2014

Energy Optimization Of A Free-Piston Enhanced Hybrid Vehicle

Kenneth Jarod Jones
North Carolina Agricultural and Technical State University

Energy Optimization of a Free-Piston Enhanced Hybrid Vehicle

Kenneth Jarod Jones

North Carolina A&T State University

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Department: Electrical and Computer Engineering

Major: Electrical Engineering

Major Professor: Dr. Gary L. Lebby

Greensboro, North Carolina

2014

School of Graduate Studies
North Carolina Agricultural and Technical State University

This is to certify that the Doctoral Dissertation of

Kenneth Jarod Jones

has met the dissertation requirements of

North Carolina Agricultural and Technical State University

Greensboro, North Carolina
2014

Approved by:

Dr. Gary L. Lebby Major Professor	Dr. Robert Li Committee Member
Dr. Christopher Doss Committee Member	Dr. Corey Graves Committee Member
Dr. John C. Kelly, Jr. Department Chairperson	Dr. Albert Esterline Committee Member

Dr. Sanjiv Sarin
Dean of Graduate Studies

Dedication

This thesis is dedicated to my father, Bobby L. Jones, my mother, Jeanette C. Jones, my sister, Tiffani J. Jones, and my grandparents, Willie and Imogene Jones, Sr. and Clarence and Inez Curry.

Biographical Sketch

Kenneth Jarod Jones was born on May 21, 1981, in Dayton, OH to Bobby and Jeanette Jones. He is the elder brother of Tiffani J. Jones and the beloved grandson of Willie and Imogene Jones, Sr. and Clarence and Inez Curry. In 2003, he received the Bachelor of Science degree in Electrical Engineering from North Carolina Agricultural and Technical University in Greensboro, North Carolina. In 2007, he received the Master of Science degree in Electrical Engineering from North Carolina Agricultural and Technical University in Greensboro, North Carolina. He is a candidate for the Doctor of Philosophy degree in Electrical Engineering.

Kenneth is a former member of the Blue and Gold Marching Machine, a brother of the Iota Zeta chapter of Kappa Kappa Psi National Honorary Band Fraternity, a member of Eta Kappa Nu, and a full product of the university. He is aggie born, aggie bred, and when he dies he will be aggie dead. AGGIE PRIDE!

Acknowledgements

I would like to thank Dr. Gary L. Lebby for being my advisor and for the guidance and wisdom he has provided me throughout my studies here at the university. I would also like to thank Dr. Robert Li, Dr. Christopher Doss, Dr. Corey Graves, Dr. Albert Esterline, and Dr. Ronald Pedroni for being on my thesis committee; and the faculty and staff members of the Electrical and Computer Engineering Department at North Carolina Agricultural and Technical State University for your support.

Thank you to my Aggie family! There are too many of you to list. Just know that I appreciate your support. I would like to give a special thank you to my roommates, Wen Fang, Xiaojue Tao, and Emily Fang. Thank you for your support!

Finally, I would like to offer words of thanks and gratitude to all of my colleagues in the Biological Inspired Engineering and Energy (BIEES) Laboratory for their friendship, support, and assistance. I would especially like to thank Wen Fang, Joseph Derrick Tabron, Charles Winley, and Audley Darmand for helping me during my progression through the electrical engineering doctoral program here at North Carolina Agricultural and Technical State University.

Table of Contents

List of Tables

Table

List of Figures

Figure

Abstract

The world's heavy dependence on vehicles which utilize hydrocarbon fuels as a primary power source has renewed the interest in electric and hybrid vehicles for industrial, commercial, and public use. The primary objective of this movement is centered on increased efficiency in energy usage for transportation. Hybrid vehicles which utilize an internal combustion engine as a linear generator, converting energy stored in hydrocarbon fuels into electrical power, could serve as a transitioning technology. The free-piston enhanced hybrid vehicle (FPHV) could potentially fill this role.

The present work contains a theoretical and numerical approach to analyzing and optimizing the usage of energy within a free-piston enhanced hybrid vehicle given urban, highway, and aggressive driving profiles. An electromechanical model representing the free-piston hybrid vehicle is developed and optimized based on mass of energy sources carried.

Chapter 1
Introduction

Over the past few decades energy has become an increasingly polarizing topic worldwide. The rise of newly industrialized and modernized countries has increased market demand for energy resources and caused their costs to soar. In particular, the rising cost of oil has led to increased gas prices. Between 2001 and 2012, the national average for gasoline in the United States increased by 250% going from $1.46 per gallon to $3.67 per gallon[1]. The world's heavy dependence on vehicles which utilize hydrocarbon fuels as a primary power source has renewed interest in electric and hybrid vehicles for industrial, commercial, and public use. The primary objective of this movement is centered around increased efficiency in energy usage for transportation[2].

Pure electric vehicles are vehicles that utilize an energy storage component, such as a battery, as the sole energy source for propulsion; while hybrid vehicles are characterized by the utilization of multiple energy sources[3, 4]. New technological developments focused around energy storage have allowed pure electric vehicles to gain some traction within the market as a viable alternative to internal combustion engine (ICE) based vehicles; however the infrastructure to fully support these vehicles is underdeveloped within most industrialized countries[5, 6]. Even though a pure electric vehicle may get similar mileage to that of an ICE vehicle, the ability to quickly refuel ICE vehicles and the established infrastructure of hydrocarbon focused fueling stations across most countries can not be overlooked from an economic standpoint. The established fueling infrastructure represents a surmountable investment that will un-

doubtedly stagger the growth of the pure electric vehicle industry. Alternatively, hybrid vehicles could potentially play an important role in bridging the gap between the established hydrocarbon fuel network and the burgeoning pure electric fueling network, given their multiple power source nature.

Hybrid vehicles which utilize an internal combustion engine as a linear generator, converting energy stored in hydrocarbon fuels into electrical power, could serve as a transitioning technology. The free-piston enhanced hybrid vehicle (FPHV) could potentially fill this role. A FPHV uses a linear generator to convert fuel and supply electrical power to an induction motor which serves as the motor vehicle's engine[3]. By existing within both networks the FPHV is able to effectively utilize the current fueling infrastructure devoted to ICE vehicles and at the same time take advantage of new developments for the electric car. This dissertation will focus on the optimization of energy use and storage within the FPHV such that its design and operation are effective for everyday public transportation.

The second chapter of this study will present a literature review about dynamics of vehicle propulsion; after which the third chapter will take an in-depth look at free-piston generator. The fourth chapter will present the simulated annealing algorithm. The fifth chapter covers the electrical drive system and power electronics located within the free-piston hybrid vehicle. The sixth chapter looks at the internal design of the free-piston hybrid vehicle and the development and implementation of the energy optimization algorithm. Chapter seven analyzes the results of the optimization algorithm. Lastly, chapter eight provides conclusions formed during the study and suggest areas where additional work should be performed.

Chapter 2
Literature Review

2.1 Vehicle Dynamics

A hybrid vehicle is defined as a vehicle with at least two different energy converters and two different energy storage systems which are primarily used for vehicle propulsion[3]. The primary objective for a system using multiple energy sources is system efficiency. The developers of hybrid vehicles seek to utilize the advantages unique to each energy subsystem within the vehicle while minimizing their disadvantages by balancing one subsystem against the other based on established goals. In order to develop a viable hybrid vehicle the basic dynamics surrounding vehicle propulsion must be considered.

2.1.1 Vehicle Propulsion

Newton's second law states that the net forces acting on an object are equal to the mass of that object multiplied by its acceleration. Using Equation 2.1, a basic equation for the acceleration of a vehicle can be derived for a given environment; where m represents the mass of the object, a represents its acceleration, and F represents the forces acting on it.

$$\sum F = ma \tag{2.1}$$

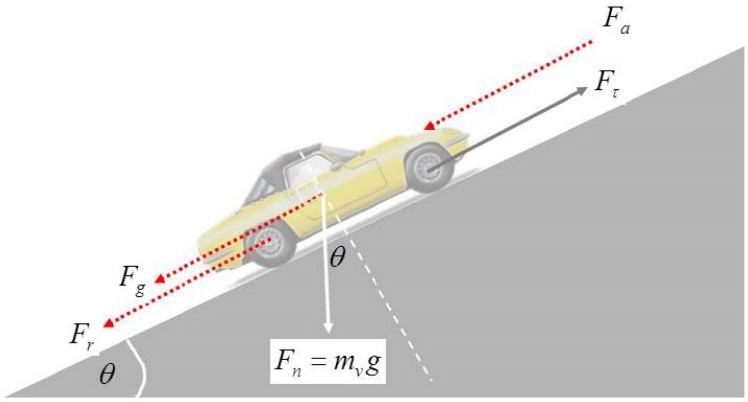


FIGURE 2.1: *Forces acting on a vehicle in motion.*

Equation 2.2 represents the summation of forces acting on the vehicle according to Newton's second law. The mass of the vehicle, m_v, multiplied by the vehicle's acceleration is equal to the difference of traction force, F_τ, applied by the tires and the summation of environmental forces acting upon the vehicle[7]. Figure 2.1 shows the environmental forces acting on the vehicle: rolling friction (F_r), normal force (F_g), and air drag (F_a).

$$m_v \frac{d}{dt} v(t) = F_\tau(t) - (F_r(t) + F_g(t) + F_a(t)) \tag{2.2}$$

Rolling friction, displayed in Equation 2.3, is the horizontal component of the normal force acting on the vehicle; where the tires serve as the point of contact. The coefficient of rolling friction, c_r, represents the bond between the driving surface and tires of the vehicle; while θ represents the inline angle of the road and g represents gravity. Equation 2.4 yields the vertical component of the normal force acting on the vehicle. Equation 2.5 models the aerodynamic resistance acting on the vehicle[7]. The model is a function of air density (ρ), the vehicle's frontal area (A_f), the vehicle's velocity (v), and the drag coefficient (c_d).

$$F_r(t) = c_r m_v g \cos(\theta(t)) \tag{2.3}$$

$$F_g(t) = m_v g \sin(\theta(t)) \tag{2.4}$$

$$F_a(t) = .5\rho_a c_d A_f (v(t))^2 \tag{2.5}$$

Equation 2.6 represents the force applied for vehicle propulsion. The force is derived from the torque generated by the motor which is delivered to the tires from the drive-train. This torque is then applied to the road through the tires for propulsion. The motor torque, gear ratio with drive-train efficiency, and tire radius are represented by τ, μ_{gear}, and r_w, respectively.

$$F_\tau(t) = \frac{\mu_{gear}\tau(t)}{r_w} \tag{2.6}$$

Having defined the forces affecting the vehicle, Equation 2.2 can now be written as a first-order differential equation representing the acceleration of the vehicle. Using Euler's method, an equation for modeling vehicle propulsion can be generated from Equation 2.7. With Equations 2.7 and 2.8, we are able to simulate the velocity of a vehicle being driven through various environments such as urban or highway traffic.

$$\frac{dv}{dt} = f(t, v(t)) = \frac{F_\tau(t) - (F_r(t) + F_g(t) + F_a(t))}{m_v} \tag{2.7}$$

$$v_{n+1} = v_n + f(t_n, v_n)\Delta t \tag{2.8}$$

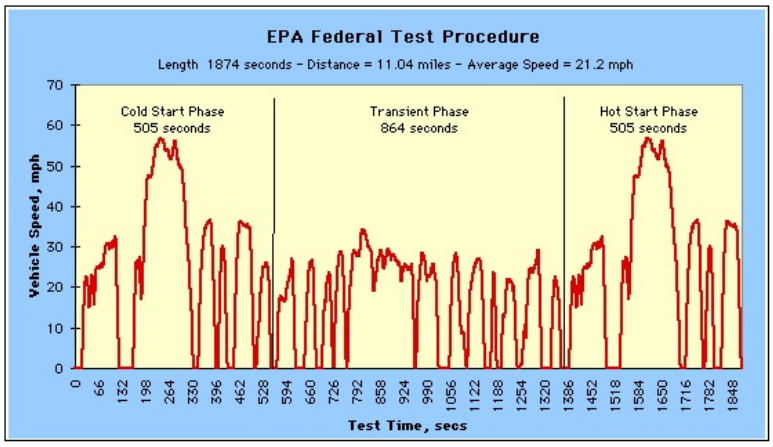
Consequently, the mechanical power, P_m, being exerted to in order to propel the vehicle can be derived using Equation 2.9.

$$P_m(t) = F_\tau(t) v(t) \tag{2.9}$$

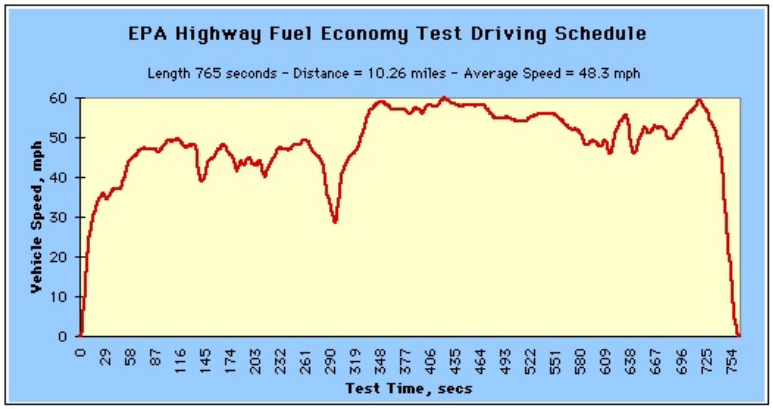
2.1.2 Fuel Economy Testing

Fuel economy tests within the United States are standardized and regulated by the Environmental Protection Agency (EPA)[8]. These tests consist of standardized speed and elevation driving schedules for urban areas and highways in order to evaluate how a vehicle consumes fuel and produces pollutant emissions during typical operation[7]. The tests are carried out in a controlled environments; where control variables like temperature and humidity can be set and monitored. Typically, chassis dynamo-meters are used to emulate force at the wheels of the vehicle during testing; thus representing the vehicle energy losses that would normally be seen if the vehicle were moving according to the selected driving schedule[7]. Multiple driving schedules have been designed to test the fuel economy of vehicles. This study will focus on using the following schedules to evaluate fuel economy: the federal test procedure 75 driving schedule (FTP-75), the highway fuel economy test driving schedule (HWFET), and the supplemental federal test procedure driving schedule (US06).

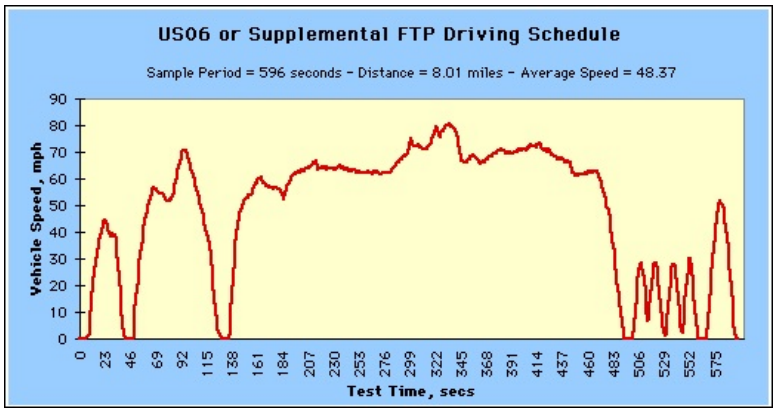
The FTP-75 shown in Figure 2.2a is structured to evaluate driving within an urban area; where vehicle movement typically occurs in short intervals with frequent stops throughout[8]. It is broken into three phases. The first and third phase account for the vehicle's emissions during cold and hot starts of the engine respectively. The second phase accounts for fuel consumption during an area where the vehicle velocity is relatively similar across a stop and go traffic pattern. Figure 2.2b displays the highway fuel economy test driving schedule which is patterned after typical highway driving across a ten mile stretch.



(A) *The Federal Test Procedure 75 driving schedule.*



(B) *The highway fuel economy driving schedule.*



(C) *The US06 FTP driving schedule.*

FIGURE 2.2: *The federal fuel economy test driving schedules.*

TABLE 2.1: *EPA Standardized Weights for Fuel Economy Testing of Hybrid Vehicles*

	FTP-75	**HWFET**	**US06**
City	90%	-	10%
Highway	-	21%	79%

Figure 2.2c contains the US06 FTP driving schedule which represents an aggressive driving profile. A combination of the phases in the driving schedules mentioned are used in order to measure the fuel economy for a given vehicle. The combination and formula for weighting each phase is determined by the EPA[8]. Table 2.1 displays the weights standardized by the EPA for testing the fuel economy of hybrid vehicles with the overall fuel economy of the vehicle being represented as a combination both; where 55% of the overall fuel economy is city fuel economy and 45% is highway fuel economy[9].

2.2 Hybrid Electric Vehicles

Hybrid electric vehicles are characterized by the configuration of their primary and secondary power sources for vehicle propulsion. The topology of a hybrid vehicle will fall into one of the following categories: series hybrid, parallel hybrid, or series-parallel hybrid.

2.2.1 Series Hybrid

In the series hybrid configuration, the electric motor is the sole source of propulsion for the vehicle. As shown in Figure 2.3, the energy sources which provide power to the motor include an energy storage device along with a primary power generator that converts energy from a specified form into electricity[3]. The inclusion of the energy storage device within the series hybrid topology prevents the need for the primary power unit to operate at powers level similar to that of the electric motor. Ideally, the

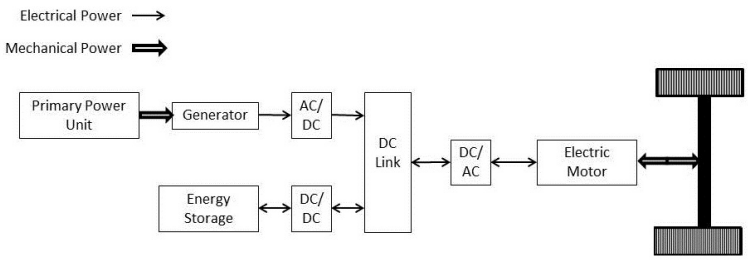


FIGURE 2.3: *The series hybrid topology.*

energy storage can be sized such that the primary power unit can be operated within range the that is more efficient for energy conversion given the technology utilized.

2.2.2 Parallel Hybrid

Within the parallel hybrid topology both the primary power unit and the electric motor are capable of propelling the vehicle through the drive-shaft[7]. Using this topology the electric motor and primary power unit can assist one another in handling the load for propulsion. Typically the electric motor is chosen to help provide additional torque when needed[3]. Figure 2.4 also shows that the electric motor can also be used as a generator to recharge the energy storage unit of the vehicle. While this topology does provide additional assistance to the primary power unit concerning torque; the assistance is only helpful at low speeds. The primary power unit must still be capable of propelling the vehicle at high speeds without the assistance of the electric motor. Unlike the series topology, the primary power unit is not shielded from the demands of the driving loaded. The primary power unit can also not serve as a generator for the onboard energy storage unit; thus leaving regenerative braking, plug-in fueling, and unit replacement as the only options for recharging the energy storage unit.

