Cycle-to-Cycle TDC Control of an Opposed-Cylinder Free Piston Engine Using H_∞ and μ Synthesis

Yuto Watanabe
yuto.watanabe@gmail.com
Purdue University, ME675 Final Project
Spring 2022

I. Introduction

Conventional internal combustion engines control the motion of a piston mechanically, using a crank and crankshaft mechanism as a means of converting translational power to rotational power. This mechanism fully constrains the trajectory of the piston motion aside from the speed of each cycle. An alternative to this design is a free-piston engine (FPE). Instead of using rigid mechanisms, the piston motion in a FPE is controlled by applying variable forces to the piston by other means such as a linear motors, pneumatics, or hydraulics. [1] Such a design is advantageous for reducing piston friction losses and achieving finer combustion control through variable compression ratios, which allows for more efficient operation with lower emissions.

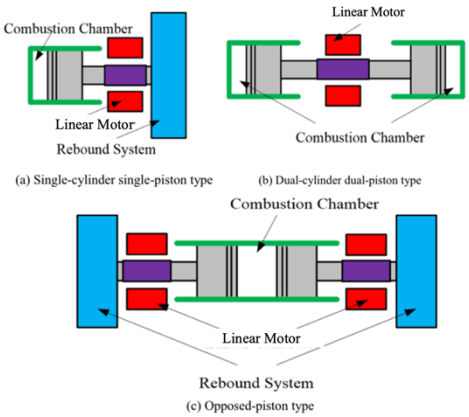


Figure 1. Schematic of typical FPE configurations. Diagram adapted from [2].

There are several configurations of FPE that are actively being researched, varying in the number and arrangement of combustion chambers and rebound systems, which changes the piston control strategy. While each of the configurations have different challenges, this paper will focus on the opposed-piston configuration which has two independent pistons actuated and applying forces separately via two motors. The MIMO nature of the system containing two pistons with coupled dynamics presents an interesting use-case for applying and analyzing linear, multivariable control techniques such as H_∞ and μ-synthesis. A comparison to using other methods such as PID control and LQR control is also discussed later in the paper.

II. System Description

In an opposed-piston type FPE, each engine cycle consists of the following steps of reciprocating piston motion:

1. Both pistons move towards each-other in the center, compressing a gas.
2. Near the top dead center (TDC) position the mixture is ignited
3. Pistons retract away from each-other while the mixture is expanded.
4. At bottom dead center (BDC) a rebound system, typically a mechanical or gas spring, forces the pistons to return towards TDC.

In general there are a multitude of control objectives that must be satisfied in each cycle regarding piston position. For instance, the TDC position must be controlled tightly to set the compression ratio. Between TDC and BDC, the piston trajectory may be controlled as it impacts cylinder intake and exhaust timing.[3] The motion of the two opposing pistons must be synchronized to minimize force imbalance resulting in external vibrations. [2] Lastly, the force applied by the motors must consider the energy addition from combustion, to extract an appropriate amount of energy from the system as electrical power.

III. Objective

This paper will focus on developing a discrete, cycle-to-cycle controller for controlling the piston TDC positions and piston synchronization. A nonlinear, continuous-time plant model is used to develop a linearized cycle-to-cycle model for a particular operating point. A controller will then be developed around the linearized operating point. The controller will be evaluated for its reference tracking performance as well its robustness using the linearized plant model. Lastly, the controller performance using the nonlinear model is briefly discussed.

IV. System and Controller Model

A simplified FPE model is used as a basis for developing and evaluating control algorithms. This paper investigates a FPE configuration in which the rebound systems to return the pistons to the central position are gas springs attached to the end of each piston, as shown in Figure 2. The gas in the combustion chamber is modeled as being in a closed volume, and it is assumed that no combustion occurs in the cylinder.

This condition simplifies the model as no heat release from combustion needs to be considered. It closely resembles the conditions during motoring prior to fuel injection, and as such, it is the operating point under consideration. Lastly, the model includes an ideal actuator such as a motor that can apply variable forces to each of the pistons based on commands from a controller.

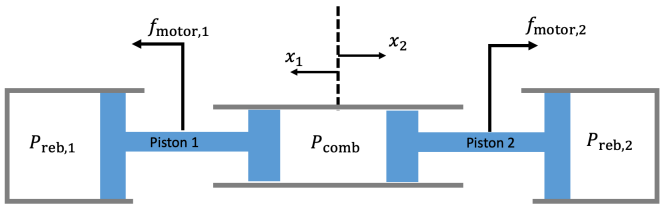


Figure 2. FPE model

Two numerical models were developed in the process of controller design. The first was a continuous-time dynamic model of the pistons that involved nonlinear dynamics. Later, the continuous model is linearized about a fixed cyclic operating point to create a cycle-to-cycle linear model.

