\mathbf{v}|^2 = \dot{x}_2^2 + \dot{y}_2^2 = \left(R^2 + r^2 + 2Rr\cos\phi\right)\dot{\theta}^2 + r^2\dot{\phi}^2 + \left(2r^2 + 2Rr\cos\phi\right)\dot{\theta}\dot{\phi}. \tag{2.62}$$

Insertion of (2.62) in (2.61) leads to:

$$T = (J+I)\frac{\dot{\theta}^2}{2} + \frac{m}{2}\left(R^2 + r^2 + 2Rr\cos\phi\right)\dot{\theta}^2 + \frac{m}{2}r^2\dot{\phi}^2 + \left(mr^2 + mRr\cos\phi\right)\dot{\theta}\dot{\phi}. \tag{2.63}$$

Using Lagrangian mechanics, as in the case of the Simple type pendulum absorber, the equations of motion are:

$$\begin{aligned} &\left(J + I + mR^2 + mr^2 + 2mRr\cos\phi\right)\ddot{\theta} \\ &+ \left(mr\left(r + R\cos\phi\right)\right)\ddot{\phi} - 2mRr\sin\left(\phi\right)\dot{\theta}\dot{\phi} - mRr\sin\left(\phi\right)\dot{\phi}^2 \quad = \quad T_a - c_s\dot{\theta} \end{aligned} \tag{2.64}$$

$$\left(mr\left(r + R\cos\left(\phi\right)\right)\right)\ddot{\theta} + mr^2\ddot{\phi} + mRr\sin\left(\phi\right)\dot{\theta}^2 \quad = \quad -c_a\dot{\phi} \tag{2.65}$$

The tuning order of the Bifilar type CPVA is found as:

$$n = \sqrt{\frac{R}{r}}. \tag{2.66}$$

The optimal pendulum mass is found in a similar way as with the Simple type CPVA, Equation (2.60), [18].

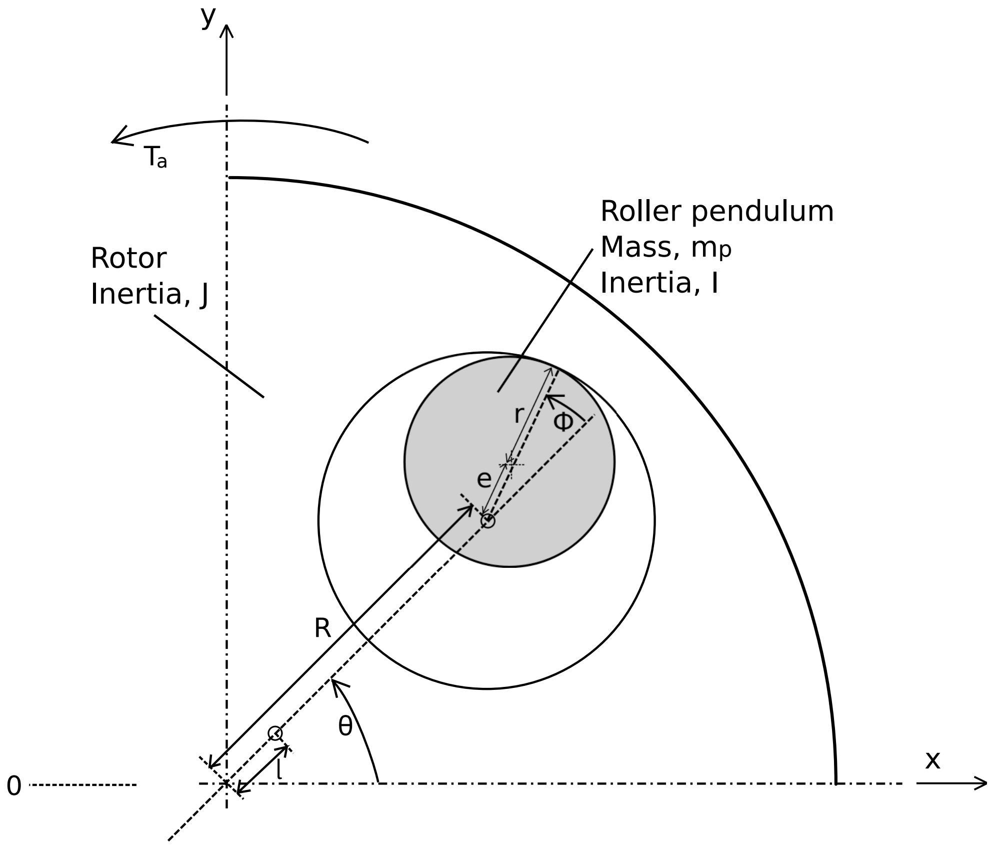


Figure 2.8: *Free body diagram of the Salomon Type CPVA unit. The rotor has inertia J and the roller inertia I. The rotor and roller angles are θ and ϕ respectively.* [19]

2.7.3 Roll Form (Salomon) Type

The Salomon CPVA is a roll-form absorber, the pendulum in this case consists of a solid cylinder roller. The roller rolls around a circular track that is formed by boring a hole at an appropriate location on the rotor, see Figure 2.8. This type is suitable for use in flywheels, gearwheels, crankweb balance weights etc. The layout of the CPVA results in a small active pendulum mass, but if it is implemented in an already existing machine component, it serves its purpose well.

From Figure 2.8, the kinetic energy is found to be:

$$\begin{aligned} T \;=\; & \frac{1}{2}J\dot{\theta}^2 + \frac{1}{2}M_0 l^2\dot{\theta}^2 + \frac{1}{2}I\left(\dot{\theta} - \frac{e}{r}\dot{\phi}\right)^2 + \frac{1}{2}m\left(R^2 + 2\,R\,e\cos\phi + e^2\right)\dot{\theta}^2 \\ & + \frac{1}{2}m\,e^2\dot{\phi}^2 + m\,e\left(R\cos\phi + e\right)\dot{\theta}\dot{\phi} \end{aligned} \tag{2.67}$$

where e is the distance between the bored hole centre and the roller centre, and I is the roller inertia. It is assumed that the roller constantly is in contact with the hole wall during operation.

Using Lagrangian mechanics, as in the case of the Simple type pendulum absorber, the equations of motion are:

$$\begin{aligned} & r\left(J + I + me^2 + M_0 l^2 + mR^2 + 2mRe\cos(\phi)\right)\ddot{\theta} \\ & \qquad + e\left(-I + mre + mRr\cos(\phi)\right)\ddot{\phi} \\ & + \frac{1}{r}\left(-mRre\sin(\phi)\,\dot{\phi}^2 - 2mRre\sin(\phi)\,\dot{\theta}\dot{\phi}\right) \;=\; T_a - c_s\dot{\theta} \end{aligned} \tag{2.68}$$

$$\frac{e}{r}\left(-I + mre + mRr\cos(\phi)\right)\ddot{\theta}$$
$$+\frac{e^2}{r^2}\left(I + mr^2\right)\ddot{\phi} + mRe\sin(\phi)\dot{\theta}^2 = -c_a\dot{\phi} \tag{2.69}$$

The tuning order of the Salomon CPVA is found as:

$$n = \sqrt{\frac{1}{e\left(\left(\frac{I}{mr^2}\right)^2 + 1\right)}} \tag{2.70}$$

and with $r = d/2$, $e = (D-d)/2$:

$$n = \sqrt{\frac{2}{(D-d)\left(\left(\frac{4I}{md^2}\right)^2 + 1\right)}} \tag{2.71}$$

where D is the diameter of the bored hole, and d is the diameter of the roller, [19].

2.7.4 Cycloidal Path Type

The cycloid path CPVA will be modeled as a rotor with an attached point-mass pendulum. The path will be chosen as the one derived by Huygens, found in the work by Emmerson [20]. This path is that of a cycloid pendulum, which is the tautochrone in a gravity field, see section 2.5.

In the local cycloid pendulum coordinate system and subtracting r from the y-coordinate to fit on the rotor, see Figure 2.9, the coordinates are:

$$x_c = \frac{1}{4}r\left(2\phi_c + \sin(2\phi_c)\right) \tag{2.72}$$
$$y_c = \frac{1}{4}r\left(1 - \cos(2\phi_c)\right) - r. \tag{2.73}$$

A rotation matrix is used to obtain coordinates in the inertial system, $\pi/2$ is added to θ just to fit with previous pendulum geometry. Also, the rotation around the rotor is added, and the coordinates become:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\theta + \pi/2) & -\sin(\theta + \pi/2) \\ \sin(\theta + \pi/2) & \cos(\theta + \pi/2) \end{bmatrix} \begin{bmatrix} x_c \\ y_c \end{bmatrix} + \begin{bmatrix} R\cos(\theta) \\ R\sin(\theta) \end{bmatrix}. \tag{2.74}$$

Substitution of (2.72), (2.73) in (2.74) yields:

$$x = -\left(-r + \frac{1}{4}r\left(1 - \cos(2\phi_c)\right)\right)\cos(\theta) -$$
$$\frac{1}{4}r\left(2\phi_c + \sin(2\phi_c)\right)\sin(\theta) + R\cos(\theta) \tag{2.75}$$
$$y = \frac{1}{4}r\cos(\theta)\left(2\phi_c + \sin(2\phi_c)\right) -$$
$$\left(-r + \frac{1}{4}r\left(1 - \cos(2\phi_c)\right)\right)\sin(\theta) + R\sin(\theta). \tag{2.76}$$

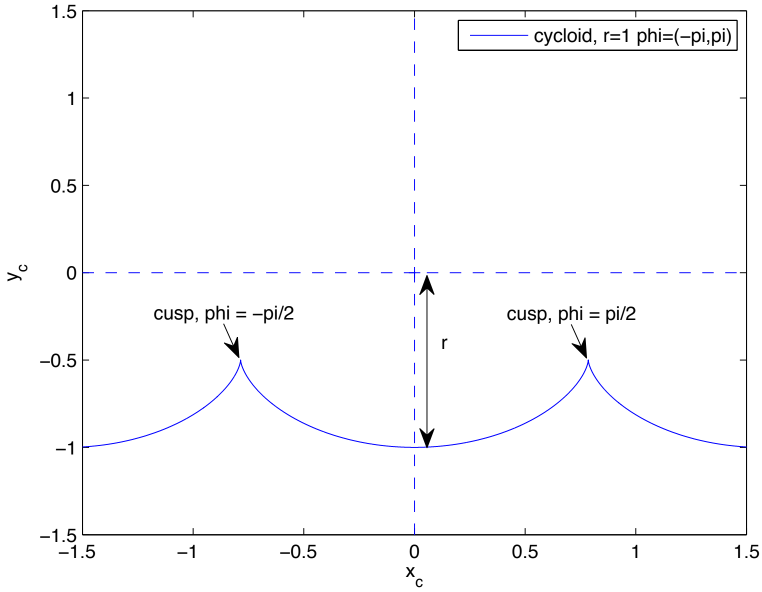


Figure 2.9: *The cycloid curve drawn with Equations (2.72) and (2.73), $r = 1$.*

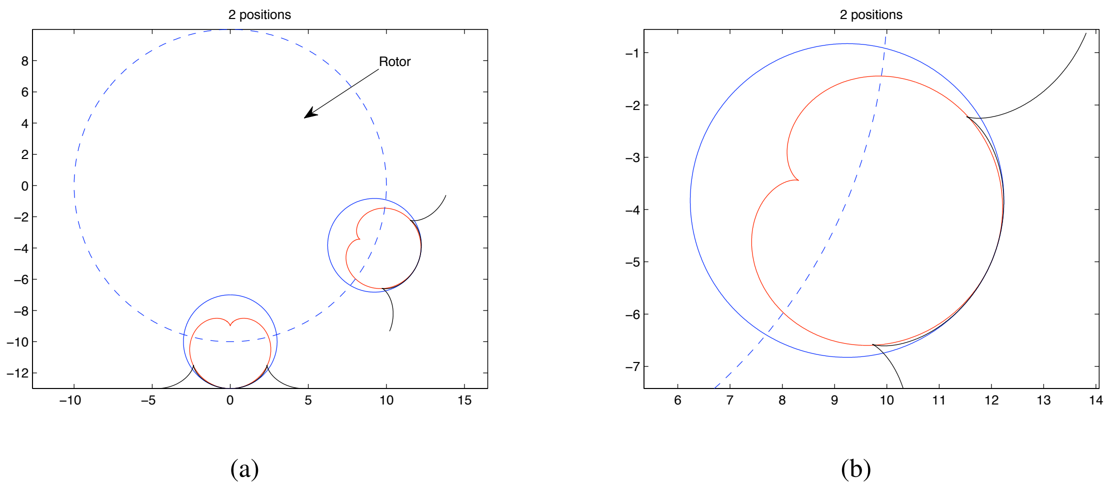


Figure 2.10: *Illustration of the cycloidal and epicycloidal paths, for CPVA types in sections 2.7.4, 2.7.5, for Matlab code see Appendix A.2. (a) The pendulum attachment point around the rotor is shown as the dotted line. The circular, cycloidal, and epicycloidal paths are shown at two positions on the rotor. (b) The circular, cycloidal (wave-pattern), and epicycloidal (heart-shaped).*

The paths of both the cycloid and the epicycloid are shown in Figure 2.10. The Matlab code may be found in Appendix A.2. Using the same method with Lagrangian mechanics as previous CPVA types, the equations of motion, with $q_1 = \theta$, $q_2 = \phi_c$, are:

$$
\begin{aligned}
&[\frac{1}{2}mRr\cos(2\phi_c) + (J + mr^2 + \frac{3mRr}{2} + mR^2)\cos(2\phi_c) + \\
&\frac{1}{4}mr^2\phi_c^2 + \frac{1}{4}mr^2\phi_c\sin(2\phi_c)]\ddot{\theta} + \\
&[\frac{mr^2}{2} + \frac{mRr}{2} + \frac{1}{2}mr^2\cos(2\phi_c) + \\
&\frac{1}{2}mRr\cos(2\phi_c) + \frac{1}{4}mr^2\phi_c\sin(2\phi_c)]\ddot{\phi}_c + \\
&\frac{1}{2}mr^2\cos(2\phi_c)\,\phi_c\dot{\phi}_c^2 - \frac{1}{4}mr(3r + 4R)\sin(2\phi_c)\,\dot{\phi}_c^2 + \\
&mr^2\cos^2(\phi_c)\,\phi_c\dot{\theta}\dot{\phi}_c - \frac{1}{2}mr(r + 2R)\sin(2\phi_c)\,\dot{\theta}\dot{\phi}_c = T_a - c_s\dot{\theta}
\end{aligned}
\tag{2.77}
$$

$$
\begin{aligned}
&[\frac{mr^2}{2} + \frac{mRr}{2} + \frac{1}{2}mr^2\cos(2\phi_c) + \frac{1}{2}mRr\cos(2\phi_c) + \\
&\frac{1}{2}mr^2\cos(\phi_c)\,\phi_c\sin(\phi_c)]\ddot{\theta} + [\frac{mr^2}{2} + \frac{1}{2}mr^2\cos(2\phi_c)]\ddot{\phi}_c \\
&-\frac{1}{2}mr^2\sin(2\phi_c)\,\dot{\phi}_c^2 - \frac{1}{2}mr^2\cos^2(\phi_c)\,\phi_c\dot{\theta}^2 + \\
&\frac{1}{4}mr^2\sin(2\phi_c)\,\dot{\theta}^2 + \frac{1}{2}mRr\sin(2\phi_c)\,\dot{\theta}^2 = 0.
\end{aligned}
\tag{2.78}
$$

For tuning and optimal pendulum mass of cycloidal pendulum, the equations of the Simple type pendulum absorber can be used, at least for first estimations.

