Master of Science Thesis in Electrical Engineering
Department of Electrical Engineering, Linköping University, 2020

Hybrid Vehicle Control Benchmark

Ruchit Bhikadiya

Master of Science Thesis in Electrical Engineering

Hybrid Vehicle Control Benchmark:

Ruchit Bhikadiya

LiTH-ISY-EX–20/5349–SE

Supervisor: **Rafael Klüppel Smijtink**
Volvo Group
Kristoffer Ekberg
ISY, Linköping university

Examiner: **Lars Eriksson**
ISY, Linköping university

Division of Vehicular Systems
Department of Electrical Engineering
Linköping University
SE-581 83 Linköping, Sweden

Abstract

The new emission regulations for new trucks was made to decrease the CO_2 emissions by 30% from 2020 to 2030. One of the solutions is hybridizing the truck powertrain with 48V or 600V that can recover brake energy with electrical machines and batteries. The control of this hybrid powertrain is key to increase fuel efficiency. The idea behind this approach is to combine two different power sources, an internal combustion engine and a battery driven electric machine, and use both to provide tractive forces to the vehicle. This approach requires a HEV controller to operate the power flow within the systems.

The HEV controller is the key to maximize fuel savings which contains an energy management strategy. It uses the knowledge of the road profile ahead by GPS and maps, and strongly interacts with the control of the cruise speed, automated gear shifts, powertrain modes and state of charge. In this master thesis, the dynamic programming strategy is used as predictive energy management for hybrid electric truck in forward- facing simulation environment. An analysis of predictive energy management is thus done for receding and full horizon length on flat and hilly drive cycle, where fuel consumption and recuperation energy will be regarded as the primary factor. An another important factor to consider is the powertrain mode of the vehicle with different penalty values. The result from horizon study indicates that the long receding horizon length has a benefit to store more recuperative energy. The fuel consumption is decreased for all drive cycle in the comparison with existing Volvo's strategy.

Acknowledgments

First, I would like to thank my examiner Prof. Lars Eriksson for accepting my master thesis at the Vehicular Systems division, Linköping university. A special thank to my supervisor Kristoffer Ekberg, for his help, feedback and suggestions on improving this thesis.

At Volvo group, I am grateful to my supervisor, Rafael Klüppel Smijtink for giving me an opportunity to perform this thesis at Volvo group, and also for his infinite help and interesting discussion throughout the thesis. Lastly, I would like to thank all team members at Volvo group.

Linköping, November 2020
Ruchit Bhikadiya

Contents

Notation

General notations

Variable	Representing
x_{tot}	Total traveled distance [m]
t	Time [s]
s	Position [m]
E	Kinetic energy [J]
$P_{auxiliary}$	Auxiliaries consumed power [W]
E_{fuel}	Fuel energy [J]
P_{fuel}]	Fuel power [W
F_{fuel}	Fuel force [N]
Q_{LHV}	Lower heating value [J/g]

Vehicle script model notations

Variable	Representing
$m_{total,s}$	Total vehicle mass including all inertia [kg]
$v_{vehicle,s}$	Vehicle speed [m/s]
$s_{vehicle,s}$	Vehicle distance [m]
$F_{t,s}$	Total force at wheel [N]
$F_{b,s}$	Brake force at wheel [N]
$F_{a,s}$	Aerodynamics resistance force [N]
$F_{r,s}$	Rolling resistance force [N]
$F_{g,s}$	Gravitational force [N]
$\omega_{wheel,s}$	Angular speed at wheel [rad/s]
ρ_{air}	Air density [kg/m^3]
A_f	Vehicle frontal area [m^2]
c_d	Air drag coefficient [-]
g	Gravitational acceleration [m/s^2]
α	Road grade [rad]
c_r	Rolling resistance coefficient [-]
$i_{finaldrive}$	Final drive ratio [-]

Engine script model notations

Variable	Representing
$F_{ice,s}$	Engine force at the wheel $[N]$
$P_{ice,s}$	Engine power $[W]$
$T_{ice,s}$	Engine torque $[Nm]$
$\omega_{ice,s}$	Engine speed at transmission's input shaft $[rad/s]$
ρ_{diesel}	Density of diesel $[kg/m^3]$
n_{cyl}	Number of cylinder [-]
n_r	Number of crankshafts [-]

Machine script model notations

Variable	Representing
$F_{em,s}$	Machine force at the wheel $[N]$
$P_{em,s}$	Machine power $[W]$
$T_{em,s}$	Machine torque $[Nm]$
$\omega_{em,s}$	Engine speed at transmission's input shaft $[rad/s]$
$P_{motorloss}$	Power loss in machine $[N]$
$i_{reduction_{em}}$	Reduction ratio of electric machine [-]

Battery script model notations

Variable	Representing
SoC	State of charge [%]
η_{coul}	Battery coulombic efficiency [-]
Q_{Ah}	Battery charge capacity $[Ah]$
$I_{batt,s}$	Circuit current $[A]$
R_i	Circuit resistance $[ohm]$
$V_{batt,s}$	Circuit voltage $[V]$
$V_{oc,s}$	Open circuit voltage $[V]$
$P_{batt,s}$	Battery power $[W]$

Transmission script model notations

Variable	Representing
$T_{GB,s}$	Torque at transmission's output shaft $[Nm]$
$\omega_{GB,s}$	Angular speed at transmission's output shaft $[rad/s]$
$\eta_{GB_{ice,s}}$	Transmission efficiency of engine [-]
$\eta_{GB_{em,s}}$	Transmission efficiency of machine [-]
i_{ice}	Gear ratio from engine to wheel [-]
i_{em}	Gear ratio from machine to wheel [-]
r_w	Radius of wheel $[m]$

HEV controller- dynamic programming

Variable	Representing
$x_{k,c}$	Continuous state
$x_{k,d}$	Discontinuous state
$u_{k,c}$	Continuous control
$u_{k,c}$	Discontinuous control
k	Stage
g_k	final Step cost
$g_{k,t}$	Transition mode cost
$g_{k,s}$	Transition mode remaining cost
F_{ice}	Engine force at the wheel $[N]$
T_{ice}	Engine torque $[Nm]$
F_{em}	Machine force at the wheel $[N]$
T_{em}	Machine torque $[Nm]$
P_{em}	Machine power $[W]$
F_t	Total force at wheel $[N]$
I_{batt}	Circuit current $[A]$
P_{fuel}	Fuel power $[J/s]$
n_{cyl}	Number of cylinder [-]
n_r	Number of crankshafts [-]

Abbreviations

Abbreviation	Full form
HEV	Hybrid electric vehicle
BEV	Battery electric vehicle
ICE	Internal combustion engine
EM	Electric machine
APU	Auxiliary power unit
EMS	Energy management strategy
DP	Dynamic programming
ECMS	Equivalent consumption minimization strategy
RB	Rule-based
A-ECMS	Adaptive equivalent consumption minimization strategy
PI	Proportional–integral
FC	Fuel consumption
PT-modes	Powertrain modes

1 Introduction

The increased amount of greenhouse gases in the environment has enhanced the need for a new regulation in the commercial vehicle segment. This regulation demands for a reduced emission of CO_2 by 30% by 2030 [1]. There are many alternative solutions that are fuel cell vehicles, battery electric vehicles, hybrid electric vehicles and alternative fuels such as bio-diesel. In this thesis, only hybrid electric vehicle will be covered. The HEV truck is powered by both internal combustion engine and electric machine which uses electric energy from the battery.

1.1 Purpose and goal

The purpose of this master thesis is to improve a dynamic programming based controller which is used for hybrid electric vehicle benchmarking, by implementing different modes of vehicle and sensitivity analysis of the additional constraints. The thesis main objective is to control the state of charge, powertrain mode, torque split, velocity and gear-shifting on a given driving cycle [1] which is treated as input that consists velocity and altitude. Through a literature review, the impact of different energy management strategies such as rule-based, dynamic programming, predictive control management, and equivalent-consumption minimization strategies shall be investigated on the basis of fuel economy and recuperation energy, and a suitable one is to be selected. The goal is to improve and implement the dynamic programming platform to enable better benchmarking and sizing of the powertrain in order to minimize fuel consumption and achieving full recuperative (braking) energy with charge sustaining ability.

[1] Driving cycle is provided by Volvo and it's a measured cycle on specific route.

1.2 Hybrid electric vehicle

Hybrid electric vehicle's powertrain is combination of an engine and an electric machine. In general, an engine works as fuel converter or irreversible prime mover. Electric prime movers contain electric machine which could function both as a motor and generator. Electro-chemical battery is use to store electric energy and attach with electric machine. One of the main advantages for developing HEV powertrain for trucks are to recuperate energy during deceleration and to drive truck in purely electric mode that gives zero real-time emissions. Hybrid-electric vehicles are classified into three main types [8].

- Parallel hybrid
- Series hybrid
- Series-parallel, or combined hybrid

1.2.1 Parallel hybrid

Parallel hybrid electric vehicle (HEV) contains both an internal combustion engine (ICE) and an electric machine (EM) which can supply the traction power either alone or in combination. There are two energy sources, fuel tank and battery connected to the engine and electric machine respectively. The Electric machine works as a motor to provide torque to the gearbox by means of a clutch and also as a generator to store recuperative energy during deceleration from wheels during braking. The engine is mechanically linked to the drive train. Typically, The ICE can be turned off during low power demand and the vehicle operates with pure electric drive. Also, during high power demand, both engine and electric machine works at the same time, which is referred to as hybrid mode.

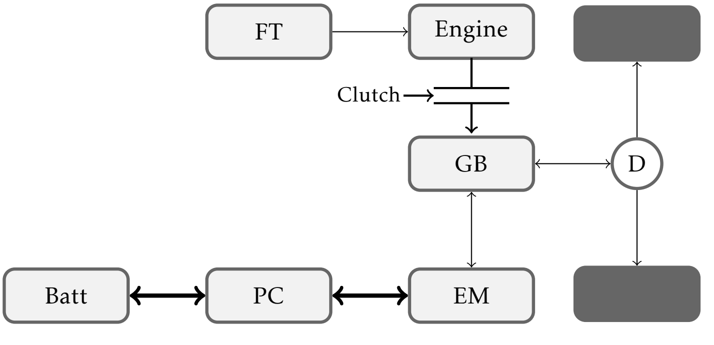


Figure 1.1: *Configuration of Parallel Hybrid. FT: fuel tank, GB: gear box, D: differential, Batt: battery, PC: power converter, EM: electric machine, parallel two lines: clutch. Bold lines: electrical link, solid lines: mechanical link. Double side arrow shows regenerative braking path.*

The mild hybrid concept is most simple parallel hybridization which contains an ICE engine powertrain with a low voltage electric motor. The mild hybrid's essential prime mover is IC engine; its battery does not need high energy storage capacity since the main role is only automatic engine stop-and-start. Parallel HEV configuration is widely used in the heavy trucks. During high power demand, both engine and electric machine are necessary to be operated in to order to fulfill drive power request. For heavy-trucks, when traveling on down-hill slope braking energy assist the truck to maintain a constant speed and that braking energy can be stored in the battery.

1.2.2 Series hybrid

Series hybrid powertrain configuration uses the IC engine as an auxiliary power unit (APU) to provide extra driving range of a battery-powered electric vehicle. The battery and generator are both connected to electric machines which provides power to the wheel. The engine drives a generator, producing electrical power that adds to the electrical power coming from the energy storage system; i.e. battery and then transmitted to the electric machine. The power produced by the generator can be used to to charge battery. In regenerative braking case, energy is stored directly into the battery using the electric machine as generator. Moreover, the power requirement of vehicle is not related to the engine operation and generates an additional degree of freedom, thus the engine can be operated at high efficiencies and less emissions. The IC engine is mechanically decoupled from the drive axle, so series hybrid configuration does not need clutch and transmission. On the other hand, series hybrid configuration contains two energy conversion; i.e., from electrical to mechanical in the machine and from mechanical to electrical in the generator, which results in a loss of efficiency. Series HEV is very useful into stop-go driving or city driving, but in some cases, a series hybrid electric vehicle consumes more fuel than a conventional vehicle, especially in highway driving.

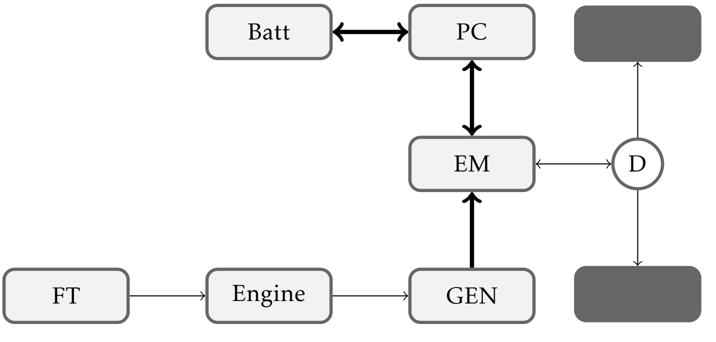


Figure 1.2: *Configuration of series hybrid. FT: fuel tank, D: differential, Batt: battery, PC: power converter, EM: electric machine, GEN: generator. Bold lines: electrical link, solid lines: mechanical link. Double side arrow shows regenerative braking path.*

1.2.3 Series-parallel, or combined hybrid

The combined HEV shown in Figure 1.3, is combination of both parallel and series hybrid concepts that contains one engine and two electric motors where one acts as a motor (EM) and other acts as a generator (GEN). As in a parallel HEV, motor and engine can provide power to the wheels in collaboration. The other motor, a generator can re-charge the battery via the engine or regenerative braking. A power-split device (PSD), which contains a planetary gear set.