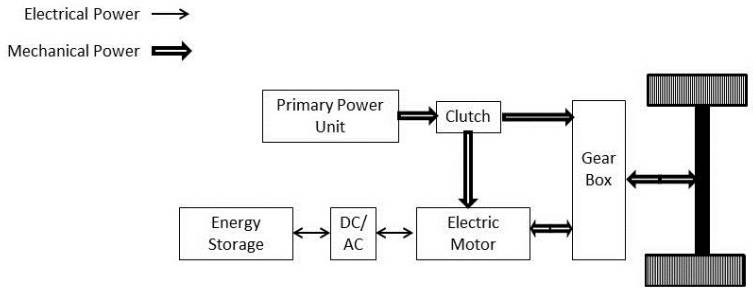


FIGURE 2.4: *The parallel hybrid topology.*

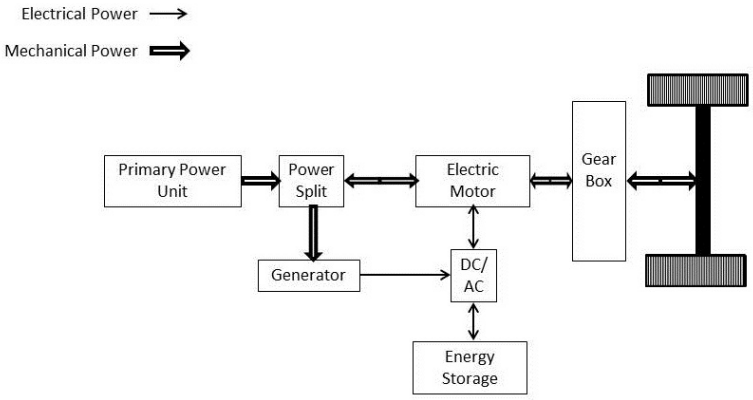


FIGURE 2.5: *The series-parallel hybrid topology.*

2.2.3 Series-Parallel Hybrid

The series-parallel hybrid topology combines features from both the series and parallel topologies. This topology has been widely used in the Toyota Prius with much success[3]. The primary power unit must still adhere to the power requirements necessary for driving the vehicle; however the electric motor can now be used to as a secondary propulsion device. As shown in Figure 2.5, this allows the primary power unit to serve as an additional generator while the electric motor propels the vehicle. Essentially the electric motor and primary power unit can be chosen to drive the car at speeds which are most suitable for each device.

2.3 Free-Piston Hybrid Vehicle

The free-piston hybrid vehicle can be classified as a series hybrid. The FPHV benefits from this topology due to the isolation of the of the primary power unit. An operating frequency can be chosen for the free piston generator that focuses on the efficiency of energy conversion; while the energy can be sized to balance the power demands of the electric motor. This research will focus on determining the attributes for the energy storage elements in the series topology such that the FPHV is a viable and energy efficient vehicle which can be used for everyday travel.

Chapter 3
Free Piston Generator

3.1 Background

A free-piston generator consists of a linear combustion engine combined with a linear electric machine which can convert hydrocarbon fuel into electricity[10]. The piston is considered free due to the motion of the piston being unrestricted by a crankshaft, which is normally seen in internal combustion engines. The removal of the crankshaft allows the piston to easily vary its stroke length; however it allows requires stricter control than the typical ICE. In 1928, Raul Pescara invented the first free-piston engine which he described as a spark ignition air compressor[11]. The dual piston free-piston generator was patented in the United States by Sydney Baruch in 1959[12]. Baruch's design, shown in Figure 3.6, places magnets with matching poles 120 degrees out of phase of one another on a movable translator. The translator lies between two opposing pistons and passes through coupled coils to generate electricity.

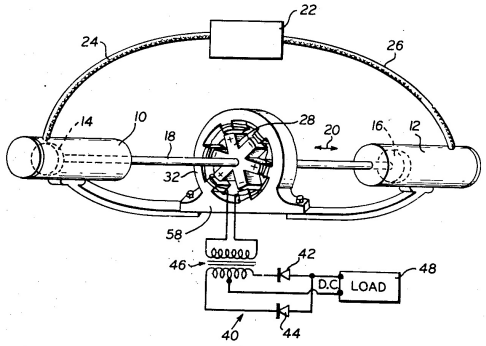


FIGURE 3.6: *A dual piston free-piston generator patented by Sydney Baruch.*

3.2 FPG Modeling

Free piston generators are commonly configured in one of two ways. The first being a single piston design where the bounce back chamber opposite of the piston is a hydraulic cylinder containing some gas, liquid, or spring which is used to rebound the translator[13]. The second configuration is a dual piston design where the bounce back chamber is replaced with another piston and combustion chamber. This study will focus on the dual piston configuration. The modeling of the dual piston FPG can be broken up into two parts: the modeling of the thermodynamics of combustion within the piston chambers for motion and the modeling of the electrodynamics of the system.

The dual piston configuration of the free piston generator, shown in Figure 3.7, consists of two opposing pistons attached to a translator. Alternating combustion within the chambers applies force to the pistons and moves the translator back and forth; thus passing the magnets by the coils and inducing an electromagnetic field in the coils[14]. Using Newton's second law Equation 3.10 can be formulated, where m_p represents the mass of the pistons,x is the displacement of the piston along the stroke path, F_e is an expansion force in one of the chambers, F_c is the opposing compression force, F_m is electromagnetic force from the generator, and F_f is friction. This study will consider the motion of the piston to be frictionless and focus on modeling the combustion in the piston chambers that supply the compression and expansion forces.

$$m_p \frac{d^2x}{dt^2} = F_e - F_c - F_m - F_f, \tag{3.10}$$

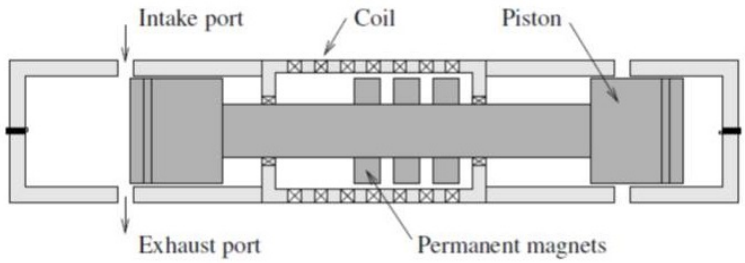


FIGURE 3.7: *A dual piston free-piston generator.*

3.2.1 Thermodynamics of a FPG

In order to model the expansion forces in the combustion chamber we must model the combustion process itself and examine how it performs work upon the piston. The first law of thermodynamics, shown in Equation, states that the change of internal energy within a system is the sum of the change in heat supplied to the system, dQ, and work done by the system, dW[15].

$$dU = dQ - dW \tag{3.11}$$

If it is assumed that complete combustion occurs within the chamber such that all fuel is consumed, the combustion process can be modeled as a closed system and Equation 3.11 can be rewritten as Equation 3.12.

$$dU = dQ - PdV \tag{3.12}$$

Equation 3.12 defines the change in work as a change in volume occurring at a constant pressure. Assuming that the gas combusting in the cylinders are ideal, equations for the change in pressure and temperature for each chamber can be derived using the ideal gas law shown in Equation 3.13. The ideal gas law represents the behavior of the gas in the chamber by approximating the relationships between pressure, volume, mass, and temperature of the specific gas. C_v is the specific heat for a constant volume, C_p is the specific heat for a constant pressure, and γ is the specific heat ratio.

$$mC_v(\gamma-1)T = PV \tag{3.13}$$

$$\gamma = \frac{C_p}{C_v} \tag{3.14}$$

By taking the derivative of Equation 3.13 with respect to time we arrive at Equation 3.15 which also represents the change of internal energy in the system given Equation 3.16 represents the internal energy within an ideal gas[16].

$$mC_v\frac{dT}{dt} = \frac{1}{(\gamma-1)}\left(P\frac{dV}{dt} + V\frac{dP}{dt}\right) \tag{3.15}$$

$$U = mC_v\Delta T, \tag{3.16}$$

Equations 3.17 and 3.18 represent the change of pressure and temperature within a combustion chamber where P represents the chamber pressure, V is the chamber volume, T is the temperature of the gas in the chamber, m is the mass of the gas in the chamber, and Q is the heat in the chamber[10, 17].

$$\frac{dP}{dt} = \frac{\gamma-1}{V}\frac{dQ}{dt} - \frac{\gamma}{V}P\frac{dV}{dt} \tag{3.17}$$

$$\frac{dT}{dt} = \frac{1}{mC_v}\frac{dQ}{dt} + \frac{T(1-\gamma)}{V}\frac{dV}{dt} \tag{3.18}$$

Now that equations for the rate of change in pressure and temperature a formula must be derived which represents the rate of heat release over the duration of the combustion. The overall release of heat use in Equations 3.17 and 3.18 can be separated into two portions: the heat released by the combustion, Q_c, and the heat absorbed by the chamber walls, Q_l, resulting in a loss of heat from the system.

$$\frac{dQ}{dt} = \frac{\partial Q_c}{\partial t} - \frac{\partial Q_l}{\partial t} \tag{3.19}$$

The derivative of the Wiebe function has been found to be a suitable estimator for the release rate of heat during combustion[10, 18]. The Wiebe coefficients, a and b are determined experimentally for the Wiebe function displayed in Equation 3.20; where $t_0$represents the start time of the combustion and t_d represents the burn duration of the combustion. The value of burn duration and the Wiebe coefficients are dependent upon the specific fuel being used in the combustion process[19]. Q_{in} in Equation 3.21 represents the total amount of heat added during combustion.

$$\chi(t) = 1 - \exp\left(-a\left(\frac{t-t_0}{t_d}\right)^{1+b}\right) \tag{3.20}$$

$$\frac{Q_c}{dt} = Q_{in}\frac{\partial \chi(t)}{\partial t} \tag{3.21}$$

Given a specific fuel, the heat of combustion,ΔH_c, can be calculated by taking the difference between the heat of formation for the products and reactants of a combustion process utilizing the specified fuel. Q_{in} can then be determined with Equation 3.22; where m is the mass of the fuel used in the chemical reaction.

$$Q_{in} = m\Delta H_c \tag{3.22}$$

The loss of heat in Equation 3.19 is the result of heat being absorbed by the walls of the combustion chamber. The rate of heat absorbed by the wall is expressed in Equation 3.23; where A is the area of the chamber wall, T is the temperature of the chamber gas, and T_w is the temperature of the chamber wall.

$$\frac{\partial Q_l}{\partial t} = \alpha A(T - T_w) \tag{3.23}$$

The heat transfer coefficient, α, is a widely used heuristic that approximates how the wall absorbs heat during combustion in internal combustion engines[20, 18, 17]. In

Equation 3.24, P represents the chamber pressure, V is the chamber volume, T is the temperature of the gas in the chamber, and $\bar{v}$ is the average speed of the piston.

$$\alpha = 130V^{-0.06}\left(\frac{P}{10^5}\right)^{0.8}T^{-0.4}\left(\bar{v}+1.4\right)^{0.8}. \tag{3.24}$$

The temperature of the wall, T_w, does not remain constant during repeated combustion cycles. The heat absorbed by the wall gradually changes the temperature of the wall. We can model this behavior as a lumped heat capacitance, in which the absorbed heat can be thought of as current charging a capacitor; where the wall eventually reaches a steady state when it absorbs enough heat[21]. Equation 3.25 represents the change in temperature for the wall of the chamber. The parameters C_w and R_w represent the thermal capacitance and thermal resistance of the chamber walls.

$$\frac{dT_w}{dt} = \frac{1}{C_w}\frac{dQ_l}{dt} - \frac{T_w}{R_wC_w} \tag{3.25}$$

The thermal capacitance, shown in Equation 3.26, is a function of the volume of the engine block, V_{block}, the density of the material used to make the engine block, ρ, and a heat conduction constant, c, which is determined by the material used to construct the engine block.

$$C_w = \rho c V_{block} \tag{3.26}$$

The thermal resistance, displayed in Equation 3.27, is a function of the outside area of the engine block, A_{block}, and a heat transfer coefficient, h, which is determined by the airflow outside of the block.

$$R_w = \frac{1}{hA_{block}} \tag{3.27}$$

3.2.2 Electrodynamics of a FPG

The FPG is a linear generator combined with an internal combustion engine which serves to propel the prime mover of the generator. A common configuration for the linear generator in an FPG can be seen in Figure 3.8[10]. Permanent magnets with alternating poles are placed along the prime mover of the generator; so that they may induce an electromotive force when passing by the coils on the stator[22]. Using Faraday's law, shown in Equation 3.28, the electromotive force produced by the generator can be modeled; where N is the number of turns in the coil and Φ is the magnetic flux[14, 18].

$$E = -N\frac{d\Phi}{dx}\frac{dx}{dt} \tag{3.28}$$

Equation 3.29 represents the Lorentz force equation for a stationary wire; where I is electrical current, L is the length of the wire, and B is the magnetic field, a relationship between the electromotive force generated and the electromagnetic force acting on the generator can be derived.

$$\vec{F} = I\vec{L} \times \vec{B} = -BiL\vec{a}_x \tag{3.29}$$

Writing Equation 3.28 in terms of B and L yields Equation 3.30, which can be substituted into Equation 3.28[23, 24]. This generates Equation 3.31 which yields the electromagnetic force acting on the generator.

$$E = BL\frac{dx}{dt} \tag{3.30}$$

$$F_m = \frac{E \cdot I}{\frac{dx}{dt}} \tag{3.31}$$

The electrical current passing through the load can be determined with Equation 3.32; where Z is the impedance of the stator and V_{sys} is the voltage level of the system.

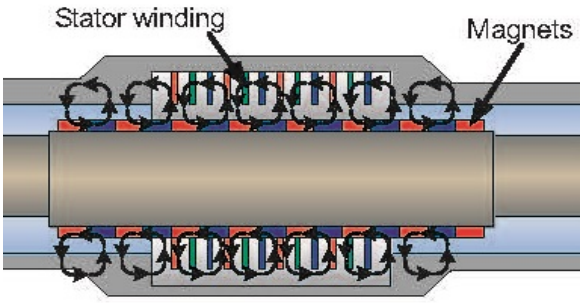


FIGURE 3.8: *The schematic of stator and prime mover for an FPG.*

Equations 3.31 and 3.32 show that the force acting on the generator from the system is proportional to the amount of power being absorbed by the system.

$$I = \frac{E - V_{sys}}{Z} \tag{3.32}$$

3.2.3 Motion of a FPG

Now that the basic dynamics of the system have been explored, a state space model can be developed to represent the motion of the FPG. Equation 3.28 clearly shows that the electromotive force generated by the generator is primarily dependent on the speed of the pistons. Equation 3.10 can be rewritten to form a first order differential equation modeling the speed of the piston.

$$\frac{d^2x}{dt^2} = \frac{F_e - F_c - F_m}{m_p} \tag{3.33}$$

We will maintain the assumption that the system being modeled is frictionless. The electromagnetic force, F_m, was defined in Equation 3.31; leaving the compression force, F_c, and the expansion force, F_e, as undefined. Pressure is defined as force per unit area; therefore both the expansion and compression forces can be derived from the pressure in each chamber using Equation 3.34, where A_p is the area of the bore of the piston.

$$F = PA_p \tag{3.34}$$

Given the reciprocative nature between the chambers when applying force to the pistons, the expansion and compression forces will be relabeled with a numerical subscript to represent the chamber of the force instead of the type of force. Now that all of the forces in Equation 3.33 have been identified, a basic state space model representing the movement of the FPG can be formed. Equations 3.35, 3.36, and 3.37 represent the state of the gas in each chamber during combustion. The volume for the chambers shown in Equation 3.36 alternate signs with another due to their connection.

$$\frac{dP_i}{dt} = \frac{\gamma - 1}{V_i}\frac{dQ_i}{dt} - \frac{\gamma}{V_i}P_i\frac{dV_i}{dt} \tag{3.35}$$

$$\frac{dV_i}{dt} = \pm A_p\frac{dx}{dt} \tag{3.36}$$

$$\frac{dT_i}{dt} = \frac{1}{mC_v}\frac{dQ_i}{dt} + \frac{T_i(1-\gamma)}{V_i}\frac{dV_i}{dt} \tag{3.37}$$

With states defined for the change in pressure, Equation 3.34 can be substituted into Equation 3.33 to yield Equation 3.38 which represents a model for the acceleration of the pistons in the generator based on the pressure in each chamber.

$$\frac{d^2x}{dt^2} = \frac{A_p(P_1 - P_2) - F_m}{m_p} \tag{3.38}$$

3.3 Simulation of a FPG

Using the first-order differential equations described in the previous section, a discrete state space model shown in Equations 3.39 and 3.40 can be generated[25]. The states, $x[k]$, represent the heat released, the change in pressure, in temperature, and in volume of the gas in each combustion chamber; along with the acceleration and velocity of the piston. The input, $u[k]$, for the systems represents the amount of fuel placed into each chamber for combustion. Parameters A, B, C, and D are generated

from the first-order differential equations for thermodynamics, electrodynamics, and motion.

$$x[k+1] = Ax[k] + Bu[k] \tag{3.39}$$

$$y[k] = Cx[k] + Du[k] \tag{3.40}$$

The output, $y[k]$, for the system is the voltage waveform generated by the motion of the piston. It should be noted that the magnitude of the waveform is proportional to the product of the magnetic field and the length of the wire to which current is being induced, shown in Equation 3.30. In order to determine the peak voltage the back-electromagnetic field constant, K_E, representing the product of the magnetic field and the length of the wire must be determined. This can be done by choosing an obtainable operating frequency for the FPG and a desired peak voltage to occur along the half periods within that frequency. Equation 3.41 demonstrates how to obtain the back-electromagnetic field constant for Equation 3.42. The parameters Δt and L represent the time it takes to complete a stroke and the stroke-length, respectively. It should be noted that the compression ratio of the piston chamber affects the length of the stroke. The compression ratio represents the relationship between the minimum and maximum values of volume for the combustion chambers[19].

$$K_E = \frac{V_{Peak}\Delta t}{L} \tag{3.41}$$

$$E = K_E \frac{dx}{dt} \tag{3.42}$$

Once the voltage peak has been determined the amount of power absorbed by the system needs to be chosen as it has an effect on the operation of the system as seen in Equation 3.31[17]. If it is assumed that the load is purely resistive and that the FPG

voltage waveform is similar to that of a sine wave; then the power absorbed by the load can determined using Equation 3.43. V_M is the magnitude of the voltage waveform and R_L is the load resistance.

$$P = \frac{V_M^2}{2R_L} \tag{3.43}$$

Chapter 4
Electric Drive System of a Free-Piston Hybrid Vehicle

4.1 Drive System Components

In this chapter the basic components that constitute the electric drive system of the free-piston hybrid vehicle will be discussed. The free-piston hybrid vehicle (FPHV) is classified as a series hybrid[26]. Figure 4.9 resembles the diagram for a series hybrid presented in Chapter ; the free piston generator has been substituted in as the primary power unit of the vehicle. The electric drive system is responsible for converting and delivering power to the necessary components in order to propel the vehicle. The drive system consists of a free-piston generator, an energy storage component, an electric motor, a full-wave rectifier, a resonant inverter, a DC-DC bi-directional converter, and a 3-phase pulse-width modulation (PWM) inverter for DC to AC conversion.

The free piston generator is assumed to be the main source of power for the system. The energy storage component serves to assist the FPG in managing energy within the system. The electric motor converts electrical power into mechanical power and propels the vehicle. The full-wave rectifier converts the AC power generated by the FPG into DC power. This helps lessen the need for the FPG to generate power at a frequency specified by the electric motor; thus enabling the FPG to operate at a frequency more suited for energy conversion[27]. The DC-DC bi-directional converter allows energy to travel back and forth to the energy storage unit[28]. This enables the controller to balance the energy within the system. The DC-link receives and stores energy from all generators connected to it. It holds just enough energy to achieve

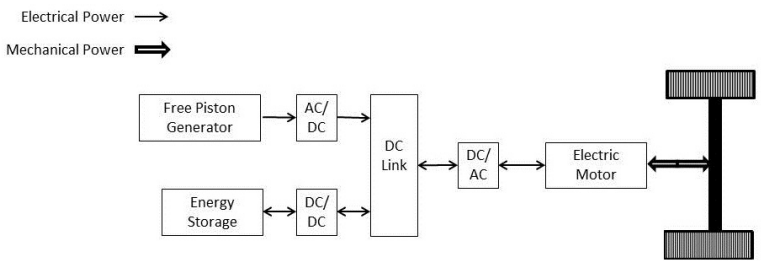


FIGURE 4.9: *The free-piston hybrid vehicle drive system block diagram.*

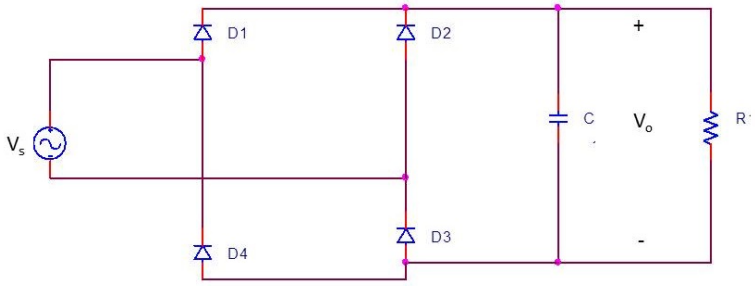


FIGURE 4.10: *A single-phase full wave rectifier with smoothing capacitor.*

maximum power in the motor for one cycle. The 3-phase PWM inverter draws on energy stored in the DC-Link and converts it into AC power; delivering it to the electric motor at the necessary frequency in order to drive the motor.