2.7.5 Epicycloidal (Cardoid) Path Type

The epicycloid used for simulations is the epicycloid of degree $a/b = k = 1$, which is fitted for placement on the rotor. This type of curve is also called a cardoid. In the pendulum coordinate system, the coordinates are from [11], with $R_{epi} = k \cdot (r/3)$:

$$x_{epi} = \left(R_{epi} + \frac{r}{3}\right)\cos(\phi_{epi}) + \frac{1}{3}r\cos\left(\frac{3\left(R_{epi} + \frac{r}{3}\right)(\phi_{epi} + \pi)}{r}\right) \tag{2.79}$$

$$y_{epi} = \left(R_{epi} + \frac{r}{3}\right)\sin(\phi_{epi}) + \frac{1}{3}r\sin\left(\frac{3\left(R_{epi} + \frac{r}{3}\right)(\phi_{epi} + \pi)}{r}\right). \tag{2.80}$$

Using a rotation matrix and adding the rotor positions, yields the coordinates in the inertial system:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x_{epi} \\ y_{epi} \end{bmatrix} + \begin{bmatrix} R\cos(\theta) \\ R\sin(\theta) \end{bmatrix} \tag{2.81}$$

which yields:

$$
\begin{aligned}
x &= \frac{1}{3}(3R\cos(\theta) + (3R_{epi} + r)\cos(\phi_{epi} + \theta) \\
&\quad + r\cos(\frac{(3R_{epi} + r)(\phi_{epi} + \pi)}{r} + \theta)) && (2.82) \\
y &= \frac{1}{3}(3R\sin(\theta) + (3R_{epi} + r)\sin(\phi_{epi} + \theta) \\
&\quad + r\sin(\frac{(3R_{epi} + r)(\phi_{epi} + \pi)}{r} + \theta)). && (2.83)
\end{aligned}
$$

Lagrangian mechanics, with $q_1 = \theta$, $q_2 = \phi_{epi}$, leads to equations of motion:

$$
\begin{aligned}
&[\frac{1}{9}(9J + 9mR_{epi}^2 + 6mR_{epi}^2 r + 2mr^2 + 9mR^2 \\
&+6m(3R_{epi} + r)R\cos(\phi_{epi}) \\
&+6mRr\cos\left(\frac{(3R_{epi} + r)}{r}(\phi_{epi} + \pi)\right) \\
&+6R_{epi}mr\cos\left(\frac{3R_{epi}\pi + r\pi + 3R_{epi}\phi_{epi}}{r}\right) \\
&+2mr^2\cos\left(\frac{3R_{epi}\pi + r\pi + 3R_{epi}\phi_{epi}}{r}\right)]\ddot{\theta} \\
&-[\frac{1}{9}m(3R_{epi} + r)(-3R_{epi} - 2r - 3R\cos(\phi_{epi}) + \\
&(3R_{epi} + 2r)\cos\left(\frac{3R_{epi}}{r}(\phi_{epi} + \pi)\right) \\
&-3R\cos\left(\frac{(3R_{epi} + r)}{r}(\phi_{epi} + \pi)\right)]\ddot{\phi}_{epi} \\
&+\frac{1}{3r}m(3R_{epi} + r)[-Rr\sin(\phi_{epi}) \\
&+R_{epi}(3R_{epi} + 2r)\sin\left(\frac{3R_{epi}}{r}(\phi_{epi} + \pi)\right) \\
&-(3R_{epi} + r)R\sin\left(\frac{(3R_{epi} + r)}{r}(\phi_{epi} + \pi)\right)\dot{\phi}_{epi}^2 \\
&+\frac{2}{3}m(3R_{epi} + r)[-R\sin(\phi_{epi}) \\
&+R_{epi}\sin\left(\frac{3R_{epi}}{r}(\phi_{epi} + \pi)\right) \\
&-R\sin\left(\frac{(3R_{epi} + r)}{r}(\phi_{epi} + \pi)\right)]\dot{\theta}\dot{\phi}_{epi} = T_a - c_s\dot{\theta} \qquad (2.84)
\end{aligned}
$$

$$
\begin{aligned}
&[\frac{2}{9}m\left(3R_{epi}+r\right)\sin\left(\frac{3R_{epi}}{2r}\left(\phi_{epi}+\pi\right)\right) \\
&\left(\left(3R_{epi}+2r\right)\sin\left(\frac{3R_{epi}}{2r}\left(\phi_{epi}+\pi\right)\right)\right. \\
&+3R\sin\left(\frac{3R_{epi}\pi+3R_{epi}\phi_{epi}+2r\phi_{epi}}{2r}\right)]\ddot{\theta} \\
&+[\frac{4}{9}m\left(3R_{epi}+r\right)^2\sin^2\left(\frac{3R_{epi}}{2r}\left(\phi_{epi}+\pi\right)\right)]\ddot{\phi}_{epi} \\
&+\frac{2mR_{epi}\left(3R_{epi}+r\right)^2\cos\left(\frac{3R_{epi}}{2r}\left(\phi_{epi}+\pi\right)\right)\sin\left(\frac{3R_{epi}}{2r}\left(\phi_{epi}+\pi\right)\right)\dot{\phi}_{epi}^2}{3r} \\
&-\frac{2}{3}m\left(3R_{epi}+r\right)[R_{epi}\cos\left(\frac{3R_{epi}}{2r}\left(\phi_{epi}+\pi\right)\right)+ \\
&R\cos\left(\frac{3R_{epi}\pi+3R_{epi}\phi_{epi}+2r\phi_{epi}}{2r}\right)]\sin\left(\frac{3R_{epi}}{2r}\left(\phi_{epi}+\pi\right)\right)\dot{\theta}^2 = 0 \qquad (2.85)
\end{aligned}
$$

For tuning and optimal pendulum mass of the epicycloidal pendulum, the equations of the Simple type pendulum absorber can be used, at least for first estimations.

2.7.6 Bifilar Type with Rollers and General-Path Representation (Epicycloid Path)

This type of CPVA unit from Denman [9] is more realistic and combines some of the properties of the previous types in this work, it is bifilar, has rollers, and the pendulum centre of mass travels on a general path C, see Figure 2.11. A general path means that the path is adjusted by a single parameter λ, to form a whole range of curves including circular, cycloid, and epicycloid. To describe the curve, this expression is used:

$$
\rho^2 = \rho_0^2 - \lambda^2 s^2 \qquad (2.86)
$$

where ρ is the radius of curvature along curve C, and ρ_0 is the radius of curvature at point $(0, -c)$, where $s = 0$, $\theta_2 = 0$. Variable s is the arc length variable along C, and θ_2 is the tangent angle to curve C ($\theta_2 = \arctan\left(dy/dx\right)$). Parameter λ can be given any value $0 \leq \lambda \leq 1$, where $\lambda = 0$ yields a circle, $\lambda = 1$ a cycloid, and $\lambda = \sqrt{n^2/\left(n^2+1\right)}$ an epicycloid with its fixed circle (see Figure 2.5) centered at the rotor centre, of radius $c - \rho_0$. This type of epicycloid curve is chosen for simulations in this work, since it has the *tautochronic* property (see section 2.5), meaning that the peroid of the pendulum is independent of the amplitude.

If the generalized coordinates are chosen as $q_1 = \theta$, $q_2 = s$, and the path of curve C is chosen with $\lambda = \sqrt{n^2/\left(n^2+1\right)}$, the kinetic energy of the system is

$$
\begin{aligned}
T = {} & \frac{1}{2}\left(\frac{i_R}{2a^2}+m_p+\frac{m_R}{2}\right)\dot{s}^2+[-\frac{i_R}{a^2}+\frac{1}{2}\left(2D-c\right)m_R\cos\left(\theta_2\right)+ \\
& \left(m_p+\frac{m_R}{2}\right)\sqrt{c^2-n^2\left(1+n^2\right)s^2}]\dot{\theta}\dot{s}+ \\
& \frac{1}{2}[J+I+2i_R+m_p\left(c^2-n^2s^2\right)+2m_R(\left(\frac{c}{2}-D\right)^2+H_0^2+ \\
& \left(\frac{c}{2}-D\right)y_P+\frac{1}{4}\left(c^2-n^2s^2\right))]\dot{\theta}^2
\end{aligned} \qquad (2.87)
$$

where i_R is moment of inertia for roller, a is roller radius, m_p is mass of pendulum, m_R is mass of roller, n is order of tuning, J is rotor moment of inertia, I is pendulum moment of inertia, y_P

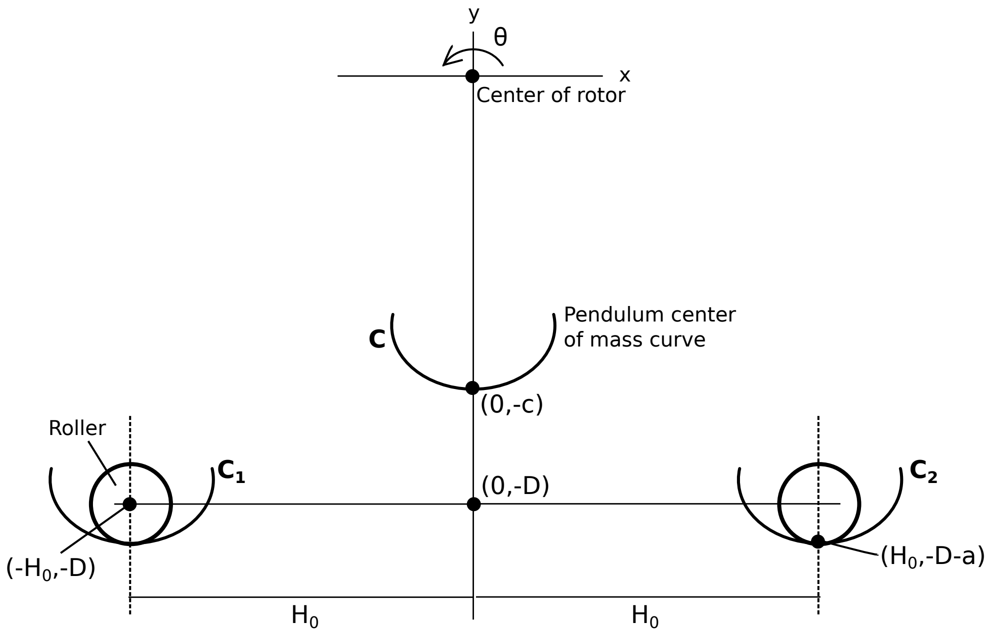


Figure 2.11: *Sketch of Bifilar Type with Rollers and General-Path representation. The pendulum travels on curves C_1 and C_2 cut out in the rotor unit, and the pendulum center of mass travels on curve C.* [9]

is the y position of pendulum centre of mass on curve C, and other constants as defined in Figure 2.11.

Further, θ_2 and y_P must be expressed as function of s:

$$\theta_2 = \frac{1}{\lambda} \arcsin\left(\frac{\lambda s}{\rho_0}\right) \tag{2.88}$$

$$y_P = -c - \frac{\rho_0}{1-\lambda^2}\left(\cos(\theta_2)\cos(\lambda\theta_2) + \frac{\lambda^2 s}{\rho_0}\sin(\theta_2) - 1\right). \tag{2.89}$$

Substitution of (2.88) in (2.89) creates an expression for y_P as a function of s. The radius of curvature at point $(0, -c)$, ρ_0, is determined by the tuning condition:

$$\rho_0 = \frac{m_P c + m_R D}{(n^2+1)\left(m_P + \frac{m_R}{2}\right) + n^2 \frac{i_R}{2a^2}}. \tag{2.90}$$

If (2.90) is inserted in (2.88), (2.89), and these new expressions for θ_2 and y_P are inserted in (2.87), an expression for the kinetic energy is found that is ready for insertion in Lagrange's equations of motion. The equations though, become long and can therefore be found in Appendix C.

It can be seen that this type of model has some advantages. In comparison with the other types, the path can be chosen easily with the parameter λ, and the tuning is also easily set with parameter n. Also, here the path can be set to have tautochronous motion, which is a major advantage, [9].

3 LITERATURE REVIEW

In this chapter, a presentation of previous studies and some of the findings of those will be included. The literature considered is from the time period between 1934 and 2009, and mainly includes books, theses, and journal articles.

The first source is the classic book on mechanical vibrations by Hartog [5], which contains basic information about CPVAs. It also contains an entire chapter about vibrations in multi-cylinder engines, which was useful for this thesis, see section 2.1. This book is a recurring reference for many in the mechanical vibration field. Further, the German articles by von A. Silbar and Desoyer [18] and von P. R. Paslay and Silbar [19] are used as sources for the Simple, Bifilar, and Salomon type absorbers. They describe the dynamics, equations of motion, tuning condition, pendulum mass estimation etc.

Newland [16] presented a thesis focusing on non-linear vibrations with applications to CPVAs. This work incorporates Bifilar type absorbers used for vibration reduction in high performance aircraft. This study describes the linear assumption with small pendulum angles well, and also the non-linear jump behavior experienced when the pendulum amplitude is too large. It also describes the stability effects of positive mistuning, and the application of CPVA units together with up to 5 degrees-of-freedom systems. The book by Wilson [2] is a very good source for the newcomer to CPVAs. It contains a thorough description of the history of the most common types of CPVAs, with sketches and pictures. There is also extensive material on theory and applications to multi-cylinder engines, it is a very thorough text with an entire chapter focusing on CPVAs only.

Making a big leap, from 1968 to 1992, Denman [9] presents an article about Tautochronic Bifilar CPVAs for engines. In this article a very convenient way of modeling CPVAs is shown, where the path of the pendulum may be prescribed with a general path representation. This article also explains that the pendulum may travel on such a path, that its behavior will behave linearly regardless of its amplitude, and that this path is an epicycloid. A path that causes this behavior is called a tautochrone. This method is used also in this thesis work, see section 2.7.6.

In an article by Albright et al. [21], Roller type pendulum absorbers are used successfully on V-8 engine crankshafts for drag racing, demonstrating the capability of CPVAs in practice. Lee and Shaw [22] draws some important conclusions when using tautochronic CPVAs to suppress vibrations. The findings are that several pendulum absorbers that are tuned to different harmonic orders must be used, a single absorber for the primary harmonic torque will not be enough. Articles by authors Chao et al. [14, 23], Lee and Shaw [24], Chao and Shaw [25, 26] are focused on analyzing the behavior of CPVA units with several attached absorbers. The non-unison motion of the attached absorbers is an interesting subject; it might have negative effects on the performance and sometimes positive effects. It is good to understand this matter since CPVAs almost exclusively are built with several pendulums in practice.

The thesis by Alsuwaiyan [15] and the article of Alsuwaiyan and Shaw [27] treat the subject of stability in great detail, both the non-linear jump behavior and bifurcation to non-unison experienced in CPVA units with several absorbers. The stability is analyzed with perturbation theory; both positive mistuning and general paths are considered. In a paper by Vidal et al. [28] a similar problem is solved as in this thesis work. CPVAs are integrated in the flywheel, and this is modeled in a Simulink application where the performance can be depicted.

The thesis of Nester [17] puts the theory developed in previous work to test. Circular path CPVAs are tested experimentally with both a single-absorber setup and a four-absorber setup, and compared with theoretical models. Also, a experimental vehicle is used for tests where CPVAs are attached to the crankshaft. In the article by Nester et al. [29], tests are performed on a variable displacement engine that can operate both in V-8 and V-4 mode. CPVAs are used to absorb vibrations, especially in V-4 mode, where a primary second order torque is present.

The results show that the absorbers are effective in this application. Further research has been conducted in the works of Alsuwaiyan and Shaw [12], Haddow and Shaw [13], Shaw et al. [30], Pfabe and Woernle [31], Ishida and Inoue [32], that also relate back and build on previous findings.

Regarding the subjects of cycloids, epicycloids, and tautochrones the works of Rana and Joag [10], Emmerson [20], Gauld [33] have been used. The first-named work has a quite thorough description of tautochrones in e.g. gravity fields and centrifugal fields. To understand how Dual Mass Flywheels operate and are modeled, the papers of Albers [4], Fidlin and Seebacher [34], Schaper et al. [35] is very helpful. For simulations in Simdrive, the documents of Prejawa [36, 37] and [38] are useful. The works of Cederwall and Salomonson [6], Boström [7], Thomson and Dahleh [8] have also been useful for vibration theory and rigid body dynamics with Lagrangian mechanics formulation.

4 METHODOLOGY – MODELING AND SIMULATION

This chapter contains a detailed explanation of the modeling and simulation procedures in this work. The modeling procedure for the CPVA unit is described, followed by a description of the simulation models developed in Simdrive. Finally, there is a description of how simulations are performed. The CPVA unit is programmed manually in C/C++, compiled, and connected to the Simdrive simulation model environment. The simulation models created for this work are the; Verification model, Sine-signal torque, Advanced model, each with their own purpose. The Advanced model is a similar to a car powertrain, and is used to investigate the use of CPVAs in the Dual Mass Flywheel. Simulations are run through the speed range of a typical diesel car engine.

4.1 CPVA Modeling

4.1.1 Derivation of Equations of Motion for Simdrive or Matlab

The first step of creating a CPVA model is to create a free body diagram, like the ones shown in Figure 2.6, 2.7, and 2.8. Since the Lagrangian method is used, the primary objective of these free body diagrams is to express the kinetic energy. An analysis of all forces in the system is therefore unnecessary. With aid of the free body diagram, it is possible to choose suitable generalized coordinates, and since the CPVA essentially is a rotating device, polar coordinates are convenient. It is important to notice here that further calculations for e.g. the Simple type CPVA are manageable by hand, but for some of the other types, e.g. the cycloidal and epicycloidal CPVA it is strongly recommended to use a software such as *Mathematica*. Calculations become several pages long, and even for the Simple type CPVA it is easy to make small mistakes. Furthermore, there is a command in *Mathematica*, called *CForm* that create C/C++ code directly. The general method on how to express the Lagrangian equations of motion of the second kind is described in section 2.2, and the reformulation to a system of first-order ODEs is described in section 2.3. Specific information for every CPVA type is provided in section 2.7. The general method for the CPVA modeling is:

1. Create a free body diagram of the CPVA unit
2. Choose and express suitable generalized coordinates for the CPVA in a polar coordinate system
3. Express the total kinetic energy of the CPVA unit in the generalized coordinates
4. Use Lagrangian mechanics to express Lagrange's equations of motion of the second kind
5. Introduce new variables for the time derivatives to form a system of first-order ODEs. This form is suitable for a Runge-Kutta solver such as *ode45* in *Matlab*
6. Solve the first-order system to obtain the second time derivatives (accelerations) of the generalized coordinates explicitly
7. Insert the expressions of acceleration in the C/C++ code, and adapt the code for the specific CPVA element

4.1.2 C/C++ Programming and Compilation to dll-file in Visual Studio

An element for simulations of CPVA elements in *Simdrive* already exists, and allows for simulations of Simple, Bifilar, or Salomon type CPVAs, which are described in section 2.7. However, some other types of CPVAs need to be tested as well, and therefore a customizable element is used. This customizable element in *Simdrive* is named *External rotary element* and belongs to the category of *Rheologic elements*. This element is connected to a *dynamic link library (dll)* file that preferably is programmed in C/C++, and compiled in e.g. *Visual studio*. The dynamic properties of the CPVA are described in this element and are communicated to *Simdrive*. The most important contents in the C/C++ file are:

- Output of physical quantities, e. g. *Angle*, units, e. g. *Degree* and type of result, e. g. *Time*
- Parameter inputs, values like *Pendulum length*, *Pendulum mass* that can be set directly in the input table of the *External rotary element*
- Export of variables, e .g. *Angle*, *Angular speed*, *Pendulum angle* etc.
- A comparison of parameter names set in the table of the *External rotary element* with the parameter names in the code, just to make sure that the right parameters are used and set
- Initialization of variables for the calculations in the element
- Setup of start values for the calculations in the element
- Calculation of accelerations and torques in the element

The programming is performed by the use of an already existing C/C++ file as a template, provided by *Contecs Eng.* Changes have to be made according to the above list. The most important part is to include the dynamics of the CPVA. To achieve this, variables for pendulum angle, pendulum angular speed, and pendulum angular acceleration are introduced, as an addition to the already existing variables for the rotor angle. The CPVA dynamics can then be described by 6 variables, angle, angular speed, and angular acceleration for both rotor and pendulum.