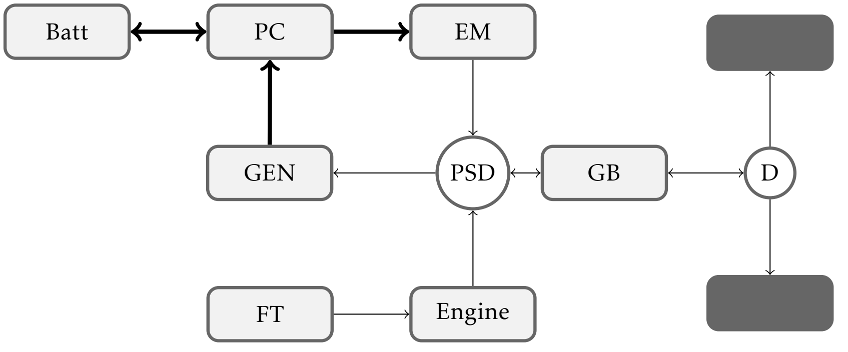


__Figure 1.3:__ Configuration of series-parallel, or combined hybrid. FT: fuel tank, D: differential, GB: gear-box, Batt: battery, PC: power converter, EM: electric machine, GEN: generator. Bold lines: electrical link, solid lines: mechanical link. Double side arrow shows regenerative braking path.

1.3 HEV supervisory control

In all types of HEVs, a supervisory controller is necessary to execute several tasks in order to fulfill requirements of driver and vehicle components with their status, and provides the best solution to the vehicle components through energy management strategy. Besides the consideration of component limits like maximum power or maximum temperature, its main goal is generally to achieve a lower energy consumption of the vehicle [8]. In contrast to conventional vehicles, energy management strategies of HEVs are more complex due to a higher degrees of freedom and constraints. The energy management strategy (EMS) is the part of a HEV controller which considers some input signals and feedback signals such as desired vehicle speed and/or driver request and based on the value of input signals decides output signals like torque for ICE and EM.

A parallel HEV can be operated in following powertrain modes:

- Conventional IC engine drive mode
- Hybrid drive mode [2]
- Pure electric drive mode
- Open driveline with both the engine and machine turned off

Energy management strategy determines the amount of energy taken from and send to the electric battery that can store energy for each vehicle driving condition [8]. Generally, it is classified into main two types; i.e. Heuristic based and Advanced control. More details about the approach are explained in section 2.

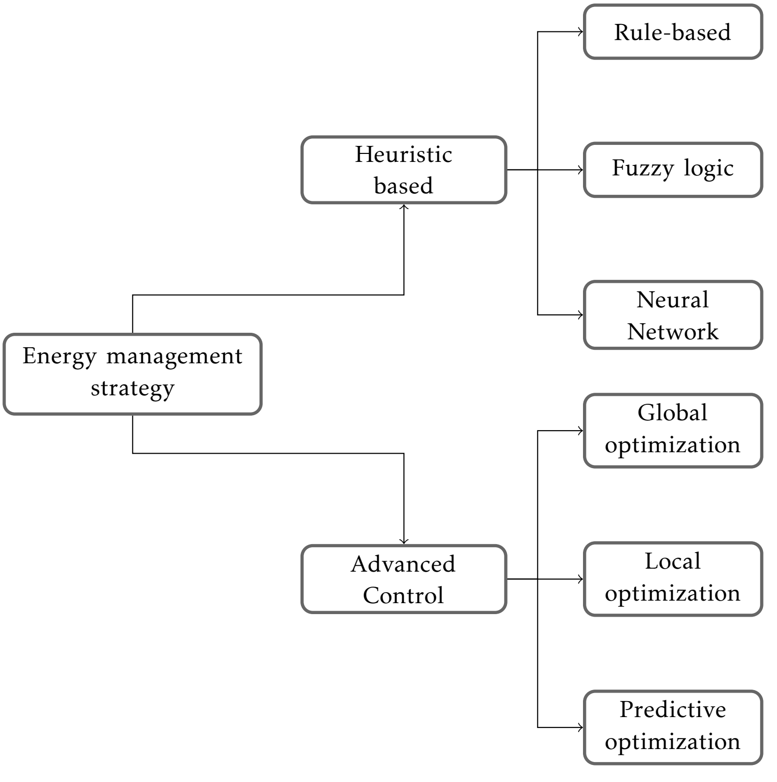


Figure 1.4: *Different energy management strategies.*

[2]In hybrid drive mode, engine and electric machine both operate at the same time and deliver power to the wheel.

1.4 HEV modelling concepts

In the context of vehicle powertrain simulation, there are two types modelling concepts are categorized as forward-facing and backward-facing model. They indicates conceptual direction for data flowing from input to output through given modelled system [8]. Both forward-facing and backward-facing modelling concepts consist powertrain components environment which can be described by model based or script based. However, script based environment is used in this thesis.

Forward-facing models represent correct causal nature of real-word events for dynamic models. A forward-facing model contains driver model which sends desired torque and braking torque to the HEV controller in an effort to follow the desired speed from driving cycle as closely as possible. A basic driver model uses one or more PI-control to achieve torque demand with desired reference speed and then transmit the commands to the HEV controller. Furthermore, the HEV controller contains energy management strategy that distributes torque demand to the engine and electric machine.

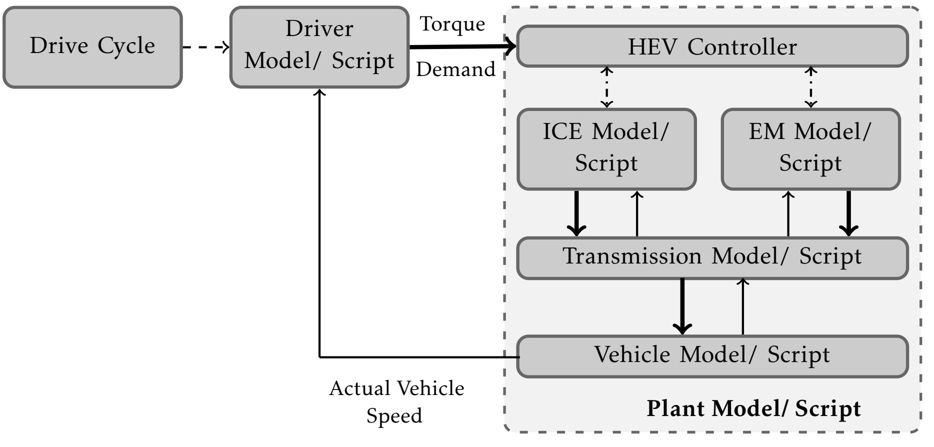


Figure 1.5: *Forward-facing vehicle script model. Bold lines and thin lines represent torque flow and speed flow between corresponding models respectively. Dashed line shows data transfer between components.*

The torque produced by the ICE and EM propagate through transmission and drive-line before ending up as torque applied at wheels. The plant model is shown in Figure 1.5 which consists HEV controller, ICE and EM model, transmission model and vehicle model. The vehicle speed which results from the applied torque at wheel is propagated through the drive-train, transmission, and returns to the ICE and EM as angular velocity. The torque and speed are used to determine power inputs and outputs of the components, resulting energy and fuel consumption. In addition, Forward-facing models provide recognition to the vehicle model drivebility and the limits of the physical system is taken into consideration. Typically, it is used for control system development that provides link between driver torque request and the powertrain components.

In a backward-facing model, see Figure 1.6, the main principle assumption is that the vehicle model exactly follows the demand from the drive cycle. Through speed from drive cycle, torque at the wheel is determined and propagated back to the powertrain through the drive-shaft and transmission, along with angular velocity. Based on powertrain torque and speed, resulting energy and fuel consumption can be determined. The outputs torque and speed are constrained by the drive cycle that becomes a backward-facing model as acausal and can not be use in realistic control system. However, it is useful for determining operating trends and performing analysis of powertrain under different conditions.

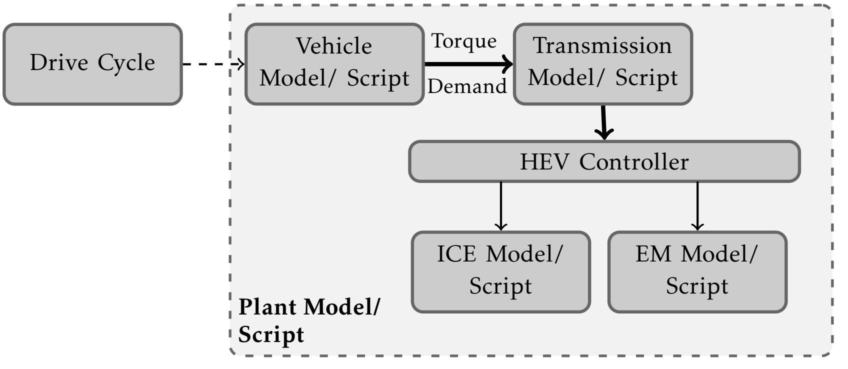


Figure 1.6: *Backward-facing vehicle model. Bold lines represents torque flow and speed flow together between corresponding models. Dashed line shows data transfer between components.*

1.5 Problem formulation

The current simulation platform being used in Volvo, uses a forward modeling approach as shown in Figure 1.5. The tools used for the simulation in the thesis are Matlab Simulink [3]. The HEV controller which contains predictive energy management with feedback controller that is optimized engine torque and machine torque in order to minimize fuel consumption. In order to set benchmark for Volvo's strategy, the different optimal control strategies are analyzed on the basis of fuel economy with different modes of the vehicle. The dynamic programming was given by Volvo and uses as predictive control in this thesis for the hybrid electric vehicle benchmarking which takes several powertrain modes into account. The major problem which should be taken into consideration is that the control strategy should store the full recuperation energy on down-hill that will be briefly explained in the next part. The second problem is to minimize fuel consumption while considering the discrete modes which can engage or disengage the engine and electric machine with gear-shifting.

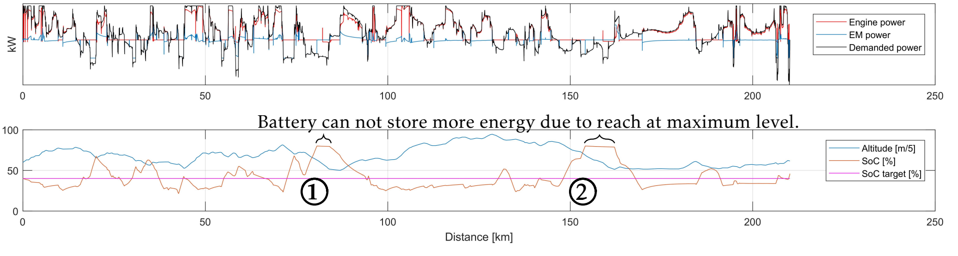


Figure 1.7: Shows the simulation result of Volvo's forward-facing model with predictive control along full driving mission. Bold number 1 and 2 represents first long down-hill and second long down-hill region respectively.

Figure 1.7 shows the simulation result for the current simulation platform. Initially, the battery state of charge (SoC) is 40%. However, the battery has been fully charged two times on down-hills and the state of charge reaches its maximum level (80%), so it can not store more energy as per power demand [Black legend] requirement shown in Figure 1.7 (Electric machine [Blue legend] shut down when SoC reaches at maximum level).

[3]http://www.mathworks.se/

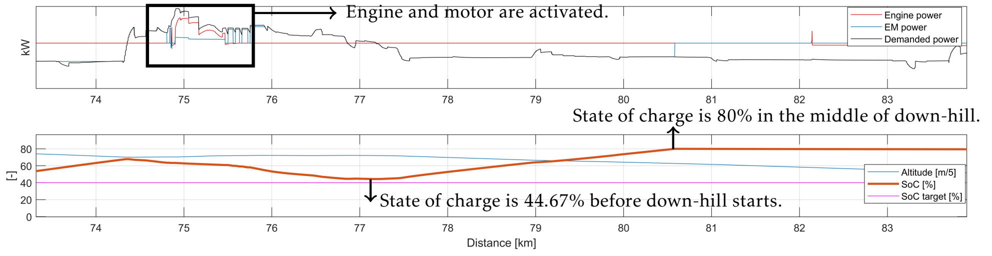


Figure 1.8: *Shows the plot of engine, electric machine, demand torque and SoC vs distance for first long down-hill region. Volvo's forward-facing model with predictive control is used for simulation on provided driving cycle.*

Figure 1.8 represents the results for the first long down-hill. Both engine and machine are activated and deliver the power to the wheels before the down-hill comes (Square block is shown in Figure 1.8). The state of charge is 44.67% before down-hill (Up-hill) and then the battery becomes fully charged (SoC- 80%) in the middle of the down-hill. So, the battery can't store remaining recuperative energy as per torque demand [Black legend]. If whole down-hill contains 100% recuperative energy then battery has stored only 52% and remaining 48% lost because of reaching at maximum level. For full drive cycle, the battery has stored 72% energy of demanded energy and 28% lost along full driving mission due to reaching the maximum level of storing energy capacity of battery. However, the fuel consumption is also affected by the lost energy that the battery has not stored.

In order to overcome this problem, one of the solutions are that the control strategy should store lost energy on down-hill by shutting off the engine and enable pure electric drive before down-hill. So, the battery can reach to its minimum level before down-hill and then battery can store full recuperative energy on down-hill. Moreover, the recuperative energy can be used later along the driving mission by electric machine which could be decreased fuel consumption. The main part of the thesis to control the mode of the vehicle in dynamic programming. i.e. to set the status of engine (on/off) and the status of electric machine (on/off).