A. Continuous Time Model

Each piston experience forces from the rebound chamber gases, the combustion chamber gasses, the forces applied by the motor, and frictional forces. The equation of motion for piston 1 is given by:

$$m\ddot{x_1} = \sum F_1$$

$$m\ddot{x_1} = F_{\text{mot},1} + F_{\text{reb},1} + F_{\text{comb}} + F_{\text{fric},1}$$

In the nominal condition the system is symmetric about the centerline, and as such when the coordinate systems for each of the pistons are also mirrored, the equations of motion for the two pistons are identical aside from the subscripts that indicate the piston.

The gas spring forces for the combustion chamber (F_{comb}) and rebound chamber (F_{reb}) are modeled as in-cylinder pressure multiplied by the cylinder cross-sectional area (A_{cross}):

$$F = PA_{cross}$$

Atmospheric pressure is neglected as its impact is minimal. The in-cylinder gas is modeled to be an ideal gas compressed and expanded through an adiabatic and isentropic process within a closed volume. Thus, the pressure-volume relationship can be given by:

$$P = P_0\left(\frac{V_0}{V}\right)^{\gamma}$$

where P and V are the chamber pressure and volume, while P_0 and V_0 are the initial conditions. γ is a adiabatic gas constant. This ideal model neglects the effect of heat transfer, or mass transfer through leaks in the piston rings, which would be present on a real system. [4]

Frictional forces are crudely modeled to be proportional to piston velocity, given by:

$$F_{\text{fric}} = -b\dot{x}$$

Given the nonlinearity in the gas spring forces, the differential equation governing piston motion has no closed-form solution. Therefore they are solved numerically in MATLAB. Numerical parameters necessary for the simulation, such as geometry, are taken from the design of the FPLA (free piston engine – linear alternator) by Sandia National Laboratories. While the FPLA differs from the system being analyzed, in ways such as the ability to pneumatically actuate the bounce chamber and inability to apply a driving force to the piston, it provides a hardware system design that is plausible in order to demonstrate the controller performance in a realistic manner. Key model parameters are listed in Table 1.

Table 1: Key model parameters adopted from the FPLA system [5]. Additional model parameters are listed in *Table 3* in the Appendix.

Parameter	Value
Combustion chamber trapped volume	1.97 L
Stroke (per piston)	220 mm
Combustion cylinder bore	81.15 mm
Bounce chamber bore	73.46 mm
Piston mass	4.9 kg
Operating cycle frequency	32 Hz

The nonlinear time-invariant system is implemented in the form:

$$\dot{Y} = f(Y, u)$$

$$Y = [x_1, v_1, x_2, v_2]^T$$

$$u = \left[F_{\text{mot},1}, F_{\text{mot},2}\right]^T$$

where Y is the system state, and v_1, v_2 are the respective piston velocities.

B. Control Architecture

A piston motion control strategy for an opposed cylinder FPE is investigated by Zulkifli, et al., which demonstrates that applying a rectangular force profile to a piston is a viable strategy to start piston motion and achieve motoring. The authors show a relatively small force applied by a motor in the direction of piston motion can excite the bouncing mode of the single piston.[4] Others such as Zhang, et al., also propose using a similar rectangular force profile. [6] This paper proposes a method in which the use of a rectangular force profile is extended to an opposed piston FPE, with two pistons rather than one.

The piston motion is said to be in synchronization when the two pistons are both moving towards each other or away from each other. This is the resonance mode that is to be excited when operating an opposed piston FPE, and it can be achieved by alternating the direction of the force applied to the pistons based on this the relative motion of the pistons. When the pistons are moving closer, the forces are directed to drive the pistons closer, and vice versa. The transition in the direction of the forces is given by events in which the average piston velocity crosses 0:

$$0 = \frac{v_1+v_2}{2}$$

A zero crossing from negative to positive is considered a TDC event and the start of the expansion stroke, while a zero crossing from positive to negative is considered a BDC event and the start of the compression stroke. The direction of the

force command for switches simultaneously for both at TDC and BDC, resulting in a rectangular force profile as shown in Figure 3. The magnitude of the rectangular profile is the quantity to be determined by the controller, and it can be changed independently for each piston from cycle-to-cycle, with a cycle beginning at TDC.

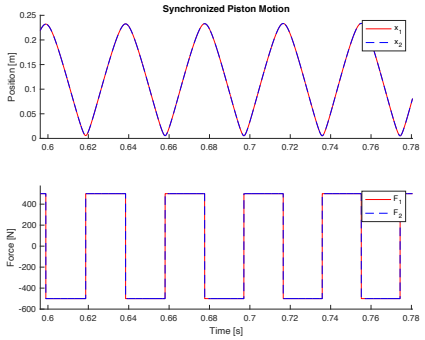


Figure 3. Piston motion in perfect synchronization

If the system is mirrored perfectly then the pistons remain synchronized when an equal of force is applied to both pistons. However, as soon as some asymmetry is introduced the piston synchronization becomes unstable. Shown in Figure 4 is a simulation in which symmetric forces are initially applied to both piston, but at $t = 1$ a small force asymmetry is introduced for a cycle, after which the force profile is made symmetric again. The asymmetry in the piston TDC position oscillates and grows unstably over time.