This is how the calculations in the C/C++ file are performed, starting with:

```
double T_prev_J=C1*(Yn2[0][0]-p_phi[0])+B1*(Yn2[0][1]-p_phi[1]);
double T_J_next=C2*(n_phi[0]-Yn2[0][0])+B2*(n_phi[1]-Yn2[0][1]);
```

The torque exerted on the CPVA (External rotary element), *T_prev_J* and *T_J_next* are the torques acting on the CPVA by the previous and next element in the model, i.e. the adjacent elements. C1, C2 are stiffness constants and B1, B2 are the damping constants to specify the characteristics of the connections between the CPVA and the next/previous element. The Yn2[][] arrays are storing variables according to Equation (2.17), and the *p_phi[]* (previous), *n_phi[]* (next) are stored in a similar way. When these torques are calculated, the net applied torque on the CPVA is calculated as:

```
double Ta=-T_prev_J+T_J_next;
```

The applied torque, T_a, is then included in the calculations of the rotor and pendulum accelerations, which are calculated according to Equations (2.18), (2.19). This is an example for the Simple type CPVA:

```
Yn2[0][2]=((I + m*pow(r,2))*Ta - cs*(I + m*pow(r,2))*Yn2[0][1] + m*r*(I + m*pow(r,2))*R*pow(Yn2
    [1][1],2)*sin(Yn2[1][0]) + m*r*R*pow(Yn2[0][1],2)*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0]))*
    sin(Yn2[1][0]) + Yn2[1][1]*(ca*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0])) + 2*m*r*(I + m*pow(r
    ,2))*R*Yn2[0][1]*sin(Yn2[1][0])))/((I + m*pow(r,2))*(J + m*pow(R,2)) - pow(m,2)*pow(r,2)*
    pow(R,2)*pow(cos(Yn2[1][0]),2));

Yn2[1][2]= (-(Ta*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0]))) + cs*Yn2[0][1]*(I + m*pow(r,2) + m*r*
    R*cos(Yn2[1][0])) - m*r*R*pow(Yn2[1][1],2)*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0]))*sin(Yn2
    [1][0]) - m*r*R*pow(Yn2[0][1],2)*(I + J + m*pow(r,2) + m*pow(R,2) + 2*m*r*R*cos(Yn2[1][0]))
    *sin(Yn2[1][0]) - Yn2[1][1]*(ca*(I + J + m*pow(r,2) + m*pow(R,2) + 2*m*r*R*cos(Yn2[1][0]))
    + 2*m*r*R*Yn2[0][1]*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0]))*sin(Yn2[1][0])))/((I + m*pow(r
    ,2))*(J + m*pow(R,2)) - pow(m,2)*pow(r,2)*pow(R,2)*pow(cos(Yn2[1][0]),2));
```

An example of a complete C/C++ code for compilation to a dll-file is provided in Appendix B, which is commented thoroughly. It can be noted that no actual calculations take place in the dll-file, it is merely instructions for the solver built into Simdrive.

In order to compile this C/C++ code file into a dll-file, Visual studio 2010 is used. A project is created in Visual studio of DLL type, and the C/C++-file is used as a source file to that project. Instructions for Visual studio are provided in Appendix D.

4.1.3 Implementation of dll-file in Simdrive

When all the steps in section 4.1.1 and 4.1.2 are completed, the dll-file is ready to be connected to the *External rotary element* which then functions as a complete CPVA unit. The dll-file is simply selected directly from the *External rotary element*. As briefly mentioned earlier, there is a table in the *External rotary element* where all parameters can be set, e.g. *Pendulum mass*, *Number of pendulum masses*, *Moment of inertia*, *Viscous damping coefficient* etc. It is important that the parameters used are the correct ones for the specific CPVA element. As explained earlier and as shown in Appendix B, the dll-file is programmed in a such a manner that irrelevant parameters cannot be set.

Some comments should be made about the compatibility of the variable names, variable units and so on, between the dll and Simdrive. For the purpose of being able to compare results efficiently and directly in Simdrive it is good to have the same variable names and units. Any issues with units have been avoided in this work, but for variable names it can be more difficult. A blank space in a variable name was not easy to program, so the commonly used *Rotary speed* became *Rotary_speed* in the CPVA for example. Usually these two types of variables can be compared directly anyhow, but the general strategy should be to give new variables the same names as the already existing ones in Simdrive. That is why the *Pendulum angle* is given the name *Pendelwinkel* for example.

4.2 Simdrive Modeling

4.2.1 Verification Model

The Verification model, shown in Figure 4.1, is created to check and validate the customized CPVA element (External rotary element) against the already existing CPVA elements in Simdrive. The model is symmetrical about the middle, and the two halves are also entirely disconnected. This is a convenient way to simulate, the Simdrive CPVA and the customizable CPVA

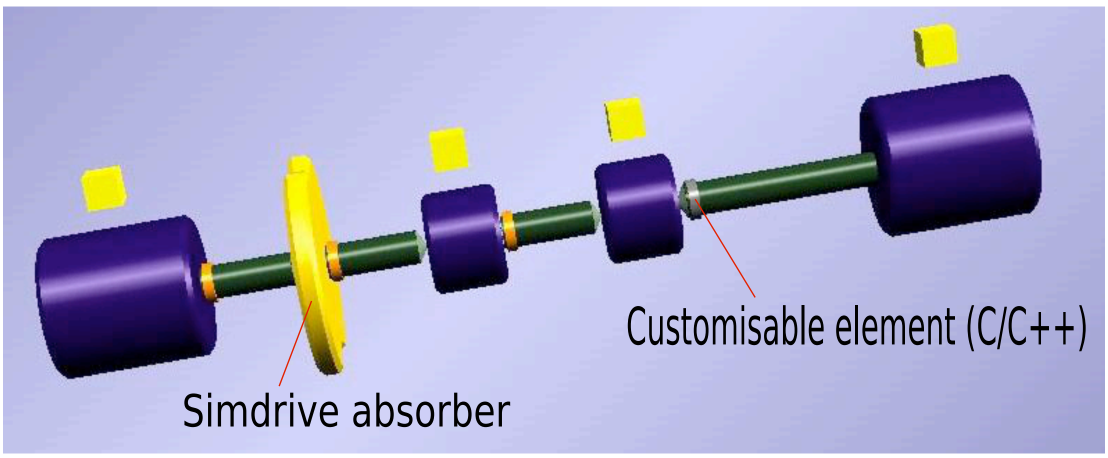


Figure 4.1: *Verification model in Simdrive for comparison of the customized CPVA and already existing CPVA elements in Simdrive.*

can be run simultaneously and the results can be compared directly. The rotary masses in the middle are identical and are driven by an n:th order sine-signal torque, and the bigger rotary masses at both ends hold constant rotary speed. The model is simulated for a short time period, say two seconds, and the two CPVAs can be compared. The elements that do already exist in Simdrive is the Simple, Bifilar, and Salomon type. The results from the customized dll CPVA show perfect agreement with the mentioned types, which indicates that the derivations of equations of motion, programming in C/C++, compilation, and implementation should be correct. Suitable variables to compare are e.g. *Pendulum angle* and *Rotary speed*.

4.2.2 Model with Sine-signal Torque

The Sine-signal torque model, shown in Figure 4.2, is quite similar to the Verification model, described in section 4.2.1, but here all the customized types of CPVAs are tested, and they are tested against a regular flywheel. Also, the model is here tested in load intervals, chosen at different rotary speeds. The rotary speeds simulated are $(700, 800, ..., 4600, 4700)$ rpm, representing the typical range of a diesel car engine. The object of investigation in this work is a four-cylinder, four-stroke engine, which will have a dominant second order torque component. Therefore, the sine-signal of the simple engines in the model is of second order, and the CPVA is tuned close to second order to absorb the irregularities caused by the engine.

This relatively simple model is very convenient to use, and good for getting acquainted with the basic characteristics of the CPVA. All the parameters can be changed quickly and the simulations also run fast. It also gives some idea of the absorbing capabilities of the CPVA, compared to a regular flywheel. Most of the parameters can be tuned and adjusted here, before the CPVA is inserted into a more advanced model. All different types of CPVAs have been run in this model, and the results are displayed in Figure 5.1 – 5.6.

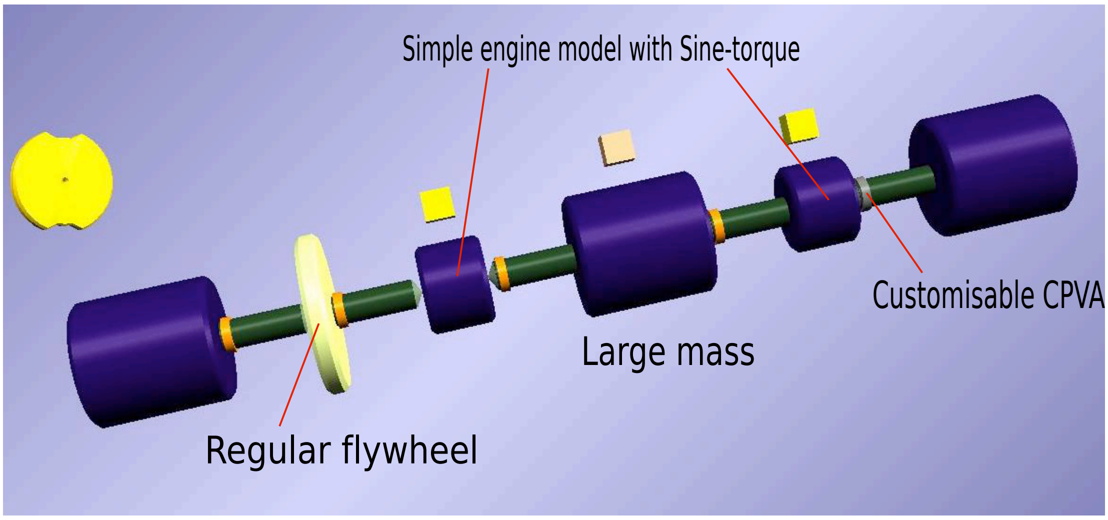


Figure 4.2: *Sine-signal torque model in Simdrive for comparison of the customized CPVA and a regular flywheel.*

4.2.3 Advanced Model with DMF, FEAD, Diesel Engine Torque

The Advanced model, shown in Figure 4.3, is created to capture the behavior of a real car powertrain. The simulation load intervals are the same as for the Sine-signal torque model, i.e. $(700, 800, ..., 4600, 4700)$ rpm. Here, the engine is modeled in a more realistic way, by including individual cylinders with individual input torque signals. The cylinders are shown in Figure 4.3 and input torque is specified in the square boxes above the cylinders. The cylinders are approximated as rotary masses that are connected to each other with flexible connections.

A belt-driven FEAD (Front Engine Accessory Drives) system is also included in the model, shown to the left in Figure 4.3. It is connected to the crankshaft with a decoupler unit. The unit that is most important to model is the Dual Mass Flywheel (DMF), where the CPVAs are placed. A detailed description of theory for the DMF will not be included in this report, instead a study of works from Albers [4], Fidlin and Seebacher [34], Schaper et al. [35] is recommended. The basic principle of a DMF is to use two flywheels instead of just one, and connect the flywheels with a weak arc spring. This decouples the engine from the rest of the powertrain and cause in the order of 90% vibration reduction. The DMF is commonly built so the arc spring rate will be proportional to the rotary speed of the DMF itself. This is because the spring will be affected by the centrifugal force; friction against the primary flywheel will prevent the spring from compressing. The DMF will thus become stiffer when the rotary speed increases, a property that CPVAs also possess.

To model a DMF in every detail is a lengthy and complicated process, some of the difficulties include many degrees-of-freedom in the spring and an understanding of the friction characteristics between the arc springs and the housing. Therefore, a simplified model of the DMF will be used in this work. There exists an element in the Engine module package of Simdrive, but since this module was not available at the time of this work another method has been used. The solution is shown in Figure 4.4, a close-up of the DMF. The primary and secondary flywheels are connected by two rheologic elements that possess the characteristics of the arc spring. The properties of the rheologic elements are shown in Figure 4.5. In simulations, the properties of these elements are added together. The values are assigned as *torque – relative angle*, showing the torque required to twist the flywheels in a certain relative angle to each other.

In the first rheologic element, the *Rotary Nonlinear Stiffness (from file)*, a component of the total torque is assigned, that is not dependent on the rotary speed. The torque is here negative for negative wind-up angles, and positive for positive wind-up angles. In the second rheologic

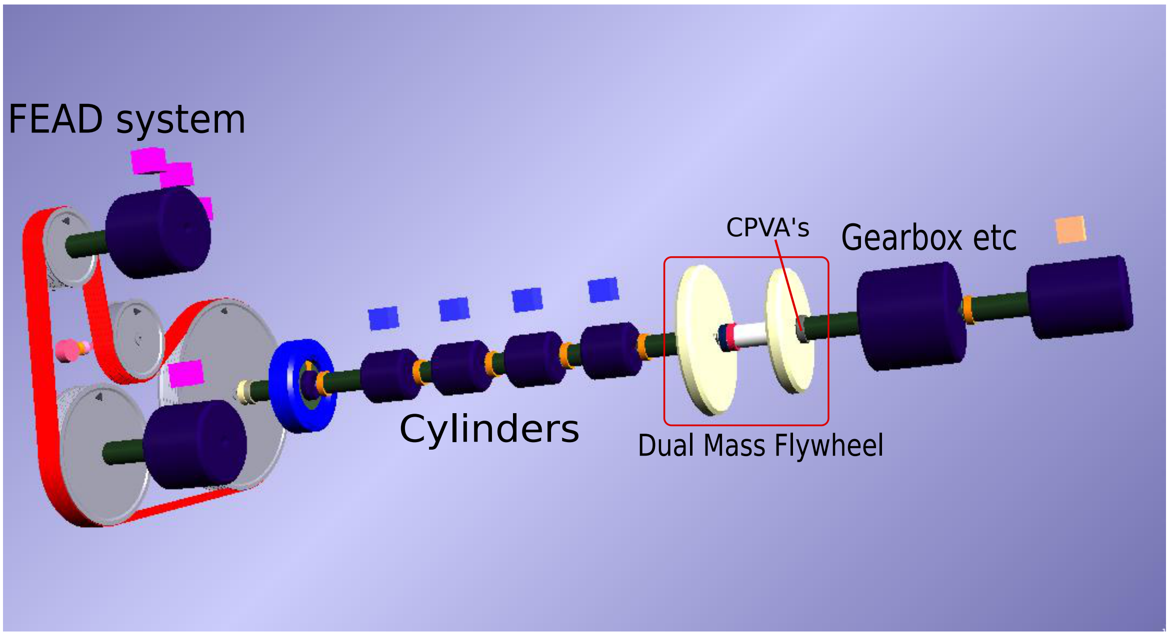


Figure 4.3: *Advanced model in Simdrive for more realistic simulation of the powertrain. The model includes FEAD system, cylinders, realistic input torque, and DMF (Dual Mass Flywheel) with CPVAs.*