4.2 Full-Wave Rectifier

The FPHV draws its power from two different energy sources. Each source is attached to the DC-link using a power electronics unit to control or converter the energy. The FPG is attached using a full-wave rectifier. The full-wave rectifier serves to convert the voltage from the FPG, an AC source, into a DC source. Figure 4.10 shows a full-wave rectifier with a smoothing capacitor attached across the load resistor. The capacitor produces a voltage which is fundamentally DC. It contains a slight ripple due to the voltage level of the source dropping below that of the capacitor during each half period of the sinusoidal wave[29].

Equation 4.44 presents the relationship between a sinusoidal source and an output

voltage, v_o, for a full-wave rectifier given a parallel RC load. The output voltage follows the source voltage while the diodes are forward bias. When the source voltage drops below the output voltage the diodes become reverse bias. The capacitor provides the voltage to the load resistor at this point. This causes a voltage ripple across the DC voltage. The size of the ripple can be determined using Equation 4.45; where V_m is the magnitude of the sinusoid, f is its operating frequency, R is the load resistor, and C is the smoothing capacitor[29].

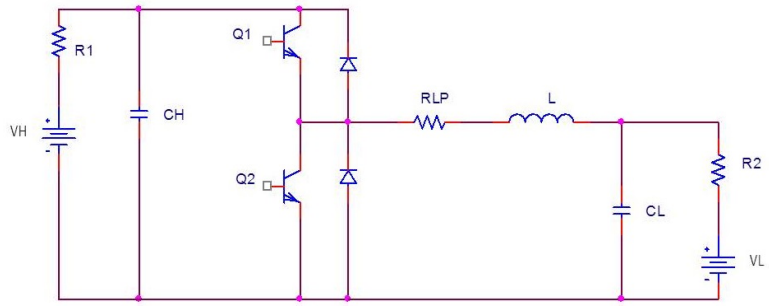
$$v_o(\omega t) = \begin{cases} |V_m \sin(\omega t)|; & one\,diode\,pair\,on \\ (V_m \sin\theta)\exp\left(\frac{-(\omega t-\theta)}{\omega RC}\right); & diodes\,off \end{cases} \tag{4.44}$$

$$\Delta V_o = \frac{V_m}{2fRC} \tag{4.45}$$

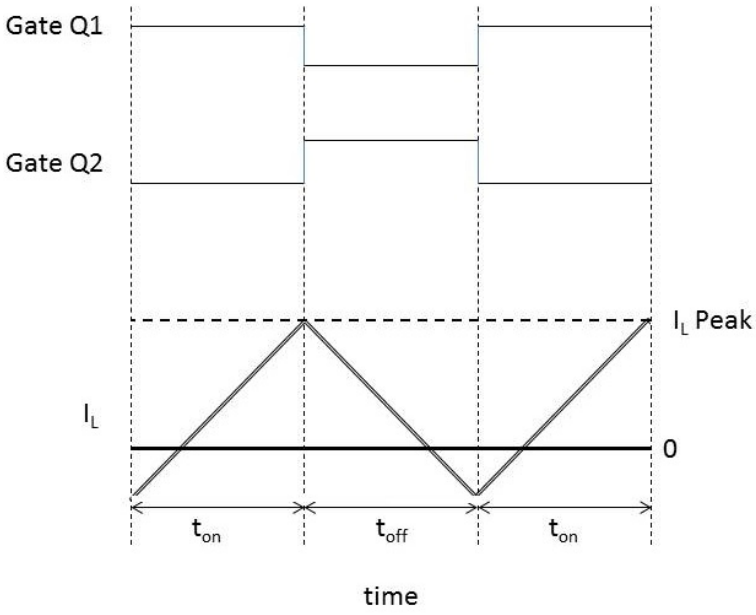
4.3 DC-DC Bi-directional Converter

The second source the FPHV draws power from is the energy storage element, which can refer to a battery or ultra-capacitor. This element needs to be capable of both sending and receiving energy thus it is connected to the DC-link using a bi-directional converter. The DC-DC bi-directional converter is a controllable power electronic unit that allows the charging and discharging of energy between two storage elements[30]. This study will focus on the half-bridge buck/boost converter shown in Figure 4.11a. This converter utilizes two transistors with feedback diodes in order to control the current passing through the inductor. Using a unified duty ratio, in which the opening and closing of gates $Q1$ and $Q2$ alternate between one another, the DC-DC converter is able to increase, decrease, or reverse the flow of current across the inductor[31].

The duty ratio refers to the amount of time in which $Q1$ is open while $Q2$ is closed in respect to the collective time allotted for the aforementioned state and the state in which $Q1$ is closed while $Q2$ is open. The state in which $Q1$ is open and $Q2$ is



(A) *A half-bridge buck boost converter.*



(B) *The switching diagram of a half-bridge buck boost converter.*

FIGURE 4.11: *A single-phase DC-DC bi-directional converter with switching chart.*

closed shall be referred to as t_{on}; while the state in which $Q1$ is closed while $Q2$ is open will be referred to as t_{off}.

$$D \equiv \frac{t_{on}}{t_{on} + t_{off}} \tag{4.46}$$

Figure 4.11b illustrates how the current in the inductor reacts during each state. While t_{on} is occurring the current increases to its peak value. When t_{off} occurs the current decreases and will eventually reverse its direction of flow. It is clear from the graph that by adjusting the duty ratio of the gates the bi-directional DC-DC converter

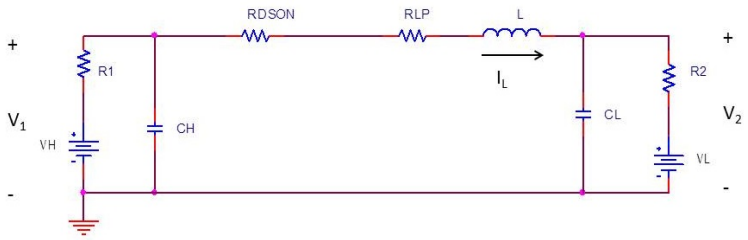


FIGURE 4.12: *The bi-directional converter during t_{on}.*

can transfer energy back and forth to energy storage elements that are connected to it. Now that the states of the DC-DC converter have been identified a closer look can be taken to understand how the circuit shown in Figure 4.11a operates based on which state is active.

The converter contains three energy storage elements: an inductor, a capacitor for the high voltage side, and a capacitor for the low voltage side. Figure 4.12 is the circuit equivalent of the converter during t_{on}. The circuit can be written as a first-order differential equation, seen in Equation 4.48, where v_1 and v_2 represent the high and low voltage sides, respectively. The resistance of inductor and the transistor are given by R_{LP} and R_{dson}. Performing Kirchhoff's current law (KCL) at the high and low voltage capacitors yields Equation 4.48 which showcases how current passing from one node to the other affects energy stored in the capacitors[32].

$$L\frac{di}{dt} + i_L\left(R_{dson} + R_{LP}\right) = v_1 - v_2 \tag{4.47}$$

$$\begin{cases} C_H \frac{dv_1}{dt} = -\left(i_L + \frac{v_1 - V_H}{R_1}\right) \\ C_L \frac{dv_2}{dt} = i_L - \frac{v_2 - V_L}{R_2} \end{cases} \tag{4.48}$$

During t_{off} the high and low voltage sides of the converter are isolated from one another. Performing Kirchhoff's voltage law on the circuit equivalent for t_{off} shown in Figure 4.13 reveals that the voltage across the inductor is strictly a function of the low capacitor voltage during this state. Additionally, performing a KCL at each on the

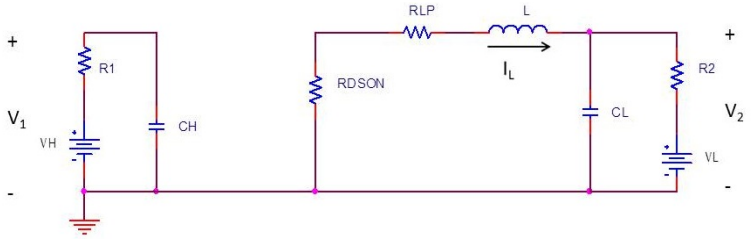


FIGURE 4.13: *The bi-directional converter during* t_{off}.

voltage nodes reinforces Equation 4.49, showing that the change in voltage across the low voltage capacitor is dependent on current in the inductor and current being drawn by an independent source[32]. Meanwhile, Equation 4.50 shows that the voltage across the high voltage capacitor is unaffected by current in the inductor.

$$L\frac{di}{dt} + i_L(R_{dson} + R_{LP}) = -v_2 \tag{4.49}$$

$$\begin{cases} C_H \frac{dv_1}{dt} = -\frac{v_1 - V_H}{R_1} \\ C_L \frac{dv_2}{dt} = i_L - \frac{v_2 - V_L}{R_2} \end{cases} \tag{4.50}$$

Using state space averaging, a set of differential equations can be formed from Equations 4.47-4.50 that will represent the change of current and voltage within the inductor and storage capacitors across both states, simultaneously[32]. If Equation 4.51 is allowed to represent the percentage of time that the converter spends in t_{off} over the period of the duty ratio; then the aforementioned equations can be combined to form the differential equations given in Equation 4.52.

$$D' = 1 - D \tag{4.51}$$

$$\begin{cases} L\frac{di}{dt} = D(v_1 - v_2) - i_L(R_{dson} + R_{LP}) + D'(-v_2) \\ C_H \frac{dv_1}{dt} = -D\left(i_L + \frac{v_1 - V_H}{R_1}\right) - D'\left(\frac{v_1 - V_H}{R_1}\right) \\ C_L \frac{dv_2}{dt} = i_L - \frac{v_2 - V_L}{R_2} \end{cases} \tag{4.52}$$

Now by replacing D' with Equation 4.51 the like terms in Equation 4.52 can be combined. This allows the differential equations to be written in a general form with respect to the initial definition of the duty ratio given in Equation 4.46. The transfer of energy during either state of the duty ratio can now be assessed with Equation 4.53.

$$\begin{cases} L\frac{di}{dt} = Dv_1 - v_2 - i_L\left(R_{dson} + R_{LP}\right) \\ C_H\frac{dv_1}{dt} = -Di_L - \frac{v_1 - V_H}{R_1} \\ C_L\frac{dv_2}{dt} = i_L - \frac{v_2 - V_L}{R_2} \end{cases} \tag{4.53}$$

4.4 DC-Link with 3-Phase Pulse-width Modulation Inverter

The purpose of the 3-phase PWM inverter is to convert energy from the DC-Link into 3-phase power suitable for driving the motor. The 3-phase PWM inverter, shown in Figure 4.14 uses alternating transistor pairs in a six-step switching strategy to draw energy from the DC-link and generate an AC voltage. The output frequency of the voltage is dependent upon the switching frequency of the transistors; while the magnitude of the voltage is dependent on the voltage of the DC source[29]. The DC source is chosen so that it can meet the max load demand and max response time demanded due to the switching frequency of the transistors. The power demand seen by the DC source from one complete switching cycle is the equivalent of the power load demand of the motor plus losses.

Typically the voltage level of a DC-link is kept within a particular range that is suitable for powering devices attached to it. For the FPHV this range is determined by the operating voltage of the motor and control strategy used to operate it. This study will consider a DC-link in which the voltage is kept constant at a specified voltage. Given this assumption, power being supplied to the motor is dependent upon the switching frequency of the transistors. This must be taken into consideration when sizing the capacitor that will serve as the DC-link. Ideally, the capacitor should be able

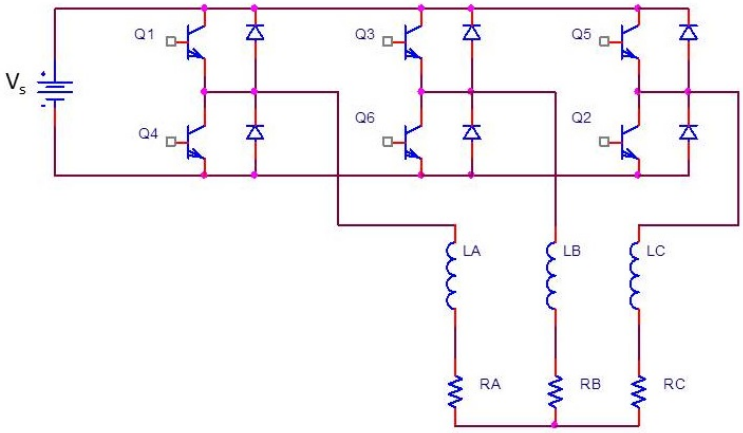


FIGURE 4.14: *A 3-phase PWM inverter.*

to supply max power to the motor which occurs at a certain frequency. The electrical frequency that the transistors must operate is proportional to the number of poles and the synchronous speed of the motor. Using Equation 4.54 the operating frequency of the transistors can be determined for any give synchronous speed[29, 33].

$$f = \frac{poles * \omega_s}{4\pi} \tag{4.54}$$

The instantaneous power supplied or absorbed by a capacitor is given by Equation 4.55. In order to solve for the capacitor, the derivative of the voltage across the capacitor is written as the empirical differences between known values of voltage and time. The power supplied is considered to be at maximum value along with the charge level of the capacitor. Equation 4.56 now represents the relationship between the transfer of energy and the size of the capacitor.

$$P(t) = Cv(t)\frac{dv}{dt} \tag{4.55}$$

$$C = \frac{P_{max}\Delta t}{v_{max}\Delta v} \tag{4.56}$$

Equation 4.56 can be rewritten with the desired values Δt and Δv which represent the maximum voltage level of the capacitor and the electrical frequency at which max-

imum power is drawn from the DC-link. The maximum power is adjusted by a factor μ to account for power losses which occur due to switching and motor slip.

$$C = \frac{\mu P_{max}}{v_{max}^2 f_{Pmax}} \tag{4.57}$$

4.5 FPHV Power Flow

With any electrical system understanding how power flows throughout that system is of great importance. If the system's generators can not meet the energy needs of various power drawing elements on time, then the system will fail. With respect to the FPHV the DC-link must always be filled with enough charge to meet the load demands of the motor; else the motor will not apply torque to the tires when needed. Power flow within the FPHV can be separated into two phases: charging and discharging of the DC-Link. The charging phase entails the FPG and energy storage unit supplying energy to the DC-Link; while the discharging phase consists of the electric motor drawing power from the DC-link. Though these two phases occur simultaneously, they will be evaluated individually and combined once each phase has been modeled.

The FPG and the energy storage element both serve as energy sources for the DC-Link; however the rate at which they can deliver energy varies considerably. The FPG can be seen a sinusoidal of relatively slow response time when compared to the capabilities of an ultra-capacitor or battery. As such, the FPG serves as a continuous source that over time needs assistance from the fast responding energy storage.

Based on the intended usage of the energy storage element the charging of the DC-link will be evaluated using only the FPG at this time. Assuming the capacitance of the DC-Link is known, a resistance, R, can be chosen to achieve a desired output voltage within a given time period. By applying the Laplace transform to Equations 4.58 and 4.59 an equation for the output voltage can be derived[34]. Given the circuit shown in Figure 4.15, it is clear that the Laplace model shown in Equation 4.60 should

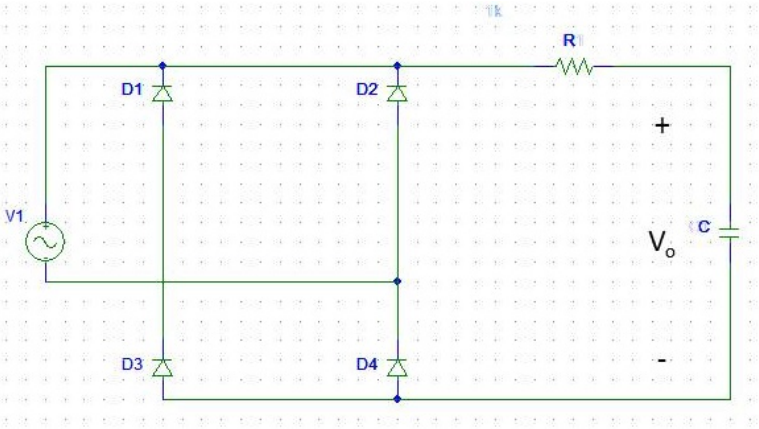


FIGURE 4.15: *The single-phase full wave rectifier attached to DC-Link.*

only be evaluated up until V_o reaches its peak voltage. Past that point the diodes will become reverse bias and no longer feed current to the capacitor.

$$V_s(t) = V_m \sin(\omega t) \tag{4.58}$$

$$\frac{dv_c(t)}{dt} + \frac{v_c(t)}{RC} = \frac{V_s(t)}{RC} \tag{4.59}$$

$$V_o(s) = \frac{\frac{1}{RC} V_m \omega}{\left(s + \frac{1}{RC}\right)\left(s^2 + \omega^2\right)} \tag{4.60}$$

With the parameters for the DC-Link chosen, Equation 4.59 can be revisited in order to include the current supplied by the energy storage element. The current, i_{es}, is restricted to the limits and capabilities of the energy storage element supplying it; therefore care must be taken to assure that the energy storage is sized correctly to effectively assist the FPG in meeting the load demand.

$$\frac{dv_c(t)}{dt} = \frac{V_s(t)}{RC} - \frac{v_c(t)}{RC} + \frac{i_{ES}(t)}{C} \tag{4.61}$$

Discharging the DC-link is dependent on the load demand of the motor. Assuming that the control strategy for the motor is based on maintaining a relatively constant voltage at the DC-link, the change of voltage in the capacitor can be determined with Equation 4.62. The current, i_o, is obtained by dividing the power load demand by the constant DC-link voltage. The parameter, μ, takes into account the efficiency of the motor when drawing power.

$$\frac{dv_c(t)}{dt} = -\frac{i_{Load}(t)}{C} \tag{4.62}$$

$$i_{Load}(t) = \frac{\mu P_{Load}(t)}{V_{DC}} \tag{4.63}$$

Using Equations 4.61 and 4.62 a general equation for the transfer of energy within the DC-link can be written. Essentially the transfer of energy across the link is the sum of the first-order differential equations of the two phases.

$$\frac{dv_c(t)}{dt} = \frac{V_s(t)}{RC} - \frac{v_c(t)}{RC} + \frac{i_{ES}(t)}{C} - \frac{i_{Load}(t)}{C} \tag{4.64}$$

Chapter 5
System Design for Energy Optimization

5.1 System Overview

Utilizing the information covered in the previous chapters a system design for an FPHV can be assembled. In order to design the system assumptions must be made about the vehicle concerning mass and performance. For this study the 2014 Toyota Prius V and the 2013 Volkswagen Jetta SEL have been chosen as base models to derive vehicle specifications from. They are highly recognizable compact vehicles currently available on the consumer market. Table 5.2 shows the specifications of the vehicles chosen[35, 36].