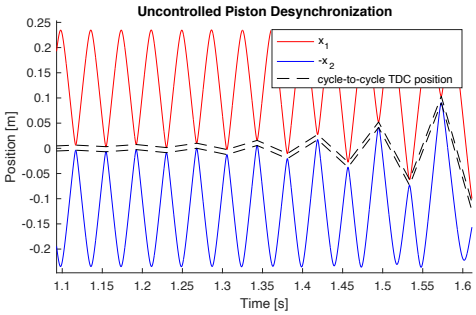


Figure 4. Unstable synchronization behavior. Note the sign of x_2 is inverted to illustrate the oscillations.

C. Linearized Cycle-to-Cycle Model

A piston force control strategy is previously introduced to allow for cycle-to-cycle force control of the pistons. Assuming the control variable is also cyclic in nature, the model, as well as the controller can be discretized with time steps governed by the TDC event frequency. Cycle-to-cycle discretization is common practice in engine controller development.[7] As the intention of this paper is to develop controllers using H_∞ and μ synthesis ,which require linear time-invariant (LTI) models, the nonlinear cycle-to-cycle model is then linearized about an operating point that is a system equilibrium.

In the context of the discretized nonlinear model, the system dynamics is be described by:

$$Y[k+1] = f(Y[k], u[k])$$
$$Y = \left[x_{\mathrm{TDC},1}, x_{\mathrm{TDC},2}, \Delta v_{\mathrm{TDC}}\right]^T$$
$$\Delta v_{\mathrm{TDC}} = \frac{1}{2}\left(v_{TDC,1} - v_{TDC,2}\right)$$

Where Y is the system state at TDC, k indicates the TDC cycle step. In comparison to the continuous time model, the discrete model contains one less element in the state as the discretization adds a constraint to the states. At TDC, the piston velocities must satisfy $v_{\mathrm{TDC},1} + v_{\mathrm{TDC},2} = 0$, hence specifying both would over-constrain the system.

The operating point for linearization was chosen to have a peak pressure of approximately 110 bar to match the motoring conditions achieved during the operation of the FPLA by Sandia National Laboratories. According to the linearization, the dynamics in the vicinity of this operating point is described by:

$$Y[K+1] - Y_o = G(u[k] - u_o)$$
$$u_o = [500\ 500]^T$$
$$Y_o = [0.006\ 0.006\ 0]^T$$

Where G is the linearized system transfer function, and u_o, Y_o are the system inputs and outputs at equilibrium. This translates to both pistons having a TDC position of 6 mm, and the motors applying a 500 N force equally on both pistons. The linearized discrete plant has real poles at (0.77, -1.42, -0.51), indicating that the system is 3rd order and has an unstable pole.

V. H_∞ and μ Synthesis Controller Design

Controller design using H_∞ and μ-synthesis is done by formulating performance requirements and plant dynamics into a LTI system that can be used to guide the optimization process. Once the controllers are synthesized, the closed-loop performance of the system is evaluated on the linearized discrete plant.

A. Reference Tracking Controller Synthesis

When reference tracking, the goal of the controller is to generate a sequence of commands to the system to control the system outputs to converges to the reference, and then hold the reference at steady-state. While conventional controllability of a system guarantees the feasibility of the former, it does not guarantee the latter, as all outputs must be controlled independently across time. This notion of independent output control is introduced by Skogestad as functional controllability, and it limits the number of outputs available for reference tracking to the number of control inputs to the system. [8]

Consequently, not all elements in the system outputs y $(x_{\mathrm{TDC},1}, x_{\mathrm{TDC},2}, \Delta v_{\mathrm{TDC}})$ can be specified for reference tracking as the system only has 2 control inputs. Thus the number of control variables is reduced by combining $x_{\mathrm{tdc},1}, x_{\mathrm{tdc},2}$ in the form of an average TDC position, $\bar{x}_{\mathrm{TDC}}$. For a FPE system, controlling the average TDC position is functionally not very different from independent TDC control as either allows specifying the compression ratio. In addition to average TDC position, TDC velocity differential (Δv_{tdc}) is kept as one of the control variables to allow us to explore controlled desynchronization behavior.