1.6 Expected results

The literature review on different energy management strategies will set a background of benchmarking for Volvo's HEV model. The strength and limitation of different EMS according to following set of features such as fuel consumption, recuperated energy and the charge sustaining ability will be investigated. Furthermore, the expected result from this thesis is that fuel economy savings will be achieved through the dynamic programming as predictive control.

1.7 Outline

The thesis will be divided into six chapters.

- **Chapter 1**:- covers the introduction with the explanation of different hybrid electric vehicle configurations and the EMS classification. It also contains purpose and goal with the problem formulation.
- **Chapter 2**:- will give a detailed description of related research of each energy management strategy with the conclusion.
- **Chapter 3**:- describes the hybrid electric vehicle with the vehicle plant script with explanation of all components.
- **Chapter 4**:- covers the explanation of HEV controller including dynamic programming.
- **Chapter 5**:- contains the simulation results with discussion part.
- **Chapter 6**:- consists the conclusion part.
- **Chapter 7**:- contains suggestions for the future work.

2

Related Research

In recent years, a hybrid powertrain control is significant research topic in the area of electromobility. Managing the engine and electric machine through energy management strategy in their efficient way is complex topic and requires a significant analysis [8]. According to section 1.3, the energy management strategies (EMS) are classified into two types: (1) Heuristic based strategy and (2) Advanced control. It is noted that the EMS can include a mixture of various techniques (offline and online) for improving the fuel economy and performance. Thus, in this thesis, the main focus is on global optimization strategy, local optimization strategy and rule based strategy.

2.1 Global optimization strategy

Global optimization strategies are non-causal and find out an optimal solution for the dynamic nature of the system over a predefined driving cycle. Due to non-casual nature, they cannot be directly used for the real-time applications. Although, the offline optimal solutions can be obtained under a given drive cycle, which can provide a benchmark for other online energy management strategies. The global optimization strategies are dynamic programming (DP), genetic algorithms, game theory, robust control, convex optimization and stochastic dynamic programming, which use to find the global optimal solutions [22]. The dynamic programming strategy will be briefly introduced and discussed in the following paragraphs.

The dynamic programming (DP) technique is based on *Bellman's principle of optimality* to achieve the global optimal results. In dynamic programming, The objective is to find the best control input from the control input grid that makes the objective function minimum or maximum at every time step, so that the state trajectory from state vector will be guaranteed optimal over a given driving cycle can be obtained and often used for benchmark purposes [13]. However, the continuous states are implemented in a discrete framework.

[19] used the dynamic programming for the optimization over a given certain time period. This method can be used to minimize fuel consumption in the presence of a soft constraints or hard constraints on the value of SoC. In addition, DP is required the grid for time and state variable (SoC). Therefore, the optimal trajectory of SoC can be calculated only for the discretized value for SoC and time. But due to discretization of state variable, SoC value is either be interpolated or approximated to the nearest available value of state variable grid. However, fine discretization can be reduced dependency on interpolation. Since, the computational burden increases exponentially with number of state variables, [17] used one approach which can reduce computational time by splitting the mission into the series of time sections. For each of these time section, an optimization problem can be solved. In the end, this approach leads sub-optimal results.

[9] introduced one approach which is the economic driving. A velocity is varied in such a way that the fuel economy can be increased. In this approach, The parallel HEV has been used where velocity trajectory, gear shift, torque split are optimized with the dynamic programming. There are total five states; SoC, velocity, actual gear, clutch (open and closed) and engine state (on and off). All state and control variables are the distance dependent. The results showed 4.3% fuel economy increased compare to the fixed velocity DP solution.

However, It is not possible use DP as a real-time control strategy since the DP is the backward approach that means the solution can be obtained only offline and having a priori knowledge of the entire driving cycle or road gradient is necessary which is not possible in real driving conditions. Instead, the dynamic programming technique is used during the design stages of vehicle in order to compare the performance of other control techniques. But, DP can be use in real-time control and that new thinking approach was made by [7] and [21] where the dynamic programming is used to calculate reference SoC and optimal equivalence factor for ECMS respectively for the real-time control.

[7] is used DP to calculate SoC reference in adaptive model predictive control. Based on GPS and ITS, modal driving cycle is generated for a given horizon with the acceleration, constant speed and deceleration sections. According to modal driving cycle, DP is used to calculate SoC reference trajectory, and based on that the parameter adaptive algorithm control is adjusted to make the real SoC following the SoC reference trajectory. The objective of DP algorithm is to minimize the fuel consumption with the best control input over a given time horizon.

In conclusion, the dynamic programming is a numerical method which is often used for offline simulation to get optimal results for a given driving cycle and to set the benchmark for real-time control. However, DP can be used to calculate

SoC reference trajectory for a certain horizon in real-time predictive control strategy. But, the computational burden increases exponentially with the number of state variables. [12] is introduced analytical solution to the dynamic programming. The main focus of the paper is to reduced computational demand by using real-time approximation of the gridded cost-to-go and derive an analytical solution for optimal torque-split at each point in the time and state grid. There are two different approximations was used; a real-time linear approximation and a quadratic spline approximation. The results shows a reduced computational burden with slight degradation in the fuel economy.

2.2 Local optimization strategy

Local optimization strategies is used to find instantaneous minimization of a cost function, taking into consideration both the engine and battery. These EMS can provide the best performance at each instant without a prior knowledge of drive cycle. They are easy to implement in real-time control. However, only local optimal results can be achieved. There are many instantaneous optimization EMSs, such as equivalent consumption management strategy (ECMS), adaptive-ECMS and robust control. In following section, ECMS and adaptive-ECMS methods are introduced and discussed.

[15] and [19] are introduced the concept of equivalent consumption management strategy (ECMS) that has a less computational burden compare to DP. This approach is based on pontryagin minimum principle where a *Hamiltonian function* is minimized at each time. In *Hamiltonian function*, the fuel and battery power both is taken into consideration and the equivalence factor converts the battery power into equivalent fuel power and added to the actual fuel power in order to maintain charge sustaining capability.

The optimization problem is straight-forward; for every time t, the Hamiltonian H must be minimized with respect to the control variable $u(t)$. In addition, $\lambda(t)$ is the co-state or equivalence factor that can be described by the *Euler-Lagrange* equation. Based on assumption, if SoC dynamics (the internal battery parameters) are independent of the SoC, then co-state or equivalence factor can be shown to be piece-wise constant along the driving mission. In general, an equivalence factor value is depended on the driving condition along the mission and constant value of equivalence factor is always different for every driving mission which is considered as a key issue.

As mentioned earlier, the performance of an ECMS is depended on the equivalence factor. Therefore, how to tune equivalence factor is the significant research to improve the performance of an ECMS. [14] proposed a new method A-ECMS which is based an adaptation law of an equivalence factor which is used as feedback SoC from vehicle plant model, and change equivalence factor through state feedback controller according to conditions and reduced fuel consumption. The PI controller is commonly used as state feedback controller. However, the PI parameters need to be an adjusted properly. [20] is introduced a new approach for adjusting an equivalence factor according to the coefficient of charging and discharging of the battery that shows great robustness and the fuel consumption is reduced up to 30% of the conventional equivalence factor value.

Another different approach is described by [21] where optimal equivalence factor is calculated based on the dynamic programming. Based on a given driving cycle, optimal equivalence factor with SoC and power demand is calculated and then used by ECMS strategy through look-up table. Quasi-static approach is considered. SoC trajectory through ECMS is very similar to dynamic programming's SoC trajectory. But, final SoC is not strictly same to initial one but it is very close. In addition, the fuel consumption is compared with benchmark method, dynamic programming. The major drawback is that only one driving cycle is used by both DP and ECMS. By comparing results only on one drive cycle is not giving complete overview regarding drive cycle sensitivity.

2.3 Predictive control strategy

The main purpose of predictive EMS is to optimized the power-split with minimizing fuel consumption by utilizing predictive information up to a certain horizon length. This energy management strategy can be used in real-time control and also gives the sub-optimal results. This strategy requires future drive cycle information such as a future velocity and road topography. [5] is introduced model predictive control for the energy management system of hybrid electric vehicles. Optimal machine torque and engine torque are decided for each sample time up to the future time horizon. These optimal values are provided to the plant model according to the current time or position.

[10] is introduced a novel predictive energy management strategy for hybrid electric trucks. This control scheme has three layers which contains optimization problem. Top layer is calculated the kinetic and electric energy in a convex optimization problem. The selection of the gear and powertrain mode such as hybrid and pure electric mode are optimized in a lower layer with dynamic programming while the lowest control layer only takes real-time decision such as torque-split through equivalence factor. In addition, the equivalence factor calculated by a non-linear state feedback controller where the equivalence factor is adjusted through the feedback from current estimated battery energy state and battery energy trajectory from the top layer control. Furthermore, [3] is introduced a time-varying predictive reference trajectory of the battery SoC and maximized recuperated energy through a quadratic programming. In recent years, further improvements in energy management have led to include the vehicle speed and engine operating points in optimization that reduce the fuel consumption. [6]

2.4 Heuristic-based strategy

Heuristic-based strategies can be divided into rule-based controllers and fuzzy logic controllers which are based on the logical rules and fuzzy logic respectively. The rules are decided based on the driver power demand, battery SoC, and vehicle velocity through 'if-then' structure. Based on these rules, the power-split can be performed to satisfy the driver power demand with charge sustaining ability. The idea behind the power-split in rule based strategy is to always operate the engine at a high efficiency. This method does not require prior knowledge of the drive cycle. Therefore, it can be implement on real-time. Due to lack of future information on the drive cycle, this method cannot be tuned which makes this method less adaptable. However, the rule based controllers are easy to implement and have less computational burden.

A typical rule based approach is based on the torque demand, vehicle sped and state of charge [8]:

- If the state of charge is too low, the engine is forced to recharge the battery.
- If the state of charge is too high, only motor is used to satisfy torque demand and the engine is shut off.
- If the torque demand is higher than maximum torque of engine, the motor is used to assist the engine.
- If vehicle speed is below certain value, the motor is used alone.
- If the vehicle speed is above the threshold value and torque demand is below the maximum engine at current engne speed, the engine alone is used.

[18] is proposed rule based strategy which is based on SoC value and power demand. To obtain optimal torque split, efficiency maps were used. Also, an engine is used to charge the battery when the efficiency is high as possible. However, the rules or transition conditions give sub-optimal results and because of that they do not have always the charge sustaining ability. Besides, the fuzzy logic controllers and rule-based controllers can be optimized through different optimization algorithms such as dynamic programming that improve control performance and better fuel economy. Nevertheless, the global optimum in different conditions can not be guaranteed.

[16] is introduced new rule based strategy based on dynamic programming. The parallel HEV is used with forward modeling. From dynamic programming results, velocity, state of charge, power demand, and optimal torque of the engine were added into the rules criteria. In addition, there are three modes that control the power sources i.e. electric only, engine mode only and hybrid mode. The fuel consumption is 1.7% higher than the DP results.

In conclusion, each energy management strategy set a background for fuel consumption, controlling SoC and speed, and mode of vehicle. The benchmark strategy- dynamic programming will be use in this thesis as a predictive control.

3

Hybrid Electric Vehicle

This chapter introduces the concept of the Volvo's hybrid electric with the general power flow through their powertrain components. The hybrid electric vehicle is implemented in MATLAB with the forward modeling approach. The approach is divided into two segments: the HEV controller and vehicle plant script. The HEV controller contains energy management strategy and vehicle plant script consists different powertrain component scripts such as the engine script, machine script, battery script, transmission script with clutch, wheel and driveshaft script, and vehicle scrip which work as a feedback control.

In this chapter, the general explanation of the HEV controller and inputs-outputs of the vehicle plant script will be explained.

3.1 Volvo hybrid electric truck

In this thesis, the reference vehicle is Volvo's FH13 long-haul truck, developed by Volvo AB. It forms the basis of many truck application which is used for the long haul transport, construction transport, and heavy transport. The drivetrain of this vehicle consists a conventional diesel engine (ICE), automated clutch, twelve speed automated manual transmission, and final drive. As explained in the introduction chapter, the parallel hybrid electric vehicle is the most general purpose and most promising concept when hybridising a truck. Therefore, the studied configuration in this thesis is Volvo's FH13 long-haul truck with the electric machine which is connected to the transmission's counter shaft through the clutch in parallel combination. The energy accumulator is a Li-ion battery pack which is connected to the electric machine through an inverter. There is also possibility to decouple an engine and motor with a clutch. The Volvo parallel hybrid electric truck is shown in Figure 3.1.

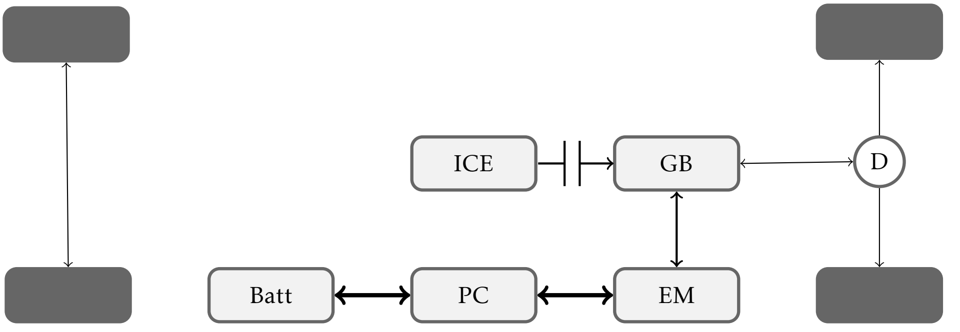


Figure 3.1: *Configuration of Volvo Parallel Hybrid. ICE: Internal combustion engine, GB: gear box, D: differential, Batt: battery, PC: power converter, EM: electric machine, parallel two lines: clutch. Bold lines: electrical link, solid lines: mechanical link. Double side arrow shows regenerative braking path.*

3.2 Drive cycle

The important input for the vehicle model is the drive cycles which is shown in Figure 3.2. The drive cycle provides inputs to the HEV controller and vehicle plant script model. The HEV controller uses a drive cycle as input for look-ahead information (Predictive control). The inputs are the desired set speed of the vehicle over distance, altitude and initial state of charge of the battery. In this thesis, three types of drive cycle are used, i.e. the flat, hilly-1 and hilly-2 drive cycle measured on a predominantly flat road, short hilly road and long hilly road respectively.