TABLE 5.2: *Vehicle Specifications*

	2014 Toyota Prius V	**2013 VW Jetta SEL**
Fuel Capacity (gal)	11.9	14.5
Curb Weight (lbs)	3274	3070
Wheel Diameter (in)	16	16
EPA City (mpg)	44	25
EPA Highway (mpg)	40	36
Horsepower ($hp@maxrpm$)	98@5200	170@5000
Torque ($lb-ft.@maxrpm$)	105@4000	184@1500
Drag Coefficient	0.25	0.30
Payload (lbs)	915	1067

With the test vehicles established focus will be placed towards developing an FPG based electrical power system that will propel each vehicle. An AC induction motor is selected to serve as the single propulsion engine. The AC-150, shown in

TABLE 5.3: *Electric Motor Specifications*

	AC-150
Motor Type	AC Induction
Poles	4
Peak Power (kW)	150
Continuous Power (kW)	40
Max Speed (rpm)	13000
Max Torque (Nm)	225
Mass (kg)	80
Road Load Efficiency	.86

FIGURE 5.16: *The AC Propulsion AC-150 electric motor.*

Figure 5.16, developed by AC Propulsion, Inc. is capable of delivering approximately 201 horsepower at 8000 rpm[37]. Table 5.3 displays the technical specifications for the AC-150, which exceeds the engine specifications for the Prius and fall slightly short of the engine specifications for the Jetta.

5.1.1 Driving Load Demand

In order to assure that the chosen motor can provide the necessary power, potential power load demands will be calculated based on the EPA driving profiles discussed in Chapter . The force applied by rolling friction and air drag the power load demand can be estimated with Equation 5.65. The rolling friction force, F_r, and the air drag force, F_a , represent the forces that must be overcome while moving the vehicle. The

total force needed to move the vehicle includes the sum of the aforementioned external forces in addition to the force that would move the vehicle absent those external forces. If it is assumed that the vehicle is being driven on a flat surface then forward propulsion is only caused when the motor applies torque to the wheels; increasing the vehicle's acceleration. This assumption leads to the conclusion that power is only demanded when the acceleration of the vehicle is positive.

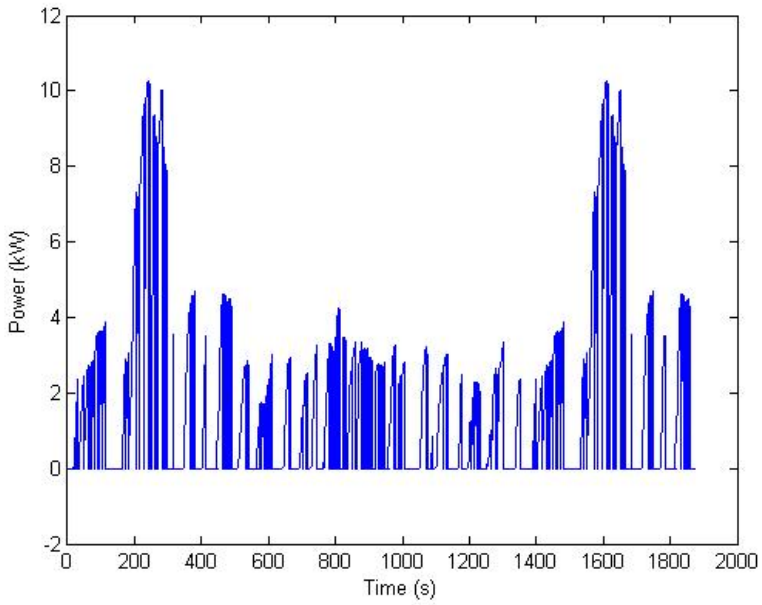
$$P_{Load}(t) = \begin{cases} \left(F_r(t) + F_a(t) + m\frac{dv}{dt}\right) v(t); & \frac{dv}{dt} > 0 \\ 0; & \frac{dv}{dt} \leq 0 \end{cases} \tag{5.65}$$

Figures 5.17 and 5.18 illustrate the theoretical power load demand needed to propel the Prius and the Jetta in an urban environment. The EPA FTP-75 and US06 were used to simulate both moderate and aggressive driving for each vehicle. Notice the peak power for each graph does not exceed 86% of the peak power of the electric motor. The AC-150 should be suitable for providing propulsion for each of the vehicles.

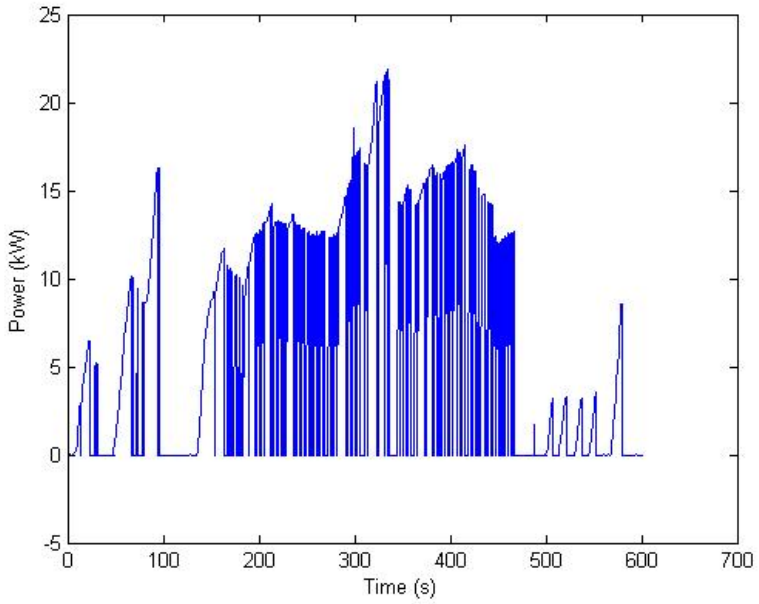
With the motor chosen, the capacitance at the DC-Link can be sized based on the power specifications shown in Table 5.3 and the efficiency map of the AC-150 shown in Figure 5.19[37]. The efficiency map shows that the peak power for the electric motor occurs at $8000rpm$. Using Equation 4.57, the size of the capacitor for the system can be determined; assuming that the operating voltage of the DC-link is a constant $336V$.

5.2 Power Generation

When determining the power generation units for the FPHV the focus is placed on meeting the power demands to move the vehicle; while minimizing the impact the stored energy being carried has on the system. Pempek's Free Piston Power Pack (FP3) will be modeled as the FPG for the test vehicles. The FP3 consists of four $25kW$ free piston modules in pack operating at $30Hz$ with a peak power of $100kW$[38]. Table 5.4 lists the specifications for the individual modules.

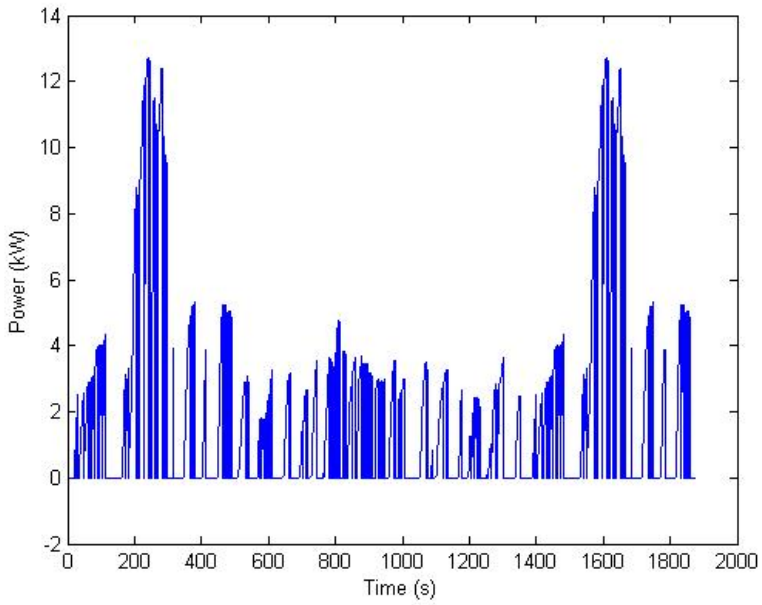


(A) *The FTP-75 power load for a Toyota Prius.*

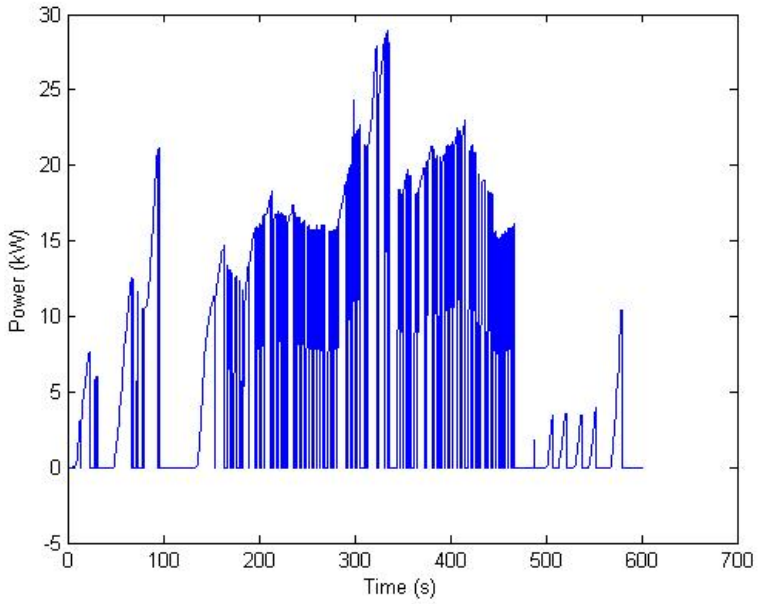


(B) *The US06 power load for a Toyota Prius.*

FIGURE 5.17: *The EPA test based power load demands for the Toyota Prius.*



(A) *The FTP-75 power load for a VW Jetta.*



(B) *The US06 power load for a VW Jetta.*

FIGURE 5.18: *The EPA test based power load demands for the VW Jetta.*

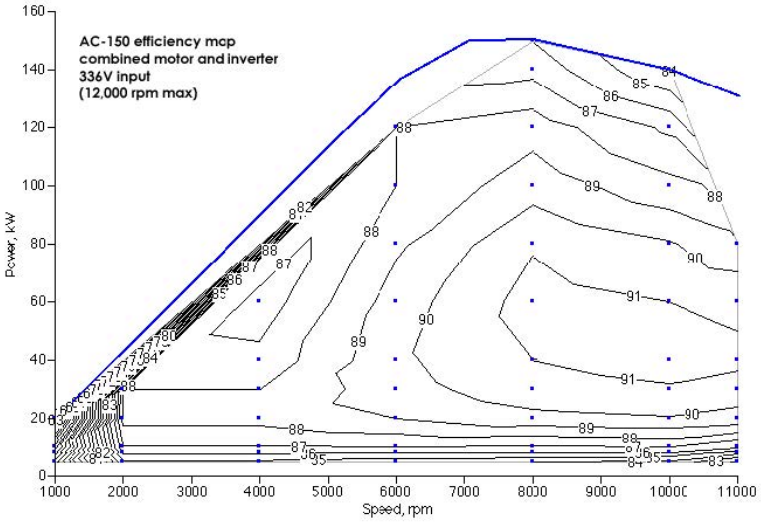
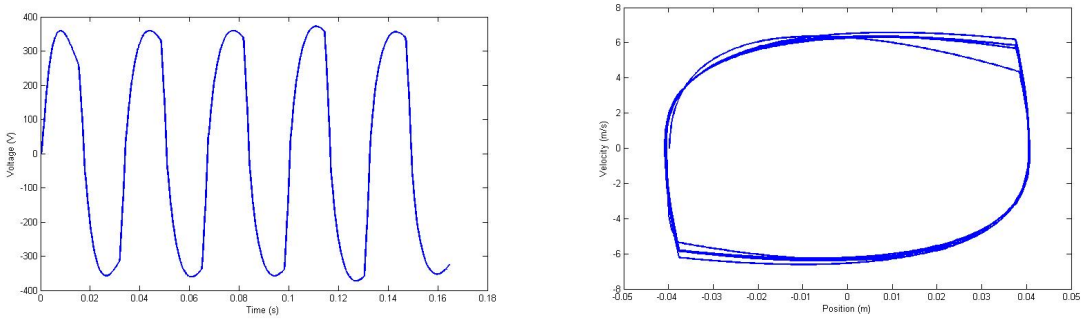


FIGURE 5.19: *The efficiency map of the AC-150.*

TABLE 5.4: *Free Piston Generator Specifications*

	25 kW FPG
Bore (m)	.066
Stoke Length (m)	.103
Compression Ratio	9.8
Mass (kg)	5.2
Material	Aluminum
Operating Frequency (Hz)	30

Using the equations discussed in Chapter 2.3, the movement of an individual module can be described with a discrete state space model. Setting the resistive load to 2.4404Ω and the back-electromagnetic field constant to 75 yields a voltage waveform with a peak value of $350V$ when operating around $30Hz$. Figure 5.20 illustrates the simulated individual free-piston module. With the capacitance and voltage level of the DC-link known and the peak voltage of the FPG chosen, a circuit representing the charging of the DC-link from the FPG's perspective can be designed using Equation 4.60. The FPG can loosely be modeled as a sinusoidal voltage source with a magnitude of $350V$ operating at a frequency of $30Hz$.



(A) *The FPG voltage waveform.* (B) *The piston velocity as a function of position.*

FIGURE 5.20: *The simulation of a 25 kW free piston generator.*

$$V_s(t) = 350\sin(60\pi t) \tag{5.66}$$

$$V_o(s) = \frac{\frac{21000\omega}{RC}}{\left(s+\frac{1}{RC}\right)\left(s^2+(60\pi)^2\right)} \tag{5.67}$$

This study will be assume that the FPG is capable of delivering $150kW$ at a frequency of $30Hz$. Now a resistance for the DC-link can be determined that will place the voltage level of the DC-link at $336V$ when the FPG reaches its peak voltage. Figure 5.21 illustrates the voltage across the DC-Link given the resistance is .475Ω and the capacitance is $3.3mF$.

5.2.1 Energy Storage

Though the scaled FPG can supply $150kW$ to the DC-link, the rate at which it supplies energy is much slower than the potential rate of energy consumption from the electric motor. This is the primary reason an additional energy unit capable of fast response is needed. This study will consider energy packs built using the following cells: Panasonic's NC18650 lithium ion battery and Maxwell Technologies BCAP3000 ultra-capacitor. Table 5.5 lists the characteristics for a single cell of each energy storage

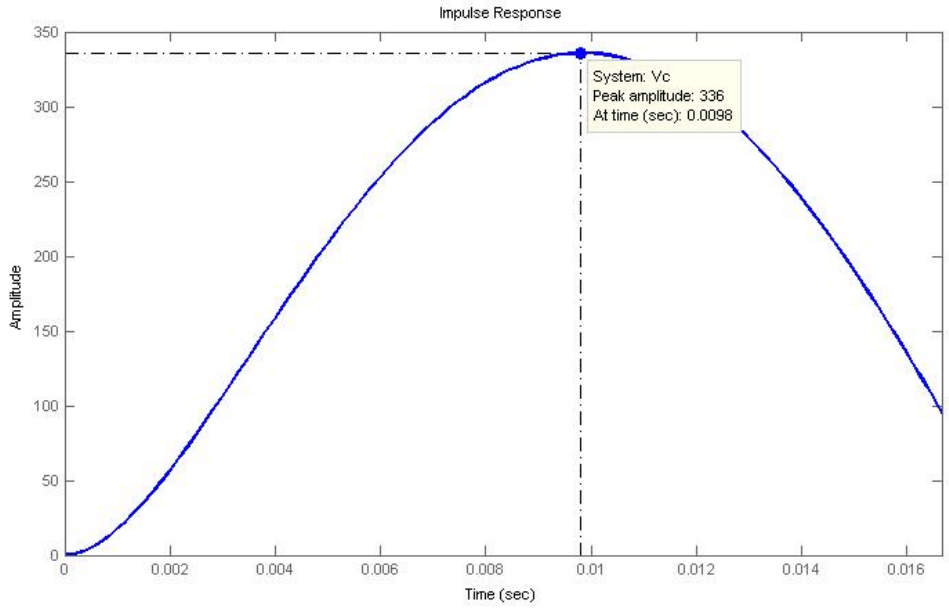


FIGURE 5.21: *The impulse response for the voltage across the DC-link capacitor.*

TABLE 5.5: *Energy Storage Element Specifications*

	NC18650	**BCAP3000**
Storage Type	Lithium Ion Battery	Ultra-capacitor
Energy Storage Capacity (Wh)	10.44	3.04
Max Continuous Current (A)	.55	210
Rated Voltage (V)	3.6	2.7
Mass (g)	45	510

element[39, 40]. The energy pack will consist of only one type of cell. The size of the pack will be addressed during the energy optimization of the FPHV.

5.3 Energy Optimization for System Design

Optimization of the FPHV energy system revolves around correctly sizing the energy stored on board relative to the amount of mass it adds such that the FPHV can maximize its travel distance while minimizing fuel consumption. Since the energy system in the FPHV is co-dependent on receiving power from a hydrocarbon based source and the energy storage element optimization must occur simultaneously over

TABLE 5.6: *Hydrocarbon Fuel Characteristics*

	Gasoline	**Diesel**	**Methanol**	**Ethanol**
Molecular Mass (g)	114.23	170.34	32.04	46.07
Density (gL^{-1})	720	820	790	789
Heat of Combustion ($KJmol^{-1}$)	5470.10	8086.50	726	1366.70
Energy Density (Whg^{-1})	13.30	13.18	6.29	8.24

both sources. Table 5.6 lists the characteristics of the hydrocarbon fuels that will power the FPG[41]. In order to relate hydrocarbon fuels to the energy storage elements the energy density of the fuels are written in terms of kilowatt-hours per gram. Through the fuel's energy density any amount of fuel the vehicle carries can be represented in terms of stored energy with the same units as energy within the energy storage elements. Furthermore this transformation will prove useful when evaluating the fitness of the system.

5.3.1 Simulated Annealing

The FPHV will be optimized using simulated annealing. Simulated annealing is a local search optimization technique. The technique is based off of physical annealing with solids, in which a crystalline solid is heated and then cooled very slowly until it achieves a crystal lattice configuration that is free of crystal defects[42]. When performed properly the resulting crystal is typically assessed to be of higher quality than the original. Using heuristics and a structured heating and cooling schedule this process has been replicated mathematically with great success; particularly when it comes to solving optimization problems of a discrete nature[43, 44].

Applying simulated annealing to solve an optimization problem requires that the heating, cooling, and objective function be tailored for that particular problem. As a result determining the heating and cooling schedules along with starting and stopping temperatures can be difficult, resulting in some trial and error until suitable parameters

are found. Even though parameters are unique to the problem being solved the basic methodology for simulated annealing does not change. Simulated annealing uses rapid heating and slow cooling processes to comb an input space and find an optimal solution for a given objective. In order to begin the process a starting point within the input space is chosen for the initial position of the search parameter, $\vec{h}$. The starting point represents a configuration of the parameters in the optimization problem. The solution parameter, $\vec{S}$ is also initialized to the same position as the search parameter.

$$\begin{aligned} \vec{h} &= initial\,state \\ \vec{S} &= \vec{h} \end{aligned} \tag{5.68}$$

After initialization, the search parameter is altered by a step parameter, Δh, as shown in Equation 5.69. The adjusted parameter, $\vec{h}_{new}$, is a potential solution for the problem that now must be assessed by the objective function, $F(x)$, and evaluated against the solution parameter. It is important to note that $\vec{h}_{new}$ must lie within the input space; if it does not it should be discarded. The step parameter should be specifically tailored for the input space that is to be searched.

$$h_{new} = h_{old} + \Delta h \tag{5.69}$$

Equation 5.70 evaluates the potential new solution and the chosen solution parameter. If the difference, ρ, is less than zero, then h_{new}, is set as the new solution; assuming the overall goal is to minimize the objective function.

$$\rho = F(h_{new}) - F(S) \tag{5.70}$$

$$\begin{cases} \vec{S} = \vec{h}_{new}; & \rho < 0 \end{cases} \tag{5.71}$$

If the suggested solution given by the search parameter is not accepted it is passed through an acceptance probability which is based on the temperature, T, of the system.