The transformation required for reference tracking above is described by:

$$y' = T_{\mathrm{track}} y$$

$$\begin{bmatrix} \bar{x}_{\text{tdc}} \\ \Delta v_{\text{tdc}} \end{bmatrix} = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{\text{TDC},1} \\ x_{\text{TDC},2} \\ \Delta v_{\text{TDC}} \end{bmatrix}$$

A system H is formulated as shown in Figure 5 to represent the optimization objective in the standard LTI plant configuration expected by the H∞ synthesis algorithm. The reference command r is the input to the system, and the weighted tracking error e_1 is the output, which makes P analogous to a weighted sensitivity function for the closed loop system.

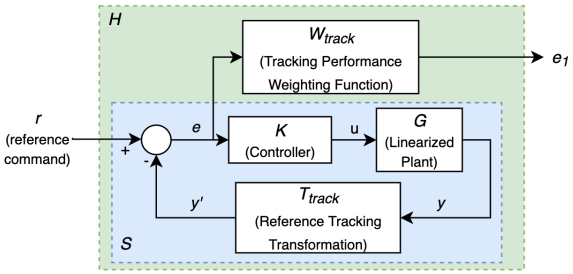


Figure 5. H_∞-Synthesis Plant System Diagram

A weighting function W_{track} is designed in the frequency domain to have the closed-loop system meet the reference tracking performance requirements listed below. Figure 6 shows the design of the resulting weighting function.

1. Steady-state tracking error of $\bar{x}_{\text{tdc}}$ and Δv_{tdc}must be less than 0.01 at steady state for a constant, unit reference command.
2. Tracking error of $\bar{x}_{\text{tdc}}$ and Δv_{tdc}must be less than 0.1 for unit amplitude sinusoidal reference inputs with frequency less than 0.1 rad/sec.

H∞-synthesis is run in MATLAB using the system H as the input, and successfully generated a 7th order controller.

Reference Tracking Results

The closed loop system with the H∞ controller integrated is inspected to verify that the performance requirements are met. The system is analyzed using the discrete, linearized plant model; evaluation using the nonlinear model is done in Section VII. The singular values of system sensitivity function, S, is smaller in magnitude than the maximum error requirements listed above, indicating that all requirements are satisfied.

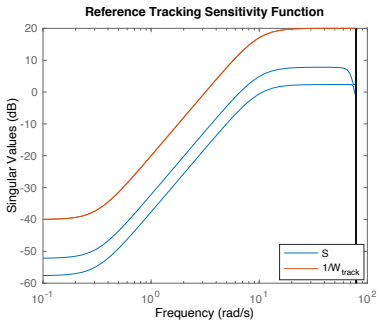


Figure 6. Closed-loop sensitivity function for reference tracking

The step response (Figure 7) of the sensitivity function reveals that the system output converges to the reference time with little steady-state error and a settling time of less than 1.5 seconds for both $\bar{x}_{\text{tdc}}$ and Δv_{tdc}, which are acceptable results.

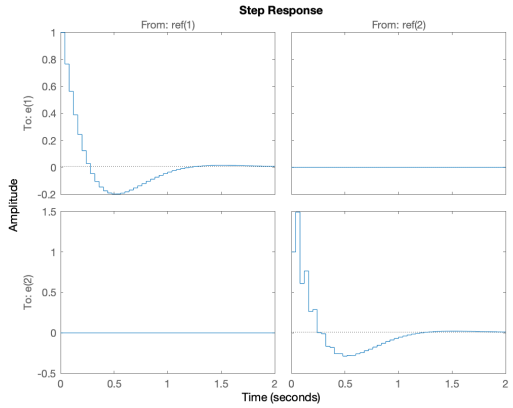


Figure 7. Step response of closed-loop sensitivity function. Reference tracking errors converge to 0 with acceptable overshoot and negligible coupling.

B. Untracked Output Convergence

While the controller settles the system outputs $\bar{x}_{\text{tdc}}$ and Δv_{tdc} to a steady-state value with satisfactory speed, the settling time for the system states is observed to be significantly longer. When a Δv_{tdc} reference step input is applied to the system, the difference in the piston TDC positions (Δx_{TDC}) oscillates between each cycle and does not settle until 12 seconds later, despite average TDC positions ($\bar{x}_{\text{tdc}}$) settling within 1.5 seconds. This effect is shown in Figure 8. Cycle-to-cycle oscillations in Δx_{TDC} could have unintended side-effects on the FPE system performance, hence it is undesirable for these transients to persist for durations much longer than the settling time of reference tracking.