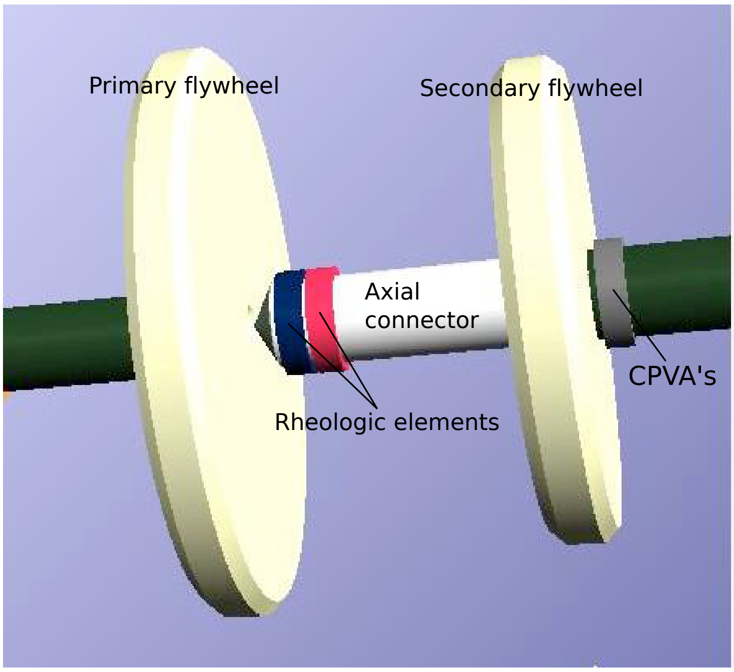
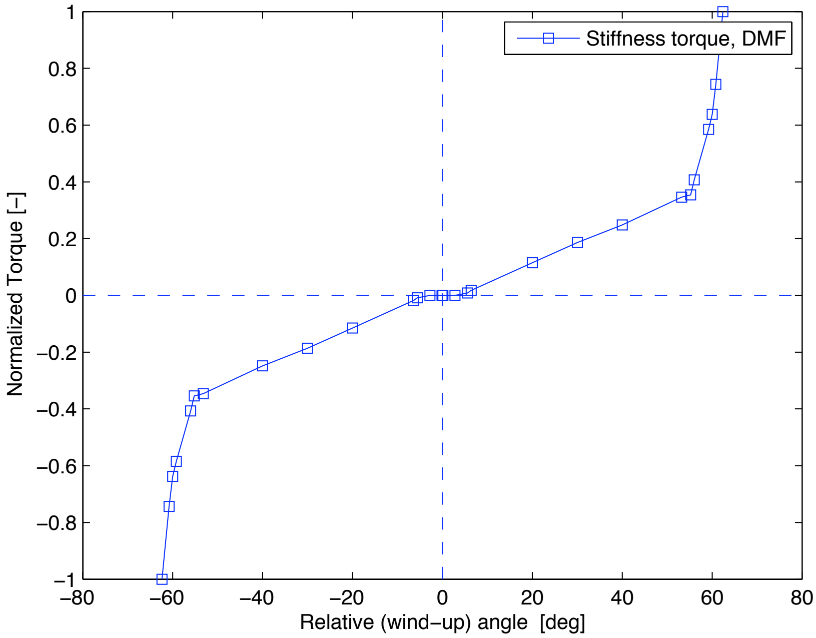
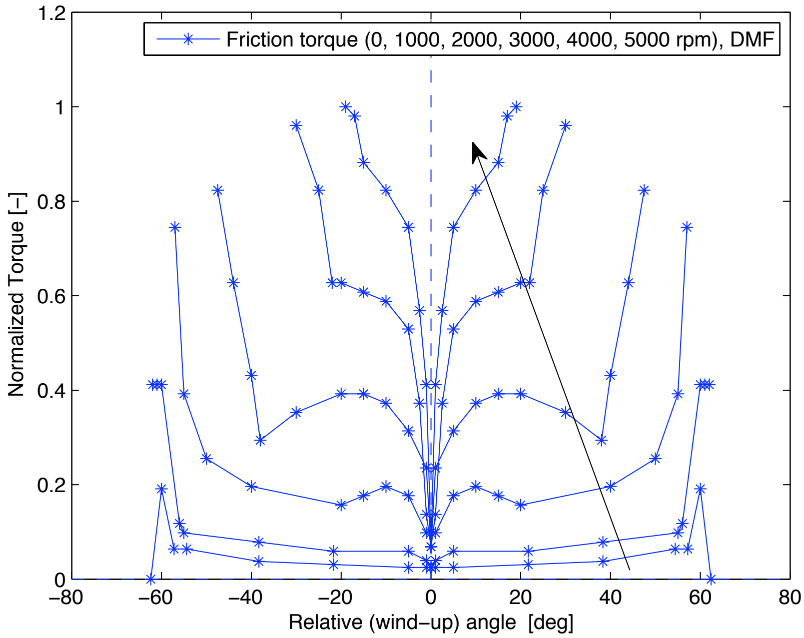


Figure 4.4: *Close-up of Dual Mass Flywheel in the Advanced model. The primary and secondary flywheels are connected with an axial connector with two rheologic elements; a Rotary Nonlinear Stiffness (from file) and a Rotary Friction Map.*



(a)



(b)

Figure 4.5: *DMF modeling with rheologic elements in Simdrive.* (a) *Stiffness torque of DMF (Rotary Nonlinear Stiffness (from file)).* (b) *Friction torque of DMF for 0, 1000, 2000, 3000, 4000, 5000 rpm in the direction of the arrow (Rotary Friction Map).*

element, the *Rotary Friction Map*, the remainder of the torque is given with a rotary speed dependence. Values for every 1000 rpm is given, and Simdrive will interpolate between these automatically. Note that the torque should be positive for both positive and negative relative angles. The values used are taken from a DMF simulation model, from German manufacturer LuK.

The rest of the powertrain is much simplified, and all remaining components in it are included in the rotary mass marked with "Gearbox etc", in Figure 4.3. The moment of inertia of this rotary mass, and the stiffness of which it is connected to the DMF is very crucial to the dynamic behavior of the whole model. Changes in these parameters will determine the vibration isolation level of the DMF. They should be chosen to give an isolation level of about 80 - 90%, if isolation level is defined as $(1 - A_{sec}/A_{prim}) * 100$ where A_{prim} and A_{sec} are the peak-to-peak vibration amplitudes of the primary and secondary side.

Finally, the rightmost element in the model holds constant levels of rotary speed, as mentioned earlier $(700, 800, ..., 4600, 4700)$ rpm. These are set in load intervals for simulation, in the square box over the element.

4.3 Simulations in Simdrive

As mentioned in previous sections, the simulations are run in load intervals for the Sine-signal torque and Advanced model. This is very convenient since results can be given in the rotary domain, and as FFT order analysis. The behavior of the system can then easily be studied over the entire engine speed range. In general, simulation times for the Verification model is a couple of minutes, for the Sine-signal torque model about 15 minutes, and for the Advanced model about an hour. A suitable strategy for the Advanced model is then to try different parameters on the CPVA unit, and make quicker simulations with only a few selected load intervals. When the CPVA parameters seem to give good results, a proper run with all load intervals is performed.

The main guidelines for parameter tuning in simulations are the ones pointed out in Chapter 2. The CPVA unit should be tuned to the harmonic order that is troublesome, and when it is a circular path type overtuning should be introduced. In general, more overtuning is needed when the CPVA operates with higher amplitudes, otherwise it will become unstable and reach the jump condition. The general tendency for the pendulum amplitude is that it will be higher for low engine speeds, a consequence of its design. Troubles both relating to performance and simulations are therefore usually encountered in the lower engine speed range. The performance might be an issue for all CPVA types simulated in this work, but for the alternate paths there are also numerical issues. For high amplitudes, the cycloidal and epicycloidal pendulum paths will reach cusps since these curves repeat themselves, which causes a numerical issue and stops simulation. Some load intervals have to be excluded in the lower range due to this fact.

Tuning and pendulum mass being the most obvious parameters to affect the behavior of the CPVA unit, there are still several other parameters and parameter changes that might have a big influence. The only way to reach a good design is to try different setups and run many simulations. The parameters though, should be within a reasonable range, e.g. the CPVAs in the Advanced model should fit in an already existing flywheel on the market. In this work, the parameters have been chosen with consideration to space in general, and in the Advanced model the pendulum mass is restricted to 1 kg, a reasonable value for a DMF in a car powertrain. A parameter that, according to previous work in the field, has a big influence on performance is the damping of the absorber motion. This has not been investigated in this work, but a damping coefficient for this has been included in the models.

5 RESULTS FROM SIMULATIONS

5.1 Model with Sine-signal Torque

The results for the Sine-signal torque model is shown in Figure 5.1 – 5.6. The parameters used are found in Appendix E. The results are of type FFT – second order analysis for rotary speed, and thus show how well the CPVA unit absorbs vibrations of that order. The results are discussed in Chapter 6.

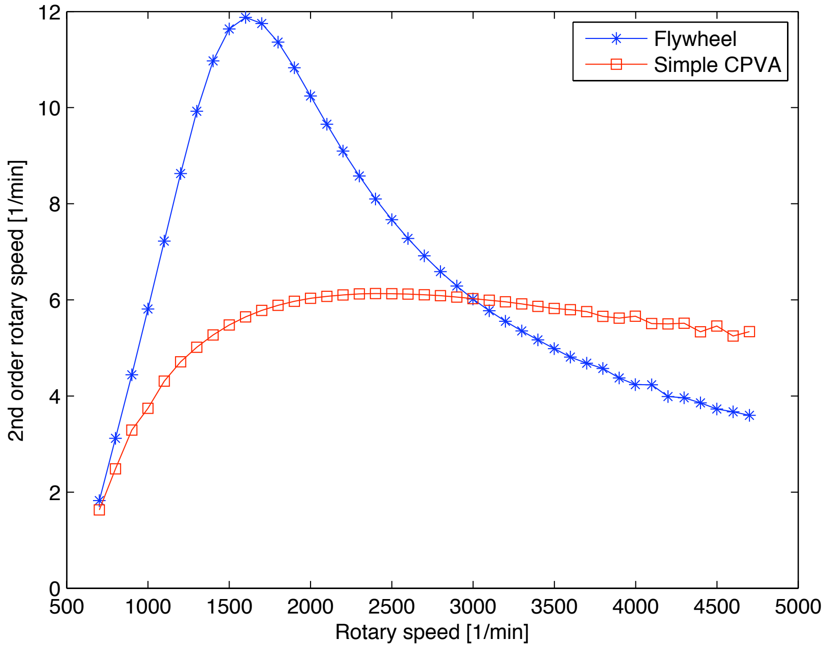


Figure 5.1: *Results of the Simple Type CPVA vs. Flywheel. Applied torque is* $T_e = 60 + 50 \cdot \sin(2\theta)$.

Vibrations are reduced in the lower engine speed range with the Simple Type, but over about 3000 rpm stability is lost and the CPVAs instead enhance vibrations.

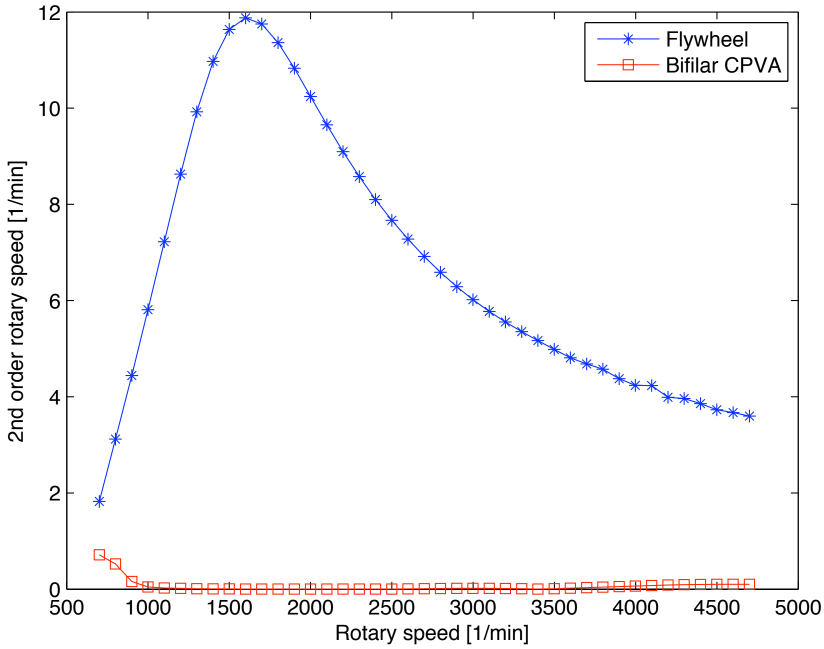


Figure 5.2: *Results of the Bifilar Type CPVA vs. Flywheel. Applied torque is* $T_e = 60 + 50 \cdot \sin(2\theta)$.

The Bifilar Type maintains stability over the entire speed range and also shows good performance.

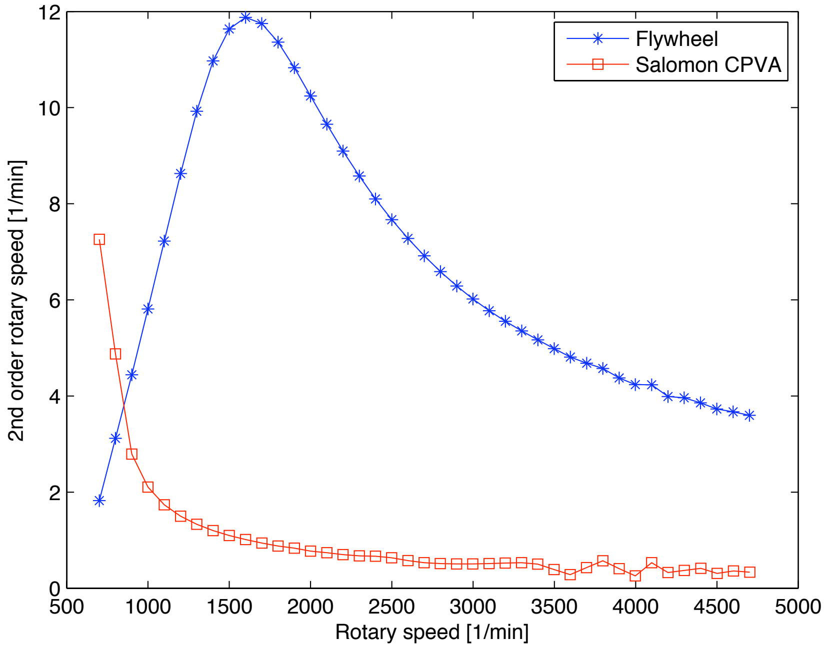


Figure 5.3: *Results of the Salomon Type CPVA vs. Flywheel. Applied torque is* $T_e = 60 + 50 \cdot \sin(2\theta)$.

The Salomon Type shows decent performance over most of the speed range, except at low speeds where stability is lost.

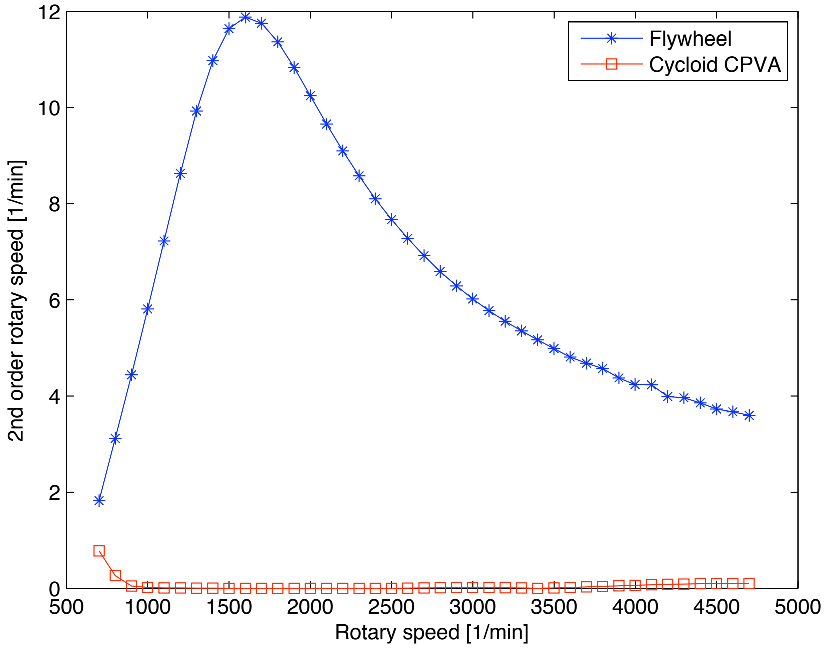


Figure 5.4: *Results of the Cycloid Path Type CPVA vs. Flywheel. Applied torque is* $T_e = 60 + 50 \cdot \sin(2\theta)$.

The Cycloid Type shows good performance in the entire speed range, similar to the Bifilar Type.

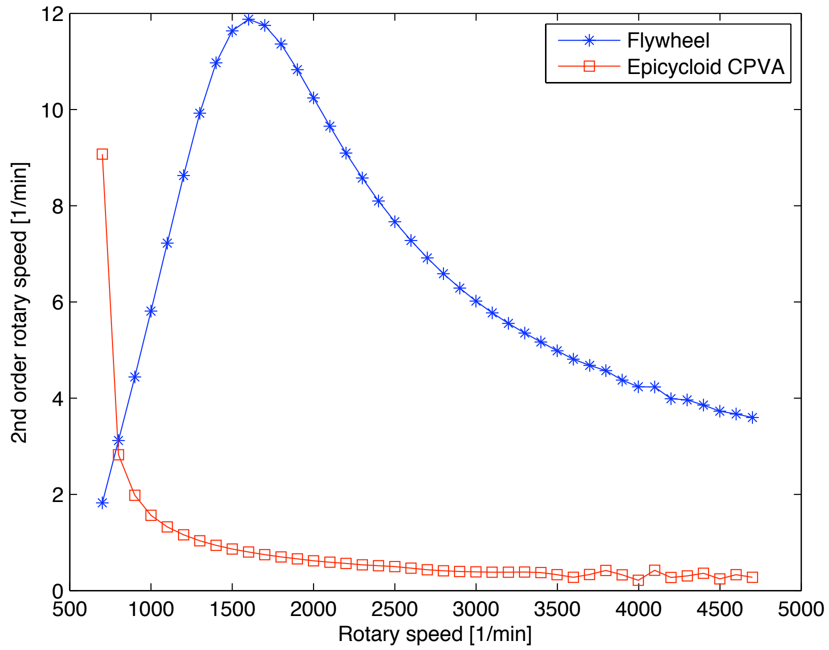


Figure 5.5: *Results of the Epicycloid (Cardoid) Path Type CPVA vs. Flywheel. Applied torque is* $T_e = 60 + 50 \cdot \sin(2\theta)$.

The Epicycloid (Cardoid) Type shows the same characteristics as the Salomon Type with decent performance except for lower engine speeds.