If a randomly generated number between zero and one is less than $P(\rho,T)$; then h_{new} is set as the new solution. This heuristic serves to help the process break out of any local extrema. The Boltzmann constant, k, is unique to the relationship under evaluation and determines how active the acceptance probability becomes at high and low temperatures[42].

$$\alpha \in [0,1] \tag{5.72}$$

$$P(\rho,T) = \exp\left(\frac{-\rho}{kT}\right) \tag{5.73}$$

$$\begin{cases} \vec{S} = \vec{h}_{new}; & \alpha < P(\rho,T) \\ \vec{S} = \vec{S}; & otherwise \end{cases} \tag{5.74}$$

Once this step is completed the system is evaluated to see whether it has reached equilibrium. Equilibrium for the process is defined as maintaining a particular solution over a predetermined length of trial solutions while in the temperature cooling cycle of the process. It can also be defined as a point where the system idles with a low level of activity or acceptance while in the cooling phase. This type of equilibrium can be seen in Figure 5.22. When equilibrium is achieved the position of solution parameter is considered the final solution to the objective[45].

The acceptance probability is driven entirely by the rapid heating and slow cooling of the process. High temperatures cause it to accept solutions at a feverishly high rate; while low temperatures have the opposite effect. In essence the high volume of acceptances can be described as boiling the solution. Once the boiling point has been reached, a slow cool down of the system can begin by decreasing the temperature, T, until the system freezes. An initial temperature is chosen as a starting point for the heat process which will increase and decrease according the heating schedule chosen for the given problem. The heating and cooling schedules are defined with individual heating

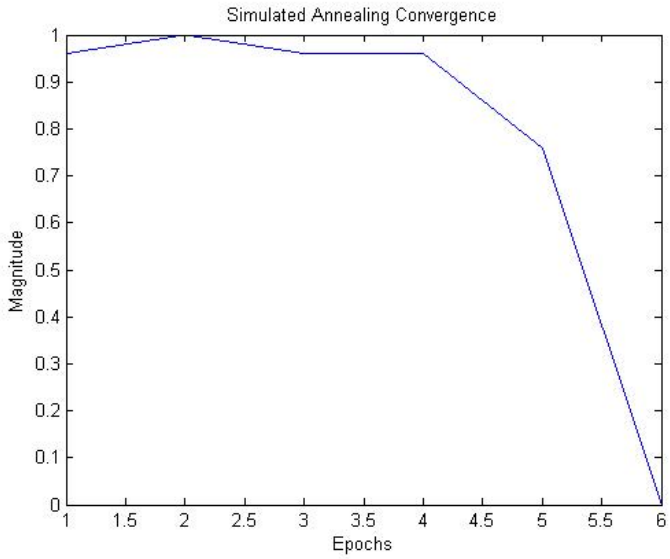


FIGURE 5.22: *The convergence of a simulated annealing algorithm towards a solution.*

and cooling rates and a common temperature step, ΔT. The temperature of the system is adjusted after a pre-determined number of possible solutions has been evaluated.

$$T(n+1) = T(n) + r_{heat} * \Delta T \tag{5.75}$$

$$T(n+1) = T(n) - r_{cool} * \Delta T \tag{5.76}$$

The boiling point and freeze point of the system can be described using batch processing in which the activity or rate of acceptance is evaluated or a given number of trials to determine the system state[46]. By doing this one can also set reasonable boiling and freezing points that are achievable with respect to the problem.

5.3.2 Application of Simulated Annealing

Equation 5.65 has shown that a change in mass will directly affect the power demand load. Since the amounts of stored energy can also be related in terms of mass it is reasonable to define the search parameter, $\vec{h}$, in terms of mass; where m_{fuel} is the

mass of the fuel and m_{ES} is the mass of the energy storage unit.

$$\vec{h} = \begin{bmatrix} m_{fuel} \\ m_{ES} \end{bmatrix} \tag{5.77}$$

The mass of the vehicle must be adjusted to reflect its new configuration. This entails mathematically removing the current propulsion system and installing the FPHV propulsion system by adjusting the mass of the vehicle. Equation 5.78 calculates the adjusted mass of the vehicle. The mass of the vehicle at curb weight, m_{curb}, represents the vehicle's mass with all driving fluids and components intact for driving. By removing the estimated mass for the internal combustion engine, m_{ice}, and the original fuel tank, m_{tank}, and then adding the mass of the FPHV system and the fuels from Equation 5.77, an estimate for the mass of the FPHV can be made.

$$m_{car} = m_{curb} - m_{ice} - m_{tank} + m_{fphv_{sys}} + m_{fuel} + m_{ES} \tag{5.78}$$

This study estimates that the internal combustion engine in both vehicles weighs 250 pounds with the Prius electric motor adding another 125 pounds to its curb weight, respectively. The mass of the original fuel tank can be determined from the specifications given in Tables 5.2 and 5.6.

5.3.3 Fitness Function Development

In order to evaluate the system for energy efficiency, the contributions that the FPG and the energy storage individually make towards supplying power to meet load demands must be assessed. The energy contributed by each element can be determined through the following steps. First, given a driving schedule and a vehicle whose mass has been adjusted to reflect the energy stored on-board, a schedule representing the power load demand of the vehicle while performing the schedule can be generated

using Equation 5.65. Using the assumption that the DC-link should maintain a constant operation voltage of $336V$, the load demand can be written as current using Equation 4.63. Since the efficiency of the motor at load is 86%, the power demand across the DC-link is increased by a factor of 1.14.

$$i_{Load}(t) = \frac{1.14 P_{Load}(t)}{336} \tag{5.79}$$

With the load current defined a discrete state space model can be written to represent energy transfer within the system using Equation 4.64. Now since the target voltage level of the DC-link is known an inverse problem can be set up using Euler's formula. It should be noted that the FPG is considered to be continuously running; thus Equation 5.80 holds for energy transferred to the DC-link from the FPG, as long as its voltage level is greater than the DC-link. Therefore, the value of Equation 5.80 is known and the power delivered by the FPG can be determined. The energy provided by the energy storage, which will balance the power to maintain the DC=link voltage at a given instance in time is calculated in Equation 5.81.

$$\frac{dv_{cFPG}(t)}{dt} = \frac{V_s(t)}{RC} - \frac{v_c(t)}{RC} \tag{5.80}$$

$$\frac{dv_{cES}(t)}{dt} = \frac{i_{Load}(t)}{C} - \frac{dv_{cFPG}(t)}{dt} - \frac{336}{\Delta t} \tag{5.81}$$

Using the voltage rates, power supplied by each of the energy sources can be determined with Equations 5.82 and 5.83 where C is the capacitance of the DC-link.

$$P_{FPG}(t) = C v_c(t) \frac{dv_{cFPG}(t)}{dt} \tag{5.82}$$

$$P_{ES}(t) = C v_c(t) \frac{dv_{cES}(t)}{dt} \tag{5.83}$$

Once the power supplied by each generator has been found, the energy supplied from each generator can be determined with Equation 5.84[25]. With the contribution of each energy source known, a ratio can be generated that represents how many times the generators can supply each contribution given the total amount of stored energy on-board. The amount fuel available in the source is adjusted by a factor, η, in order to account for the source's efficiency during the conversion process. Studies have shown that FPGs similar to the size being considered can reach up to 44% in energy convergence efficiency[47].

$$E = \sum P(t_i)\,\Delta t_i \tag{5.84}$$

$$\varepsilon = \frac{\eta E_{SourceTotal}}{E_{Drive}} \tag{5.85}$$

Once the ratios have been found the reciprocal of the smallest ratio is chosen as the system fitness grade as shown in Equation 5.86. The smallest ratio is chosen do to the co-dependency of the generation units when supplying power. If one source is depleted then the FPHV will not function. Based on this grading system, solutions which perform well will yield grades closer to zero.

$$\alpha = \frac{1}{\min(\varepsilon_{FPG}, \varepsilon_{ES})} \tag{5.86}$$

The current fitness function allows for the evaluation of only a single driving schedule. In order to perform comprehensive optimization, this study will utilize the FTP-75, HWFET, and the US06 in a weighted fitness function. Equation 5.87 displays the new composite grade that will be used to evaluate the system across the three driving schedules. The best solutions will still generate low scores when being evaluated. The overall goal of the optimization will be to minimize the grade of the solution across the three driving schedules.

$$\overline{A} = .4\alpha_{FTP-75} + .4\alpha_{HWFET} + .2\alpha_{US06} \tag{5.87}$$

5.3.4 System Constraints

Though the optimization algorithm has been developed there are some constraints that must be adhere to when evaluating solutions for the system. First, the entire mass of stored energy must remain under $205kg$. This is to ensure that optimization doesn't maximize the energy system at the complete expense of payload. Secondly, the current sent by the energy storage elements can not exceed the maximum current for the pack. Traditionally, the cells in the power pack are structured to meet specific current, voltage, and power requirements based on the energy that must be delivered; because this study is evaluating a variety of possible configurations, maximum current will have a generalized definition. Equation 5.88 represents the generalization of max current for this study in which the number of cells, N, multiplied by the maximum continuous current of an individual cell yields the maximum current for the pack.

$$i_{max} = N i_{max\,cont} \tag{5.88}$$

Current supplied from the energy storage at any given time is given by Equation 5.89. Solutions that fail to stay under the maximum current are discarded.

$$i_{ES}(t) = C\frac{dv_{cES}(t)}{dt} \tag{5.89}$$

5.3.5 Processing Constraints

The computationally expensive nature of simulated annealing requires that some sacrifices be made concerning complex modeling. For this reason once a voltage wave-

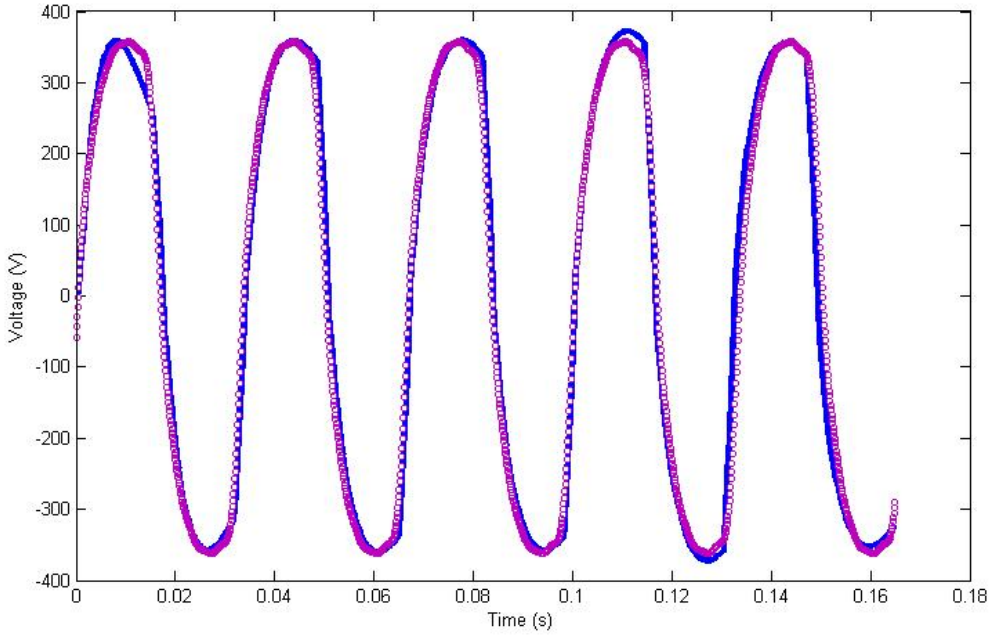


FIGURE 5.23: *The Fourier series model and the discrete state space model of a 25 kW free piston generator.*

form was generated using the FPG model described in Chapter 2.3, a Fourier series model of the waveform was generated in order to represent the FPG voltage. The Fourier series model initially contained five harmonics; but was expanded to use five odd harmonics when it was observed that the even harmonics were relatively small in comparison to the odd harmonics. Equation 5.90 represents a Fourier series model. In order to determine the coefficients for the model a Fourier matrix is developed for the time length of the voltage waveform with the chosen harmonics.

$$f(t) = a_0 + \sum_{n=0}^{k} a_{2n+1} \cos n\omega_0 t + b_{2n+1} \sin n\omega_0 t \tag{5.90}$$

$$F = \begin{bmatrix} 1 & \cos\omega_0 t_1 & \sin\omega_0 t_1 & \cdots & \sin 9\omega_0 t_1 & \cos 9\omega_0 t_1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & \cos\omega_0 t_n & \sin\omega_0 t_n & \cdots & \sin 9\omega_0 t_n & \cos 9\omega_0 t_n \end{bmatrix} \tag{5.91}$$

Using the Fourier matrix, Equation 5.92 is established. The coefficients can now be solved for by performing a pseudo-inverse on Equation 5.92 yielding Equation 5.93. The Fourier series model, shown in Figure 5.23, can now represent the FPG voltage waveform at any given instance in time.

$$V_{FPG}(t) = F(t) * w \tag{5.92}$$

$$w = (F^{\intercal}F)^{-1} F^{\intercal} V_{FPG} \tag{5.93}$$

Chapter 6
Results

6.1 Experimental Setup

The optimization described in Chapter 4.5 was performed using simulated annealing. A batch mode configuration was used when evaluating the state of the optimization algorithm before each temperature update. Each epoch in batch mode consisted of twenty-five potential solutions. The algorithm was deemed to be in a boiling state if the acceptance rate of solutions across an epoch was greater than 75% and in a freezing state if the rate was less than 20%. A total of sixteen configurations were passed through the simulated annealing for optimization. Each representing a unique combination from the vehicles, energy storage units, and hydrocarbon fuels given in Chapter 4.5. It should be noted that the optimization process only considers systems which utilize both power generators.

6.2 Energy Optimization Results

Tables 6.7 and 6.10 present the solutions obtained for all sixteen configurations. The first thing noticed from the solutions is that the mass chosen for the energy storage pack remains relatively consistent for both types of vehicles. This suggests that the energy contribution from the lithium ion battery might be uniform for vehicles of this particular size and fluctuations in the mass of hydrocarbon fuels have no bearing upon it.

TABLE 6.7: *Stored Energy Solutions for FPHV with a Lithium Ion Battery*

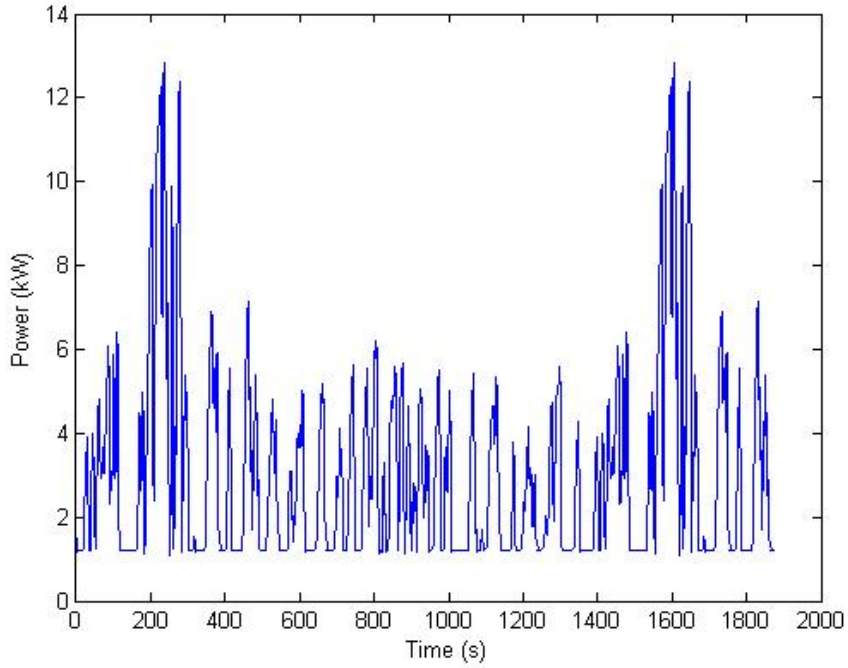
	Fuel Mass (kg)	**Battery Mass** (kg)
Gasoline	49.0	91.0
Diesel	56.2	90.8
Methanol	53.7	91.6
Ethanol	86.6	92.9

(A) *2013 VW Jetta*

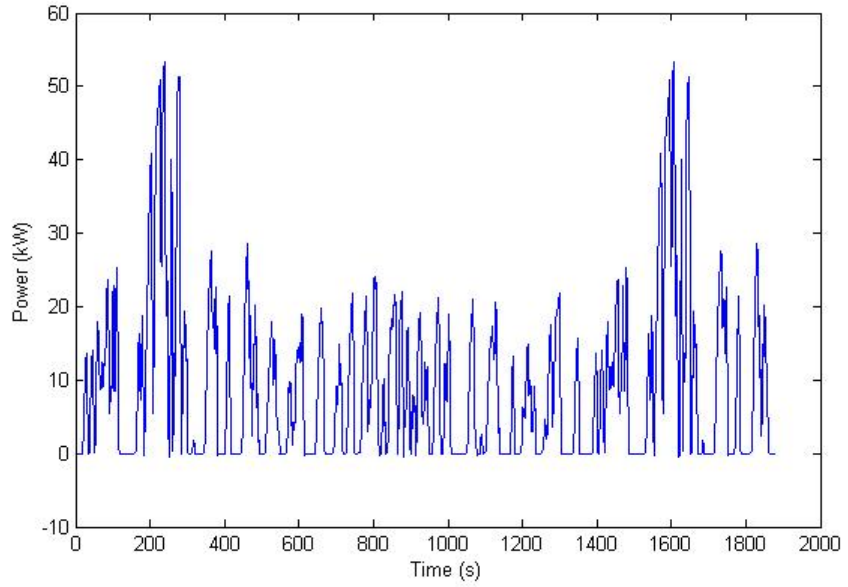
	Fuel Mass (kg)	**Battery Mass** (kg)
Gasoline	87.2	91.2
Diesel	70.6	91.4
Methanol	60.6	91.6
Ethanol	88.0	92.7

(B) *2014 Toyota Prius*

Based on this observation and how the hydrocarbon fuel levels fluctuate for the lithium ion vehicles, its probable that the lithium ion battery pack is serving as the primary generator in the vehicle. A quick look at the power supplied for one of the solutions utilizing a lithium ion battery confirms our theory. Figure 6.24 shows a comparison of the power supplied by the FPG and power supplied by the lithium battery. The magnitude of power delivered from the battery is four times greater than that of the FPG confirming that it is supplying the majority of energy for propulsion. Table 6.8 displays the average energy supplied from each generator across all configurations utilizing the lithium ion battery. The average energy supplied by the lithium battery is about four times greater than that of the energy supplied by the FPG; corroborating the previous observations. The table also shows an increase in the power supplied by the FPG across the FTP-75 driving schedule.



(A) *FPG using gasoline.*



(B) *Lithium ion battery.*

FIGURE 6.24: *Power supplied by generators for 2013 Jetta FPHV using lithium ion battery and gasoline as energy sources.*

TABLE 6.8: *Average Energy Supplied by Generator for FPHV with a Lithium Ion Battery*

	FPG (kWh)	**Battery** (kWh)
FTP-75	1.53	4.45
HWYFET	1.20	4.45
US06	0.97	3.55

(A) *2014 Toyota Prius*

	FPG (kWh)	**Battery** (kWh)
FTP-75	1.59	4.73
HWYFET	1.31	4.93
US06	1.06	3.80

(B) *2013 VW Jetta*

FTP-75 contains the slowest speeds of the driving schedule. Given the data it is suspected that the lithium ion battery has defaulted to serving as the primary power generation source do to its ability to quickly respond to the power load demand. The decrease of energy supplied by the FPG at high speeds reinforces this theory given that high speeds would result in the motor pulling current at a higher frequency.