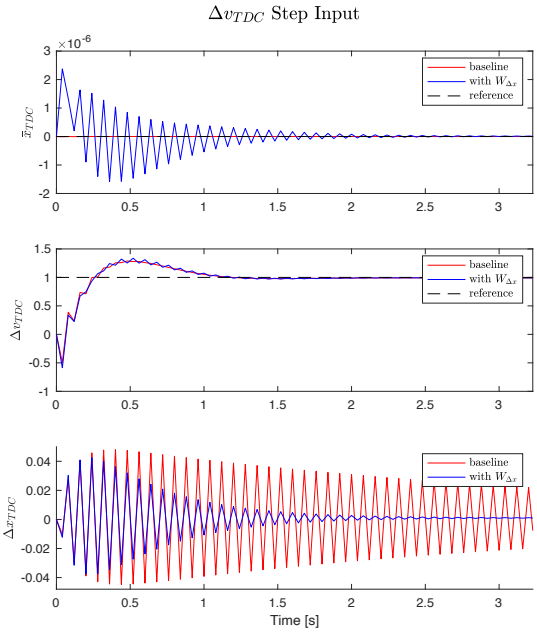


Figure 8. System response to step change in Δv_{TDC} reference. In the baseline case, Δx_{TDC} has a much longer settling time than Δv_{TDC} being commanded.

To mitigate this behavior, an additional output (e_2) is added to the H∞-synthesis plant H, which would penalize high frequency output of Δx_{TDC} during the controller synthesis optimization. A weighting function $W_{\Delta x}$ is added to accompany this output, as well as a transformation matrix to calculate Δx_{TDC}.

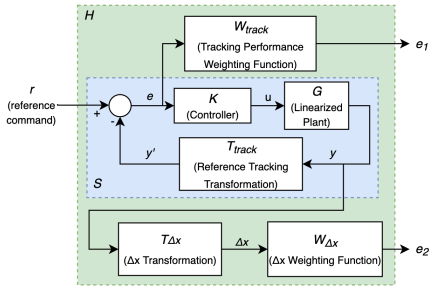


Figure 9. Controller synthesis system with Δx_{TDC} penalization

The transformation to derive Δx_{TDC} is given by:

$$\Delta x_{\mathrm{TDC}} = T_{\Delta x} y$$

$$\Delta x_{\mathrm{TDC}} = [0.5 \quad -0.5 \quad 0] \begin{bmatrix} x_{\mathrm{tdc},1} \\ x_{\mathrm{tdc},2} \\ \Delta v_{\mathrm{tdc}} \end{bmatrix}$$

The weighting function $W_{\Delta x}$ is a constant that is tuned to trade-off between the settling time of Δx_{TDC} and the magnitude of the transients in $\bar{x}_{\mathrm{tdc}}$.

A new controller is synthesized with this new plant formulation. The new controller has an improved Δx_{TDC} settling time, reducing the original 12 seconds to 2 seconds. The performance level achieved by H∞-synthesis, gamma, has increased from 0.188 to 0.247, indicating that this additional weighting does impact the margin in the optimization to meeting performance requirements, but its effect is not detrimental the value is still less than 1.

C. Robust Stability and Performance

One of the downsides of H∞-synthesis is its sensitivity to plant modelling errors; that is, if the model behavior is not sufficiently close to the actual plant behavior, the controller will not function as intended. We introduce μ-synthesis as a technique that can be applied to quantify the plant model uncertainty and develop a controller that can meet performance and stability requirements within those expected uncertainties.

Quantifying plant uncertainties is a challenging task, as intimate knowledge of the plant is necessary to know the unmodeled behavior and possible model parameter ranges. Hardware manufacturing tolerances, operating conditions, and actuator system behavior could factor into this. For this paper, a blanket approach is taken to assume that there is $\pm 25\%$ uncertainty in the magnitude of the frequency response for each of the outputs of the plant model G. The uncertainty is applied to the outputs of the nominal plant model in block U as shown in Figure 10.

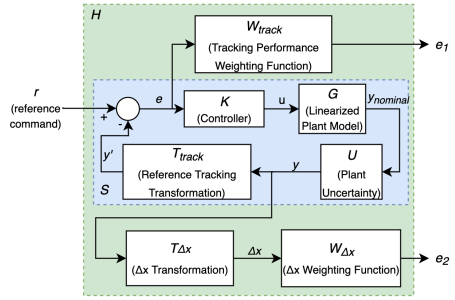


Figure 10. Controller synthesis system with model uncertainty

μ-synthesis is run in MATLAB to develop a new controller while considering these model uncertainties, in addition to the previous performance requirements. Result summary (Table 2) show that the robust performance and stability of the μ-synthesis controller is significantly improved over the H∞-synthesis controller, and in fact it can handle almost twice the magnitude of uncertainty expected. The downsides are the increased order of the controller, which corresponds to an increase in roughly 17 times in computational cost.

Table 2. Comparison of H∞-synthesis and μ ∞-Synthesis controllers

Metric	H∞-syn	μ-syn	LQI
Gamma (Nominal Performance Level)	0.24	0.517	4.9
Robust Performance Margin	1.07	1.92	0
Robust Stability Margin	1.35	2.12	0.66
Controller Order	7	29	2

VI. Alternative Controller Formulations

Other controller architectures were explored to assess the advantages and disadvantages of using H∞ and μ-synthesis for this application.