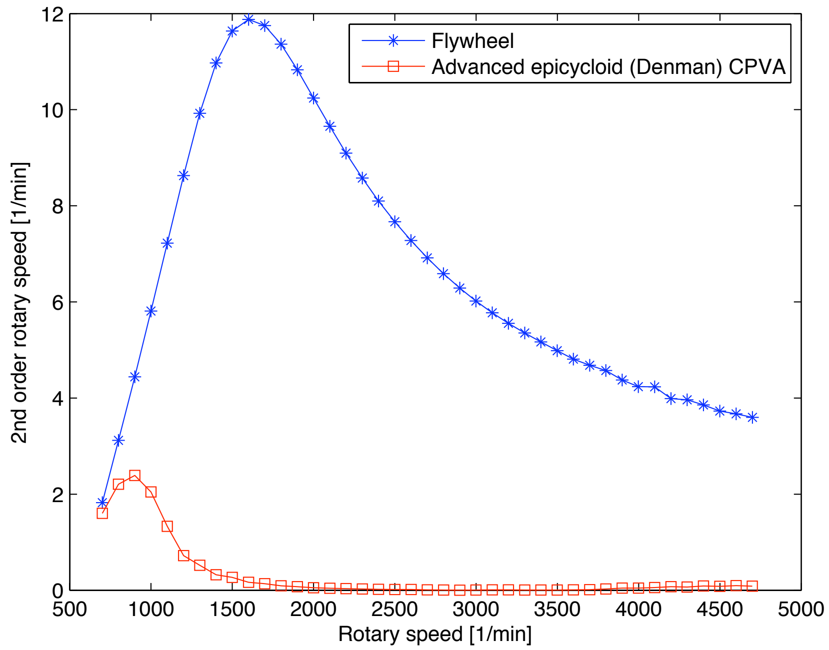


Figure 5.6: *Results of the Bifilar Type CPVA with Rollers and General Path vs. Flywheel. Applied torque is* $T_e = 60 + 50 \cdot \sin(2\theta)$.

The final type has almost the performance of Bifilar and Cycloid Type, but below about 1500 rpm the performance is worse, although with maintained stability.

5.2 Advanced Model with DMF, FEAD, Diesel Engine Torque

The results of the Advanced model are presented in this section and are, as in the previously shown Sine-signal model, FFT – 2nd order rotary speed results. In Figure 5.7, the results for the primary and secondary flywheel are shown, without any CPVAs. It can be seen that the isolation level is about 90% in the 2nd order domain. Figure 5.8 – 5.13 shows how the secondary flywheel is affected when different types of CPVAs are attached to it. The parameters used are found in Appendix E.

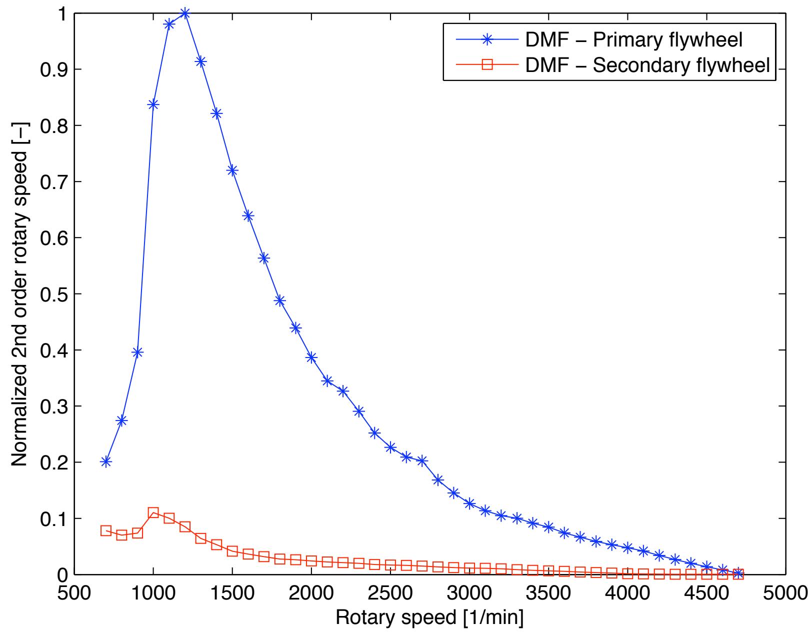


Figure 5.7: *Results in the Advanced model. Comparison of 2nd order rotary speed at primary and secondary flywheel.*

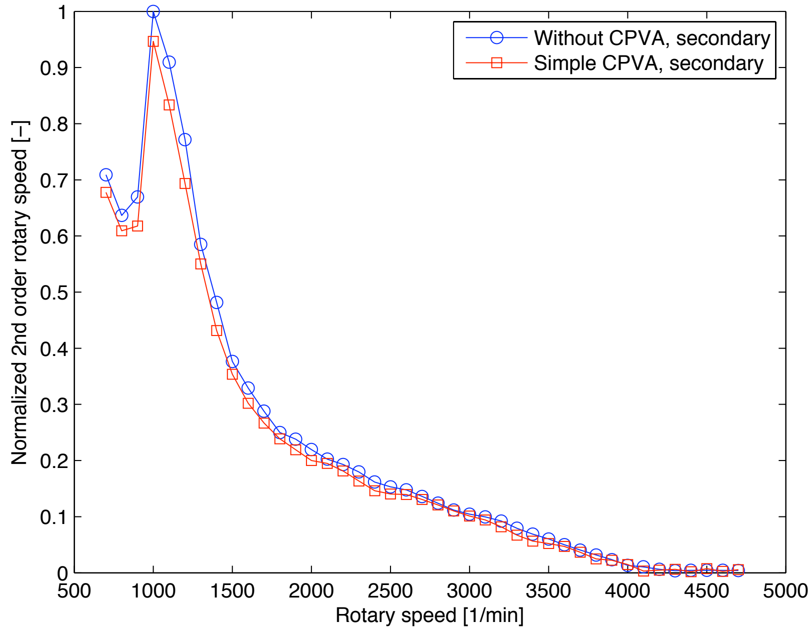


Figure 5.8: *Advanced model results – without CPVA and with Simple CPVA, secondary side.*

The Simple Type seems to have little effect on the vibrations of the secondary side in this case. The results are virtually the same as without use of CPVAs.

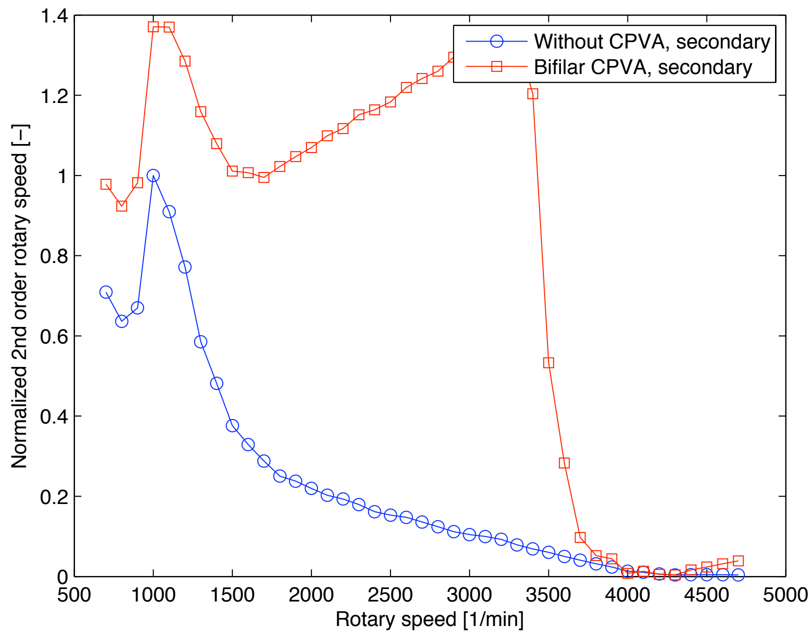


Figure 5.9: *Advanced model results – without CPVA and with Bifilar CPVA, secondary side.*

The Bifilar Type in this case gives results that are highly unsatisfactory. The vibrations are enhanced for the majority of the speed range.

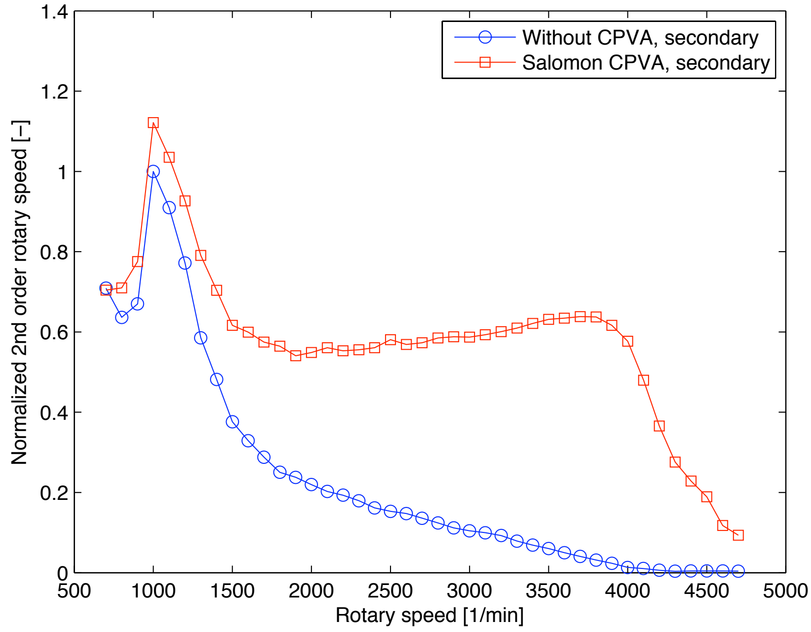


Figure 5.10: *Advanced model results – without CPVA and with Salomon CPVA, secondary side.*

The Salomon type also performs poorly. Vibrations are enhanced in the entire speed range.

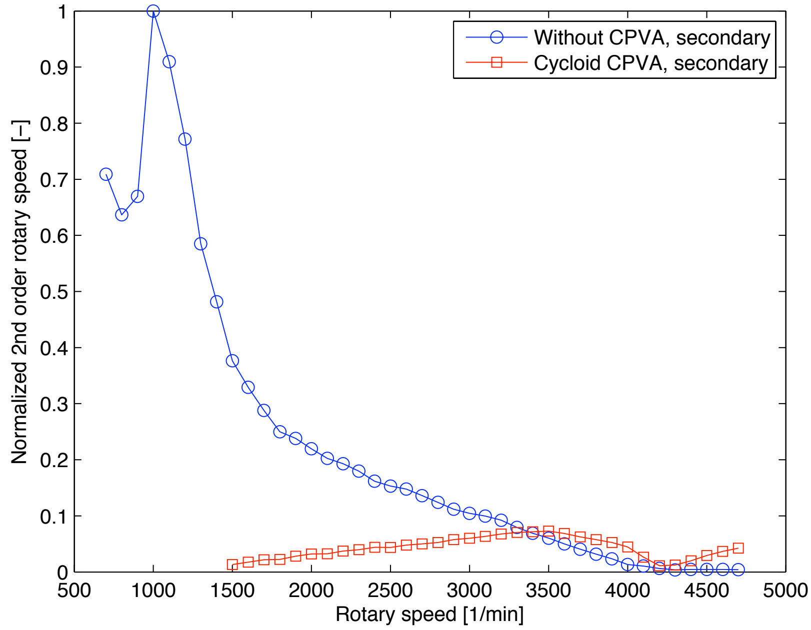


Figure 5.11: *Advanced model results – without CPVA and with Cycloid Path CPVA, secondary side.*

The Cycloid Type manages to reduce vibrations in the middle speed range, but enhances in the higher range. Simulations below 1500 rpm could not be run due to the reach of a cusp in the curve.

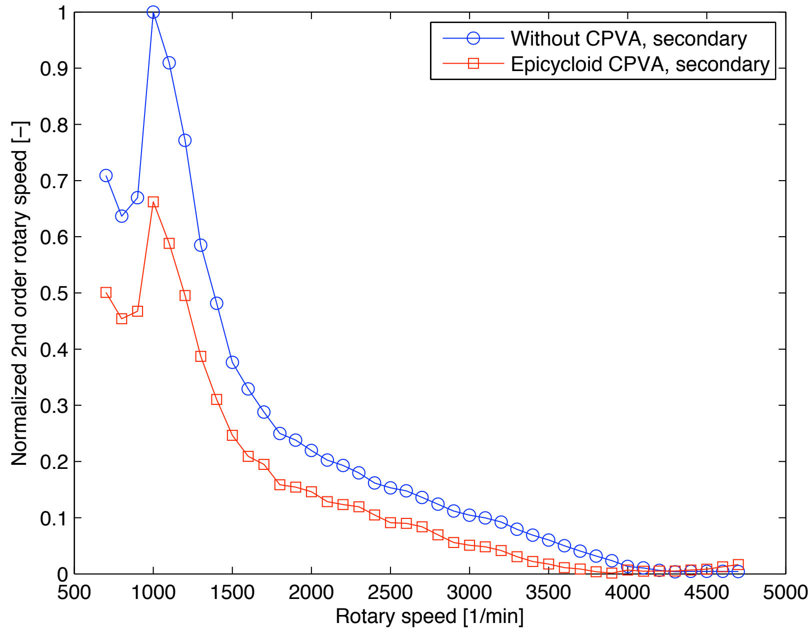


Figure 5.12: *Advanced model results – without CPVA and with Epicycloid CPVA, secondary side.*

The Epicycloid (Cardoid) Type seems to maintain stability and reduces vibrations to some extent, in the entire engine speed range.

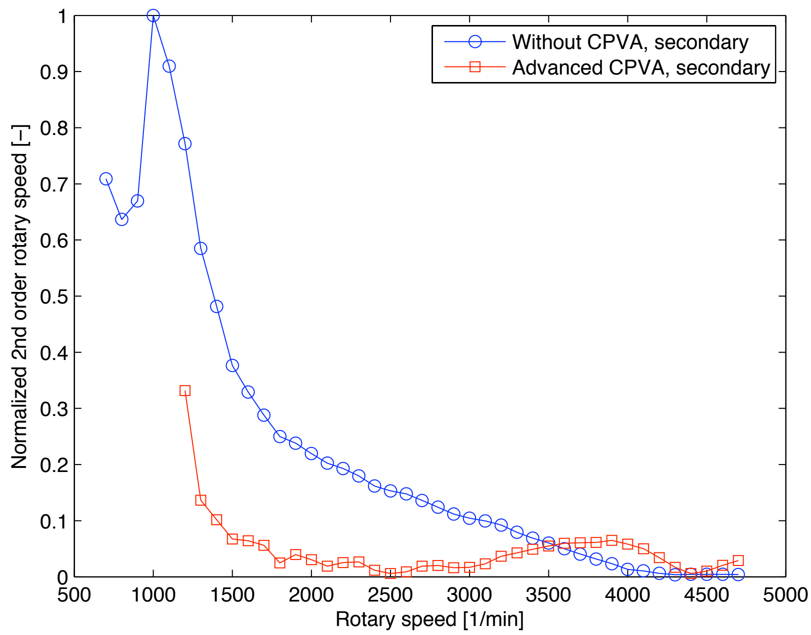


Figure 5.13: *Advanced model results – without CPVA and with Bifilar Type CPVA with Rollers and General Path, secondary side.*

The Bifilar Type with Rollers shows good performance in the middle speed range of the engine, but as with the Cycloid Type, vibrations are enhanced in the high range. Simulations below about 1200 rpm could not be run due to the reach of a cusp in the curve.

6 DISCUSSION

In this chapter, both the modeling and simulations performed in this work is discussed. The modeling process proved to be more time consuming than expected, mainly because the CPVA elements had to be programmed manually, and left a relatively small amount of time for simulations. The simulation results are therefore not fully optimized, more time is needed to adjust the parameters and fully utilize the potential of every CPVA type.

6.1 Modeling

Regarding the validity of the CPVA models created in Simdrive, it may be assumed that the models for Simple, Bifilar, and Salomon are modeled correctly. This is based on the fact that they give the same results as the already existing Simple, Bifilar, and Salomon CPVA elements in Simdrive. It can therefore also be assumed that the remaining models, Cycloid path, Epicycloid (cardoid) path, and Bifilar with Rollers and General Path, have been modeled correctly, assuming that no small errors have been made. In retrospect, the Cycloid and Epicycloid models are maybe not very convenient for simulations, and were made in an early phase of this work, when the understanding of curve types were not complete. Also, the Epicycloid model is really just one specific path in the whole range of different epicycloid paths. It is more convenient to have easy access to all possible paths. Therefore, the path representation as in section 2.7.6 presented by Denman [9] is much more convenient. In this model it is also easy to adjust the path, so it will have neither softening nor hardening non-linear behavior, the tautochronic condition.

The goals set up for this thesis are to create a Verification, Sine-signal, and Advanced model for simulations. The Verification model was used mostly under the development of CPVA models, and proved very helpful in that process. The derivation of equations as a process is quite long and sensitive to errors, and the same holds for the C/C++ programming. The Verification model and existing CPVA models in Simdrive were therefore a great help.

The Sine-signal model, as a simplified powertrain model, is nevertheless quite useful for investigations. It can be thought of as an experimental setup, a test bench where the behavior of the CPVA unit can be studied, with simple disturbing torques. Even though simple, this model is useful to estimate relationships between magnitude of disturbing torques, inertia of rotor, pendulum mass, and other CPVA measurements and to obtain an intuitive feel for them.

The Advanced model is more similar to a real car powertrain, the most important additions here is the realistic engine torque and the Dual Mass Flywheel (DMF). The CPVAs are here put in their right environment, as described in section 1.3. The realistic engine torque is important to include, since the effects of having a torque signal with multiple harmonic content can affect the CPVAs in different ways. The DMF is also important because it makes it possible to capture the isolation effect from primary to secondary side, ensuring that the CPVAs are subjected to a reasonable amount of torque. It should be pointed out though, that the model is still much simplified, especially on the secondary side. The clutch, gearbox, differential, driveshaft and other parts are lumped into one unit, and significant effects might be missed.