With the battery acting as the primary generator it will also serve as the limiting factor concerning functionality for the vehicle. Table 6.9 relays the theoretical travel distance for each vehicle given the particular driving schedule profile. The travel distance is determined by taking the limiting energy ratio from Equation 5.85 and multiplying it by the distance traveled within the driving schedule.

$$Dist_{travel} = \min\left(\varepsilon_{FPG}, \varepsilon_{ES}\right) * Dist_{schedule} \tag{6.94}$$

The system's driving distance seems to typically fall between 40 and 50 miles for both vehicles. Seeing as the FPHV are carrying more hydrocarbon fuel than their ICE counterparts the conclusion can be made that the benefits of carrying the hydrocarbon fuel are not being fully realized.

Table 6.10 displays the solutions obtained for the FPHVs equipped with ultra-

TABLE 6.9: *Travel Distance for FPHV with a Lithium Ion Battery*

	FTP-75 (*mi*)	**HWYFET** (*mi*)	**US06** (*mi*)
Gasoline	52.2	48.5	47.5
Diesel	52.6	48.8	47.7
Methanol	52.9	49.3	48.3
Ethanol	52.6	48.8	47.7

(A) *2014 Toyota Prius*

	FTP-75 (*mi*)	**HWYFET** (*mi*)	**US06** (*mi*)
Gasoline	49.12	43.91	44.37
Diesel	48.90	43.81	44.21
Methanol	49.23	44.11	44.61
Ethanol	50.45	44.93	45.33

(B) *2013 VW Jetta*

TABLE 6.10: *Stored Energy Solutions for FPHV with an Ultra-capacitor*

	Fuel Mass (*kg*)	**Capacitor Mass** (*kg*)
Gasoline	65.5	93.9
Diesel	87.0	81.0
Methanol	87.0	81.0
Ethanol	69.7	71.2

(A) *2013 VW Jetta*

	Fuel Mass (*kg*)	**Capacitor Mass** (*kg*)
Gasoline	44.3	81.9
Diesel	44.5	94.5
Methanol	66.7	64.2
Ethanol	86.2	86.6

(B) *2014 Toyota Prius V*

capacitors. Based on observations from the vehicles using lithium ion batteries it is suspected that the FPHVs with ultra-capacitors would follow a similar pattern where the energy storage unit serves as the primary energy source. This is certainly the case as data will soon elude to this; however, one should take notice at the mass totals given for the ultra-capacitors. Considering the energy density of the ultra-capacitor relative to that of the lithium battery, the values shown in Table 6.10 for energy storage pack mass

TABLE 6.11: *Travel Distance for FPHV with an Ultra-capacitor*

	FTP-75 (*mi*)	**HWYFET (*mi*)**	**US06 (*mi*)**
Gasoline	1.32	1.12	1.12
Diesel	1.10	0.92	0.96
Methanol	1.10	0.92	0.96
Ethanol	0.88	0.82	0.88

(A) *2013 VW Jetta*

	FTP-75 (*mi*)	**HWYFET (*mi*)**	**US06 (*mi*)**
Gasoline	1.21	1.12	1.04
Diesel	1.32	1.23	1.20
Methanol	0.88	0.82	0.88
Ethanol	1.21	1.12	1.12

(B) *2014 Toyota Prius*

suggest that current ultra-capacitors might not be capable of performing as a primary power source for FPHVs.

Table 6.11 confirms that ultra-capacitors are not yet up to the task for being a primary power source in compact vehicles. A majority of the projected travel distances are only slightly over one mile; with some falling much shorter than that. None of the configurations are capable of completing their respective driving schedules; as a result there seem to be no viable ultra-capacitor solutions for a FPHV based on a compact vehicle design using given the constraints that were placed upon the system in this study's optimization process.

Given the results shown in Tables 6.8 and 6.9 the viable solutions obtained during the optimization process where re-evaluated with the operating frequency of the FPG adjusted to $60Hz$ and $120Hz$. This was done in order to observe how an increase in generator frequency for the FPG would affect the energy balance and performance of the system. Tables 6.12 and 6.13 display the average energy supplied by each energy source across the driving schedules.

TABLE 6.12: *Average Energy Supplied by Generators for FPHV with a Lithium Ion Battery and FPG Operating at 60Hz*

	FPG (kWh)	**Battery** (kWh)
FTP-75	2.70	2.39
HWYFET	2.12	2.52
US06	1.88	2.20

(A) *2014 Toyota Prius*

	FPG (kWh)	**Battery** (kWh)
FTP-75	2.79	2.60
HWYFET	2.39	2.89
US06	2.18	2.52

(B) *2013 VW Jetta*

TABLE 6.13: *Average Energy Supplied by Generators for FPHV with a Lithium Ion Battery and FPG Operating at 120Hz*

	FPG (kWh)	**Battery** (kWh)
FTP-75	3.52	1.54
HWYFET	2.17	1.45
US06	1.90	1.31

(A) *2014 Toyota Prius*

	FPG (kWh)	**Battery** (kWh)
FTP-75	3.63	1.65
HWYFET	2.40	1.68
US06	2.14	1.54

(B) *2013 VW Jetta*

When the operating frequency of the FPG is set to $60Hz$ the energy consumption is almost evenly distributed across both sources; however setting the frequency to $120Hz$ results in the FPG taking a majority share of the load and becoming the primary generator. The change in FPG operating frequency has had a noticeable effect on the energy supplied by each generation source. This effect also extends to the overall performance of the vehicle. Tables 6.14 and 6.15 display the theoretical travel distances for FPHVs with free-piston generators operating at $60Hz$ and $120Hz$. The FPHVs us-

TABLE 6.14: *Travel Distance for FPHV with a Lithium Ion Battery and FPG Operating at 60Hz*

	FTP-75 (*mi*)	**HWYFET** (*mi*)	**US06** (*mi*)
Gasoline	97.3	86.8	76.9
Diesel	97.5	87.0	77.1
Methanol	40.4	47.6	41.8
Ethanol	60.9	71.8	63.2

(A) *2014 Toyota Prius*

	FTP-75 (*mi*)	**HWYFET** (*mi*)	**US06** (*mi*)
Gasoline	89.6	74.7	67.6
Diesel	89.4	74.5	67.4
Methanol	53.9	58.5	49.9
Ethanol	50.3	54.5	46.5

(B) *2013 VW Jetta*

ing gasoline and diesel have doubled and tripled their travel distance across the driving schedules when operating at $60Hz$ and $120Hz$, respectively. The travel distance of the FPHVs using methanol and ethanol as a energy sources remained relatively unchanged at $60Hz$; but decreased at $120Hz$. The drastic changes in travel distance are likely attributed to the fuel density of the hydrocarbon fuel. Both gasoline and diesel have a considerably higher energy density when compared to methanol and ethanol. When the operating frequency of the FPG was increased the high energy density of the fuel became an important factor in determine overall system performance.

TABLE 6.15: *Travel Distance for FPHV with a Lithium Ion Battery and FPG Operating at 120Hz*

	FTP-75 (*mi*)	**HWYFET** (*mi*)	**US06** (*mi*)
Gasoline	151.6	149.6	129.4
Diesel	151.9	150.0	129.6
Methanol	31.1	46.4	41.8
Ethanol	47.0	70.2	63.2

(A) *2014 Toyota Prius*

	FTP-75 (*mi*)	**HWYFET** (*mi*)	**US06** (*mi*)
Gasoline	137.1	127.4	112.7
Diesel	136.7	127.1	112.5
Methanol	18.4	25.7	22.5
Ethanol	17.2	24.0	20.9

(B) *2013 VW Jetta*

Chapter 7
Conclusion and Future Work

7.1 Conclusion

The initial goal of the study looked to deliver an optimized free-piston hybrid vehicle that could take advantage of today's hydrocarbon infrastructure, along with the emerging electric vehicle industry, and serve as a transitioning technology between the two. Though the study was able to produce a solution for the FPHVs containing lithium ion based batteries, some of the glaring short-comings of today's technology were insurmountable. The energy density of the energy storage unit and the frequency of the power generated by the FPG were the two factors that had significant impacts during the optimization process for the FPHV.

The energy density of the ultra-capacitor is the factor that currently makes it a poor choice for energy storage in small passenger vehicles. The ultra-capacitors weight relative to that of the car placed it out of range for a viable solution. Until the ultra-capacitors energy density is at least comparable to the current lithium battery it can not serve as a viable energy storage solution within a compact vehicle. This problem also points out the main advantage hydrocarbon fuels and vehicles using internal combustion engines have over series hybrid and electric vehicles. Even though the energy conversion efficiency for internal combustion engine is relatively low, its use of hydrocarbons fuels as an energy source maintains the ICE's superior status amongst viable solutions for vehicle propulsion. This is due to the energy densities of the hydrocarbons fuels greatly exceeding those of all electrical energy sources currently available.

The second factor affecting the FPHV deals with the frequency of the power generated by the FPG. It was believed that the FPG would need some assistance from the energy storage to propel the vehicle; however the amount of assistance that the FPG needed was greatly underestimated. In order for it serve as an effective co-generation source the frequency of power generation for the FPG must be increased. This study has shown that when the FPG operates at low frequencies the FPHV becomes primarily dependent upon the energy storage unit. In order to utilize the full benefits of the hydrocarbon fuel it is carrying the FPG speed must be increased. Increasing the speed of the FPG to $60Hz$ and $120Hz$ resulted in drastic improvements in the energy usage concerning the energy storage and lengthened the vehicles travel distance; however care must be take in choosing the hydrocarbon fuel source as the increased speed consumes energy at a faster rate.

7.2 Future Work

The need for energy storage devices with greater energy density is a key area for all power consuming devices. The storage device that is able to contain large quantities of power relative to its mass would revolutionize the power industry and change the world.

The operating frequency of the FPG might prove to be the more approachable problem of the two mentioned. Motion control of the piston is a primary concern for the FPG. The design of the system makes it difficult to control at high speeds. A closer study of the physics surrounding the movement and power generation of the FPG could lead to a design and control mechanism that could increase the performance of the FPG to levels which may be more suitable for driving applications.

Bibliography

[1] "United states: All grades, areas and formulations," tech. rep., 1000 Independence Ave., SW Washington, DC 20585, April 2013.

[2] S. Golbluff, "Optimization of a plug-in hybrid vehicle," Master's thesis, Georgia Institute of Technology, 2006.

[3] Hansson, Analysis and Control of a Hybrid Vehicle Powered by a Free-Piston Energy Converter. PhD thesis, Royal Institute of Technology, 2006.

[4] M. Ortuzar, J. Dixon, and J. Moreno, "Design, construction, and performance of a buck-boost converter for an ultracapacitor-based auxiliary energy system for electric vehicles," 2003.

[5] Y. Chiang, "Building a better battery," Science, vol. 330, 2010.

[6] M. Duduta, B. Ho, V. Wood, P. Limthongkul, V. Brunini, W. Carter, and Y. Chiang, "Semi-solid lithium rechargeable flow battery," 2011.

[7] L. Guzzella and A. Sciarretta, Vehicle Propulsion Systems. 2005.

[8] "Environmental protection agency federal register," November 2009.

[9] M. Meyer, "Understanding the challenges in hev 5-cycle fuel economy calculations based on dynamometer test data," Master's thesis, Virginia Polytechnic Institute and State University, 2011.

[10] J. Hansson, "Minimizing power pulsations in a free piston energy converter," in 11th European Conference on Power Electronics and Applications, 2005.

[11] R. Mikalsen and A. Roskilly, "A review of free-piston engine history and applications," 2009.

[12] S. Baruch, "Power sources," 1959.

[13] Mikalsen and Roskilly, "The control of a free-piston engine generator. part 1: Fundamental analyses," Elsevier, 2010.

[14] B. Guru and H. Hiziroglu, Electromagnetic Field Theory Fundamentals. PWS Publishing Company, 1998.

[15] R. Serway and J. Faughn, College Physics. Thomson Brooks/Cole, 2003.

[16] D. Kondepudi, Introduction to Modern Thermodynamics. John Wley and Sons, 2008.

[17] Mikalsen and Roskilly, "The design and simulation of a two-stroke free-piston compression ignition engine for electrical power generation," Els, 2009.

[18] J. Mao, Z. Zuo, and H.Feng, Thermodynamics - Interaction Studies - Solids, Liquids and Gases, ch. Dimensionless Parametric Analysis of Spark Ignited Free-Piston Linear Alternator, pp. 271–299. InTech, 2011.

[19] T. C. de Melo, G. Machado, R. M. Jr, C. Belchior, and P. Pereira, "Thermodynamic modeling of compression, combustion and expansion processes of gasoline, ethanol and natural gas with experimental validation on a flexible fuel engine," 2007.

[20] J. Hansson, M. Leskell, F. Carlsson, and C. Sadarangani, "Operational strategies for a free piston energy converter," in 5th International Symposium on Linear Drives for Industry Applications, 2005.

[21] J. Holman, Heat Transfer. McGraw-Hill, 2002.

[22] W. Arshad, C. Sadarangani, T. Backstrom, and P. Thelin, "Finding an appropriate electrical machine for a free piston generator," 2002.

[23] I. Boldea, Variable Speed Generators. CRC Press, 2006.

[24] C. Chiang, J. Yang, S. Lan, T. Shei, W. Chang, and B. Chen, "Dynamic modeling of si/hcci free-piston engine generators," in 6th IEEE Conference on Industrial Electronics and Applications, 2011.

[25] B. Lathi, Signal Processing and Linear Systems. Berkeley-Cambridge Press, 1998.

[26] Arshad, Backstrom, Thelin, and Sadarangani, "Intergrated free-piston generators: An overview," 2002.

[27] A. Lidozzi and L. Solero, "Power balance control of multiple input dc-dc power converter for hybrid vehicles," 2004.

[28] S. Jalbrzykowski and T. Citko, "A bidirectional dc-dc converter for renewable energy systems," Bulletin of the Polish Academy of Sciences, vol. 57, pp. 363–368, 2009.

[29] D. Hart, Introduction to Power Electronics. Prentice Hall, 1997.

[30] S. Gargies, H. Wu, and C. Mi, "Isolated bi-directional dc-dc converter for hybrid electric vehicle application," 2006.

[31] D. Trowler and B. Whitaker, "Bi-directional inverter and energy storage system," tech. rep., University of Arkansas, 2008.

[32] J. Zhang, Bidirectional DC-DC Power Converter Design Optimization, Modeling and Control. PhD thesis, Virginia Polytechnic Institute and State University, 2008.

[33] A. Trzynadlowski, Control of Induction Motors. Academic Press, 2001.

[34] D. Zill and M. Cullen, Differential Equations with Boundary-Value Problems. Brooks/Cole, 1997.

[35] Toyota, "2013 toyota prius v," tech. rep., 2013.

[36] Volkswagen, "2014 jetta," tech. rep., 2013.

[37] "Ac-150 gen-2 ev power system," tech. rep., AC Propulsion, Inc.

[38] D. Carter and E. Wechner, "The free piston power pack: Sustainable power for hybrid electric vehicles," 2003.

[39] M. Technologies, "Datasheet: K2 series ultracapacitors," tech. rep., Maxwell Technologies.

[40] Panasonic, "Nnp series: Ncr18650:," tech. rep., Panasonic, 2010.

[41] D. Lide and H. Kehiaian, CRC Handbook of Thermophysical and Thermochemical Data. CRC Press, 1994.

[42] D. Henderson, S. Jacobson, and A. Johnson, "The theory and practice of simulated annealing," in Handbook of Metaheuristics, Springer US, 2003.

[43] S. Banerjee and N. Dutt, "Very fast simulated annealing for hw-sw partitioning," CECS Technical Report, 2004.

[44] S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," Science, vol. 220, pp. 671–680, 1983.

[45] E. Poupaert and Y. Deville, "Simulated annealing with estimated temperature," AI Communications, vol. 13, pp. 19–26, 2000.

[46] R. Eglese, "Simulated annealing: A tool for operational research," European Journal of Operational Research, vol. 46, pp. 271–281, 1990.

[47] T. Johnson, "Free-piston engine," tech. rep., Sandia National Laboratories, 2013.