A. PID

An attempt was made to use a combination of several PID controllers for the reference tracking problem. PID controllers are used ubiquitously across industry for their ease of implementation and maintenance, and therefore a control scheme using PID control would be beneficial. An approach was taken to decouple the MIMO system into two SISO control loops: one to tracking $\bar{x}_{\mathrm{tdc}}$, and the other to track Δv_{TDC}, as shown in Figure 11. Using intuition of the system symmetry, it can be thought that controlling $\bar{x}_{\mathrm{tdc}}$ would require increasing or decreasing the force applied to both pistons ($F_{\mathrm{motor,total}}$) by an equal amount, and controlling Δv_{TDC}, would require offsetting the forces ($F_{\mathrm{motor,diff}}$) applied on the pistons to induce asymmetry. The sum of the total and differential forces is used as the force applied to each piston ($F_{\mathrm{motor},1}, F_{\mathrm{motor},2}$).

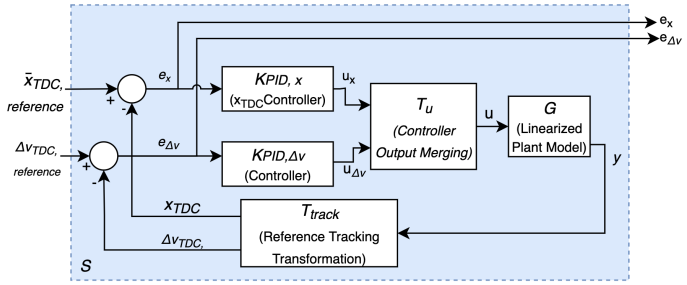


Figure 11. Decoupled system and controller

To explain the rationale for this decoupling more rigorously, the SVD decomposition of the plant is used. Consider the SVD decomposition of the system to be controlled for reference tracking:

$$U\Sigma V^* = T_{\mathrm{track}} G$$

As the left-singular vectors (column vectors of U) and right-singular vectors (column vectors of V) have constant direction

across all frequencies (Figure 15), a matrix transformation can be applied to diagonalize the system transfer function:

$$\Sigma = U_o^*(T_{\text{track}}G)V_o$$

Where U_o and V_o correspond to the fixed directions of U and V. With this transformation, the system input is transformed from $[F_{motor,1}, F_{\text{motor},2}]^T$ to $[F_{\text{motor,total}}, F_{\text{diff}}]^T$ and the PID controllers for $\bar{x}_{\text{tdc}}$ and Δv_{TDC} can be designed independently in a similar fashion as a SISO system. A MIMO controller that utilizes this decoupling to use a series of SISO controllers is an *SVD-controller.* In this case, as the directionality of U, V is independent of frequency, the controller synthesized with this transformation will be optimal, regardless of the decoupling. [9] This is a strong sign that this approach is feasible and will have acceptable performance.

Despite this, a PID controller could not be implemented to fully demonstrate this concept, as stable PID gains could not be found. While the SVD-controller eliminated the multivariable input and output coupling of the plant, the fact remains that the system has a third-order transfer function with a pole in the RHP; something that is not trivial to design a PID controller for. If a PID controller is designed to stabilize the system, it is likely that the reference tracking could be achieved, but questions about the performance, robustness, and ability to control the untracked output Δx_{TDC} still remain.

B. Linear Quadratic Integral Control

Another approach used to solve the reference tracking problem is linear quadratic control; one of the cornerstones of optimal control. Rather than using a conventional linear quadratic regulator which relies purely on gains, a linear quadratic integral (LQI) controller is selected for implementation, as the integrator is helpful in driving down steady-state error and meeting the performance requirements. For a discrete LQI controller, the controller formulation minimizes the cost function given by:

$$J(u) = \sum_{n=0}^{\infty} z^T Q z + u^T R u + 2 z^T N u$$

Where $z = [y; e_{int}]^T$ is the plant state and integrated error, and u is the controller output. Q, R, N are cost weighting parameters that are tuned. [10]

In tuning the parameters, the elements of Q corresponding to e_{int} were weighted heavily to prioritize reference tracking performance. The elements of Q corresponding to y are set comparatively smaller non-zero values as any large transients in y are to be avoided. Otherwise R and N are set to 0. The LQI controller gains were solved for in MATLAB, and the controller is integrated into the system as shown by Figure 12.