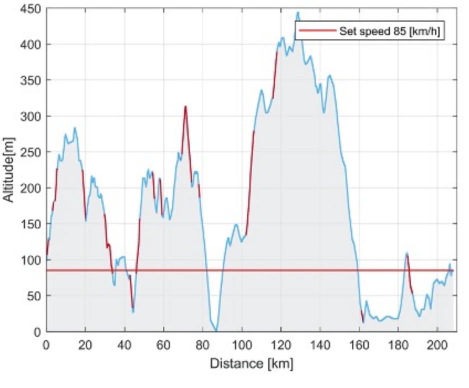


Figure 3.2: *Hilly-2 drive cycle with set speed 85 km/h and altitude. Red and blue lines in the altitude shows steep up/down hill.*

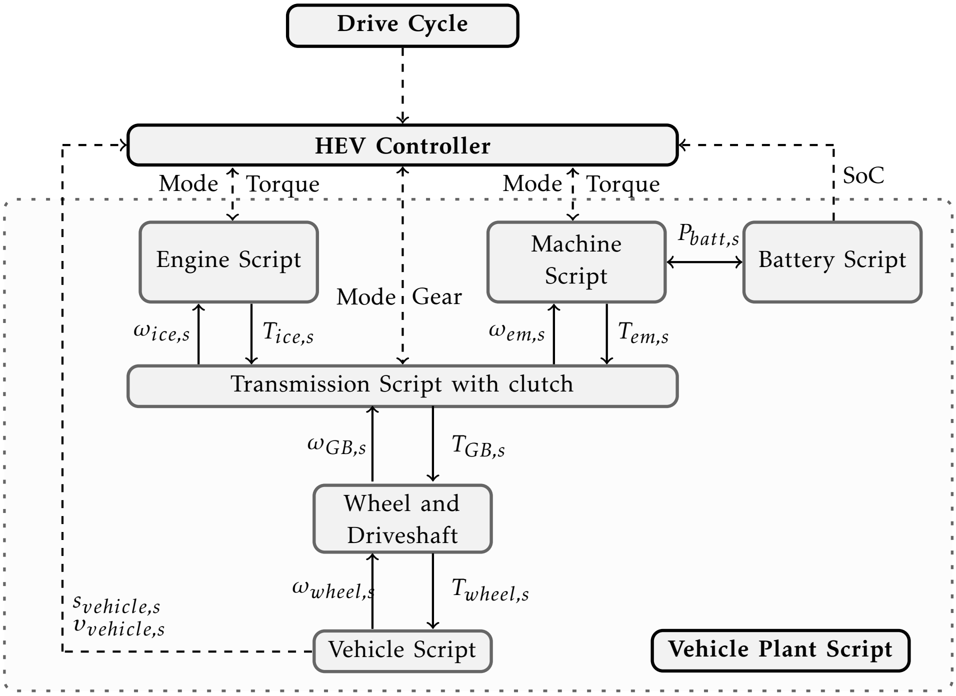


Figure 3.3: *Representation of hybrid electric vehicle script based plant model with power flow. Dotted line represents the interaction through the flow of data between the HEV controller and the HEV components. Solid lines represent power flows between HEV components.*

3.3 HEV controller

A HEV controller contains a predictive energy management strategy that optimized controls in order to minimized the fuel consumption. The first input of the HEV controller is the drive cycle. Additional inputs are a state of charge, vehicle position and speed from the battery script and vehicle script as a feedback control. The outputs are the requested powertrain mode, requested gear and torque requests for the vehicle plant script. The optimization model is minimized the energy consumption by applying the look-ahead information (Predictive control). In the dynamic programming, the total states are a kinetic energy (speed), state of charge, current gear and current powertrain modes. The control inputs are an engine torque, machine torque, selected gear and selected powertrain modes. The states and control inputs description is mention in the Section 4.

Table 3.1: *Vehicle specification*

Vehicle Parameters	**Value**
`Vehicle mass`	34.5 [Ton]
`Engine size`	13 [L] / 400 [KW]
`Electric machine`	200 [kW]
`Over-speed allowance`	+5 kph, -10 kph
`Transmission`	12 speed AMT

3.4 Vehicle plant script

In this thesis, the modeling approach is the forward-facing script model which is shown in the Figure 3.3. The working principle of script based model is considered as a feedback control loop.

The drive cycle consists of inputs for the HEV controller and the vehicle plant script model. The HEV controller consists a predictive energy management strategy, whose purpose is to optimise the controls. In addition, the HEV controller sends optimized controls to vehicle plant script model based on the feedback request i.e. the current operating conditions (current position of the vehicle). The controls are engine torque, electric machine torque, gear number, and the powertrain mode.

After receiving optimal controls from the HEV controller, the engine script and the electric machine script conveys the torque request $T_{ice,s}$ and $T_{em,s}$ on to the transmission script before ending up as a torque applied at the wheels ($T_{wheel,s}$) in the wheel and driveshaft script. Through torque supplied to the wheel, the vehicle speed ($v_{vehicle,s}$) is in return propagated to the wheel and driveshaft script, transmission script, and returns to both engine script ($\omega_{ice,s}$) and machine script ($\omega_{em,s}$) as an angular velocity. In addition, the battery script calculates state of charge (SoC) through the power of machine which comes from the machine script. The propagated vehicle speed ($v_{vehicle,s}$) and state of charge (SoC) is considered as inputs for the next feedback request and sends them back to the HEV controller. The detailed explanation of each scripts are described in the following sections.

3.4.1 Engine script

The first input of this script is the torque value and powertrain mode request from the HEV controller and provided ICE torque $T_{ice,s}$[1] as an output to the transmission script by checking the maximum torque limit of the engine.

[1] s denotes the variable for vehicle plant script.

In addition, the second input for engine script is $\omega_{ice,s}$, which is propagated back from the transmission script. Based on the $T_{ice,s}$ and $\omega_{ice,s}$, the engine script specifies the fuel consumption rate ($\dot{m}_{f,s}$ $[g/s]$) through the fuel map, as a function of its operating points which is defined by $T_{ice,s}$ and $\omega_{ice,s}$. The angular speed $\omega_{ice,s}$ is depended on the transmission script model and it is affected by changing the gear request, engagement/disengagement of the engine and opening/closing the clutch.

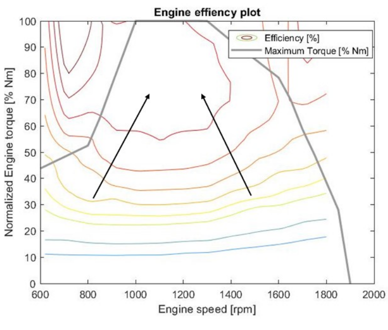
$$\dot{m}_{f,s} = Fuelmap\,(T_{ice,s}, \omega_{ice,s}) \quad [g/s] \tag{3.1}$$

The total fuel consumption is obtained by;

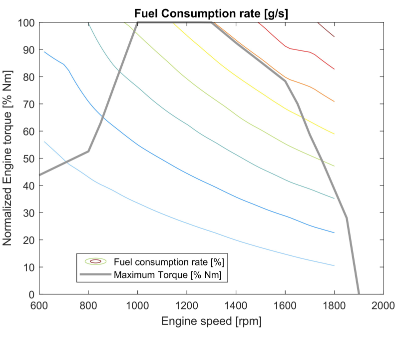
$$V_{total} = \int_0^{s_f} \frac{\dot{m}_{f,s}}{\rho_{diesel} \cdot v_{vehicle,s} \cdot 1000} \cdot ds \quad [l] \tag{3.2}$$

Where ρ_{diesel} is the density of diesel $[kg/l]$. The fuel consumption rate unit $[g/s]$ is converted into $[kg/s]$. The fuel consumption per 100 km is obtained from the total distance traveled (x_{tot}).

$$Fuel\ consumption = \frac{V_{total} \cdot 10^5}{x_{tot}} \quad [l/100\ km] \tag{3.3}$$



(a) *Engine efficiency plot with maximum torque. The arrow indicates increasing efficiency.*



(b) *Fuel consumption of diesel engine. The arrow indicates increasing fuel consumption.*

Figure 3.4: *Engine efficiency plot and fuel consumption map.*

The engine is only allowed to operate within its range limits which is defined by maximum, and minimum torque limits. The minimum torque limit is negative which is called the Volvo engine braking.

$$T_{ice,s,\min} \leq T_{ice,s} \leq T_{ice,s,\max} \tag{3.4}$$

3.4.2 Electric machine script

The electric machine considers torque request and powertrain mode request as its first input from the HEV controller and send the machine torque $T_{em,s}$ as an output. Furthermore, the electric machine script checks the maximum torque limit of the electric machine. The output $T_{em,s}$ is provided to the transmission script.

The second input is the angular speed of the electric machine $\omega_{em,s}$ which comes from the transmission script. The angular speed is affected by gear changing request or engaged/disengaged of the machine. The electric machine script sends the power of electric machine $P_{em,s}$ to the battery script, which includes all losses of the machine which includes the loss from inverter as well. This is determined by means of EM efficiency map. The efficiency map is a function of both $T_{em,s}$ torque and $\omega_{em,s}$ speed which is shown in Figure 3.5.

$$P_{em,s} = T_{em,s} \cdot \omega_{em,s} + P_{motorloss}(T_{em,s}, \omega_{em,s}) \tag{3.5}$$

Where $P_{motorloss}$ is the power loss in electric machine which is calculated by the EM efficiency map.

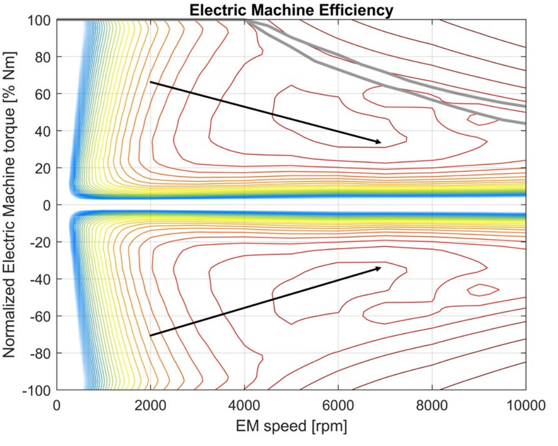


Figure 3.5: *Electric machine efficiency plot with maximum torque. The arrow indicates increasing efficiency.*

The operation of the electric machine is limited by its minimum and maximum torque limitations.

$$T_{em,s,\min} \leq T_{em,s} \leq T_{em,s,\max} \tag{3.6}$$

3.4.3 Battery script

The battery script measures the state of charge level of the battery. The input of this script is the power of machine $P_{em,s}$ which is derived from the electric machine script. The state of charge level is 20% - 80% , this is to increase the efficiency and enhance the lifetime of the battery. The output of this script is state of charge (SoC) and is provided to the HEV controller.

The power of the battery is described by Equation 3.7.

$$P_{batt,s} = P_{em,s} + P_{auxiliary} \tag{3.7}$$

Where $P_{auxiliary}$ is the power consumed by the auxiliaries and it is considered as a constant. The auxiliaries are always consumed power from the battery even if the electric machine is turned off.
The state of charge rate equation is given by Equation 3.8.

$$\frac{dSoC}{dt} = -\eta_{col}^{-sign(I_{batt,s})} \left(\frac{V_{oc,s} \pm \sqrt{V_{oc,s}^2 - 4\ P_{batt,s} R_i}}{2 R_i Q_{Ah}} \right) \tag{3.8}$$

Where η_{coul} is the columbic efficiency and it is a constant value. V_{oc} is the open circuit voltage and it is measured by look-up table (Voltage map) as a function of current state of charge. R_i is the internal resistance of the battery. Q_{Ah} is the battery charge capacity.

3.4.4 Transmission script with clutch

The inputs of this script are selected powertrain mode, engine torque ($T_{ice,s}$) and the machine torque ($T_{em,s}$), which are obtained from HEV controller, engine script and machine script respectively. Further, the transmission script calculates the output $T_{GB,s}$, which is provided to the wheel and driveshaft script. Furthermore, this script also takes selected gear from HEV controller as an input and calculates $T_{GB,s}$ according to Equation 3.9 with respective gear ratio.

Another input to the transmission script is the angular speed of the transmission $\omega_{GB,s}$ which comes from wheel and driveshaft script, and further transmits an angular speed of the engine ($\omega_{ice,s}$) and machine ($\omega_{em,s}$) to the respective scripts. Equation 3.10 takes for the calculation of the same.

In this thesis, Volvo's automated manual transmission (AMT) has been chosen with its transmission efficiency. The transmission script consists a set of different gears with different conversion ratios. This provides a torque-speed conversion from higher torque to lower torque, accordingly lower speed to higher speed. The inertia is also included in the script.