6.2 Simulations

As mentioned earlier, little relative time was left for simulations in this work, and the parameters in simulations performed are not fully optimized. The results obtained show that, when the parameters are tuned correctly, vibrations of the designated harmonic order may be virtually eliminated. Results from the Sine-signal torque model, in section 5.1, show that Bifilar, Cycloid Path, and the Bifilar Type with General Path eliminate second order vibrations almost completely, for a wide range of engine speeds. The Salomon and Epicycloid (Cardoid) Path type seems to achieve the same level of vibration absorption, with decent performance between 1500-4700 rpm. Under 1000 rpm and reaching down to 700 rpm though, the CPVA unit loses its absorbing capability and instead amplifies vibrations. Assumptions based on the theory in Chapter 2 are that the pendulum amplitudes here become too high, and that the absorber reaches the jump bifurcation condition. This might have been solved by introduction of overtuning to increase stability. The Simple type CPVA performs significantly worse here than the other types. The exact reason for this is not known but changes in parameters can increase its performance.

The results of the Advanced model shows more unpredictable behavior. First, Figure 5.7 shows the isolation properties of the Dual Mass Flywheel (DMF). In this particular setup, the isolation level is about 90%, which of course varies in practice. This level may be changed easily by parameters in the model. It seems that the model captures the isolation properties of the DMF well. As mentioned before, all CPVA units simulated in the advanced model have 1 kg of total pendulum mass, a reasonable value in practice.

Studies of Figure 5.8 – 5.13 show that results for the Simple type CPVA indicate only a slight advantage or no difference compared to the original setup with no CPVAs. The absorbers do not seem to lose their stability, neither they seem to operate satisfactory. The Bifilar type, on the other hand, seem to lose stability over the majority of speed range, from 700-4000 rpm, and only make vibrations much worse. This shows the double-faced behavior of the pendulum absorbers, they can also make things much worse. The Salomon type shows similar performance and fails to operate over the whole engine speed range. An interpretation of the results for the Bifilar and Salomon CPVA is that the pendulum amplitude is too high, and they lose stability through the jump bifurcation.

The Cycloid Path CPVA, on the other hand, seem to be capable of proper operation. Unfortunately it did not run under 1500 rpm due to numerical errors, i.e. the pendulum amplitude was too large. In the engine mid-range it manages to reduce vibrations, but over 3500 rpm it unfortunately instead enforce vibrations. The Epicycloid (Cardoid) model, compared to the Cycloid, does not have as good vibration reduction capabilities in the mid-range, but reduces vibrations steadily over the whole engine speed range. The final type, Bifilar with General Path, also experience numerical errors, below about 1200 rpm. It also shows good potential of reducing vibrations in the mid-range.

These simulations show some patterns. First, it can be observed that all CPVAs with cycloid or epicycloid paths perform well, in both the Sine-signal and Advanced model. Also, the Cycloid and Bifilar with General Path type seem to give similar results, and usually also good results. An observation for the Advanced model is that none of the regular types, Simple, Bifilar, and Salomon, seem to be able to operate properly. An interpretation of this could be that the torque exerted on the CPVAs is high relative to the pendulum mass, causing them to operate on their limit. The other types with alternate paths are the only types that can remain stable at those amplitudes. No conclusions will be drawn of any of this, since the number of simulations is too low. Worth noting is that the CPVAs not only affect the secondary side, but also the primary side. This is not within the scope of this thesis, but simulations show both decrease and increase of primary side vibrations, and not necessarily in conjunction with the secondary side. A final remark is that tests of the Simple, Bifilar, and Salomon should be performed with more overtuning, and see if that could increase stability.

7 CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

In this thesis, several types of CPVAs have been modeled and used in three types of simulation models in the software Simdrive 3D; a *Verification*, *Sine-signal torque* and *Advanced powertrain* model. The CPVA elements have been programmed manually in C/C++ and compiled to dll-files that are called by Simdrive. The already existing CPVA elements in Simdrive are the Simple, Bifilar, and Salomon type. These types have also been modeled manually and verified successfully in the verification model mentioned above. To avoid the softening non-linear behavior of circular pendulum paths, that may reduce performance and reduce stability, CPVAs with cycloidal and epicycloidal paths have also been modeled and simulated.

The results of the Sine-signal torque model show that, for the given CPVA parameters in Appendix E, every CPVA type seem to have the potential of reducing second order vibrations. Some of the CPVA models with cycloidal or epicycloidal paths manage to reduce irregularities to a level close to zero, over the major part of the engine speed range. Results of the Advanced powertrain model with CPVAs on the DMF show that the exciting torque from the engine seems high, compared to the predefined pendulum mass of 1 kg. The CPVA types with regular circular paths either do not affect vibration levels, or amplify them, which is highly undesirable. The cycloidal and epicycloidal path CPVAs though, show some potential of reducing vibrations. However, these types experience numerical issues in the lower engine speed range. It seems from simulations that the CPVA parameters must be tuned well for the designated pendulum mass. Since much time was spent on modeling in this work, little relative time was left for simulations. The parameters are therefore not optimized yet.

Some suggestions for future work are:

- Continue development of the Advanced powertrain model by adding more components.
- Introduce ”stoppers” in the CPVA models. In reality, there is usually a limit in pendulum amplitude due to physical constraints. This should be included in the model for increased realism. Also, numerical issues leading to simulation crashes could be avoided.
- Investigate the influence of parameter changes further. Try to establish some general guidelines for parameter relationships, other than just order tuning.
- Investigate the damping of the absorber, and how it affects the stability and performance. According to previous work, it has a large influence.
- Investigate the whole range of paths, from circular, through epicycloidal paths, up to cycloidal path, and how it affects stability and performance.
- Setup a model with support for several absorbers, which move independently with coordinates ϕ_1, ... , ϕ_n for n absorbers. This would be more realistic since the absorbers often do not move in unison.
- Investigate how reducing vibration levels of the primary harmonic order affect vibrations of other orders. Previous work, such as by Lee and Shaw [22], shows that reducing vibrations of the primary order might amplify vibrations of others orders, which creates a need for additional CPVAs tuned to other orders.
- Perform experimental tests of individual CPVAs, and experimental tests of CPVAs on a DMF in a car.

A final suggestion is to use the type of model with general path that can be adjusted with a single parameter, also used in previous work such as by Denman [9]. Then it is easier to adjust paths, for example to a tautochronic one, and also easier to compare results with previous work.

References

[1] Volvo Cars, August 2010. URL http://en.wikipedia.org/wiki/Volvo_Cars.

[2] W. Ker Wilson. *Practical Solution of Torsional Vibration Problems: With Examples from Marine, Electrical, Aeronautical and Automobile Engineering Practice*, volume four: Devices for Controlling Vibration. Chapman & Hall Ltd, 3rd edition, 1968.

[3] K. D. McCutcheon. No short days: The Struggle to Develop the R-2800 ”Double Wasp” Crankshaft, 2010. URL http://www.enginehistory.org.

[4] A. Albers. *Advanced Development of Dual Mass Flywheel (DMFW) Design - Noise Control for Today's Automobiles*.

[5] J. P. Den Hartog. *Mechanical Vibrations*. Dover Publications, Inc. New York, 1985.

[6] M. Cederwall and P. Salomonson. *An Introduction to Analytical Mechanics*. Chalmers University of Technology, 4th edition, April 2009.

[7] A. Boström. *Rigid Body Dynamics*. Chalmers University of Technology, October 2009.

[8] W. T. Thomson and M. D. Dahleh. *Theory of Vibration with Applications*. Prentice Hall, 5th edition, 1997.

[9] H. H. Denman. Tautochronic Bifilar Pendulum Torsion Absorbers for Reciprocating Engines. *Journal of Sound and Vibration*, 159(2):251–277, 1992.

[10] N. C. Rana and P. S. Joag. *Classical Mechanics*. Tata McGraw-Hill Publishing Company Limited, 1991.

[11] Wolfram MathWorld, December 2010. URL http://mathworld.wolfram.com.

[12] A. S. Alsuwaiyan and S. W. Shaw. Performance and Dynamic Stability of General-Path Centrifugal Pendulum Vibration Absorbers. *Journal of Sound and Vibration*, 252(5):791–815, 2002.

[13] A. G. Haddow and S. W. Shaw. Centrifugal Pendulum Vibration Absorbers: An Experimental and Theoretical Investigation. *Nonlinear Dynamics*, 34:293–307, 2003.

[14] C.-P. Chao, S. W. Shaw, and C.-T. Lee. Stability of the Unison Response for a Rotating System with Multiple Tautochronic Pendulum Vibration Absorbers. *Journal of Applied Mechanics*, 64:149–156, March 1997.

[15] A. S. Alsuwaiyan. *Performance, Stability, and Localization of Systems of Vibration Absorbers*. PhD thesis, Michigan State University, 1999.

[16] D. E. Newland. *Nonlinear Vibrations: A Comparative Study with Applications to Centrifugal Pendulum Vibration Absorbers*. PhD thesis, Massachusetts Institute of Technology, 1963.

[17] T. M. Nester. Experimental Investigation of Circular Path Centrifugal Pendulum Vibration Absorbers. Master's thesis, Michigan State University, 2002.

[18] von A. Silbar and K. Desoyer. Zur Erzielung optimaler Wirkung bei Pendel-Schwingungstilgern. *Ingenieur-Archiv*, (XXII):36–44, May 1954.

[19] von P. R. Paslay and A. Silbar. Optimale Auslegung von Salomon-Schwingungstilgern. *Ingenieur-Archiv*, (XXIV):182–187, 1956.

[20] A. Emmerson. *Things are seldom what they seem - Christian Huygens, the Pendulum and the Cycloid*.

[21] M. Albright, T. Crawford, and F. Speckhart. Dynamic Testing and Evaluation of the Torsional Vibration Absorber. *SAE*, pages 185–192, 1994.

[22] C.-T. Lee and S. W. Shaw. Torsional Vibration Reduction in Internal Combustion Engines Using Centrifugal Pendulums. In *Vibration of Nonlinear, Random, and Time-Varying Systems*, volume 3, pages 487–492, 1995.

[23] C.-P. Chao, C.-T. Lee, and S. W. Shaw. Non-Unison Dynamics of Multiple Centrifugal Pendulum Vibration Absorbers. *Journal of Sound and Vibration*, 204(5):769–794, 1997.

[24] C.-T. Lee and S. W. Shaw. The Non-Linear Dynamic Response of Paired Centrifugal Pendulum Vibration Absorbers. *Journal of Sound and Vibration*, 203(5):731–743, 1997.

[25] C.-P. Chao and S. W. Shaw. The Effects of Imperfections on the Performance of the Subharmonic Vibration Absorber System. *Journal of Sound and Vibration*, 215(5):1065–1099, 1998.

[26] C.-P. Chao and S. W. Shaw. The Dynamic Response of Multiple Pairs of Subharmonic Torsional Vibration Absorbers. *Journal of Sound and Vibration*, 231(2):411–431, 2000.

[27] A. S. Alsuwaiyan and S. W. Shaw. Localization of Free Vibration Modes in Systems of Nearly Identical Vibration Absorbers. *Journal of Sound and Vibration*, 228(3):703–711, 1999.

[28] F. Vidal, D. Drecq, and G. Louradour. P.A.R.I.S. : Pendulum Acyclism Reducer Integrated System. In *Seoul 2000 FISITA World Automotive Congress*, number F2000H219, June 2000.

[29] T. M. Nester, A. G. Haddow, and S. W. Shaw. Vibration Reduction in a Variable Displacement Engine Using Pendulum Absorbers. In *Noise & Vibration Conference and Exhibition*. SAE International, May 2003.

[30] S. W. Shaw, P. M. Schmitz, and A. G. Haddow. Tautochronic Vibration Absorbers for Rotating Systems. *Journal of Computational and Nonlinear Dynamics*, 1:283–293, October 2006.

[31] M. Pfabe and C. Woernle. Reduction of Periodic Torsional Vibration using Centrifugal Pendulum Vibration Absorbers. In *PAMM Proc. Appl. Math. Mech.*, volume 9, pages 285–286, 2009.

[32] Y. Ishida and T. Inoue. Torsional Vibration Suppression by Roller Type Centrifugal Vibration Absorbers. *Journal of Vibration and Acoustics*, 131(051012), October 2009.

[33] C. Gauld. The Treatment of Cycloidal Pendulum Motion in Newton's *Principa*. *Science & Education*, 13:663–673, 2004.

[34] A. Fidlin and R. Seebacher. DMF simulations techniques – Finding the needle in the haystack. In *LuK Symposium*, pages 55–71, 2006.

[35] U. Schaper, O. Sawodny, T. Mahl, and U. Blessing. Modeling and Torque Estimation of an Automotive Dual Mass Flywheel. In *American Control Conference*, pages 1207–1212, June 2009.

[36] L. Prejawa. *Online-Help RPM Dependent Vibration Absorber*. CONTECS Engineering Services GmbH, August 2009.

[37] L. Prejawa. *Modeling possibilities of dynamic vibration absorbers in SIMDRIVE3D*. CONTECS Engineering Services GmbH, June 2010.

[38] *Simdrive 3D User Guide*. CONTECS Engineering Services GmbH, 3.5 beta edition, January 2005.

Appendices

A Matlab Code

A.1 CPVA Dynamics Simulation Code

```
% [runsystem.m]
% Example for Simple Type CPVA
% m-file to run simulations
%

clear all
close all

R=0.17; % Radius to pendulum attachment point on rotor (PPR)
r=0.04; % Radius from PPR to centre of mass for absorber
m=1; % Mass of absorbers
J=0.5; % Moment of inertia for rotor
klambda=0.004; % Radius of gyration for absorbers

% Two initial conditions are set: The angular speed of the rotor
% and the position of the pendulum absorber
Initspeedtheta=(1000)*2*pi/60; % (rpm)*2*pi/60
Initposabsorber=(0)/180*pi;    % (degrees)/180*pi

% Calculate the harmonic order n of CPVA arrangement
n=sqrt((r*R)/(r^2+klambda^2))

% Set start and end time, and initial conditions
t0=0; tf=5; y0=[0 Initspeedtheta Initposabsorber 0]'; % [theta, thetadot phi phidot]
tspan=[t0 tf];
% Solve with ode45
[t,y]=ode45('dynsystem',tspan,y0);

% Extract angle and angular speed from y-vector
theta=y(:,1); % Rotor angle
thetadot=y(:,2); % Rotor angular speed
phi=y(:,3); % Pendulum absorber angle
phidot=y(:,4); % Pendulum absorber angular speed

%%% Plots vs. time %%%

figure(1) % Plot rotor angle
plot(t,theta/pi*180), grid
xlabel('time [s]')
ylabel('theta(t), [degrees]')

figure(2) % Plot rotor angular speed
plot(t,thetadot/pi*180), grid
xlabel('time [s]')
ylabel('thetadot(t), [degrees/s]')

figure(3) % Plot absorber angle
plot(t,phi/pi*180), grid
xlabel('time [s]')
ylabel('phi(t), [degrees]')

figure(4) % Plot absorber angular speed
plot(t,phidot/pi*180), grid
xlabel('time [s]')
ylabel('phidot(t), [degrees/s]')

%%% Position plots %%%

% Calculate positions of absorber attachment point on rotor
x1=R.*sin(theta);
y1=-R.*cos(theta);

% Calculate positions for absorber centre of mass
x2=R.*sin(theta)+r.*sin(theta+phi);
y2=-R.*cos(theta)-r.*cos(theta+phi);

% Plot the above positions
figure(5)
plot(x1,y1)
hold on
```

```
plot(x2,y2)
axis([-(R+r)*1.1 (R+r)*1.1 -(R+r)*1.1 (R+r)*1.1])
```

```
% [dynsystem.m]
% Example for Simple Type CPVA
% function file to describe dynamics of CPVA,
% called by runsystem.m
%

function ydot=dynsystem(t,y)
R=0.17;        % Radius to pendulum attachment point(PAP) on rotor
r=0.04;        % Radius from POP to centre of mass for absorber
m=1;           % Pendulum absorber mass
J=0.5;         % Moment of inertia for rotor
klambda=0.004; % Radius of gyration for absorber
A=60;          % Amplitude of constant applied torque
a=50;          % Amplitude of fluctuating applied torque
n=2;           % Harmonic order of fluctuating applied torque

Ta=(A+a*sin((n)*y(1))); % Applied external torque
ca=0.1;                 % Absorber viscous damping
cs=0;                   % Rotor viscous damping

% Constants to clean up a bit
k1=m*(R^2+r^2+klambda^2)+J;
k2=m*R*r;
k3=m*(r^2+klambda^2);

% A*ydot=f => ydot=A^(-1)*f

A=[1    0                0    0;
   0    k1+2*k2*cos(y(3))  0    k3+k2*cos(y(3))
   0    0                1    0;
   0    k3+k2*cos(y(3))  0    k3];

f=[y(2);
   k2*(2*y(2)*sin(y(3))+y(4)*sin(y(3)))*y(4)+Ta-cs*y(2);
   y(4);
   -k2*sin(y(3))*(y(2))^2-ca*y(4)];

ydot=A\f;
```