Appendix

Matlab Code for the Simulation and Optimization of a FPHV

C:\Users\Kenneth Jones\Documents\Kens_Dissertation\code\FPG_SA\fpeg_25W_specs.m Monday, November 11, 2013 5:59 AM

```
fpeg.bore = .066; %m
fpeg.stroke_length = .103; %m
fpeg.compression_ratio = 9.8;
fpeg.mass = 5.2; %kg
fpeg.bore_area = pi * (fpeg.bore/2)^2; %(m^2)
fpeg.vol_min = (fpeg.bore_area*fpeg.stroke_length)/...
(fpeg.compression_ratio-1);
fpeg.vol_max = fpeg.bore_area*fpeg.stroke_length - fpeg.vol_min;
fpeg.density = 2707; %kg/m^3 Aluminum
fpeg.heat_conduct_const = 0.896; %kJ/(kg*C)
fpeg.thermal_C = fpeg.density * fpeg.heat_conduct_const; %kJ/C
fpeg.thermal_R = 1/(75 * fpeg.bore * fpeg.stroke_length * 1.5); %C/W
fpeg.back_emf_const = 75; %V/(mps)
fpeg.internal_R = 2.4404; %ohms 12
fpeg.thrust_const = 36.3;
```

```
function [state_vars,output_Vec,Output_Vec_Data,time,flag_complete]...
= fpeg_dynamicsSA(fpeg,fuel,state_vars,output_Vec,fire_select,...
fuel_mass,freq)
%fpeg_dynamics  Simulates  the physics of an free piston generator
%   Detailed explanation goes here
% fpeg_specs
% fpeg_state_vars_cal
% fuel = fuel_select(1);

%Fuel load parameters
mole_mass = fuel.mass;
Hc = fuel.Hc;
hf = fuel.hf;
hf_co = fuel.hf_co;  hf_h2o
= fuel.hf_h2o;  density =
fuel.density;  pct_co =
fuel.percent_co;  pct_h2o =
fuel.percent_h2o;
react_rate_parms  = fuel.react_rate_parms;
a2f_ratio = fuel.air2fuel_ratio;
concentration  = fuel.conc;
Cp = fuel.Cp;
Cv = fuel.Cv;
% burn_duration  = fuel.burn_duration;
burn_duration  = 0.0006;
wiebe_coeffs  = fuel.wiebe_coeff;
min_pressure  = 1.01325e5;
max_pressure  = 1e7;

%FPEG load parameters
therm_C = fpeg.thermal_C;
% therm_R = fpeg.thermal_R;
bore = fpeg.bore;
bore_area = fpeg.bore_area;
fpeg_mass = fpeg.mass;
emf_const = fpeg.back_emf_const;
thrust_const = fpeg.thrust_const;
vol_max = fpeg.vol_max;
vol_min = fpeg.vol_min;
gamma = Cp/Cv;
flag_complete = 0;
end_marker = 10000;


%System Resistance
system_R = fpeg.internal_R;

%Initialize Sensor Outputs
fuel_set = 0;

% m_1 = 0;
% m_2 = 0;

m_1 = fuel_mass;
```

```
m_2 = fuel_mass;

P_1 = output_Vec(1);
V_1 = output_Vec(2);
T_1 = output_Vec(3);
Twall_1 = output_Vec(4);

P_2 = output_Vec(5);
V_2 = output_Vec(6);
T_2 = output_Vec(7);
Twall_2 = output_Vec(8);

Volt = output_Vec(9);
Current = output_Vec(10);
x = output_Vec(11);
dx = state_vars(12);

dQ_1 = state_vars(1);
dP_1 = state_vars(2);
dV_1 = state_vars(3);
dT_1 = state_vars(4);
dTwall_1 = state_vars(5);

dQ_2 = state_vars(6);
dP_2 = state_vars(7);
dV_2 = state_vars(8);
dT_2 = state_vars(9);
dTwall_2 = state_vars(10);

dx2 = state_vars(11);
dx = state_vars(12);

Release_P1 = 0;
Release_P2 = 0;

beta_P1 = 1;
beta_P2 = 1;

mid_P1 = P_1 + (300e3 - P_1)/2;
mid_P2 = P_2 + (300e3 - P_2)/2;

beta_P1_flag = 1;
beta_P2_flag = 1;

decay_flag = 1;
dx_zero_flag = 1;

if (fire_select == 1)
end_position = (.5*fpeg.stroke_length)  - fpeg.vol_min/...
fpeg.bore_area;
init_Release_P1 = 0;
%     init_Release_P2 = (1.01325e5 - output_Vec(5))/(2/freq);
init_Release_P2 = 0;
else
```

```
end_position = -(.5*fpeg.stroke_length) + fpeg.vol_min/...
fpeg.bore_area;
%     init_Release_P1 = (1.01325e5 - output_Vec(1))/(2/freq);
init_Release_P1 = 0;
init_Release_P2 = 0;
end


%Time
delta_t = .0001;
time = 0:delta_t:1; %seconds

% %Desired Signal
% freq = 30;
% Desired = -cos (2*pi*freq*time);
DW_1 = zeros(length(time),1);
DW_2 = zeros(length(time),1);
for n = 1:length(time)
t = time(n);

if ((t == 0) && (fire_select == 1) && (fuel_set == 0) )
init_t = t;
fuel_set = 1;
m_1 = fuel_mass;
m_2 = fuel_mass;

xb = exp(-wiebe_coeffs(1)*((time(n:end) - init_t)/...
burn_duration).^(wiebe_coeffs(2) + 1));
DW_1(n:length(time)-1) = diff(1-xb);

elseif ((t == 0) && (fire_select == 2) && (fuel_set == 0) )
init_t = t;
fuel_set = 1;
m_1 = fuel_mass;
m_2 = fuel_mass;

xb = exp(-wiebe_coeffs(1)*((time(n:end) - init_t)/...
burn_duration).^(wiebe_coeffs(2) + 1));
DW_2(n:length(time)-1) = diff(1-xb);
end

%Chamber 1 Coefficients
%     gamma(1) = Cp/Cv;
h_wall(1) = heatcoeff_wall(V_1(n),P_1(n),T_1(n),dx(n));
h_1(n) = h_wall(1);
cyl_area(1) = 2*pi*bore_area + 4*V_1(n)/bore;


%Chamber 2 Coefficients
%     gamma(2) = Cp/Cv;
h_wall(2) = heatcoeff_wall(V_2(n),P_2(n),T_2(n),dx(n));
h_2(n) = h_wall(2);
cyl_area(2) = 2*pi*bore_area + 4*V_2(n)/bore;
```

```
%NaN/imag check for wall coefficient
if ((isnan(h_wall(1))) || (abs(imag(h_wall(1)))) > 0)
if (n == 1)
h_wall(1) = 1;
else
h_wall(1) = h_1(n-1);
h_1(n) = h_wall(1);
end
end

%NaN/imag check for wall coefficient
if ((isnan(h_wall(2))) || (abs(imag(h_wall(2)))) > 0)
if (n == 1)
h_wall(2) = 1;
else
h_wall(2) = h_2(n-1);
h_2(n) = h_wall(2);
end
end

%Thermal REsistance of System
therm_R(1) = 1/(V_1(n)*h_wall(1)*cyl_area(1));
therm_R(2) = 1/(V_2(n)*h_wall(2)*cyl_area(2));


%Pressure Release Valve
if (fire_select == 1)
end_position = (.5*fpeg.stroke_length) - ...
fpeg.vol_min/fpeg.bore_area;
Release_P1 = 0;
Release_P2 = -gamma*P_2(n)/V_2(n)*state_vars(8) - init_Release_P2;
else
end_position = -(.5*fpeg.stroke_length) + ...
fpeg.vol_min/fpeg.bore_area;
Release_P1 = -gamma*P_1(n)/V_1(n)*state_vars(3) - init_Release_P1;
Release_P2 = 0;
end

if beta_P1 < 0
beta_P1 = 0;
end

if beta_P2 < 0
beta_P2 = 0;
end


%First Chamber dynamic states (A)
state_Mat = [0,0,0,0,0,0,0,0,0,0,0,0;...          %dQ_1
(gamma - 1)/V_1(n), 0, -gamma*P_1(n)/V_1(n),  ....
0, 0, 0 ,0,0,0,0,0,0;...          %dP_1
0,0,0,0,0,0,0,0,0,0,0,bore_area;...         %dV_1
1/(m_1*Cv),0,(1-gamma)*T_1(n)/V_1(n),0,0,...
```

```
0, 0, 0, 0 ,0,0,0;...         %dT_1
0,0,0,0,0,0,0,0,0,0,0,0;...          %dTwall_1

%Second Chamber dynamic states (A)
0,0,0,0,0,0,0,0,0,0,0,0;...          %dQ_2
0,0,0,0,0,(gamma  - 1)/V_2(n),  0, ...
-gamma*P_2(n)/V_2(n),  0, 0, 0,0;...         %dP_2
0,0,0,0,0,0,0,0,0,0,0,-bore_area;...         %dV_2
0,0,0,0,0,1/(m_2*Cv),  0,(1-gamma)*T_2(n)/V_2(n),   0, 0, 0,0;...
0,0,0,0,0,0,0,0,0,0,0,0;...          %dTwall_2

%Speed and acceleration of piston
0,0,0,0,0,0,0,0,0,0,0,-(thrust_const*(emf_const/system_R))/...
fpeg_mass;...         %dx2
0,0,0,0,0,0,0,0,0,0,delta_t,1;...  %dx
];

input_Vec = [DW_1(n)*m_1*Hc/mole_mass  - V_1(n)*h_wall(1)...
*cyl_area(1)*(T_1(n)  - Cel_2_Kel(Twall_1(n)));...  %dQ_1
-Release_P1;...  %dP_1
0;...
0;...
(h_wall(1)*cyl_area(1)*(T_1(n)  - Cel_2_Kel(Twall_1(n)))/...
(therm_C  * V_1(n)) - (Cel_2_Kel(Twall_1(n))/therm_R(1))/...
(therm_C  * V_1(n)));... %dTwall_1
DW_2(n)*m_2*Hc/mole_mass  - V_2(n)*h_wall(2)*cyl_area(2)...
*(T_2(n) - Cel_2_Kel(Twall_2(n)));...  %dQ_2
-Release_P2;...   %dP_2
0;...
0;...
(h_wall(2)*cyl_area(2)*(T_2(n)  - Cel_2_Kel(Twall_2(n)))/...
(therm_C  * V_2(n)) - (Cel_2_Kel(Twall_2(n))/therm_R(2))/...
(therm_C  * V_2(n)));... %dTwall_2
(P_1(n)- P_2(n))*bore_area/fpeg_mass;... %dx2
0];

output_Mat = [0,beta_P1*delta_t,0,0,0,0,0,0,0,0,0;...  %P_1
0,0,delta_t,0,0,0,0,0,0,0,0;...  %V_1
0,0,0,delta_t,0,0,0,0,0,0,0;...  %T_1
0,0,0,0,delta_t,0,0,0,0,0,0;...  %Twall_1

0,0,0,0,0,0,beta_P2*delta_t,0,0,0,0;...  %P_2
0,0,0,0,0,0,0,delta_t,0,0,0;...  %V_2
0,0,0,0,0,0,0,0,delta_t,0,0;...  %T_2
0,0,0,0,0,0,0,0,0,delta_t,0;...  %Twall_2

0,0,0,0,0,0,0,0,0,0,emf_const;... %Voltage
0,0,0,0,0,0,0,0,0,0,emf_const/system_R;...    %Current
0,0,0,0,0,0,0,0,0,0,delta_t;... %x
];

if (fire_select == 1)
input_Vec(6) = 0;
else
```

```
input_Vec(1) = 0;
end

%State Variable Calculations
next_state_vars = state_Mat * state_vars + input_Vec;

%System Variable Constraints
%Heat Chamber 1 (dQ_1)
if state_vars(1) < 0
state_vars(1) = 0;
end
%Heat Chamber 2 (dQ_2)
if state_vars(6) < 0
state_vars(6) = 0;
end

%NaN/imag check for State Variables
if ((sum(isnan(next_state_vars)) > 0) || (sum(abs(imag(next_state_vars))) > 0))
disp('Stopping Loop for Variables')
flag_complete = -1;
break;
end

%Output Reset (Voltage,Current)
output_Vec(9) = 0;
output_Vec(10) = 0;



%Output Calculations
output_Vec = output_Mat * state_vars + output_Vec;

%Releases pressure in combustion chamber
if ((fire_select == 1) && (dP_1(n) < 0) && (.9*max(P_1) > P_1(n)) )
if (decay_flag == 1)
lambda = -log(P_2(n)/P_1(n))/(inv(freq)-t);
base = P_1(n);
init_t_decay = t;
decay_flag = 0;
end
state_vars(2) = -lambda*base*exp(-lambda*(t-init_t_decay));
output_Vec(1) = base*exp(-lambda*(t-init_t_decay));
elseif ((fire_select == 2) && (dP_2(n) < 0) && (.9*max(P_2) > P_2(n)))
if (decay_flag == 1)
lambda = -log(P_1(n)/P_2(n))/(inv(freq)-t);
base = P_2(n);
init_t_decay = t;
decay_flag = 0;
end
state_vars(7) = -lambda*base*exp(-lambda*(t-init_t_decay));
output_Vec(5) = base*exp(-lambda*(t-init_t_decay));
end

%Pressue Chamber 1 (P_1)
```

```
if output_Vec(1) < 1.01325e5
output_Vec(1) = 1.01325e5;
end

%Pressue Chamber 2 (P_2)
if output_Vec(5) < 1.01325e5
output_Vec(5) = 1.01325e5;
end

%Update State Variables
%Record State Varables for viewing
dQ_1(n+1) = next_state_vars(1);
dP_1(n+1) = next_state_vars(2);
dV_1(n+1) = next_state_vars(3);
dT_1(n+1) = next_state_vars(4);
dTwall_1(n+1) = next_state_vars(5);

dQ_2(n+1) = next_state_vars(6);
dP_2(n+1) = next_state_vars(7);
dV_2(n+1) = next_state_vars(8);
dT_2(n+1) = next_state_vars(9);
dTwall_2(n+1) = next_state_vars(10);

dx2(n+1) = next_state_vars(11);
dx(n+1) = next_state_vars(12);

%Update Sensor Outputs
%Record Outputs for viewing
P_1(n+1) = output_Vec(1);
V_1(n+1) = output_Vec(2);
T_1(n+1) = output_Vec(3);
Twall_1(n+1) = output_Vec(4);

P_2(n+1) = output_Vec(5);
V_2(n+1) = output_Vec(6);
T_2(n+1) = output_Vec(7);
Twall_2(n+1) = output_Vec(8);

Volt(n+1) = output_Vec(9);
Current(n+1) = output_Vec(10);
x(n+1) = output_Vec(11);

state_vars = next_state_vars;

if ((x(n) >= end_position) && (fire_select == 1))
end_marker = n;
flag_complete = 1;
break;
elseif ((x(n) <= end_position) && (fire_select == 2))
flag_complete = 1;
end_marker = n;
break;
end
end
```

```
if (flag_complete  == 1)
P_1  =  P_1(2:end_marker+1);
V_1  =  V_1(2:end_marker+1);
T_1 = T_1(2:end_marker+1);
Twall_1 = Twall_1(2:end_marker+1);

P_2  =  P_2(2:end_marker+1);
V_2  =  V_2(2:end_marker+1);
T_2 = T_2(2:end_marker+1);
Twall_2 = Twall_2(2:end_marker+1);

Volt = Volt(2:end_marker+1);
Current = Current(2:end_marker+1);
x = x(2:end_marker+1);
time = time(2:end_marker+1);
end

Output_Vec_Data  = [P_1(:), V_1(:), T_1(:), Twall_1(:),...
P_2(:), V_2(:), T_2(:), Twall_2(:), Volt(:), Current(:), x(:)];
```

```
function [FPG_energy, ES_energy, ES_max_current] = ...
Evaluate_System(freq,LoadCurrent,R)
%Simulated Annealing Parameters

FPG_model = importdata('Gasoline_Model.mat');

motor.poles = 4;
motor.power = 150e3; % Watts
motor.constant_V = 336; % Volts
motor.max_I = motor.power/motor.constant_V;  %Amps
max_switch_freq = .5*motor.poles*rpm2hz(12000);
max_power_switch_freq = .5*motor.poles*rpm2hz(8000);
motor.C = motor.max_I/(max_power_switch_freq*motor.constant_V);

%FPG Parameters
vs_FPG.voltage_peak = 350;
vs_FPG.freq = freq;
vs_FPG.power = 100e3;
vs_FPG.C = Generate_Capacitor(vs_FPG.power/...
(max_power_switch_freq*vs_FPG.voltage_peak^2)); %0.0545F
vs_FPG.R = vs_FPG.C.esr;

%DC Link Parameters
dc_link.C = Generate_Capacitor(.0033);
dc_link.L = .0185;
% dc_link.Rc = dc_link.C.esr;
dc_link.Rc = R;
dc_link.V_limit = motor.constant_V;

[doutput, output] = Power_Sim_DCLink(FPG_model,dc_link,vs_FPG,LoadCurrent);

FPG_power_supplied = doutput(:,1);
ES_power_supplied = doutput(:,2);
ES_current_supplied = output(:,2);

clear doutput output

% [converted_FPG_power_supplied]  = power_converter(FPG_power_supplied,freq);
% [converted_ES_power_supplied]  = power_converter(ES_power_supplied,freq);
FPG_power_supplied = decimate(FPG_power_supplied,1000);
ES_power_supplied = decimate(ES_power_supplied,1000);
ES_max_current = max(ES_current_supplied);

FPG_energy = sum(FPG_power_supplied)/3600;
ES_energy = sum(ES_power_supplied)/3600;
```

```
dQ_1 = 0;
dP_1 = 0;
dV_1 = 0;
dT_1 = 0;
dTwall_1 = 0;

m_1 = 0;
P_1 = 1.01325e5; %4e6; %1.01325e5 Pa == 1 atm
V_1 = fpeg.vol_min;
T_1 = 293; %Kelvin (room temp)
Twall_1 = 20; %Celsius (room temp)
dQ_2 = 0;
dP_2 = 0;
dV_2 = 0;
dT_2 = 0;
dTwall_2 = 0;

m_2 = 0;
P_2 = 1.01325e5; %4e6; %1.01325e5 Pa == 1 atm
V_2 = fpeg.vol_max;
T_2 = 293; %Kelvin (room temp)
Twall_2 = 20; %Celsius (room temp)
dx2 = 0;
dx = 0;
x = -(.5*fpeg.stroke_length  - fpeg.vol_min/fpeg.bore_area);
Volt = 0;
Current = 0;
state_vars   = [dQ_1; dP_1; dV_1; dT_1; dTwall_1;...
                dQ_2; dP_2; dV_2; dT_2; dTwall_2;...
                dx2; dx];

output_Vec = [P_1; V_1; T_1; Twall_1; P_2;...
V_2; T_2; Twall_2; Volt; Current; x];
```

```
function [fuel] = fuel_select(type)
switch (type) case
{'gas' ,1} fuel.type =
'gasoline';
fuel.mass = 114.23; %molecular mass (g)
fuel.Hc = 5470.1e3; %Heat of combustion (J/mol)
fuel.hf = (250.1/fuel.mass)*10^3;  %Heat of formation (KJ/g)
fuel.hf_co = (393.5/44.01)*10^3;  %Heat of formation (KJ/g)
fuel.hf_h2o = (241.8/18)*10^3;  %Heat of formation (KJ/g)
fuel.density = 720; %Fuel density (g/L)
fuel.react_rate_parms  = [4.6e11, 15.098, 0.25, 1.5]; %[A, E/R, m, n]
fuel.air2fuel_ratio  = 3.5017;  %ratio per gram
fuel.conc = @conc;  %Concentration of air & fuel (mol/L)
fuel.Cp = 187.4;
fuel.Cv = fuel.Cp - 20.8;
fuel.wiebe_coeff  = [2,1]; %[coeff exp]
fuel.burn_duration  = .0006; %seconds
fuel.percent_co  = 0.6849;
fuel.percent_h2o  = 0.3151;

case {'diesel',2}
fuel.type = 'diesel';
fuel.mass = 170.34; %molecular mass (g)
fuel.Hc = 8086.5e3; %Heat of combustion (J/mol)
fuel.hf = (300.9/fuel.mass)*10^3;  %Heat of formation (KJ/g)
fuel.hf_co = (393.5/44.01)*10^3;  %Heat of formation (KJ/g)
fuel.hf_h2o = (241.8/18)*10^3;  %Heat of formation (KJ/g)
fuel.density = 820; %Fuel density (g/L)
fuel.react_rate_parms  = [3.8e11, 15.098, 0.25, 1.5]; %[A, E/R, m, n]
fuel.air2fuel_ratio  = 2.9118;  %ratio per gram
fuel.conc = @conc;  %Concentration of air & fuel (mol/L)
fuel.Cp = 232.63;
fuel.Cv = fuel.Cp - 20.8;
fuel.wiebe_coeff  = [5,2]; %[coeff exp]
fuel.burn_duration  = .0125; %seconds
fuel.percent_co  = 0.6897;
fuel.percent_h2o  = 0.3103;

case {'methane',3}
fuel.type = 'methane';
fuel.mass = 16.04; %molecular mass (g)
fuel.Hc = 890.7e3; %Heat of combustion (J/mol)
fuel.hf = (77.4/fuel.mass)*10^3;  %Heat of formation (KJ/g)
fuel.hf_co = (393.5/44.01)*10^3;  %Heat of formation (KJ/g)
fuel.hf_h2o = (241.8/18)*10^3;  %Heat of formation (KJ/g)
fuel.density = .66; %Fuel density (g/L)
fuel.react_rate_parms  = [8.3e5, 15.098, -0.3, 1.3]; %[A, E/R, m, n]
fuel.air2fuel_ratio  = 4;  %ratio per gram
fuel.conc = @conc;  %Concentration of air & fuel (mol/L)
fuel.Cp = 35.67;
fuel.Cv = fuel.Cp - 20.8;
fuel.wiebe_coeff  = [5,2]; %[coeff exp]
fuel.burn_duration  = .0125; %seconds
fuel.percent_co  = 0.5501;
```

```
case {'methanol',4}
fuel.type = 'methanol';
fuel.mass = 32.04; %molecular mass (g)
fuel.Hc = 726e3; %Heat of combustion (J/mol)
fuel.hf = (239.1/fuel.mass)*10^3;  %Heat of formation (KJ/g)
fuel.hf_co = (393.5/44.01)*10^3;  %Heat of formation (KJ/g)
fuel.hf_h2o = (241.8/18)*10^3;  %Heat of formation (KJ/g)
fuel.density = 790; %Fuel density (g/L)
fuel.react_rate_parms  = [3.2e12, 15.098, 0.25, 1.25]; %[A, E/R, m,n]
fuel.air2fuel_ratio  = 1.5;  %ratio per gram
fuel.conc = @conc;  %Concentration of air & fuel (mol/L)
fuel.Cp = 44.07;
fuel.Cv = fuel.Cp - 20.8;
fuel.wiebe_coeff  = [5,2]; %[coeff exp]
fuel.burn_duration  = .0125; %seconds
fuel.percent_co  = 0.5501;
fuel.percent_h2o  = 0.4499;

case {'ethanol',5}
fuel.type = 'ethanol';
fuel.mass = 46.07; %molecular mass (g)
fuel.Hc = 1366.7e3; %Heat of combustion (J/mol)
fuel.hf = (277.7/fuel.mass)*10^3;  %Heat of formation (KJ/g)
