Robust Design of a Valve Train Cam Phasing Controller using Virtual Prototyping Techniques

Dale Witt
Createch Inc.

Bryan Kelly
Synopsys Inc.

Reprinted From: Simulation & Modeling Mechatronics
(SP-2111)
Robust Design of a Valve Train Cam Phasing Controller using Virtual Prototyping Techniques

Dale Witt
Createch Inc.

Bryan Kelly
Synopsys Inc.

ABSTRACT
Cam phasing, or Variable Valve Timing (VVT), is an electro hydraulic and mechanical camshaft control concept managed by the vehicle's microcontroller engine management system. Development and implementation of cam phasing mechanisms is pursued by the automotive industry today because it gives measurable increase in performance, and reduction in undesired engine emissions. This paper illustrates the usage of virtual prototyping techniques to efficiently investigate cam phasing architecture control algorithm implementation to permit more robust cam phasing design. The control algorithm implementation resides in Simulink, and the virtual prototype of a complete hydraulic vane cam phaser system resides in a selected analog mixed technology simulator. Co-simulation enables the two different simulation engines to communicate, hence dynamic controller development can commence against virtual hardware. Cam phaser response is heavily dependent on engine oil temperature and pressure. As a specific conceptual example, a simple initial controller is developed that does not consider oil temperature and pressure fluctuations. An enhanced more robust controller is then implemented to account for these system variances. The ability to develop the entire control algorithm in a tool suited for that purpose, and then to test the controller with a high-fidelity system plant modeled in another simulation environment is illustrated using robust engineering techniques.

INTRODUCTION
The active control of the relative angle between the cam and crank, or cam phase control is a standard feature on many new automotive engines. The addition of a rotary vane hydraulic actuator between cam drive sprocket or pulley and the cam itself, combined with a proportional 4-way flow control valve provides the ability to continuously vary the cam phase within a practical zone of 20 to 40 cam degrees. This control zone of variable cam phase angle can be either positive, or advancing from a zero-base position, or negative to retard the cam. The control of the cam phase is accomplished using position feedbacks that compare the cam and crank angular positions, as well as a controller and associated hardware interface. The cam phasing hydraulic supply is generally provided by the engine lubrication system. The solenoid drive utilizes a Pulse-Width-Modulated (PWM) output that is common in many automotive control systems.

The basic system schematic used for the cam phasing control system simulation is shown below in Figure 1. Each portion of this system will be discussed in the next section.

Figure 1: Complete Cam Phasing Control System and Plant
The local control for a typical cam phasing system is either a PID or PI controller that can be tuned to optimize many different performance characteristics. The local controller receives a command signal to direct the cam phase angle to change to, or hold a desired position. At the highest level the block diagram shown below in Figure 2 represents the basic closed-loop control system. In this case the controller is a PID, and the Plant is the cam phaser system.
For the purpose of general system analysis and/or verification by test on actual hardware, a set of consistent command sequences are used as command inputs. An example of a command sequence is shown in Figure 3 below, where the desired cam phase angle is changed between zero and two intermediate step levels at 10 degrees and 20 degrees. The cycle sequence includes both step and ramp functions for evaluating control system behavior.

A comprehensive model of the complete cam phasing system includes all of the critical plant component models in their respective technology domains such as electronic and electrical, mechanical, hydraulic and electromagnetic. In addition, an accurate representation of the controller is necessary to ensure that analysis results are not skewed by having ideal, or close-to-ideal models for the critical system elements. Performing analysis by simulation of well-established test sequences on such a comprehensive system model yields great benefits in terms of eliminating many design and calibration sequences normally performed on actual dynamometer or vehicle test hardware. Robust engineering techniques can be easily applied in a simulation environment, where evaluating sensitivity to application temperature and engine parameters, as well as component dimensions and tolerances is very fast and inexpensive when compared to actual hardware test.

SABER was selected as the system simulator for this application. SABER is an advanced robust multi-technology domain simulation package, which includes a mature hardware description language (HDL). Its large available application specific libraries (ASLs) permits construction of multi-discipline systems, such as the cam phaser presented here. The cam phaser system implemented in SABER consisted of a single cam, cam driven pump often employed in a V6 push-rod type engine. This system embraces physically realistic models of the crank, timing chain, cam and cam phaser, solenoids and sensors, and oil pump assembly.