A.2 Circular/Cycloidal/Epicycloidal Path Code

```
%
% [pathconstruction.m]
% This m-file is a good aid for constucting pathss
%

clear all
close all

R=10; % Radius of rotor circle
r=3; % Radius of circular absorber path
k=1; % Degree of epicycloid
epiR=k*(r/3); % Constant for epicycloid

% Plot absorber paths at rotor angles defined by thetavec:
thetavec=[pi/2+pi,-pi/8];

phi=linspace(-pi,pi,1000);

% For the rotor circle
circtheta=linspace(0 , 2*pi, 1000);

% This for-loop is for epicycloid and circular path
for i=1:length(thetavec)

    % Extract rotor posistions
    theta=thetavec(i);

    % Circular path
```

```
    x2=R.*cos(theta)+r.*cos(theta+phi);
    y2=R.*sin(theta)+r.*sin(theta+phi);

    % Epicycloid equations
    xepi=-((epiR+(r/3))*cos(phi+pi)-(r/3)*cos(((r/3)+epiR)/(r/3)*(phi+pi)));
    yepi=-((epiR+(r/3))*sin(phi+pi)-(r/3)*sin(((r/3)+epiR)/(r/3)*(phi+pi)));

    % Rotation of epicycloidal path
    xprime=xepi*cos(theta)-yepi*sin(theta);
    yprime=xepi*sin(theta)+yepi*cos(theta);

    % Add the posistion on the rotor
    xepifinal=R.*cos(theta)+xprime;
    yepifinal=R.*sin(theta)+yprime;

    % Plot epicycloid path
    plot(xepifinal,yepifinal,'r')
    hold on

    % Plot circular path
    plot(x2,y2)

end

phi2=linspace(-pi,pi,1000); % Phi for cycloid

% This for-loop is for cycloid
for i=1:length(thetavec)
    theta=thetavec(i);

    % Equations for cycloid path
    xcycl=(r/4*(2*phi2+sin(2*phi2)));
    ycycl=r/4*(1-cos(2*phi2))-r;

    % Rotation of path
    xprimecycl=xcycl*cos(theta+pi/2)-ycycl*sin(theta+pi/2);
    yprimecycl=xcycl*sin(theta+pi/2)+ycycl*cos(theta+pi/2);

    % Add position on rotor
    xcyclfin=R.*cos(theta)+xprimecycl;
    ycyclfin=R.*sin(theta)+yprimecycl;

    plot(xcyclfin,ycyclfin,'black')

end

% Draw the rotor circle
xcirc=R*cos(circtheta);
ycirc=R*sin(circtheta);
plot(xcirc,ycirc,'--')
axis('equal')
title([num2str(length(thetavec)),' positions'])
```

B C/C++ Code for dll Simple Type CPVA Element in Simdrive

```
#include <string.h>
#include <stdio.h>
#include <math.h>

//Possible Output-Flags
#define TIME 1                                  //BIT 0
#define AVERAGE 2                               //BIT 1
#define MAXIMUM 4                               //BIT 2
#define MINIMUM 8                               //BIT 3
#define PEAK_TO_PEAK 16                 //BIT 4
#define FFT 64                                  //BIT 6
#define PEAK_TO_PEAK_2 1024             //BIT 10
#define ARBEITSSPIEL 2048               //BIT 11

#define NUMB_OUTPUTS 7 // Define the number of outputs to results in Simdrive
//Ouput ------------------------------------------------------------------------------------------------------------------------------
// Define the results that should be used as outputs
char Output_Functions[NUMB_OUTPUTS][20]={"Angle", "Angular_Speed", "Angular_Acc","Pendelwinkel", "pendulum_omega", "pendulum_omega_p","Rotary_Speed"};  // Changed, added pend_angle
// Define the units for those results, e.g. for "Angle" the unit will be "Degree"
char Output_Units[NUMB_OUTPUTS][20]={"Degree", "rad/s", "rad/s2","DEGREE", "rad/s", "rad/s2","1/min"};                          // Changed, added unit for pend_angle
// Choose output results types from the "Possible Output-Flags" list above
long int Output_Flags[NUMB_OUTPUTS]={TIME+FFT+PEAK_TO_PEAK_2+MAXIMUM+MINIMUM, TIME+FFT+PEAK_TO_PEAK_2+MAXIMUM+MINIMUM, TIME+FFT+PEAK_TO_PEAK_2+MAXIMUM+MINIMUM, TIME+FFT+PEAK_TO_PEAK_2+MAXIMUM+MINIMUM, TIME+FFT+PEAK_TO_PEAK_2+MAXIMUM+MINIMUM, TIME+FFT+PEAK_TO_PEAK_2+MAXIMUM+MINIMUM,TIME+FFT+PEAK_TO_PEAK_2+MAXIMUM+MINIMUM};


// This shows the structure of this DLL file. Index p stands for previous and n for next,
// referring to the prev and next element in the Simdrive model structure. C1, C2, and B1, B2 are stiffness
// and damping coefficients. The whole CPVA element corresponds to the "J" in the figure, in this code.

//            |<---------------DLL--------------->|
//-----------+++------------+-------+------------+++-----------
//Jp_simdrive|||===C1, B1===|===J===|===C2, B2===|||Jn_simdrive
//-----------+++------------+-------+------------+++-----------


// Rotor properties inputs (observe that the assigned values can be replaced in the external rotary unit table)
double J=1;                                     // moment of inertia [kgm2]     (For the whole pendular mass unit, "Moment of Inertia" in Simdrive)
double cs=0.;                           // viscous damping of the rotor [Ns/m]

// Pendular masses inputs (observe that the assigned values can be replaced in the external rotary unit table)
double mp=1;                            // mass per pendulum [kg]
double Ipend=0.004;                     // moment of inertia for each pendulum [kgm2]
double R=0.2;                           // radius from centre to absorber mount [m] (Distance Pivot-Guiding Point)
double r=0.05;                          // distance from absorber mount to absorber centre of mass [m] (Distance Guiding Pendulum)
double NumberPedMasses=2;       // Number of pedular masses [-]
double ca=0.;                           // viscous damping of absorber [Ns/m](not available in DAT mass in Simdrive)

// Constants used in calculation
double m=mp*NumberPedMasses;    // total mass of absorbers [kg] (must be initialised in init)
double I=Ipend*NumberPedMasses; // total inertia of CPVA [kgm2] (must be initialised in init)

// Other constants
double PI=acos(0.0)*2;                  // Define Pi
double C1=1000000;                              // Stiffness on 1st side (Probably numbered in the direction of x)
double B1=0.;                                   // Damping on 1st side
double C2=14000;                                // Stiffness on 2nd side
```

```
double B2=0.;                                   // Damping on 2nd side

double phi_J[3];                // This array contains Angle/Angular speed/Angular acceleration for rotor
double phi_Pendulum[3]; // Same as above but for pendulum (or more general, the second generalised coordinate)

bool used=0;



// Here, the values are exported from phi_J and phi_Pendulum and given a unit name
extern "C" __declspec (dllexport) double Angle()
{
        return phi_J[0]/PI*180;
}

extern "C" __declspec (dllexport) double Angular_Speed()
{
        return phi_J[1];
}

extern "C" __declspec (dllexport) double Angular_Acc()
{
        return phi_J[2];
}

extern "C" __declspec (dllexport) double Pendelwinkel()
{
        return phi_Pendulum[0]/PI*180;
}

extern "C" __declspec (dllexport) double pendulum_omega()
{
        return phi_Pendulum[1];
}

extern "C" __declspec (dllexport) double pendulum_omega_p()
{
        return phi_Pendulum[2];
}

extern "C" __declspec (dllexport) double Rotary_Speed()
{
        return phi_J[1]*(-60/(2*PI));
}


extern "C" __declspec (dllexport) int init_output(int ID, char **names, char **units, long int **flags)
{
        *names=&Output_Functions[0][0];
        *units=&Output_Units[0][0];
        *flags=&Output_Flags[0];
        //Return number of output results
        return NUMB_OUTPUTS;
}

//
----------------------------------------------------------------------------------------------------------------

extern "C" __declspec (dllexport) bool construct(int Simdrive_ID, int *DLL_ID)
{
        if(used!=0){
                printf("DLL wurde bereits eingeladen\n");
                return 0;
        }

        used=1;
        return 1;
}

extern "C" __declspec (dllexport) bool destruct(int ID)
{
        return 1;
}
```

```
//
    ------------------------------------------------------------------------------------------------------------

extern "C" __declspec (dllexport) bool set_value(int ID, char *name, double value)

// The purpose of this section is to compare the parameter names used as input in Simdrive
// with the ones in the code. This ensures that the right parameters are used.

{
        if(strcmp(name, "J")==0)
        {
                J=value;
                return 1;
        }

        if(strcmp(name, "R")==0)
        {
                R=value;
                return 1;
        }

        if(strcmp(name, "r")==0)
        {
                r=value;
                return 1;
        }

        if(strcmp(name, "Ipend")==0)
        {
                Ipend=value;
                return 1;
        }

        if(strcmp(name, "mp")==0)
        {
                mp=value;
                return 1;
        }

        if(strcmp(name, "ca")==0)
        {
                ca=value;
                return 1;
        }

        if(strcmp(name, "cs")==0)
        {
                cs=value;
                return 1;
        }

        if(strcmp(name, "NumberPedMasses")==0)
        {
                NumberPedMasses=value;
                return 1;
        }

        if(strcmp(name, "C1")==0)
        {
                C1=value;
                return 1;
        }

        if(strcmp(name, "B1")==0)
        {
                B1=value;
                return 1;
        }

        if(strcmp(name, "C2")==0)
        {
                C2=value;
                return 1;
        }
```

```
	if(strcmp(name, "B2")==0)
	{
		B2=value;
		return 1;
	}


	//Parameter could not be set-->Error in Simdrive if "return 0"
	return 0;
};


extern "C" __declspec (dllexport) bool init(int ID, int *numb_diffeq_order1, int *
    numb_diffeq_order2)
{
	//Set Number Differential equations 1. Order
	*numb_diffeq_order1=0;
	//Set Number Differential equations 2. Order
	// *numb_diffeq_order2=1;			Commented
	*numb_diffeq_order2=2;		//	Added

	if(J<=0){
		printf("J-Val is not valid in DLL, J>0!\n");
		return 0;
	}

	// Here, the total pendulum mass for CPVA, total inertia for CPVA and Pi is initialised
	m=NumberPedMasses*mp;
	I=Ipend*NumberPedMasses;
	PI=acos(0.0)*2;



	//Initilize
	phi_J[0]=phi_J[1]=phi_J[2]=0.;
	phi_Pendulum[0]=phi_Pendulum[1]=phi_Pendulum[2]=0.;
	return 1;
}

extern "C" __declspec (dllexport) bool set_start_val_rot(int ID, double *p_phi, double *n_phi,
    double **Yn1, double **Yn2)
{
	//**Yn1[i] ->i.th Equation 1. Order-->Yn1[i][0]==Y(t), Yn1[i][1]=dY(t)_dt
	//**Yn2[i] ->i.th Equation 2. Order-->Yn2[i][0]==Y(t), Yn2[i][1]=dY(t)_dt, Yn2[i][2]=
	    d2Y(t)_dt2

	// Set start values
	//Set Omega Start for DLL-Mass,
	Yn2[0][1]=p_phi[1];
	//Only for Output
	phi_J[1]=p_phi[1];

	return 1;
}

extern "C" __declspec (dllexport) bool set_start_val_3D(int ID, double **p_phi, double **n_phi,
     double **Yn1, double **Yn2)
{
	return set_start_val_rot(ID, p_phi[3], n_phi[3], Yn1, Yn2);
}

extern "C" __declspec (dllexport) int calc_reaction_rot(int ID, double *p_phi, double *n_phi,
    double *T_prev, double *T_next, double **Yn1, double **Yn2)
{
	//p_phi-->vector at prev_mass angle, omega, acceleration
	//n_phi-->vector at next_mass angle, omega, acceleration
	//T_prev-->Torque at prev Mass
	//T_next-->Torque at next Mass

	//Torque at prev-mass, positive torque accelerates prev_mass in positive direction
	 double T_prev_J=C1*(Yn2[0][0]-p_phi[0])+B1*(Yn2[0][1]-p_phi[1]);
	//Torque at J-Mass, positive torque accelerates J_mass in positive direction
	 double T_J_next=C2*(n_phi[0]-Yn2[0][0])+B2*(n_phi[1]-Yn2[0][1]);

	//Set Torques at P-mass and N-Mass
```

```
        *T_prev=T_prev_J;
        *T_next=-T_J_next;

        // The net applied torque is the sum of the previous and next torque
        double Ta=-T_prev_J+T_J_next;

        // Yn2[0][0]=theta  Yn2[0][1]=theta_dot   Yn2[0][2]=theta_dot_dot
        // Yn2[1][0]=phi    Yn2[1][1]=phi_dot     Yn2[1][2]=phi_dot_dot

        // This is where the dynamics of the CPVA is inserted, as rotor acceleration and
            pendulum acceleration.
        // They can be inserted directly from Mathematica.

        // Rotor acceleration
        Yn2[0][2]=((I + m*pow(r,2))*Ta - cs*(I + m*pow(r,2))*Yn2[0][1] +
     m*r*(I + m*pow(r,2))*R*pow(Yn2[1][1],2)*sin(Yn2[1][0]) +
     m*r*R*pow(Yn2[0][1],2)*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0]))*
      sin(Yn2[1][0]) + Yn2[1][1]*(ca*
        (I + m*pow(r,2) + m*r*R*cos(Yn2[1][0])) +
       2*m*r*(I + m*pow(r,2))*R*Yn2[0][1]*sin(Yn2[1][0]))))/
   ((I + m*pow(r,2))*(J + m*pow(R,2)) -
     pow(m,2)*pow(r,2)*pow(R,2)*pow(cos(Yn2[1][0]),2));

        // Pendulum acceleration
        Yn2[1][2]= (-(Ta*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0]))) +
     cs*Yn2[0][1]*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0])) -
     m*r*R*pow(Yn2[1][1],2)*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0]))*
      sin(Yn2[1][0]) - m*r*R*pow(Yn2[0][1],2)*
      (I + J + m*pow(r,2) + m*pow(R,2) + 2*m*r*R*cos(Yn2[1][0]))*
      sin(Yn2[1][0]) - Yn2[1][1]*(ca*(I + J + m*pow(r,2) +
          m*pow(R,2) + 2*m*r*R*cos(Yn2[1][0])) +
        2*m*r*R*Yn2[0][1]*(I + m*pow(r,2) + m*r*R*cos(Yn2[1][0]))*
         sin(Yn2[1][0])))/
   ((I + m*pow(r,2))*(J + m*pow(R,2)) -
     pow(m,2)*pow(r,2)*pow(R,2)*pow(cos(Yn2[1][0]),2));

        //Only for Output
        phi_J[0]=Yn2[0][0];
        phi_J[1]=Yn2[0][1];
        phi_J[2]=Yn2[0][2];

        phi_Pendulum[0]=Yn2[1][0];
        phi_Pendulum[1]=Yn2[1][1];
        phi_Pendulum[2]=Yn2[1][2];

        return 1;
};

extern "C" __declspec (dllexport) int calc_reaction_3D(int ID, double **p_phi, double **n_phi,
    double *T_prev, double *T_next, double **Yn1, double **Yn2)
{
        for(int i=0; i<6; i++) T_prev[i]=T_next[i]=0.;
        return calc_reaction_rot(ID, p_phi[3], n_phi[3], &T_prev[3], &T_next[3], Yn1, Yn2);
}
```

C Equations of Motion for Bifilar Type with Rollers and General-Path Representation (Mathematica)

Rotor equation

$$
s'[t]\left(-\frac{\left(mP+\frac{mR}{2}\right)\left(n^2+n^4\right)s[t]\,s'[t]}{\sqrt{c^2-\left(n^2+n^4\right)s[t]^2}}+\right.
$$

$$
\left((c-2\,Ddis)\,mR\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)\mathrm{Sin}\left[\frac{\mathrm{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]\right.
$$

$$
\left.\left.s'[t]\right)\Big/\left(2\,(c\,mP+Ddis\,mR)\sqrt{1-\frac{n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)^2s[t]^2}{(c\,mP+Ddis\,mR)^2\left(1+n^2\right)}}\right)\right)+
$$

$$
\left(-2\,mP\,n^2\,s[t]\,s'[t]+\frac{1}{2}\,mR\left(-2\,n^2\,s[t]\,s'[t]+2\,(c-2\,Ddis)\left(-1-n^2\right)\right.\right.
$$

$$
\left(\left(n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)\mathrm{Cos}\left[\frac{\mathrm{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]s[t]\,s'[t]\right)\Big/\right.
$$

$$
\left((c\,mP+Ddis\,mR)\left(1+n^2\right)\sqrt{1-\frac{n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)^2s[t]^2}{(c\,mP+Ddis\,mR)^2\left(1+n^2\right)}}\right)-
$$

$$\Bigg(2\,a^2\,n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)^2\,Cos\left[\frac{ArcSin\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\,\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)\,s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]$$

$$s[t]\,s'[t]\Bigg)\Big/\Bigg(\left(c\,mP+Ddis\,mR\right)\left(1+n^2\right)\left(iR\,n^2+a^2\,\left(2\,mP+mR\right)\left(1+n^2\right)\right)$$

$$\sqrt{1-\frac{n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)^2\,s[t]^2}{\left(c\,mP+Ddis\,mR\right)^2\left(1+n^2\right)}}\Bigg)+$$

$$\frac{n^2\,Sin\left[\frac{ArcSin\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\,\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)\,s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]\,s'[t]}{1+n^2}-\Bigg(2\,a^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)Sin\Bigg[$$

$$\frac{ArcSin\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\,\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)\,s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\Bigg]\,s'[t]\Bigg)\Big/\left(iR\,n^2+a^2\,\left(2\,mP+mR\right)\left(1+n^2\right)\right)\Bigg)\Bigg)\Bigg)$$

$$theta'[t]+\Bigg(-\frac{iR}{a^2}-\frac{1}{2}\left(c-2\,Ddis\right)mR\,Cos\left[\frac{ArcSin\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\,\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)\,s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]+$$

$$\left.\left(\text{mP} + \frac{\text{mR}}{2}\right) \sqrt{c^2 - \left(n^2 + n^4\right) s[t]^2}\right) s''[t] +$$

$$\left(\text{Inertia} + 2\,\text{iR} + J + \text{mP}\left(c^2 - n^2 s[t]^2\right) + \frac{1}{2}\,\text{mR}\left(c^2 + (c - 2\,\text{Ddis})^2 + 4\,\text{HO}^2 - n^2 s[t]^2 + \right.\right.$$

$$2\,(c - 2\,\text{Ddis})\left(-1 - n^2\right)\left(c - \frac{c\,n^2}{1 + n^2} - \frac{2\,a^2\,(c\,\text{mP} + \text{Ddis}\,\text{mR})}{\text{iR}\,n^2 + a^2\,(2\,\text{mP} + \text{mR})\left(1 + n^2\right)} + \right.$$

$$\left(2\,a^2\,(c\,\text{mP} + \text{Ddis}\,\text{mR})\,\text{Cos}\left[\frac{\text{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{\text{iR}\,n^2}{2\,a^2} + \left(\text{mP} + \frac{\text{mR}}{2}\right)\left(1+n^2\right)\right) s[t]}{c\,\text{mP} + \text{Ddis}\,\text{mR}}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]\right.$$

$$\left.\sqrt{1 - \frac{n^2\left(\frac{\text{iR}\,n^2}{2\,a^2} + \left(\text{mP} + \frac{\text{mR}}{2}\right)\left(1 + n^2\right)\right)^2 s[t]^2}{(c\,\text{mP} + \text{Ddis}\,\text{mR})^2\left(1 + n^2\right)}}\right) \Big/ \left(\text{iR}\,n^2 + a^2\,(2\,\text{mP} + \text{mR})\left(1 + n^2\right)\right) +$$

$$\left.\left.\left.\frac{n^2\,s[t]\,\text{Sin}\left[\frac{\text{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{\text{iR}\,n^2}{2\,a^2} + \left(\text{mP} + \frac{\text{mR}}{2}\right)\left(1+n^2\right)\right) s[t]}{c\,\text{mP} + \text{Ddis}\,\text{mR}}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]}{1 + n^2}\right)\right)\right) \text{theta}''[t] = \text{Ta} - \text{cs} * \text{theta}'[t]$$