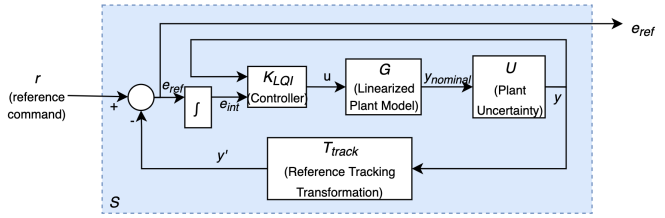
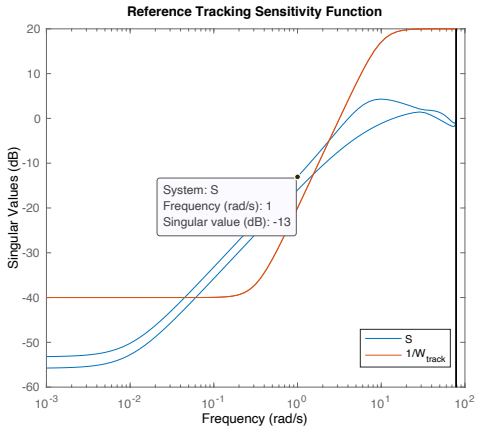


Figure 12. System diagram with LQI Controller

Stable reference tracking is achieved using the LQI controller and the nominal plant model, though the one of the performance requirements are not met. For a unit-amplitude sinusoidal reference tracking input, the amplitude of the steady-state error will be 0.22 (-13dB), which exceeds the requirement of 0.1.



The LQI controller underperforms the H_∞ and μ-synthesis in the area of robustness as well, as seen by its robust stability margin (Table 2). It is not guaranteed to be stable under the assumed model uncertainty, and the range in which it is robustly stable is less than 1/3 in magnitude compared to the controller designed with μ-synthesis.

On the other hand, the merits of the LQI controller lies in its simplicity in developing the controller. A total of 5 parameters were tuned to 2 unique values in the process of developing the LQI controller, and the controller usually resulted in stable outputs, making tuning straightforward. This was much appreciated when compared to H_∞-synthesis, which has many degrees of freedom in constructing the synthesis input plant, and often resulted in no solution being found. Despite the inferior performance of the LQI controller compared to the H_∞ synthesis controller, these qualities could be a significant driver in a decision to implement a LQI controller.

VII. Controller Performance on a Nonlinear Plant Model

Lastly, the μ-synthesis controller is tested in simulation using the nonlinear plant model. The system is exercised to track references away from its operating point to assess if the linear controller allows the controller to respond to a reasonable range of references. Results from a small sample of step inputs show the controller performs similarly with the nonlinear plant as it did with the linear plant, and raises confidence that this controller could be functional on a real system in the vicinity of this operating point.

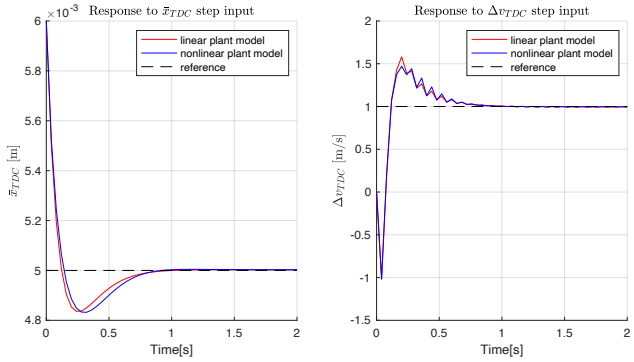


Figure 13. Comparison of step input behavior with linear vs nonlinear plant model

Simulations from a standstill, however, do not achieve favorable results (Figure 14). The controller forces exceed saturation limits, and it appears that there is an integral windup behavior that causes severe overshoot in $\bar{x}_{\mathrm{tdc}}$. More work is necessary to quantify the bounds in which this controller meets performance and stability requirements with the nonlinear model.

VIII. Conclusions

In this paper, several methods were proposed to control the piston motion of an opposed-piston FPE in motoring. As all methods relied on the use of a, linear system model, the nonlinear continuous time models were reformulated into a linearized, discrete system based on TDC events. It was shown that it is feasible to develop controllers with satisfactory performance using the H_∞ and μ-synthesis methods, as well as LQI control methods. While the H_∞ and μ-synthesis methods proved to achieve higher performing controllers with increased robustness, the LQI control methods offered a simpler path to achieve a working controller.

IX. Future Work

Much of this paper is focused around developing and evaluating controllers in the motoring operating condition, which is near-steady state; however that is only one short phase in the overall operation of a FPE, hence other operating points should be considered. While using the same μ-synthesis for the transition from stationary to motoring resulted in unacceptable performance, anti-windup techniques or gain scheduling could extend the range of controllable operating points. Other topics to be considered are controller order reduction for reduced computational time, and more detailed parametrizing plant uncertainty for improved robustness characterization.