SIMULINK was selected to model the PID controller as it is a popular environment used for the construction and simulation of control block diagrams. The two simulation engines communicate seamlessly via a SaberSimulink Cosimulation interface offered by Synopsys.

THE PLANT

The basic system plant model includes many individual component models that together have great influence on system behavior and interactions with the controller. The components can be grouped into logical divisions along various domains such as mechanical, electro-hydraulic, electro-magnetic and analog and digital electronics. In addition, there are many cross-domain interactions that must be considered for accurate system simulation. Each of these critical domain-based plant models is discussed briefly here.

MECHANICAL PLANT MODELS

The loading applied to a cam phaser actuator is a combination of angular oscillations that originate in the engine crankshaft and torsional loading from the cam connection due to valve train static and dynamic characteristics. The specific plant modeled here is a common push rod-type engine that has the oil pump driven off the cam, which provides additional torsional loading in the phaser actuator.

The critical angular oscillations that originate in the crankshaft snout are due to the torsional vibrations resulting from firing cylinders. These oscillations vary in magnitude and frequency based on engine speed and the number of engine cylinders firing, and appears as a small AC signal on top of the basic crankshaft speed. The torsional oscillations are transmitted from the crank snout through to the phaser actuator via the timing chain or belt. The timing chain provides additional dynamic effects that
are reflected in the measured speed at the cam sprocket, or phaser actuator sprocket connection. This can be seen for an engine start-up sequence where the speed levels off at 1000 RPM as shown in Figure 4 below.

Figure 4: Crank Snout and Cam Sprocket Speed with Torsional Oscillations

The output or loaded end of the phaser actuator is connected to the camshaft, which is modeled to account for static and dynamic loads due to valve train characteristics. This cam loading is sensitive to the angular position of the cam due to the cyclic intake and exhaust valve actuation by the cam lobes. In addition, this torque-angle characteristic changes with engine speed and bearing friction due to viscous drag in the bearings. The resulting torque-versus-time plot for the engine start sequence referred to earlier is shown in Figure 5.

Figure 5: Cam Torque Load

In addition to the specific cam dynamic loading there is loading from the engine oil pump that is driven by the cam in this engine. This is illustrated in the schematic showing the mechanical cam and pump connections to the phaser in Figure 6 below.

Figure 6: Cam and Pump Connection to Phaser

ELECTRO-HYDRAULIC PLANT MODELS

The hydraulic portion of the system plant includes a realistic model for the engine lubrication system, the phaser proportional flow control valve and the phaser vane actuator.

The engine lubrication system plant model includes a fixed-displacement pump that is characterized with realistic flow delivery versus pressure performance that is a function of pump speed and fluid temperature. The network of flow passages between the pump outlet and the feed point for the phaser control valve is modeled using Table Look-Up (TLU) model that includes comprehensive flow-versus-pressure differential
characteristics as a function of engine speed and fluid temperature. The TLU data can be generated using either a flow network modeled separately or from actual test data.

The proportional flow control valve model includes a realistic 4-way overlapped spool valve, electromagnet with non-linear force and inductance characteristics, as well as mechanical elements such as a bias spring and stroke-limiting hard stops.

The phaser vane actuator model includes the rotary hydraulic actuator with fluid volume compressibility, cross-port and external leak paths, and a mechanical locking pin to restrain the actuator in the home or zero phase position when pressure is low.

**ELECTRO-MAGNETIC SENSOR MODELS**

There are two speed sensors modeled in the plant to monitor crank and cam speed. These sensors are characterized to simulate Hall Effect elements that switch outputs based on rising and/or falling edges of a toothed wheel passing the sensor head. The tooth spacing is relatively fine for the crank sensor and coarse for the cam sensor. The actual tooth edge locations are known for both the crank and cam, resulting in periodic pulse updates that correspond to angular positions. The raw signal outputs for the crank and cam sensors are shown below in Figure 7. Note the higher frequency of signal updates from the crank sensor as compared to the cam sensor.

**Figure 7: Crank and Cam Speed Sensor Raw Output**

Included in the sensor outputs is the digitized speed based on the edge-to-edge timing and the known locations of each tooth. The digitized speed signals are shown for the crank and cam in Figure 8.