The dynamics of gear-shifting with the respective shift time, engagement/ disengagement of the engine and the machine, and clutch model with its opening/closing time is provided by Volvo. In addition, the slipping effect has been taken into consideration in the clutch model. Slipping occurs when the clutch opens, and this slip affects the angular speed $\omega_{ice,s}$ which in turn affects the fuel consumption calculation in the engine script. When the clutch is opened or closed, the overall inertia is affected.

$$T_{GB,s} = T_{ice,s} \cdot i_{ice} \cdot \eta_{GB_{ice,s}}^{\text{sign}(T_{ice,s})} + T_{em,s} \cdot i_{em} \cdot i_{reduction_{em}} \cdot \eta_{GB_{em,s}}^{\text{sign}(T_{em,s})} \tag{3.9}$$

The gear box efficiencies ($\eta_{GB_{ice,s}}$ and $\eta_{GB_{em,s}}$) depend on the gear number, as well as the sign of $T_{ice,s}$ and $T_{em,s}$ torque values. The transmission efficiency map is used in this thesis.

$$\begin{aligned} \omega_{ice,s} &= \omega_{GB,s} \cdot i_{ice} \\ \omega_{em,s} &= \omega_{GB,s} \cdot i_{em} \cdot i_{reduction_{em,s}} \end{aligned} \tag{3.10}$$

Where $\omega_{GB,s}$ is the angular velocity of the transmission. i_{ice} and i_{em} are the gear ratios for an ICE and EM respectively. r_w and $i_{reduction_{em}}$ are the radius of the wheel and reduction gear ratio of the EM. $\omega_{ice,s}$ and $\omega_{em,s}$ are angular velocity of the engine and the electric machine respectively.

3.4.5 Wheel and driveshaft script

The wheel and driveshaft script consists the losses of driveshaft and final-drive ratio. The first input is a torque of transmission ($T_{GB,s}$) which comes from the transmission script, and provides a torque of wheel ($T_{wheel,s}$) to the vehicle script including all losses of driveshaft and final-ratio.

The second input is the angular speed of wheel ($\omega_{wheel,s}$) which occurs from the vehicle script and send it to the transmission script in the form of $\omega_{GB,s}$ with considering driveshaft ratio. The driveshaft loss map is used to calculate the losses in the driveshaft as function of a torque and speed of wheel. The wheel torque and transmission angular speed are described in Equations 3.11 and 3.12.

$$T_{wheel,s} = T_{GB,s} \cdot i_{finaldrive} + driveshaft_{loss}(T_{wheel,s}, \omega_{wheel,s}) \tag{3.11}$$

$$\omega_{GB,s} = \omega_{wheel,s} \cdot i_{finaldrive} \tag{3.12}$$

3.4.6 Vehicle script

The vehicle script describes the nature of longitudinal vehicle dynamics. The first input of this script is the torque of the wheel ($T_{wheel,s}$) from the wheel and driveshaft script and provides vehicle velocity ($v_{vehicle,s}$) for next stage according to Equation. The $v_{vehicle,s}$ is transmitted to the HEV controller. Also, the script calculates the angular speed of the wheel ($\omega_{wheel,s}$) which provides to the wheel and driveshaft script.

The longitudinal vehicle dynamics consists four main resistance forces which are the traction force ($F_{t,s}$), braking force ($F_{b,s}$), gravitational force (F_g), rolling resistance force ($F_{r,s}$) and aerodynamic resistance force ($F_{a,s}$). The total mass of the vehicle including all components inertia is described as $m_{total,s}$. Traction force is produced by the two power sources with combination of $T_{ice,s}$ and $T_{em,s}$ which is considered as $T_{wheel,s}$ at the wheel. The Equation 3.1 which describes the longitudinal vehicle dynamics.[2]:

$$m_{total,s}\frac{d}{dt}v_{vehicle,s} = F_{t,s} - F_{b,s} - F_{a,s} - F_{r,s} - F_{g,s} \tag{3.13}$$

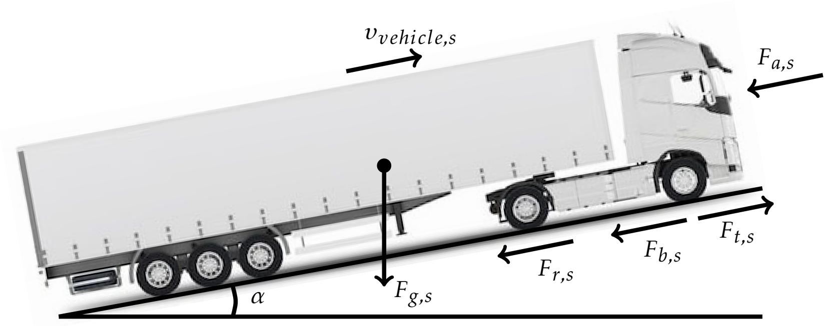


Figure 3.6: *Longitudinal Vehicle Dynamics*

The traction force ($F_{t,s}$) is calculated by torque of the wheel and is shown in Equation 3.14.

$$F_{t,s} = \frac{T_{wheel,s}}{r_w} \tag{3.14}$$

Where r_w is the radius of the vehicle. However, the angular speed of wheel is calculated by Equation 3.15.

$$\omega_{wheel,s} = \frac{v_{vehicle,s}}{r_w} \tag{3.15}$$

Aerodynamic friction losses

The aerodynamics resistance force $F_{a,s}$ is acting on the vehicle due to the viscous friction of the surrounding air on the vehicle surface and separation of the air flow due to the pressure difference between the front and the rear of the vehicle. Usually, the aerodynamics resistance force is measured by simplifying the vehicle model as a prismatic body with a frontal area A_f and constant drag coefficient c_d.

$$F_{a,s} = \frac{1}{2} \cdot \rho_{air} \cdot A_f \cdot c_d \cdot v^2 \tag{3.16}$$

Where ρ_{air} and $v_{vehicle,s}$ are the density of the air and vehicle speed respectively.

Gravitational force

In long haul trucks, the gravitation force plays major role when driving on a non-horizontal road and influences the vehicle behavior. The force is modeled as follows;

$$F_{g,s} = m_{total} \cdot g \cdot \sin(\alpha) \tag{3.17}$$

Where $m_{total,s}$ is total vehicle mass, g is the acceleration due to gravity and α is slope angle which expressed in radians.

Rolling resistance

The rolling resistance F_r is modeled as follows;

$$F_{r,s} = m_{total,s} \cdot g \cdot c_r \cdot \cos(\alpha), \quad v_{vehicle,s} > 0 \tag{3.18}$$

Where c_r is the rolling resistance friction that depends on the several parameters such as tire pressure, vehicle speed, temperature and surface but vehicle speed has small influence at lower values so c_r values is constant in this case. $m_{total,s}$ is total vehicle mass, g is the acceleration due to gravity and α is slope angle.

4

HEV controller

In this chapter, the idea behind a supervisory controller will be discussed with its layout. Regarding the introduction section, the hybrid electric vehicle consists of two power sources i.e. the engine and the electric machine. In order to operate both in an efficient way, there is the need of supervisory controller (HEV controller) that follows the model predictive control paradigm. It contains optimization method (energy management strategy) and provides the control signals to the vehicle components according to the vehicle current position. The predictive control is formulated as a full or receding horizon optimal control problem with respect to the system dynamics, control inputs and state constraints. The information about the future driving mission that means receding or full horizon input data acquires by drive cycle.

The main objective of the EMS is to minimize the fuel consumption of the vehicle. The overview of different energy management strategies that is described in Section 2. In this thesis, the EMS is used to minimize the fuel consumption over receding or full prediction horizon with the dynamic programming. The aim of the EMS is the following:

- To find the controls which is minimized the fuel consumption.

The HEV controller is classified into different blocks; Road- Re -constructor, EMS and Reference Re-constructor. All blocks are separated but strongly interact and affect each other. Their functionalities are described in next section.

4.1 The layout of HEV controller

The HEV controller consists three main blocks that are shown in Figure 4.1, *Road Re-Constructor*, *Energy Management Strategy (EMS)* and *Reference Re-Constructor*. The main input of the HEV controller is drive cycle (Look-ahead information). The other inputs are vehicle plant script outputs and current vehicle position as feedback control. The output signals from HEV controller are the torque value of engine, electric machine, selected powertrain mode and selected gear. The outputs of HEV controller are provided to the vehicle plant script.

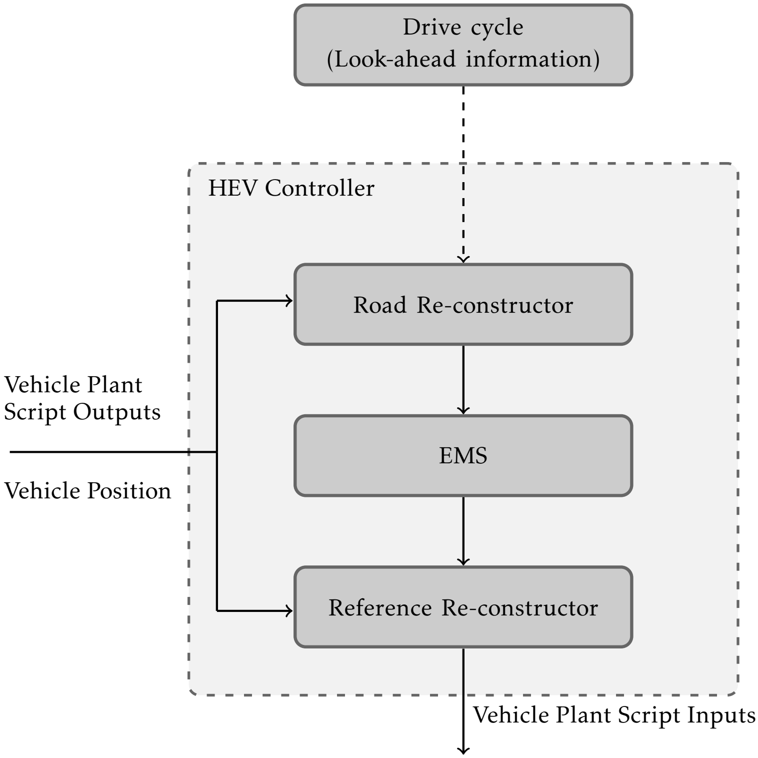


Figure 4.1: *Representation of HEV controller*

4.2 Road Re-constructor

The Road Re-constructor block takes the look-ahead information data (Drive cycle) which contains minimum set vehicle speed and topographic information. Further, it calculates the outputs of this block which are the upper and lower speed bound over the prediction horizon length with considering road curvature and road legal speed limit. Later, the speed bound is used for discretizing kinetic energy as state in the EMS. An adjusted set speed trajectory over the horizon length is also computed and considered as one of the outputs of this block. In addition, the adjusted set speed is calculated in the relation with altitude.

The functionality of this block depends on the horizon length. i.e. for receding horizon, every-time this block is activated to re-calculate the adjusted set speed trajectory with upper and lower bound limit up to receding horizon length before vehicle has covered the update length distance. During full horizon, the road re-constructor block turns on only at beginning of simulation or when vehicle starts, generates adjusted set speed trajectory with upper and lower bound limit up to full horizon length and turn it off. In addition, the final destination should be predefined for full horizon case, then GPS extract the input data for entire route and provided to this block. The outputs from this block are provided to the EMS block.

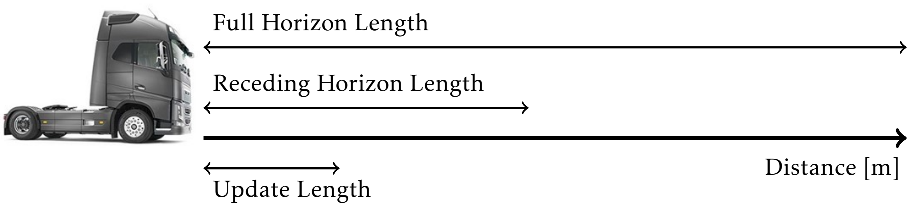


Figure 4.2: *Definition of Update length, Receding horizon length and full horizon length*

4.3 Energy management strategy (EMS)

The EMS block contains the optimization method to find optimal state and control policy trajectories that minimizes the fuel consumption based on receding horizon or full horizon length. The output trajectories are provided to the Reference Re-constructor. The optimization algorithm is based on the dynamic programming. The state variables are kinetic energy, battery state of charge, current gear and current powertrain mode. The control inputs are engine torque, electric machine torque, selected gear and selected powertrain mode.