fuel.hf_co = (393.5/44.01)*10^3;  %Heat of formation (KJ/g)
fuel.hf_h2o = (241.8/18)*10^3;  %Heat of formation (KJ/g)
fuel.density = 790; %Fuel density (g/L)
fuel.react_rate_parms  = [1.5e12, 15.098, 0.15, 1.6]; %[A, E/R, m,n]
fuel.air2fuel_ratio  = 2.08;  %ratio per gram
fuel.conc = @conc;  %Concentration of air & fuel (mol/L)
fuel.Cp = 65.59;
fuel.Cv = fuel.Cp - 20.8;
fuel.wiebe_coeff  = [2.3,3]; %[coeff exp]
fuel.burn_duration  = .0125; %seconds
fuel.percent_co  = 0.6198;
fuel.percent_h2o  = 0.3802;

otherwise
fuel.type = 'gasoline';
fuel.mass = 114.23; %molecular mass (g)
fuel.Hc = 5470.1e3; %Heat of combustion (J/mol)
fuel.hf = (250.1/fuel.mass)*10^3;  %Heat of formation (KJ/g)
fuel.hf_co = (393.5/44.01)*10^3;  %Heat of formation (KJ/g)
fuel.hf_h2o = (241.8/18)*10^3;  %Heat of formation (KJ/g)
fuel.density = 720; %Fuel density (g/L)
fuel.react_rate_parms  = [4.6e11, 15.098, 0.25, 1.5]; %[A, E/R, m,n]
fuel.air2fuel_ratio  = 3.5017;  %ratio per gram
fuel.conc = @conc;  %Concentration of air & fuel (mol/L)
fuel.Cp = 187.4;
fuel.Cv = fuel.Cp - 20.8;
fuel.wiebe_coeff  = [2,1]; %[coeff exp]
fuel.burn_duration  = .0006; %seconds
fuel.percent_co  = 0.6849;
```

```
end
end

function [ conc_air, conc_fuel ] = conc(air_mass,fuel_mass,fuel)
%Conc Summary: Calcualtes the concentration of fuel and air in a given
%mixture
vol_air = air_mass*.7; %Liters
vol_fuel = fuel_mass/fuel.density;  %Liters

conc_air = (air_mass/32)/(vol_air  + vol_fuel); %Molarity (mol/L)
conc_fuel = (fuel_mass/fuel.mass)/(vol_air  + vol_fuel);  %Molarity (mol/L)
end
```

```
clear all clc
fpeg_25W_specs
fuel = fuel_select(1);
freq_list = 30:5:60;
freq = freq_list(7);
desired_EMF = 350;
delta_t = 1/(2*freq);
stroke_length = fpeg.stroke_length;
fpeg.back_emf_const = desired_EMF/(2*stroke_length*freq);
tstart = tic;
[SA_score,fuel_amt,system_freq,flag_complete,Volt,Current]...
= Piston_Motion_DataGen(fpeg,fuel,freq);
tElapsed_sec = toc(tstart);
tElapsed_min = round(tElapsed_sec/60);
```

```
function [ out ] = heatcoeff_wall(V,P,T,speed  )
%HEATCOEFF_WALL  Determines the wall heating coeffieicient  for the engine
%   Inputs: V = volume (m^3)
%           P = pressure (pascals)
%           T = temperature (Kelvin)
%           speed = speed (m/s)

%Honenberg heat coeffecient
out = 130*V^-0.06 * (P/1e5)^0.8 * T^-0.4 * (abs(speed) + 1.4)^0.8;

end
```

```
function [LoadCurrent] = Load_Current(schedule,car,SA_mass)

load(schedule,'schedule_mph');

gravity = 9.80665; %m/s^2
d_air = 1.2041; %air density

samp_freq = 1000;

schedule_mph = interp(schedule_mph,samp_freq);

cr = car.rolling_coeff;
drag_coef = car.drag_coeff;
eff = car.gear_eff;
mass_car = car.car_mass + SA_mass; %kg
F_Area = car.frontal_area; %m^2
schedule_mps = mph2mps(schedule_mph);
Fr = cr * mass_car * gravity;
Fa = .5 * drag_coef * d_air *F_Area * (schedule_mps.^2);
schedule_acc = diff(schedule_mps);
schedule_acc = [schedule_acc;0];
schedule_power = (Fa + Fr).*schedule_mps/1000 + mass_car * schedule_acc...
.*schedule_mps/1000;

schedule_acc_ID = (schedule_acc > 0);
schedule_power2 = schedule_power.*schedule_acc_ID;
LoadCurrent = 1.14*schedule_power2/336 * 1e3;

smoothing_factor = 10;
LoadCurrent = smooth(LoadCurrent,smoothing_factor);
```

```
function [SA_score,fuel_amt,system_freq,flag_complete,Volt,Current]=...
Piston_Motion_DataGen(fpeg,fuel,freq)
fpeg_state_vars_cal

t = [];

P_1 = [];
V_1 = [];
T_1 = [];
Twall_1 = [];

P_2 = [];
V_2 = [];
T_2 = [];
Twall_2 = [];

Volt = [];
Current = [];
x = [];

dx = [];
dx2 = [];
beta = 10;

init_state = state_vars;
prev_state_vars = state_vars;
prev_output_Vec = output_Vec;
init_fuel = 1;

half_period = 1/(2*freq);
half_period_check = ones(10,1);

fire_select = 1;
fuel_amt(3) = 0;

for n = 1:3

flag_complete = 0;
fuel_mass = init_fuel;
state_vars = prev_state_vars;
beta = 10;

while ((flag_complete == 0) || ...
(abs(half_period - half_period_check(n)/half_period) > .03))

[state_vars,output_Vec,Output_Vec_Data,time,flag_complete]=...
fpeg_dynamicsSA(fpeg,fuel,state_vars,output_Vec,fire_select,fuel_mass,freq)
half_period_check(n) = time(end);

if (flag_complete == -1)
break;
elseif ((flag_complete == 0) || (half_period_check(n) >= half_period))
fuel_mass = fuel_mass + beta*1;
state_vars = prev_state_vars;
```

```
output_Vec = prev_output_Vec;
elseif (1/half_period_check(n) - 1/half_period >= 1)
fuel_mass = fuel_mass - beta*1;
state_vars = prev_state_vars;
output_Vec = prev_output_Vec;

beta = beta/2;
end
end

if (flag_complete == -1)
break;
end

Volt = [Volt; Output_Vec_Data(:,9)];
Current = [Current; Output_Vec_Data(:,10)];

prev_state_vars = [0;0;0;0;0;0;0;0;0;0;0;0;0;0];
prev_output_Vec = output_Vec;

fuel_amt(n) = fuel_mass;
init_fuel = .5 * fuel_mass;

if (fire_select == 1)
fire_select = 2;
else
fire_select = 1;
end

end

system_freq = 1/(mean(half_period_check));
Energy = system_freq*sum(abs(Volt.*Current))/5;
fuel_total = mean(fuel_amt);

SA_score = fuel_total/Energy;
```

```
Power_Sim_DCLink(FPG_model,dc_link,fpg_src_store,I_out)
%POWER_SIM Summary of this function goes here
%   Detailed explanation goes here

%Time scale
vector_length = length(I_out) + 1;
%delta_t = 1/120;
delta_t = 1e-3;
time = 0:length(I_out)-1;
time = delta_t*time;

%Load DC_link parameters
C_dc = dc_link.C.value;
L_dc = dc_link.L;
R_dc = dc_link.Rc;
V_DCLink_limit = 336;

%Load Source parameters
freq = fpg_src_store.freq;

%Initialize recording state variables
V_FPG_src(vector_length) = 0;
iL_FPG(vector_length) = 0;
% iL_ES(vector_length) = 0;
VC_DCLink(vector_length) = 0;
dVC_DCLink(vector_length) = 0;
% d2VC_DCLink(vector_length) = 0;

extra_energy = 0;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Initialize Systems States
dX_system = zeros(2,1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%System State Parameters Matrix
A = zeros(2,2);
A(1,2) = -1/(R_dc*C_dc);
A(2,1) = delta_t;
A(2,2) = 1;

%System Input Matrix
B = zeros(2,2);
B(1,1) = 1/(R_dc*C_dc);
B(2,2) = -delta_t/C_dc;

%System Output Matrix
C = zeros(2,2);
C(1,1) = 1;
C(2,2) = 1;



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
for n = 1:length(I_out)

V_FPG_src(n) = FPG_Wave(FPG_model.coeffs,time(n),freq);


%Update input parameters
U_system = [V_FPG_src(n); I_out(n)];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%State Updates for System
next_dX_system = A*dX_system + B*U_system;
X_system = C*dX_system;

if (next_dX_system(1) < 0)
next_dX_system(1) = 0;
end
%
if (next_dX_system(2) > V_DCLink_limit)
extra_energy = extra_energy +.5*C_dc*(V_DCLink_limit-next_dX_system(1))^2;
next_dX_system(2) = V_DCLink_limit;
end

%Update Output parameters
dVC_DCLink(n) = X_system(1);
VC_DCLink(n) = X_system(2);
iL_FPG(n) = C_dc*X_system(1)*(X_system(2) >= 0);

dX_system = next_dX_system;
end


V_FPG_src = V_FPG_src(1:length(I_out));
iL_FPG = iL_FPG(1:length(I_out));
VC_DCLink = VC_DCLink(1:length(I_out));
dVC_DCLink = dVC_DCLink(1:length(I_out));

iL_ES = C_dc*(V_DCLink_limit - VC_DCLink)/delta_t;
FPG_power_supplied = (iL_FPG.*VC_DCLink)/1000;
ES_power_supplied = (iL_ES.*VC_DCLink)/1000;

doutput = [FPG_power_supplied(:), ES_power_supplied(:)];
output = [iL_FPG(:), iL_ES(:), VC_DCLink(:)];

end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [ y_hat ] = FPG_Wave(coeffs,t,freq)
w = 2*pi*freq;
fseries_mat = [1, cos(w*t)', sin(w*t)', cos(3*w*t)', sin(3*w*t)',...
cos(5*w*t)', sin(5*w*t)',cos(7*w*t)', sin(7*w*t)',cos(9*w*t)',sin(9*w*t)'];
y_hat = abs(fseries_mat*coeffs);
end
```

```
function [ rate ] = rate_react(  air_mass,fuel_mass,temp,fuel)
%RATE_REACT Calculates the rate of reaction of a given amount of fuel about
%to be ignited
%Inputs:
%temp = temperature (Kelvin)
%air_mass = grams
%fuel_mass = grams

%Output
%rate (per second)
[ conc_air, conc_fuel ] = fuel.conc(air_mass,fuel_mass,fuel);
A = fuel.react_rate_parms(1);  E_R =
fuel.react_rate_parms(2);  exp_fuel =
fuel.react_rate_parms(3);  exp_air =
fuel.react_rate_parms(4);

rate = A*exp(-E_R/temp)*(fuel_mass.^exp_fuel)*(air_mass.^exp_air);
end
```

```
clc
%SA_CENTERS Performs simulated annealing to find
%the tuning variable for
%the non-linear proximal support vector machine

FTP_schedule = 'ftpdds_schedule.mat';
HWY_schedule = 'hwyftedds_schedule.mat';
US06_schedule = 'us06_schedule.mat';

gravity = 9.80665;
fuel_num = 5;
fuel = fuel_select(fuel_num);
Toyota_Prius
% VW_Jetta
fpeg_25W_specs
fpg_eff = .44;

tuning_var(1) = 30;     %freq
tuning_var(2) = 50;    %fuel_mass kg
tuning_var(3) = 75;    %energy_mass kg

%Heat of Combustion
Hc = fuel.Hc/1000;

tank_size = gal2Liter(car.gas_tank_volume); %liters
fuel_density = fuel.density;  %g/L
energy_mass_limit = lb2newt(450)/gravity;  %kg Jetta
% energy_mass_limit = lb2newt(250)/gravity;  %kg Prius


% %Panasonic Lithium Ion Cells
% %#18650
% cell_size = 3.6*2.9; %(V*Ah -> Wh)
% cell_mass = 45; %grams
% cell_max_current = .55; %A

%Maxwell Technologies Ultracapacitor
%BCAP3000
cell_size = 3.04; %(Wh)
cell_mass = 510; %grams
cell_max_current = 210; %A

old_cost = 1e16;

temp = .3;    %Initial Starting temperature

epochs = 25; %Number to run in batch while heating/cooling
%before doing
    %energy check

count = 1;

state = 'HEAT';
```

```
while ((temp > 0) && (count <= 12))

energy = 0;
for i = 1:epochs
%             temp
%Varies the tuning variable in the SVM
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%             tuning_var(1) = tuning_var(1) + randi([-1 1],1);    %freq
tuning_var(1) = 30;
tuning_var(2) = abs(tuning_var(2) + randi([-5 5],1)*rand(1));
%fuel_mass kg
tuning_var(3) = abs(tuning_var(3) + randi([-5 5],1)*rand(1));
%energy_mass kg

if ((tuning_var(1) > 60) || (tuning_var(1) < 30))
    tuning_var(1) = best_tuning_var(1);
end

if ((tuning_var(2) + tuning_var(3)) > energy_mass_limit)
    tuning_var(2) = best_tuning_var(2);
    tuning_var(3) = best_tuning_var(3);
end

if ((tuning_var(1) >= 30) && (tuning_var(1) < 34))
    R = .475;
elseif ((tuning_var(1) >= 34) && (tuning_var(1) < 39))
    R = .355;
elseif ((tuning_var(1) >= 39) && (tuning_var(1) < 50))
    R = .335;
else
    R = .235;
end

fuel_molecular_mass = 1000*tuning_var(2)/fuel_density; %grams
fuel_tank_size = (Hc*fuel_molecular_mass)/3600;  %kWh

energy_storage_mass = 1000*tuning_var(3);  %grams
energy_storage_size = (cell_size*energy_storage_mass/cell_mass)/1000; %kWh
energy_storage_max_current = cell_max_current*...
round(energy_storage_mass/cell_mass);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
SA_mass = tuning_var(2) + tuning_var(3);
[LoadCurrent_FTP85] = Load_Current(FTP_schedule,car,SA_mass);
[LoadCurrent_HWY] = Load_Current(HWY_schedule,car,SA_mass);
[LoadCurrent_US06] = Load_Current(US06_schedule,car,SA_mass);

temp_tracker(count) = temp;

%Computes performance cost for new tuning variable
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[FPG_energy, ES_energy, ES_max_current_FTP85] = ...
Evaluate_System(tuning_var(1),LoadCurrent_FTP85,R);
```

```
fuel_duration_FTP85  = (fpg_eff*fuel_tank_size)/FPG_energy;
energy_duration_FTP85  = energy_storage_size/ES_energy;

metric_FTP85  = 1/min([fuel_duration_FTP85,energy_duration_FTP85]);

[FPG_energy,  ES_energy,  ES_max_current_HWY]  = ...
Evaluate_System(tuning_var(1),LoadCurrent_HWY,R);
fuel_duration_HWY  = (fpg_eff*fuel_tank_size)/FPG_energy;
energy_duration_HWY  = energy_storage_size/ES_energy;

metric_HWY  = 1/min([fuel_duration_HWY,energy_duration_HWY]);

[FPG_energy,  ES_energy,  ES_max_current_US06]  = ...
Evaluate_System(tuning_var(1),LoadCurrent_US06,R);
fuel_duration_US06  = (fpg_eff*fuel_tank_size)/FPG_energy;
energy_duration_US06  = energy_storage_size/ES_energy;

metric_US06  = 1/min([fuel_duration_US06,energy_duration_US06]);

new_cost  = .4*metric_FTP85  + .4*metric_HWY  + .2*metric_US06;

if((ES_max_current_FTP85  > energy_storage_max_current)||...
(ES_max_current_HWY  > energy_storage_max_current)||...
(ES_max_current_US06  > energy_storage_max_current))
    new_cost  = 1e16;
end
d_cost  = new_cost  - old_cost;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Evaluates  centers  and  decides  which  set  to  keep
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if  (new_cost  <  1e16)
    if  (d_cost  <  0)
        best_tuning_var  =  tuning_var;
        old_cost  =  new_cost;
        energy  =  energy  +  1;
    elseif  (rand(1)  <  exp((-d_cost/temp)))
%                 exp((-d_cost/temp))
        best_tuning_var  =  tuning_var;
        old_cost  =  new_cost;
        energy  =  energy  +  1;
    end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

end

energy  =  energy/epochs
energy_tracker(count)  =  energy;

if  (energy  >  .75)  &&  (strcmp(state,'HEAT'))
state  =  'COOL'
end
```

```
if ((energy < .2) && (strcmp(state,'COOL')))  || (temp <= .0001)
break
end

temp = temperature_update(temp,state);
count = count + 1;
end

figure(1),plot(energy_tracker),ylabel('Magnitude'),...
title('Energy  Tracker')
figure(2),plot(temp_tracker),ylabel('Magnitude'),...
title('Temperature  Tracker')
```

```
function new_T = temperature_update(T,state)
wf = .8;    %warming factor
cf = .2;    %cooling factor
T_step = .05;   %step factor for temperature

    switch state
        case 'HEAT'
            new_T = T + wf*T_step;
        case 'COOL'
            new_T = T - cf*T_step;
    end
end
```

```
avg_person_weight  = 195;
persons = 1;
gasoline_density  = 6.073; %lbs/gal

%FPG System
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
FPG_powerpack_mass  = 100; %kg
ACProp_motor_mass  = 50; %kg
ACProp_peu_mass  = 30; %kg

%Car Constants
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
car.type = 'Toyota Prius';
car.curb_weight  = 3042;
car.engine_weight_est  = 250 + 125; %lbs (ICE/AC)
car.est_curb_weight  = car.curb_weight  - car.engine_weight_est;
car.rolling_coeff  = .015;
car.gas_tank_volume  = 11.9; %US Gallons
car.drag_coeff  = 0.25;
car.tire_radius  = inch2m(7.5);
car.car_mass_old  = lb2newt(car.est_curb_weight  - ...
(gasoline_density*car.gas_tank_volume) + persons  * avg_person_weight)/...
gravity; %kgrams
car.car_mass  = car.car_mass_old  + FPG_powerpack_mass  + ACProp_motor_mass...
+ ACProp_peu_mass;
car.frontal_area  = 22.4/10.764;  %m^2

%Gearbox
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
car.gear_ratio  = [6.5,5.9,4.1,3.7];
car.final_gear_ratio  = 1.23;
car.gear_eff  = 0.95;
```

```
avg_person_weight  = 195;  % in pounds
persons = 1;
gasoline_density  = 6.073; %lbs/gal

%FPG System
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
FPG_powerpack_mass  = 100; %kg
ACProp_motor_mass  = 50; %kg
ACProp_peu_mass  = 30; %kg


%Car Constants
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
car.type  = 'VW Jetta SEL 2011';
car.curb_weight  = 3018; % in pounds
car.engine_weight_est  = 250; %lbs
car.est_curb_weight  = car.curb_weight  - car.engine_weight_est;
car.gas_tank_volume  = 14.5; %US gallons
car.rolling_coeff  = .015;
car.drag_coeff  = 0.3;
car.frontal_area  = inch2m(70)*inch2m(57.2);  %Estimate
car.tire_radius  = inch2m(8.5);
car.car_mass_old  = lb2newt(car.est_curb_weight  -...
(gasoline_density*car.gas_tank_volume)...
+ persons  * avg_person_weight)/gravity;  %kgrams
car.car_mass  = car.car_mass_old  + FPG_powerpack_mass  +...
ACProp_motor_mass  + ACProp_peu_mass;
%Gearbox
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
car.gear_ratio  = [3.77,2.11,1.26,0.86,0.66];
car.final_gear_ratio  = 3.389;
car.gear_eff  = 0.95;

% ACProp_Engine_module
```