**Figure 8: Crank Cam Speed Sensor Speed Output**

Updates to the modulated angular displacement on a 0 to 360 degree scale for both the crank and cam are also included as sensor module outputs. This signal for both the crank and cam are shown in Figure 9 below. Note the higher resolution for the crank angle signal.

**Figure 9: Crank Cam Speed Sensor Angle Output**

**HARDWARE INTERFACE AND PROCESSING PLANT MODELS**

The hardware interface and processing modules include a PWM solenoid driver and a separate processing module to compute the cam phase angle from the speed sensor signals.

The PWM solenoid driver model receives controller outputs for commands related to PWM duty cycle and PWM frequency. The high-side coil drive circuit is cycled based on the desired PWM duty cycle at the commanded frequency, with the high driver voltage based on the battery power available. The solenoid driver output with a steady 40% duty command from the controller is shown in Figure 10 below.
The processing module reads the pulsed angle outputs from both the crank and cam speed sensors, and calculates the relative cam phase angle. This cam phase angle is output for both the actual phase angle in terms of cam degrees and crank degrees. The cam phase signal for the baseline system for in both cam and crank degrees is shown in Figure 11 below.

The baseline system model was tested to determine performance at engine speeds between 1000 RPM and 6000 RPM, and the fluid temperature ranging from 60°C to 140°C. The results indicate that control authority degrades due to component leakage and temperature effects. The degradation occurs as the fluid temperature reaches or exceeds 120°C, and when the engine speed decreases below 1500 RPM when the fluid temperature is between 80°C and 120°C. These limits, therefore, will be hard coded into the new controller model in order to limit operation to the speed and temperature ranges where control authority is assured.

**THE CONTROLLER**

The PID controllers remain the most widely used controller in industry today, primarily because they are simple, and usually easy to tune due to the reduced number of parameters that required consideration. There are several different classifications of PID algorithms which may have additional variations of theme dependent on the design engineer, the nature of the plant to be controlled, and the desired performance. For general illustrative purposes a parallel-type PID configuration was selected. Because the localized controller for the cam phaser is likely to be implemented as a microcontroller, a digital PID was adopted. The structure of a digital PID controller [3] acting upon an error signal as shown in Figure 2 may be represented by the following, where $T$ is the discrete sample time for the controller.
To reduce distortion due to time aliasing, the sample time should be less than the shortest time constant in the system. The basic idea of a PID controller is that it first reads the system state via a measurement from a cam phaser position sensor. It then subtracts the measurement from a desired reference, cam phase command signal, to generate an error signal. The error will then be managed in three ways, to handle the present, through the proportional term, to recover from the past, using the integral term, and to anticipate the future, the derivative term can be leveraged.

COSIMULATION

Simulink was selected as the environment to implement the above PID controller, as it is often a platform used for algorithm development. The Simulink controller is shown in Figure 14, and the SaberSimulink cosimulation block can also be observed therein.

The process to cosimulate Saber with Simulink is an easy three step process, as explained below and illustrated in Figure 15.

1. The controller model is first assembled in Simulink. A cosimulation interface block, named SaberCosim.mdl, must be inserted into the Simulink diagram. This block has an input, an output, and two properties defining the width (number of signals) at the input and output. The control signals are then routed through muxes and demuxes to these inputs and outputs.

2. Launch Synopsys’ SaberSimulink Cosim Tool from Saber. This is a graphical model/symbol generation tool. The Simulink controller model is parsed and the adjacent cosimulation Saber model and symbol is automatically generated. Select the PlacePart button to insert the symbol into the active Saber schematic.

3. Connect sensor and control signals that are to be routed to Simulink onto the generated Saber cosimulation interface symbol (model).