The main input of this block is speed bound from Road re-constructor block which is converted into kinetic energy grid as one of the states in the dynamic programming. The functionality of EMS depends on the input feed from the road re-constructor. Whenever the road re-constructor block is activated, the EMS is calculated optimal state and control inputs trajectories. The detailed explanation of the optimization model is described in Section 4.2

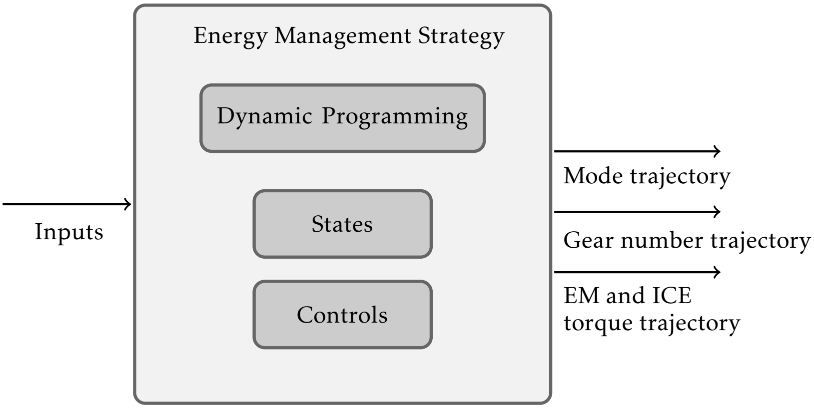


Figure 4.3: *The overall structure of the EMS.*

4.3.1 The model implementation for dynamic programming

Table 4.1: *The total states and control inputs of dynamic programming*

Number	States	Control inputs
1	Kinetic energy (E)	Engine torque (T_{ice})
2	State of charge (SoC)	Electric machine torque (T_{em})
3	Gear (γ)	Selected gear (γ_{sel})
4	Powertrain mode (p)	Selected powertrain mode (p_{sel})

Longitudinal model

In the dynamic programming, the kinetic energy is one of state. Therefore, the longitudinal model is described in this section. The task for this section is to calculate the kinetic energy for the dynamic programming algorithm, So, the control inputs T_{ice} and T_{em} are the piecewise constant inputs to the dynamic programming. Therefore, the control inputs set total demand force (F_t) to calculate kinetic energy for the next stage with the define constraints and within specified limits for the kinetic energy. The functionality of control inputs T_{ice} and T_{em} are depended on the powertrain mode which is described in following section. The state kinetic energy is defined as according to Equation 4.1.

$$E = \frac{1}{2}mv^2 \tag{4.1}$$

Therefore, an expression for the derivative of kinetic energy with respect to position from Equation 3.13 can be rewritten as;

$$\frac{dE}{ds} = F_t(s) - F_b(s) - F_a(E(s)) - F_r(\alpha(s)) - F_g(\alpha(s)) \tag{4.2}$$

The traction force F_t can be expressed as;

$$F_t = F_{ice} + F_{em} \tag{4.3}$$

Where F_{ice} and F_{em} are considered as engine force at the wheel and motor force at the wheel respectively, and can be described as;

$$F_{ice} = \frac{1}{r_w}\left(T_{ice} \cdot i_{ice} \cdot i_{finaldive} \cdot \eta_{gb_{ice}}^{\text{sign}(T_{ice})}\right) \tag{4.4}$$

$$F_{em} = \frac{1}{r_w}\left(T_{em} \cdot i_{em} \cdot i_{finaldive} \cdot i_{reduction_{em}} \cdot \eta_{gb_{em}}^{\text{sign}(T_{em})}\right) \tag{4.5}$$

Where T_{ice} denotes the engine torque and one of the control inputs. In similar way, T_{em} is the electric machine torque including all losses and considered as control input of dynamic programming. Both T_{ice} and T_{em} is limited by their maximum and minimum value, and discretized in the dynamic programming. $\eta_{gb_{em}}$ and $\eta_{gb_{ice}}$ are the efficiency of the transmission and calculated based on the transmission efficiency map as a function of gear number. The driveshaft losses are also considered in the longitudinal model.

Battery model

The state of charge is the second piece-wise constant state in the dynamic programming. Therefore, the state of charge dynamics is given by Equation 4.6.

$$S\dot{o}C = \begin{cases} -\frac{1}{\eta_{\text{coul}}}\frac{I_{batt}}{Q_{Ah}}, & I_{batt} > 0 \\ -\eta_{\text{coul}}\frac{I_{batt}}{Q_{Ah}}, & I_{batt} < 0 \end{cases} \tag{4.6}$$

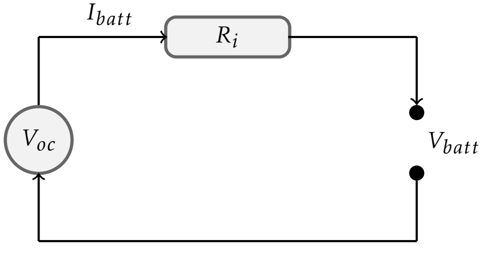


Figure 4.4: *Equivalent circuit of a battery. V_{oc} and V_{batt} represent the open-circuit voltage and the battery voltage respectively, R_i represents the internal resistance of battery, and I_{batt} express as the battery current.*

Where η_{coul} is columbic efficiency and provided by Volvo. In Figure 4.4, an equivalent circuit of the battery is shown and according to ohm's law, power of battery can be written as;

$$P_{batt} = P_{em} + P_{auxiliary} = (T_{em} \cdot \omega_{em}) + P_{auxiliary} \tag{4.7}$$

The power of battery is obtained by one of the control inputs (T_{em}). The angular speed is obtained by the state, the kinetic energy. In addition, $P_{auxiliary}$ is the constant power consumed by auxiliaries and it is always provided by the battery regardless of powertrain modes. According to Kirchoff's voltage law, $V_{batt} = V_{oc} - I_{batt} \cdot R_i$, where V_{batt} is the voltage of the battery and Equation 4.8 becomes;

$$P_{batt} = V_{oc} I_{batt} - I_{batt}^2 R_i = P_{em} + P_{auxiliary} \tag{4.8}$$

Where P_{em} is the electric machine power. By solving Equation 4.8 for I_{batt},

$$I_{batt} = \frac{V_{oc} \pm \sqrt{V_{oc}^2 - 4\left(P_{em} + P_{auxiliary}\right) R_i}}{2R_i} \tag{4.9}$$

The final state equation for SoC with time dependency is given by using equation 4.9 and 4.6,

$$\frac{dSoC}{dt} = -\eta_{col}^{-sign(I_{batt})} \left(\frac{V_{oc} \pm \sqrt{V_{oc}^2 - 4(P_{em} + P_{auxiliary}) R_i}}{2R_i Q_{Ah}} \right) \tag{4.10}$$

In this thesis, spatial coordinate which denotes the traveled distance and variables are position dependent. Therefore, the battery dynamics is given by Equation 4.11

$$\frac{dSoC}{ds} = \frac{-\eta_{col}^{-sign(I_{batt})}}{v(s)} \left(\frac{V_{oc} \pm \sqrt{V_{oc}^2 - 4\left(P_{em} + P_{auxiliary}\right) R_i}}{2R_i Q_{Ah}} \right) \tag{4.11}$$

When $P_{batt} = P_{em} + P_{auxiliary} < 0$, battery is in charging mode, since $S\dot{o}C > 0$ and positive value of P_{batt} indicates discharging of battery. Lastly, the state SoC is limited by Equation 4.12 and discretized in the dynamic programming.

$$SoC_{\min} \leq SoC \leq SoC_{\max} \tag{4.12}$$

where SoC_{min} denotes the lowest admissible state of charge and SoC_{max} the highest.

Strategic powertrain modes and gears

In this section, the different powertrain modes are explained i.e. Hybrid mode, pure electric mode with ICE off, pure ICE mode with EM off, ICE and EM both off. All these modes are discrete state in the dynamic programming and describes different driving situations. The mode of vehicle is directly related to T_{ice} and T_{em} control inputs. According to the powertrain mode, T_{ice} and T_{em} control inputs are provided to the longitudinal model and battery model in the dynamic programming. The transition from one mode to another mode is handled by dynamic programming algorithm. However, the status of engine is the important factor in

Table 4.2: *Powertrain modes*

Modes	Description	Total force
1	Hybrid	$F_{ice} + F_{em} = F_t$
2	Machine only	$F_{em} = F_t, F_{ice} = 0$
3	Engine only	$F_{ice} = F_t, F_{em} = 0$
4	Open driveline	$F_{ice} = F_{em} = F_t = 0$

order to save the fuel. The powertrain can operate in the following modes:-

Hybrid drive:- In this mode, the engine and the machine are used to propel the vehicle. Further, both T_{ice} and T_{em} control inputs are provided to the longitudinal model to define vehicle speed and to set demanded force at wheel (F_t).

Only machine drive with ICE off:- In this mode, only electric machine is activated and propel the vehicle. When pure electric mode is activated, first the clutch opens and then disengaged the engine which means no torque required from the engine. Therefore, only T_{em} control input is provided to the longitudinal model to decide the vehicle speed and to set demanded force (F_t). This mode has great benefit regarding fuel savings and emissions as well, since fuel consumption is zero. This mode activates when small hills are present or force from the electric machine can maintain set speed, or during downhill because the regenerative braking can charge battery.

Only engine drive with machine off:- In pure diesel mode, only engine propel the vehicle and machine is shut off. When pure diesel mode is activated, first the clutch opens and then disengaged the machine. Therefore, only T_{ice} control input is provided to the longitudinal model to calculate kinetic energy and total demand force (F_t). This mode activates during steep hills where more engine force required or when battery SoC level is low.

Engine and machine both off with open driveline:- In this mode, both machine and engine are off and when this mode is activated, first the clutch opens and then disengaged the engine and machine which means there is no force on the driveline from the power sources. Therefore, no control inputs T_{ice} and T_{em} are provided to the longitudinal model. Usually, this mode is activated when the vehicle is going on small down-hill where the regenerative braking is not required or vehicle can maintain the set speed.

In the dynamic programming, the current gear is one of the states and selected gear is one of the control inputs. The selected gear is directly related to the gear ratio of transmission. According to selected gear, the transmission gear ratios (i_{ice} and i_{em}) are provided to the longitudinal model in order to calculate F_{ice} and F_{em}. The gear state change is handled by the dynamic programming algorithm.

Fuel consumption

The fuel mass flow rate is obtained by the fuel consumption map which is shown in Figure 3.4(b).

$$\dot{m}_f = Fuelmap(T_{ice}, \omega_{ice}) \tag{4.13}$$

The power equivalent to the fuel flow rate $\dot{m}_f$ $[g/s]$ is,

$$P_{\text{fuel}}(t) = \frac{n_{\text{cyl}} \dot{m}_f \omega_{ice} Q_{LHV}}{n_r} \tag{4.14}$$

Where P_{fuel} $[J/s]$, ω_{ice} $[rpm]$, Q_{LHV} $[J/g]$ and $\dot{m}_f$ $[g/s]$ are fuel power, engine speed, fuel lower heating value and the diesel mass flow rate respectively. In addition, n_{cyl} and n_r are the number of cylinders and the number of crankshafts revolutions per cycle. In spatial coordinates, the fuel energy is described by Equation 4.28 which is used as cost function (g_k) in the dynamic programming.

$$\frac{dE_{\text{fuel}}}{ds} = \frac{P_{\text{fuel}}(s)}{v(s)} = F_{\text{fuel}}(s) \tag{4.15}$$

4.3.2 Formulation of dynamic programming

The dynamic programming is a optimization method that solves the optimal control problem by breaking it down into simpler sub-problems. The method was developed by Richard Bellman and based on Bellman's principle of optimality [4]. The general mathematical formulation of the continuous optimal control problem has seen in Equation 4.1.

$$\min_{u(\cdot)} \quad \phi(x(s_f)) + \int_{s_0}^{s_f} g_t(t, x(s), u(s))\, ds \tag{4.16}$$

subject to

$$\begin{aligned} &\dot{x}(s) = f(s, x(s), u(s)) \\ &x(0) = x_0 \\ &x(s) \in X(s) \\ &u(s) \in U(s) \end{aligned}$$

Where $x(t)$ is a state vector, s_i and s_f are the initial and the final position and x_0 is a fixed initial point, The end point $x(s_f)$ is free and can be any value in $\mathbf{R}^n$. $\phi(x(s_f))$ is the terminal cost. g_t is the cost function and descried in Equation 4.15. The control is piece-wise linear function that satisfies the given constraints $u(t) \in U$, for the position $s \in [s_i, s_f]$.

The general form of an optimal control problem for the discrete dynamic programming is shown in Equation 4.2. By discretizing the position trajectory s in Equation 4.1 into N number of stages which results in $\frac{s_f - s_i}{N} = h$ sized steps (The distance between the stages) and approximating $\frac{dx}{ds}$ by using the Euler forward [11].

$$\min_{u(k)} \quad \phi(x_N) + \Sigma_{k=0}^{N-1} h \cdot g_k(k, x_k, u_k) \tag{4.17}$$

subject to

$$x_{k+1} = f(k, x_k, u_k)$$
$$x_0 \text{ given, } x_k \in X_{grid}$$
$$u_k \in U_{grid}(k, x_k)$$

Where k is the integer which indicates k: th stage in the N number of the stages. However, the step size h is fixed for full and receding horizon length. Let's consider, the value of step size (h) is 50 [m] and horizon length ($s_f - s_i$) is 2000 [m]. Based on these values, the number of stages can be calculated as $N = \frac{2000}{50} = 40$. Thus, the number of stages varies based on horizon length. Furthermore, x_k and u_k are state dynamics and control variable at the stage k respectively. The g_k is the cost function of the dynamic programming.

System Dynamics: The variable $k \in 0, 1, ..., N$ represents the stages of optimization problem. The state vector x corresponds to some state space which generally depends on k. i.e. $x_k = x(k) \in X_{grid}$, where X_{grid} denotes the state space grid at stage k and $X_{grid} = \mathbf{R}^n$ for all $k = 0, 1, .., N$ but it can be the discrete set. In this EMS, the state variables are velocity, v, battery state of charge, SoC, current gear, γ and current powertrain mode, p that are denoted by Equation 4.18. Moreover, the state change (Transition mode) for gear and powertrain mode in the dynamic programming is described in Section 4.3.4.

$$x = (v, SOC, \gamma, p) \in X_{grid} = [x_{min}, x_{max}] \tag{4.18}$$

The state dynamics is given by Equation 4.19.

$$x_{k+1} = f(k, x_k, u_k) \tag{4.19}$$

where the state vector x_k, the control variable u_k and the vector f is depended on the position s. The state dynamics for kinetic energy (velocity) is described in Equation 4.5. In similar way, the state of charge dynamics is described in Equation 4.11.