X. References

[1] Y. Zhu *et al.*, "The control of an opposed hydraulic free piston engine," *Applied Energy*, vol. 126, pp. 213–220, 2014, doi: https://doi.org/10.1016/j.apenergy.2014.04.007.

[2] C. Guo, Z. Zuo, H. Feng, and T. Roskilly, "Advances in free-piston internal combustion engines: A comprehensive review," *Applied Thermal Engineering*, vol. 189, p. 116679, 2021, doi: https://doi.org/10.1016/j.applthermaleng.2021.116679.

[3] R. Mikalsen and A. P. Roskilly, "The control of a free-piston engine generator. Part 1: Fundamental analyses," *Applied Energy*, vol. 87, no. 4, pp. 1273–1280, 2010, doi: https://doi.org/10.1016/j.apenergy.2009.06.036.

[4] S. A. Zulkifli, M. N. Karsiti, and A. R. A. Aziz, "Starting of a free-piston linear engine-generator by mechanical resonance and rectangular current commutation," in *2008 IEEE Vehicle Power and Propulsion Conference*, 2008, pp. 1–7. doi: 10.1109/VPPC.2008.4677748.

[5] M. Leick and R. Moses, "Experimental Evaluation of the Free Piston Engine - Linear Alternator (FPLA)," Albuquerque, NM, and Livermore, CA (United States), Mar. 2015. doi: 10.2172/1177159.

[6] R. Yang, X. Gong, Y. Hu, and H. Chen, "Motion control of free piston engine generator based on LQR," in *2015 34th Chinese Control Conference (CCC)*, 2015, pp. 8091–8096. doi: 10.1109/ChiCC.2015.7260927.

[7] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2005.

[8] M. Hovd, R. Braatz, and S. Skogestad, "Optimalty of SVD controllers," May 1996.

[9] MATLAB, "Linear-Quadratic-Integral control." https://www.mathworks.com/help/control/ref/ss.lqi.html (accessed Apr. 30, 2022).

XI. Appendix

Table 3. Test data from the Sandia National Labs FPLA is used to generate model parameters for the continuous-time model, shown below. Values are approximate, as they are interpreted from plots in the report. For bounce chamber pressures, the mean value in the vicinity of TDC and BDC were used, as the pressures in these regions change drastically due to the FPLA's pneumatic actuation of the bouncer chamber, which is not modeled in the continuous-time model.

Parameter	**Value**
Bounce chamber BDC volume	1.5 L
Bounce chamber TDC volume	0.11 L
Combustion chamber BDC volume	2 L
Combustion chamber TDC volume	0.05 L
Bounce chamber TDC pressure	200 kPa
Bounce chamber BDC pressure	3000 kPa
Combustion chamber BDC pressure	100 kPa
Combustion chamber TDC pressure	15,000 kPa
Combustion chamber adiabatic gas constant	1.35
Bounce chamber adiabatic gas constant	1.22

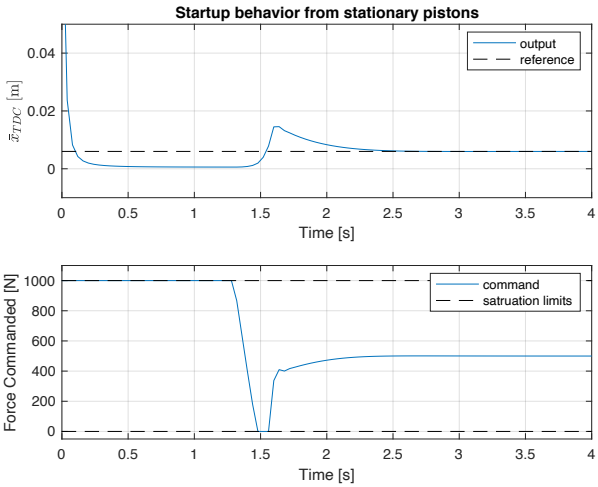


Figure 14. Startup behavior from stationary pistons using the μ-synthesis controller. Note the extreme overshoot of the TDC reference, followed by a undershoot. In a real system this is likely to cause a peak pressure that could damage hardware. The controller is not suitable for use from a stationary start.

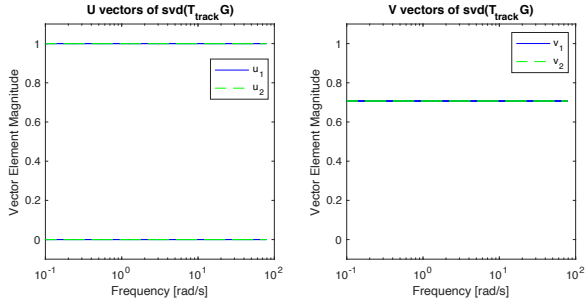


Figure 15. SVD decomposition of $T_{track}G$. Note the constant direction vectors of U and V across all frequencies.