Running the cosimulation is controlled entirely from the Saber interface, either from the Saber command line or from the Saber interface within the schematic editor. Saber simulation commands can be used according to user preference. There are no changes to Saber simulation commands or to the Saber interface to support the SaberSimulink Cosimulation.

\[
\frac{u(z)}{e(z)} = \left( K_p + \frac{K_i}{T} \frac{z+1}{z-1} + K_d \frac{z-1}{Tz} \right)
\]
TUNING PID PARAMETERS

There is a wealth of published and commercially available methods that address the sometimes-challenging task of closed-loop controller parameter tuning. Ideally the best way to find the needed PID parameters is from a mathematical model of the plant. However plants are often complex and typically not easily defined with a detailed mathematical description. Consequently an online PID tuning strategy is preferred, if not the only practical option. The most employed online PID design technique used in industry is the Ziegler-Nichols method [4]. The first step in this method is to set the integral and derivation gains to zero. The proportional gain is then increased until a sustained but stable oscillatory behavior is obtained on the output. This maximum gain, labeled the critical gain, $K_c$ and critical oscillation $P_c$, are recorded and the P, I, and D values are adjusted accordingly using Table 1-1.

Table 1-1: Ziegler-Nichols Parameter Tuning

<table>
<thead>
<tr>
<th>Controller</th>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$K_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>$0.5 \times K_c$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PD</td>
<td>$0.65 \times K_c$</td>
<td>$0.12 \times P_c$</td>
<td></td>
</tr>
<tr>
<td>PI</td>
<td>$0.45 \times K_c$</td>
<td>$0.85 \times K_c$</td>
<td></td>
</tr>
<tr>
<td>PID</td>
<td>$0.65 \times K_c$</td>
<td>$0.5 \times K_c$</td>
<td>$0.12 \times P_c$</td>
</tr>
</tbody>
</table>

Further manual tuning of the PID parameters is usually required to optimize the performance of the controller. A realistic virtual prototype of the physical plant hardware in Saber not only permits PID parameter tuning to commence efficiently, but different control algorithms can be tested and verified.

As an example of the tuning process, the below illustrates the performance with a PI controller. Here the derivative gain (D) is zero, but the proportional (P) and integral (I) gains have been adjusted to give good response for the cam phaser system when the engine is operating at medium speed, 1800 RPM, and the engine oil is 80 degree centigrade.

Some overshoot is observed, and while reduction of the proportional gain can reduce this, it also results in noticeable steady state error. This implies that the proportional (P) gain has been optimized to its operational level with its integral (I) term companion. So what can be done to reduce the overshoot? Recall that the derivative (D) term helps the controller anticipate the future, as gives an addition from the rate of change in the error to the system control input. Using the Ziegler-Nichols method and a little additional manual adjustment resulted in a 15% improvement evidenced in Figure 17.
The D term essentially behaves as a high pass filter on the error signal. Consequently, though D-control can be beneficial, it should be implemented judiciously as it easily introduces instability in a system and makes it more sensitive to noise.

CONTROLLER IMPROVEMENTS

Cam phaser hardware is a mixed-technology system, as it embraces a myriad of interacting electrical, mechanical, hydraulic, and temperature dependencies. For example, the response of the cam phaser is heavily dependent on oil pressure, which is contingent upon oil temperature and engine speed. Moreover any mechanical electro hydraulic system has physical limitations and experiences significantly reduced performance when pushed up near or beyond boundary regions of operation. Consequently a controller that possesses sensitivity to these system variables, and has other decision making algorithms, could give more robust performance for the cam phaser system over wider regions of operation.

As case in point, Figure 18 illustrates the cam phaser performance operating at 80 °C, 100 °C and 115 °C for the controller shown in Figure 14 above. Performance shows evidence of degradation as temperatures continue to rise. Further investigation showed that the virtual physical prototype captured in simulation operated very poorly when the temperature was above 120 °C, and the engine crank speed below 1500 RPM. This is similar to “real” physical cam phaser systems; an artifact of physical constraints, which may not be easily modified or may be cost effective to address. Regardless, the situation gives an interesting “what if” exercise to examine what other functionality could be implemented in the controller to help address real-world realities.

A cam phaser controller that monitors both engine oil temperature and speed can incorporate set-points or use temperature and speed profiles represented in a look-up table to provide intelligence to the controller. For example the controller can generate a fail-safe shutdown signal when the known physical system performance boundaries are reached. Figure 19 illustrates controller shutdown operation when the engine speed dips below 1500 RPM, when the oil temperature resides at 80 °C and 115 °C respectively. The shutdown operation commands the phaser to return to home, or zero phase position, which it can do at 80 °C within the window of the given phase command signal, thanks to ample oil pressure. However at the hotter 115 °C the more sluggish phaser while moving to zero phase position, is caught mid-stroke by the cam command signal after the crank speed criteria rises above back the 1500 RPM threshold.