Control variable: The variable $u_k \in U(k, x_k)$ is the control or decision variable. The set of control constraints set $U(k, x_k)$ which are the range of the control variable. The control inputs are torque of engine, torque of electric machine, selected gear and selected powertrain mode.

$$u = (T_{ice}, T_{em}, \gamma_{sel}, p_{sel}) \in U_{grid} = [u_{min}, u_{max}] \tag{4.20}$$

The set of control inputs (u) contains a sequence of functions which is shown in Equation 4.21.

$$u = \{u_0, u_1 \ldots, u_{N-1}\} \tag{4.21}$$

Cost function: The cost function is a main objective of the problem and it is minimized over each stage. In this thesis, Equation 4.15 represents the cost function ($g_k = F_{fuel}$) for the dynamic programming. The inputs of the cost function are the requested engine T_{ice} and machine torque T_{em}, the selected gear (γ_{sel}) and the selected powertrain mode (p_{sel}). While the outputs are the fuel force, the next stage kinetic energy and the state of charge. The next stage kinetic energy is calculated by the longitudinal model Equation 4.2 by obtaining total force (F_t). Furthermore, F_t is computed by control inputs (torque of engine and machine). In similar way, the next stage state of charge is calculated by Equation 4.11. The total cost function of entire path can be shown as;

$$\phi(x_N) + \Sigma_{k=0}^{N-1} h \cdot g_k\left(k, x_k, u_k\right) = \phi(x_N) + \Sigma_{k=0}^{N-1} h \cdot F_{fuel}\left(k, x_k, u_k\right) \tag{4.22}$$

Where $\phi(x(t_f))$ is the terminal cost that penalize deviation from the desired terminal state and the running cost adds a term $g_k\left(k, x_k, u_k\right)$ to the total cost at each stage.

4.3.3 The principle of optimality

An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision [11].
This means that if an optimal policy $u^* = \{u_0^*, u_1^* \ldots, u_{N-1}\}$ passes through the state x_k at stage $k = i$, and to find optimal policy from x_k to x_N by minimizing the cost function;

$$\min\ J_k\left(x_k\right) = \phi\left(x_N\right) + \Sigma_{k=0}^{N-1} g_k\left(k, x_k, u_k\right) \tag{4.23}$$

The optimal policy is shown in Equation 4.24.

$$u_i^* = \{u_i^*, u_{i+1}^* \ldots, u_{N-1}\} \tag{4.24}$$

The overall optimal policy can be find out by consecutively determining the set of optimal policies for a smaller sub-problems [11].

4.3.4 DP algorithm by backwards recursion

There are many different approaches for the implementation of dynamic programming algorithm and the selection is depends on the particular application which is to be solved. In the dynamic programming optimal control problem, It is required to available a priori knowledge of the desired path or expected operating conditions. The optimal policy can be found with dynamic programming by using the backward recursion from the terminal point and calculates the cost for each successive stage to the initial point.

Initially, the final step ($k = N-1$) is considered and then calculate the optimal policy and called 'final sub-problem'. Next, find out the optimal policy for final sub-problem that involves the final two stages ($k = N-2 : N-1$), and the process is continued until find out the optimal policy for the all stages ($k = N-1 : 1$).

With DP recursive algorithm, The optimal control problem can be solved. Equation 4.8 represents that final stage ($k = N-1$) cost,

$$J^*_{N-1}(x_{N-1}) = \min \{g_{N-1}(N-1, x_{N-1}, u_{N-1})\} \tag{4.25}$$

The minimum cost of the next stage ($k = N-2$) is therefore,

$$J^*_{N-2}(x_{N-2}) = \min \left\{g_{N-2}(N-2, x_{N-2}, u_{N-2}) + J^*_{N-1}\right\} \tag{4.26}$$

Calculating backwards from ($k = N-2$) to the initial stage $k = 0$,, the total cost can be found by recursive equation,

$$J^*_k(x_k) = \min \{g_k(k, x_k, u_k) + J^*_{k+1}\}, \quad k = \{0, \ldots, N-1\} \tag{4.27}$$

where the cost $J^*_0(x_0)$ is the optimal cost of the overall control policy u^*.

Pseudo code

The original pseudo code for the dynamic programming can be written and displayed in [8]. The cost function of dynamic programming becomes discontinuous by adding gears and powertrain modes. The updated pseudo code is shown in Algorithm 1. Therefore, $x_{k,c}$ and $x_{k,d}$ are considered as continuous state (SOC and E) and discontinuous state (γ and p) respectively. In similar way, $u_{k,c}$ and $u_{k,d}$ are continuous control input (T_{ice} and T_{em}) and discontinuous control input (γ_{sel} and p_{sel}) respectively. Let's consider, the powertrain mode state $x_{k,d} = \{1, 2, 3, 4\}$ and control $u_{k,d} = \{1, 2, 3, 4\}$. In similar way, the gear state $x_{k,d} = \{10, 11, 12\}$ and control $u_{k,d} = \{10, 11, 12\}$. The algorithm is based on backwards recursion which is counting backwards from the final stage N where it calculates final cost which is J_N.

Let's consider at stage k, *for all indexes* $i_{x,c}$ and $i_{x,d}$ in (X_{grid}), the dynamic programming algorithm computes next stage value $x_{k+1,c}$ and current step cost g_k by satisfying the state constraints *for all indexes* $i_{u,c}$ and $i_{u,d}$ in (U_{grid}), and choose optimal control. Inside these loops, first condition checks $x_{k,d} = u_{k,d}$ (i.e. if both state and control are powertrain mode 1), and calculates current step cost (g_k) and $x_{k+1,c}$. The second condition checks $x_{k,d} \neq u_{k,d}$ (i.e. if state is powertrain mode 1 and control input is powertrain mode 2). In this condition, the dynamic programming calculates current step cost (g_k) and $x_{k+1,c}$ based on the condition of flag request (switching powertrain mode or gear) according to Table 4.3. Furthermore, current step cost (g_k) is sum of $g_{k,s}$ and $g_{k,t}$ which is calculated by the transition function file. These transition files contains transmission dynamics cost with their feasibility which is explained in next paragraph. After that, the current cost g_k is added into next stage cost $J(x_{k+1,c})$ by satisfying the state constraints and minimized over all control inputs in the discrete set (U_{grid}). Lastly, the dynamic programming algorithm stores optimal control inputs ($u_{opt,c}$ and $u_{opt,d}$) and optimal cost (J_{opt}) according to the state-loops and the stages.

The transition mode consists different function files that compute the cost for engagement or disengagement the engine or electric machine using the transmission and the clutch dynamics with their respective shifting time. The clutch dynamics consists the slipping effect. Slipping occurs when the clutch opens or closes and it affects the next stage velocity state (x_{k+1}). Also, the overall inertial is affected. The clutch dynamics contains the time for opening or closing the clutch. In similar way, the transmission dynamics contains synchronizing effect and its shifting time. On the other hand, these function files also compute the cost for changing gear state by using same the transmission dynamics and the clutch dynamics. The activation of these transition mode function files depend on $engine_{flag}$, $machine_{flag}$ and gear shifting flag, since each flag request has different impact on transmission's synchronizer and opening/closing of the clutch.

Since velocity is one of the states in algorithm, the dynamic programming checks the required distance for opening/closing of the clutch or engagement/disengagement of the engine and the machine. The required distance is calculated by transmission shifting time. If the required distance is greater then step size (h) in the dynamic programming, the transition cost is considered as an infeasible cost.

Conditions of flag request for modes are shown in Table 4.3. However, gear shifting request depends on one flag request which is called gear shifting flag. When current gear is same as select gear then there is no flag request (sustain mode). In similar way, when current gear is not equal to select gear then gear shifting flag gives status and transition mode files activates. In summary, transition mode request affects the powertrain modes change and gear shifting events.

For example, if the powertrain mode one is associated to hybrid mode. Similarly, if the powertrain mode two refers to fully electric mode. During the calculation of step cost from current powertrain mode one to select powertrain mode two ($x_{k,d} = 1$ and $u_{k,d} = 2$), the engine turn off request is activated by $engine_{flag}$ by returning value zero for the engine off. According to flag request, the transition mode function files are activated and compute cost ($g_{k,t}$ and $g_{k,s}$) with next stage state value ($x_{k+1,c}$). The dynamic programming checks the required distance for the transmission and the clutch dynamics through their shifting time with feasibility. After transition mode cost calculation and feasibility, both costs $g_{k,t}$ and $g_{k,s}$ are added together after satisfying state constraints.

As can be seen in Algorithm 1, the next stage cost $J(x_{k+1,c})$ that has been calculated in the previous iteration and denotes the optimal cost-to-go at stage $K + 1$ for the state $x_{k+1,c}$. However, there is in general no guarantee that $x_{k+1,c} \in X_{grid}$, since X_{grid} is only defined for discrete values of the continuous state $x_{k,c}$. This is shown in Figure, where a control input $u_{k,c}$ is applied at state $x_{k,c}$ and leads to a state $x_{k+1,c}$. The method to find the cost-to-go is to use a linear interpolation. A linear interpolation between the values $J(x_{k,c}, k + 1)$ and $J(x_{k+1,c}, k + 1)$, which are already known from the previous iteration, can be used. In this thesis, custom interpolation functions which are optimized for speed and state of charge.

Table 4.3: *Conditions of get flags request function file for engine on or off, and machine on or off in the transition mode. The status 0 implicates a turned off the engine or machine, and 1 indicates turn on the engine or machine.*

Current powertain mode ($x_{k,d}$)	**Select powertain mode ($u_{k,d}$)**	$engine_{flag}$	$machine_{flag}$
1 or 3	2 or 4	0	-
2 or 4	1 or 3	1	-
3 or 4	1 or 2	-	1
1 or 2	3 or 4	-	0

[a] There is no flag request when $x_{k,d} = u_{k,d}$ (sustain mode), since no powertrain mode change during this case and it sustain in current powertrain mode only.

Algorithm 1: Standard Pseudo Code (Scalar Implementation)

```
initialization;
for all indexes i_{x,c} and i_{x,d} in X_grid do
 |  J_N = φ(x_N)
end
for k = N-1 to 1 do
 |  for all indexes i_{x,c} and i_{x,d} in X_grid do
 |   |  x_{k,c} = X_grid(i_{x,c}, k) ;
 |   |  x_{k,d} = X_grid(i_{x,d}, k) ;
 |   |  reset J_k;
 |   |  for all indexes i_{u,c} and i_{u,d} in U_grid do
 |   |   |  u_{k,c} = U_grid(i_{u,c}, k + 1) ;
 |   |   |  u_{k,d} = U_grid(i_{u,d}, k + 1) ;
 |   |   |  if x_{k,d} = u_{k,d} then
 |   |   |   |  /* Sustain mode                              */
 |   |   |   |  [x_{k+1,c} g_k] = f(x_{k,c}, u_{k,c}, k, ...);
 |   |   |   |  /* Check feasibility of state constraints */
 |   |   |  else if x_{k,d} ≠ u_{k,d} then
 |   |   |   |  /* Transition mode files activate           */
 |   |   |   |  [x_{k+1,c} g_{k,t}] = f(x_{k,c}, u_{k,c}, k, ...);
 |   |   |   |  /* Check feasibility of state constraints */
 |   |   |   |  [x_{k+1,c} g_{k,s}] = f(x_{k,c}, u_{k,c}, k, ...);
 |   |   |   |  /* Check feasibility of state constraints */
 |   |   |   |  g_k = g_{k,s} + g_{k,t};
 |   |   |  else
 |   |   |   |  break ;
 |   |   |  end
 |   |   |  J_k = g_k + J(x_{k+1,c}, k + 1);
 |   |   |  if J_k < previous J_k then
 |   |   |   |  J_opt = J_k;
 |   |   |   |  u_{opt,c} = u_{k,c};
 |   |   |   |  u_{opt,d} = u_{k,d};
 |   |   |  end
 |   |  end
 |   |  J(i_{x,c}, i_{x,d}, k) = J_opt;
 |   |  U(i_{x,c}, i_{x,d}, k) = u_{opt,c};
 |   |  U(i_{x,c}, i_{x,d}, k) = u_{opt,d};
 |  end
end
```