Pendulum equation

$$-\left(-\frac{\left(\text{mP}+\frac{\text{mR}}{2}\right)\left(n^2+n^4\right)s[t]}{\sqrt{c^2-\left(n^2+n^4\right)s[t]^2}}+\right.$$

$$\left((c-2\,\text{Ddis})\,\text{mR}\left(\frac{\text{iR}\,n^2}{2\,a^2}+\left(\text{mP}+\frac{\text{mR}}{2}\right)\left(1+n^2\right)\right)\text{Sin}\left[\frac{\text{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{\text{iR}\,n^2}{2\,a^2}+\left(\text{mP}+\frac{\text{mR}}{2}\right)\left(1+n^2\right)\right)s[t]}{c\,\text{mP}+\text{Ddis}\,\text{mR}}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]\right)\Big/$$

$$\left(2\,(c\,\text{mP}+\text{Ddis}\,\text{mR})\sqrt{1-\frac{n^2\left(\frac{\text{iR}\,n^2}{2\,a^2}+\left(\text{mP}+\frac{\text{mR}}{2}\right)\left(1+n^2\right)\right)^2 s[t]^2}{(c\,\text{mP}+\text{Ddis}\,\text{mR})^2\left(1+n^2\right)}}\right)\right)s'[t]\,\text{theta}'[t]+$$

$$\left(-\frac{\left(\text{mP}+\frac{\text{mR}}{2}\right)\left(n^2+n^4\right)s[t]\,s'[t]}{\sqrt{c^2-\left(n^2+n^4\right)s[t]^2}}+\left((c-2\,\text{Ddis})\,\text{mR}\left(\frac{\text{iR}\,n^2}{2\,a^2}+\left(\text{mP}+\frac{\text{mR}}{2}\right)\left(1+n^2\right)\right)\right.\right.$$

$$\left.\text{Sin}\left[\frac{\text{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{\text{iR}\,n^2}{2\,a^2}+\left(\text{mP}+\frac{\text{mR}}{2}\right)\left(1+n^2\right)\right)s[t]}{c\,\text{mP}+\text{Ddis}\,\text{mR}}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]s'[t]\right)\Big/$$

$$\left.\left(2\,(c\,\text{mP}+\text{Ddis}\,\text{mR})\sqrt{1-\frac{n^2\left(\frac{\text{iR}\,n^2}{2\,a^2}+\left(\text{mP}+\frac{\text{mR}}{2}\right)\left(1+n^2\right)\right)^2 s[t]^2}{(c\,\text{mP}+\text{Ddis}\,\text{mR})^2\left(1+n^2\right)}}\right)\right)\text{theta}'[t]-$$

$$\frac{1}{2}\left(-2\,\text{mP}\,n^2\,s[t]+\frac{1}{2}\,\text{mR}\left(-2\,n^2\,s[t]+2\,(c-2\,\text{Ddis})\left(-1-n^2\right)\right.\right.$$

$$\left(\left(n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)\mathrm{Cos}\left[\frac{\mathrm{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]s[t]\right)\Big/\right.$$

$$\left((c\,mP+Ddis\,mR)\left(1+n^2\right)\sqrt{1-\frac{n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)^2 s[t]^2}{(c\,mP+Ddis\,mR)^2\left(1+n^2\right)}}\right)-$$

$$\left(2\,a^2\,n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)^2\mathrm{Cos}\left[\frac{\mathrm{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]s[t]\right)\Big/$$

$$\left((c\,mP+Ddis\,mR)\left(1+n^2\right)\left(iR\,n^2+a^2\,(2\,mP+mR)\left(1+n^2\right)\right)\right.$$

$$\left.\sqrt{1-\frac{n^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)^2 s[t]^2}{(c\,mP+Ddis\,mR)^2\left(1+n^2\right)}}\right)+\frac{n^2\,\mathrm{Sin}\left[\frac{\mathrm{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]}{1+n^2}-$$

$$\left(2\,a^2\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)\mathrm{Sin}\left[\frac{\mathrm{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\left(\frac{iR\,n^2}{2\,a^2}+\left(mP+\frac{mR}{2}\right)\left(1+n^2\right)\right)s[t]}{c\,mP+Ddis\,mR}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right]\right)\Big/$$

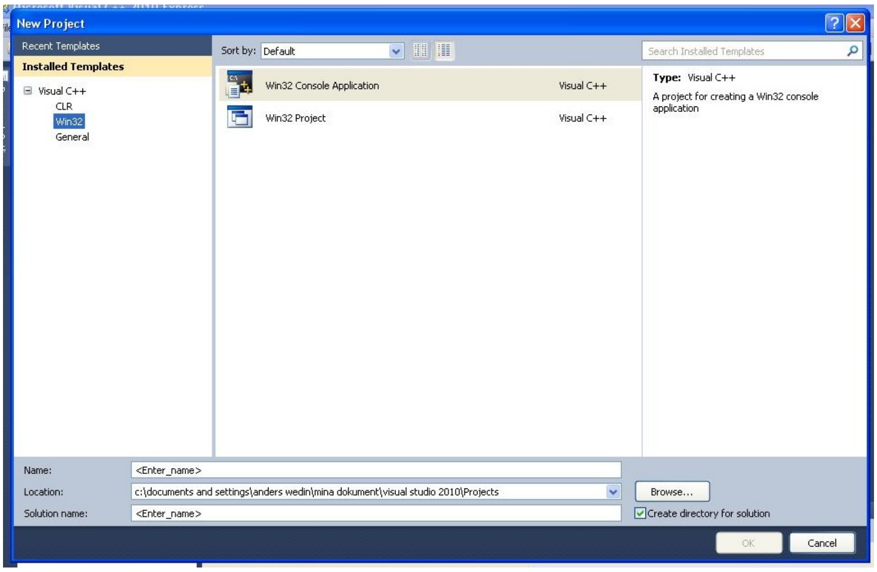
$$\left.\left.\left.\left(\mathrm{iR}\, n^2 + a^2\,(2\,\mathrm{mP} + \mathrm{mR})\,\left(1 + n^2\right)\right)\right)\right)\right)\ \mathrm{theta}'[t]^2 + \frac{\left(\mathrm{iR} + a^2\,(2\,\mathrm{mP} + \mathrm{mR})\right)\, s''[t]}{2\,a^2} +$$

$$\left(-\frac{\mathrm{iR}}{a^2} - \frac{1}{2}\,(c - 2\,\mathrm{Ddis})\,\mathrm{mR}\,\mathrm{Cos}\left[\frac{\mathrm{ArcSin}\left[\frac{\sqrt{\frac{n^2}{1+n^2}}\,\left(\frac{\mathrm{iR}\,n^2}{2\,a^2} + \left(\mathrm{mP} + \frac{\mathrm{mR}}{2}\right)\,\left(1+n^2\right)\right)\, s[t]}{c\,\mathrm{mP} + \mathrm{Ddis}\,\mathrm{mR}}\right]}{\sqrt{\frac{n^2}{1+n^2}}}\right] + \right.$$

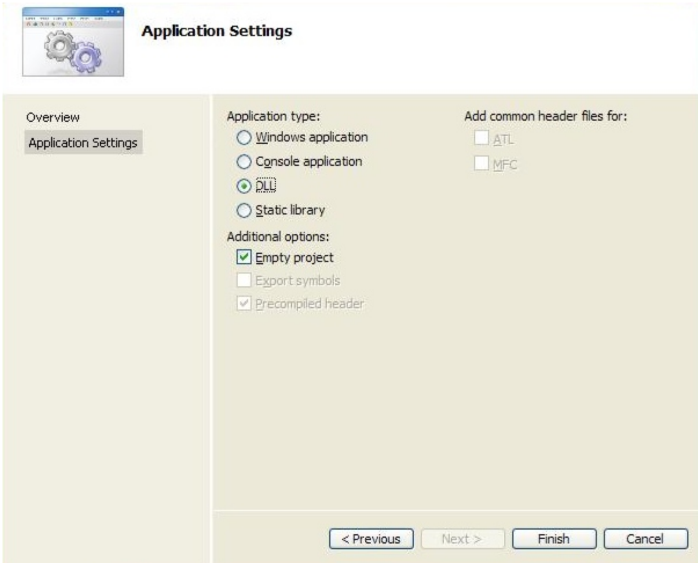
$$\left.\left(\mathrm{mP} + \frac{\mathrm{mR}}{2}\right)\,\sqrt{c^2 - \left(n^2 + n^4\right)\, s[t]^2}\right)\ \mathrm{theta}''[t] = -\mathrm{ca} * s'[t]$$

D Visual Studio Instructions

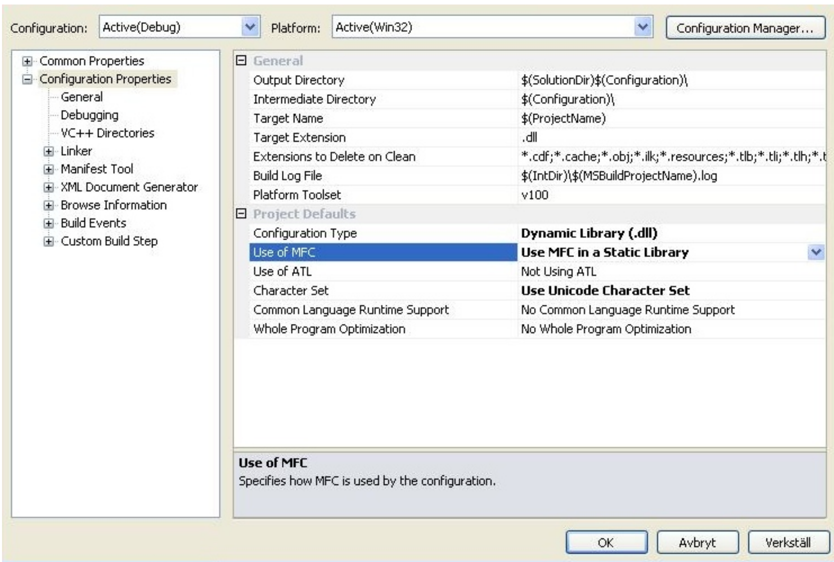
1. Choose **File – New – Project...** from the menu bar
2. In the *Field templates*, pick *Win32*
3. Pick *Win32 Console Application* and a name for the project. Click *Ok*. See Figure D.1 (a) for steps 1 – 3
4. In the wizard: Click *Next*. Pick Application type: *DLL*. Additional options: *Empty project*. Click *Finish*. See Figure D.1 (b)
5. To add C/C++ file: Right-click on **Source files – Add – Existing item...**
6. Settings for dll: Choose **Project – Properties** from the menu bar. Pick *Configuration properties – Use of MFC – Use MFC in a static library*. See Figure D.1 (c)



(a)



(b)



(c)

Figure D.1: *Visual studio instructions and settings.* (a) *Steps 1 – 3.* (b) *Step 4.* (c) *Step 6.*

E CPVA Parameters in Simdrive Simulations

The parameters used in simulations of the Sine-signal torque model and Advanced model are provided in Table E.1 – E.6.

Table E.1: *Parameters for Simple Type CPVA.*

Parameter	**Unit**	**Assigned value**	
		Sine-signal	Advanced
J	$kg \cdot m^2$	0.0375	0.02
c_s	$Nm \cdot s/rad$	0	0
m_p	kg	2	0.25
I_{pend}	$kg \cdot m^2$	0.004	0.02
R	m	0.1	0.2
r	m	0.025	0.049
$NumberPedMasses$	-	2	4
c_a	$Nm \cdot s/rad$	0	0
$C1$	Nm/rad	$14 \cdot 10^3$	$1 \cdot 10^6$
$B1$	$Nm \cdot s/rad$	0	0
$C2$	Nm/rad	$1 \cdot 10^6$	$1 \cdot 10^9$
$B2$	$Nm \cdot s/rad$	0	0

Table E.2: *Parameters for Bifilar Type CPVA.*

Parameter	**Unit**	**Assigned value**	
		Sine-signal	Advanced
J	$kg \cdot m^2$	0.0375	0.02
c_s	$Nm \cdot s/rad$	0	0
m_p	kg	2	0.25
I_{pend}	$kg \cdot m^2$	0.004	0.02
R	m	0.1	0.2
r	m	0.025	0.049
$NumberPedMasses$	-	2	4
c_a	$Nm \cdot s/rad$	0	0
$C1$	Nm/rad	$14 \cdot 10^3$	$1 \cdot 10^6$
$B1$	$Nm \cdot s/rad$	0	0
$C2$	Nm/rad	$1 \cdot 10^6$	$1 \cdot 10^9$
$B2$	$Nm \cdot s/rad$	0	0

Table E.3: *Parameters for Salomon Type CPVA.*

Parameter	Unit	Assigned value	
		Sine-signal	Advanced
J	$kg \cdot m^2$	0.0375	0.02
c_s	$Nm \cdot s/rad$	0	0
M_0		0	0
l		0	0
m_p	kg	2	0.25
I_{pend}	$kg \cdot m^2$	0.004	0.0004
R	m	0.3	0.3
D	m	0.2258	0.1
d	m	0.128	0.054
$NumberPedMasses$	-	2	4
$C1$	Nm/rad	$14 \cdot 10^3$	$1 \cdot 10^6$
$B1$	$Nm \cdot s/rad$	0	0
$C2$	Nm/rad	$1 \cdot 10^6$	$1 \cdot 10^9$
$B2$	$Nm \cdot s/rad$	0	0

Table E.4: *Parameters for Cycloid Path Type CPVA.*

Parameter	Unit	Assigned value	
		Sine-signal	Advanced
J	$kg \cdot m^2$	0.0375	0.02
c_s	$Nm \cdot s/rad$	0	0
m_p	kg	2	0.25
I_{pend}	$kg \cdot m^2$	0.004	0.0001
R	m	0.1	0.4
r	m	0.025	0.098
$NumberPedMasses$	-	2	4
c_a	$Nm \cdot s/rad$	0	0
$C1$	Nm/rad	$14 \cdot 10^3$	$1 \cdot 10^6$
$B1$	$Nm \cdot s/rad$	0	0
$C2$	Nm/rad	$1 \cdot 10^6$	$1 \cdot 10^9$
$B2$	$Nm \cdot s/rad$	0	0

Table E.5: *Parameters for Epiycloid (Cardoid) Path Type CPVA.*

Parameter	Unit	Assigned value	
		Sine-signal	Advanced
J	$kg \cdot m^2$	0.0375	0.02
c_s	$Nm \cdot s/rad$	0	0
m_p	kg	2	0.25
I_{pend}	$kg \cdot m^2$	0.004	0.0001
R	m	0.1	0.2
r	m	0.025	0.049
$NumberPedMasses$	-	2	4
c_a	$Nm \cdot s/rad$	0	0
$C1$	Nm/rad	$14 \cdot 10^3$	$1 \cdot 10^6$
$B1$	$Nm \cdot s/rad$	0	0
$C2$	Nm/rad	$1 \cdot 10^6$	$1 \cdot 10^9$
$B2$	$Nm \cdot s/rad$	0	0

Table E.6: *Parameters for Bifilar Type with Rollers and General-Path Representation CPVA.*

Parameter	Unit	Assigned value	
		Sine-signal	Advanced
J	$kg \cdot m^2$	0.0375	0.02
c_s	$Nm \cdot s/rad$	0	0
m_p	kg	2	1
I_{pend}	$kg \cdot m^2$	0.004	0.01
i_R	$kg \cdot m^2$	0	0
a	m	0.00635	0.01
m_R	kg	0	0
H_0	m	0.04	0.1
D_{dis}	m	0.06535	0.25
c	m	0.0573355	0.2
n	$-$	2	2.02
c_a	$Nm \cdot s/rad$	0	0
$C1$	Nm/rad	$14 \cdot 10^3$	$1 \cdot 10^6$
$B1$	$Nm \cdot s/rad$	0	0
$C2$	Nm/rad	$1 \cdot 10^6$	$1 \cdot 10^9$
$B2$	$Nm \cdot s/rad$	0	0