Other controller improvements were investigated. The earlier controller uses a static cam phaser duty cycle bias to command a 40% drive (as opposed to 50%) to account for residual torque in the cam phaser system. Feed forward implementation of temperature sensitivity and system disturbance was investigated to determine if such modifications could result in any slight improvements in the system performance at higher temperatures. This can be observed in Figure 20.
The final form of the controller is shown in Figure 21. Controller improvements can continue to evolve and consider both cam phaser system and controller specific particulars to further help ensure overall system robustness.

The complete design ensemble was implemented in the Saber simulator. This included the complete Saber plant model and the tuned controller running in Simulink using the cosimulation feature. The exercise presented here includes a comprehensive sweep of the engine speed and fluid temperature to examine and quantify how the step response time is affected.

The boundaries for the engine speed and fluid temperature sweeps were held within the operating zones discussed for the controller logic above. That is, the engine speed minimum level was set to 1500 RPM while the fluid temperature was kept at or below 120°C for the tested command sequence. The resulting measurements for the 20° to 10° step occurring at time=2.0 seconds is shown below in Figure 22.

The results indicate large step response sensitivity to fluid temperature at levels higher than 100°C. The sensitivity to engine speed is less-apparent and varies in part due to cam load torque variations.

This high sensitivity to oil temperatures above 100°C provides a good target for additional investigation. The normal range of operating temperature can approach the 120°C to 140°C range, and most certainly will exceed 100°C. There are, therefore, tangible benefits to design modifications that improve high temperature performance. Recall that one of the controller modifications involved shutting down the control system when the temperature increased beyond 120°C, which actually limits the operational zone rather than solves the problem. To solve the problem, additional parametric and statistical simulations can be run to understand which of the components, and specifically which dimensions or characteristics are responsible for the high sensitivity to oil temperature. This exercise is left for future work.

CONCLUSION

This paper presents advances in the modeling process for automotive cam phasing systems and components. A comprehensive plant is modeled using Saber, and includes the critical components and sub-systems that directly affect system performance. A controller for the system is initially modeled within the Saber environment, and then replaced by a digital version modeled within Simulink. The Saber plant model is simulated in concert...
with the Simulink controller using the SaberSimulink cosimulation tool.

Several improvements are added to the controller to compensate for realistic inefficiencies in the system plant, and the controller is tuned to optimize response and stability.

In this paper a critical step in robust design was undertaken, that is, determining the outer boundaries for satisfactory operation. The improved controller’s oil temperature and crank speed monitor and shutdown features address the boundary conditions. This provides illustration of an initial step, and shows the convenience and realizable benefits of virtual prototyping techniques to address hardware issues.

The final system model was exercised to evaluate the sensitivity of a basic step response time to both engine speed and fluid temperature. A nested simulation analysis was performed where the system was subjected to temperature range over 40 degrees centigrade over a wide but nominal operating speed range.

Future work includes additional parametric and statistical simulations to evaluate system characteristic sensitivities to critical dimensions in the individual plant components, as well as adding additional logic and sensor inputs to the controller.

REFERENCES

1. An application of nonlinear PID control to a class of truck ABS problems, Jiang, F.; Gao, Z.; Decision and Control, Proceedings of 40th IEEE Conference 4-7 Dec. 2001 Page(s):516 - 521 vol.1

CONTACT

Dale Witt is the founder of Createch, Inc. which is located in Brush Prairie, WA. Createch, Inc. provides technical services related to model development and laboratory test for model characterization and validation of hydraulic, mechanical and electromagnetic components and systems. Dale can be reached at createch@pacifier.com

Bryan Kelly holds BS and M.S. degrees in physics from Iowa State University (1993). He joined Synopsys in 1994, and since that time, Bryan has worked in software modeling and tool development spanning many different technology disciplines. Bryan's current work assignment is in Synopsys R&D where he has responsibilities for modeling development and analysis of motor/generators, motor drives and control systems. Bryan can be reached at bryan.kelly@synopsys.com.

ACRONYMS

VVT: Variable Valve Timing
PWM: Pulse Width Modulation
RPM: Revolutions per Minute
°C: Degrees Centigrade