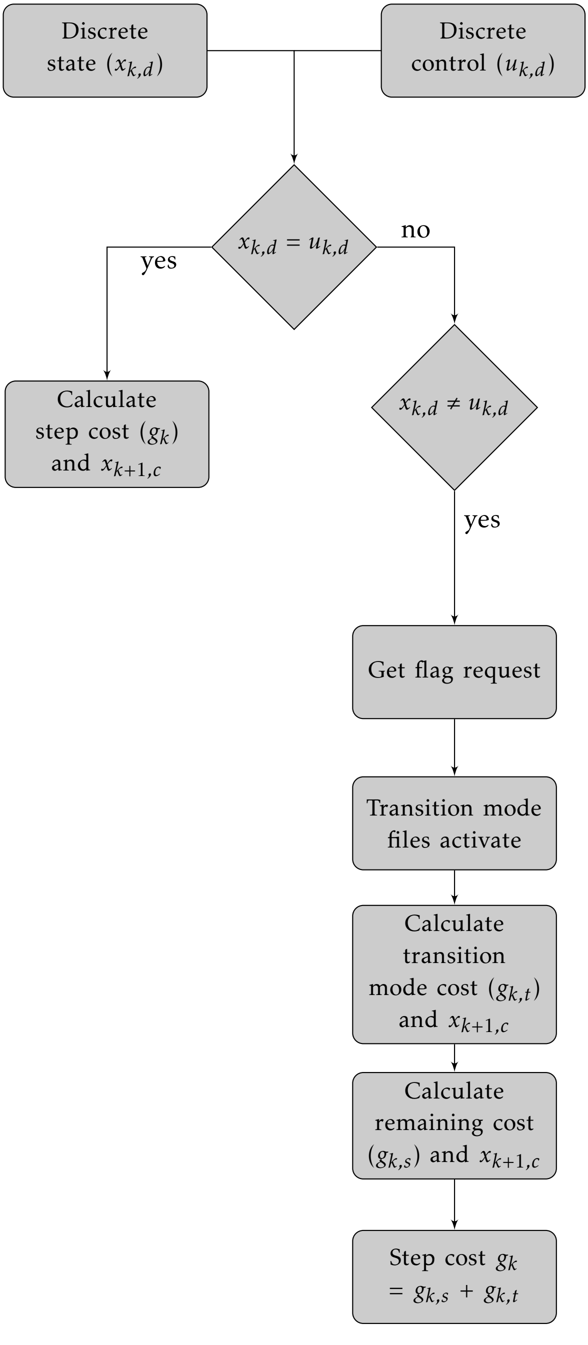


Figure 4.5: *Flow chart of how to calculate step cost with discrete state and control, i.e. powertrain modes. Dynamic programming algorithm check conditions for satisfying state constraints after calculation of both transition mode cost and step cost.*

4.3.5 Tuning the strategic powertrain modes

In this section, the influence of penalizing powertrain modes are analyzed. As earlier described, there are four powertrain modes which decide the vehicle mode. In the dynamic programming, the transition mode cost affects the powertrain mode changes and/or gear shifting events since not all dynamics are modeled. In spite of having that cost, sometimes frequent mode changes are observed which decreases the driveability and shown in Figure 4.6. Therefore, two penalties are used, one is for engine on/off and second is for electric machine on/off. The main aim of using these penalties is to reduce frequent engine on/off request or/and electric machine on/off request in very short instance. In the dynamic programming, the powertrain modes are considered as the state and engine on/off condition is penalized by the transition from a electric mode or full de-clutch mode to hybrid mode or only diesel mode and vice versa. Same for the electric machine on/off condition when the powertrain mode changes from hybrid mode or electric mode to a only diesel mode or full de-clutch mode and vice versa. Furthermore, the state cost with penalties is defined as Equation 4.11;

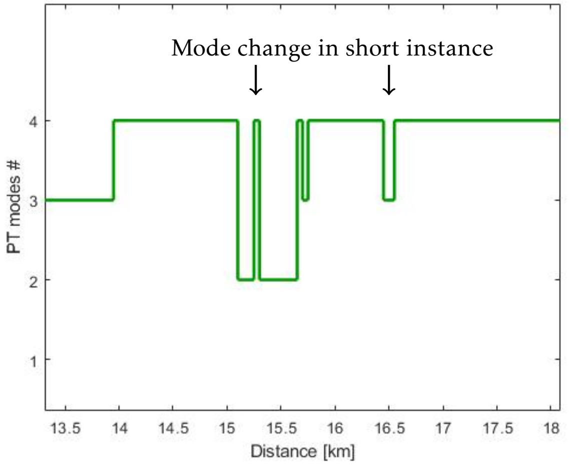


Figure 4.6: *Powetrain mode selection without penalties*

$$J_u\,(x_k, u_k) = g_k + f_{ice}(x_{k,d}, u_{k,d}) + f_{em}(x_{k,d}, u_{k,d}) \tag{4.28}$$

Where f_{ice} and f_{em} are engine on/off penalty function and electric machine on/off penalty function respectively.

During calculation of step cost from current powertrain mode one to select powertrain mode two, engine turn off request is activated by $engine_{flag}$ by returning value zero for engine off and then penalty will be added to the step cost. If $engine_{flag} \neq 1\ or\ 0$ then current powertrain mode is equal to select powertrain mode and no penalty is required.

$$f_{ice}(x_{k,d}, u_{k,d}) = \begin{cases} Pen_{\text{ice}}, & \text{if } engine_{flag} = 1 \; or \; 0 \\ 0, & \text{if } engine_{flag} \neq 1 \; or \; 0 \end{cases} \tag{4.29}$$

$$f_{em}(x_{k,d}, u_{k,d}) = \begin{cases} Pen_{\text{em}}, & \text{if } machine_{flag} = 1 \; or \; 0 \\ 0, & \text{if } machine_{flag} \neq 1 \; or \; 0 \end{cases} \tag{4.30}$$

Where Pen_{ice} and Pen_{em} are considered as numerical penalty value for a powertrain mode changes.

4.4 Reference Re-constructor

The Reference Re-constructor is received optimal state dynamics and control input trajectory from the EMS block based on receding or full horizon. This block provides optimal torque for the engine and electric machine, gear and powertrain mode with respect to the vehicle position to find current request from the vehicle script plant. This block contains interpolation function which is provided single value from control input trajectory.

4.5 Computational time

The dynamic programming has high computational burden due to high number of states and control inputs. The EMS block function file is converted into C++ MEX file. Creating C++ MEX-file lead to acceleration of algorithms. Full C++ MEX file runs in the MATLAB environment. Its inputs and outputs are the same as normal MATLAB function files. This MEX-file reduces computational time about 80%. For receding horizon, the computational time is higher than full horizon due to re-calculation of state trajectory according to the updated length.

5

Results and Analysis

After implementation of different powertrain modes into the dynamic programming algorithm, this chapter gives the results on different driving cycles. Firstly, the results from the HEV controller on different driving cycle is analyzed and selection of powertrain modes penalty has been studied in order to increase the driving performance. Secondly, the dynamic programming based model has been simulated for different receding horizon lengths and then full horizon. The comparison made between different horizon length on the basis of fuel consumption. Also, the behavior of state of charge of the battery has been studied for different horizon length. As a last step, fuel consumption of different drive cycle is described with relative to the base value of existing Volvo's supervisory control strategy.

Since the kinetic energy and state of charge both are states in the dynamic programming, it is not necessary to have the average speed along the full mission is 85 *km/h*. The speed can be decreased on the up-hill to avoid gear down-shift. In similar way, the speed can be increased on the down-hill to store recuperative energy. Therefore, the speed varies based on the slope and average speed along the mission is not always 85 *km/h*. In similar way, the state of charge is not always 40% at the end of the cycle which can be seen in the results; section 5.3.1. The two equations which is used in the Volvo gives compensated value of fuel consumption on the basis of state of charge (40%) and velocity (85 *km/h*).

5.1 Powertrain modes penalty study

In this case study, the selection of numerical penalty has been analyzed, since they have significant impact on the total number of mode changes and fuel consumption. In this study, for each of the two penalty parameters, one value is

Table 5.1: *Simulation results for flat drive cycle over full horizon with best five optimal penalties. Objectives are fuel consumption [l/100 km] and total number of powertrain mode change. Fuel consumption is normalized with respect to highest fuel consumption case of the optimization result.*

Case	Pen_{engine}	$Pen_{machine}$	FC [%]	Total mode change
1	22000	45000	99.38	70
2	40000	60000	99.43	64
3	10714	120000	99.52	63
4	10714	52000	99.29	72
5	46000	94000	99.57	55

[a] Set speed for the simulation is 85 [km/h].

determined. To decide numerical value for penalties, Volvo's inbuilt software has been used with fifteen different values of Pen_{ice}, from 0 to 60000, and Pen_{em}, from 0 to 120000. Both penalties, Pen_{ice} and Pen_{em} are optimized together in the Volvo's inbuilt software. The main objectives of the optimization are fuel consumption and number of powertrain mode changes.

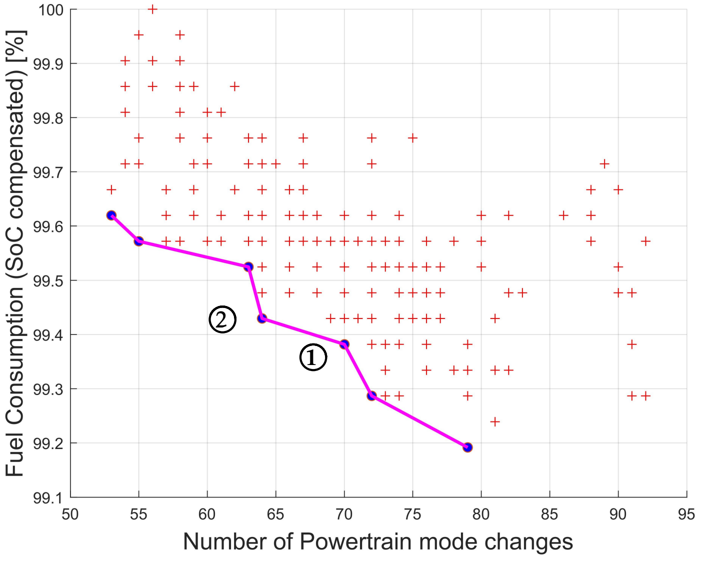


Figure 5.1: *Pareto Estimation over flat drive cycle. Bold numbers represent the best solutions in terms of fuel consumption and total mode changes, and penalty values i.e.,* Pen_{engine} *and* $Pen_{machine}$ *are shown in Table 5.1. Case 1 is considered as best solution.*

Table 5.2: *Simulation results for hilly-2 drive cycle over full horizon with best five optimal penalties. Objectives are fuel consumption [l/100 km] and total number of powertrain mode change. Fuel consumption is normalized with respect to highest fuel consumption case of the optimization result.*

Case	Pen_{engine}	$Pen_{machine}$	FC [%]	Total mode change
1	25000	120000	99.42	128
2	25000	102857	99.51	126
3	0	111428	99.37	141
4	17857	77142	99.24	146
5	46428	102857	99.60	124

[a] Set speed for the simulation is 85 $[km/h]$.

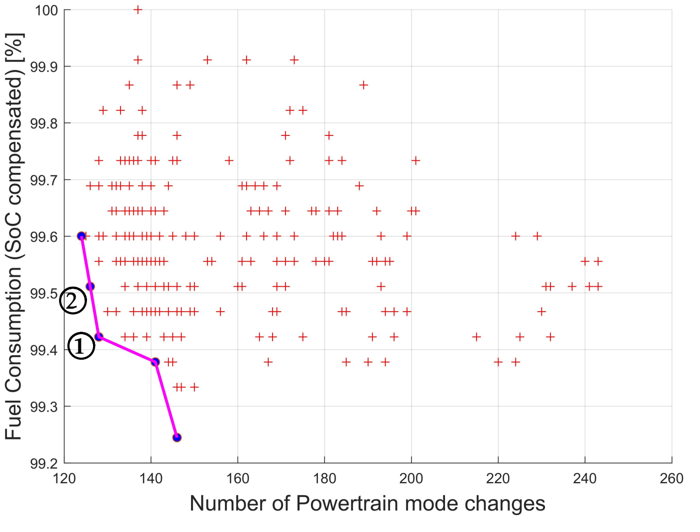


Figure 5.2: *Pareto Estimation over hilly-2 drive cycle. Bold numbers represent the best solutions in terms of fuel consumption and total mode changes, and penalty values i.e., Pen_{engine} and $Pen_{machine}$ are shown in Table 5.1. Case 1 is considered as best solution.*

As shown in Figure 5.3, different penalty values have significant influence on the total number of mode changes. Thus, the total number of powertrain mode changes is reduced from 95 to 65 with increasing penalty value between the different simulation on flat drive cycle. For simulated penalty range, 0.8 % variation in the fuel consumption is observed for same drive cycle. In similar way, the total number of mode changes is decreased from 240 to 125 and fuel consumption variation is 0.75% for hilly-2 drive cycle. So, one value is selected for both penalties Pen_{ice} and Pen_{em}. This selection is done by means of pareto estimation. The optimal values of penalties are shown as case 1 in pareto estimation (Figure 5.1 and 5.2). To conclude on the penalty value Pen_{ice}, the average value of these two drive cycle's results case 1 from Table 5.2 and 5.3 are considered. For Pen_{em}, the same average method is considered. This method could be extended for even more cycles to be more representative of real driving conditions. For the ICE on/off condition, penalty $Pen_{ice} = 25000$ is selected. In a similar way, penalty $Pen_{em} = 85000$ is selected for the EM on/off condition.

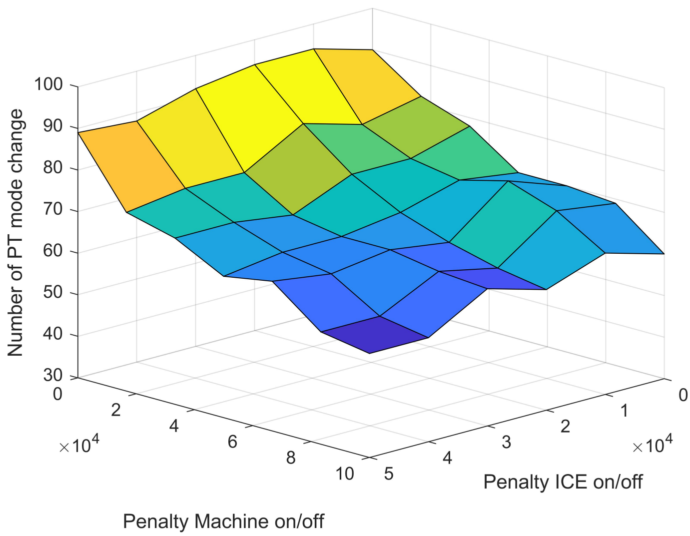
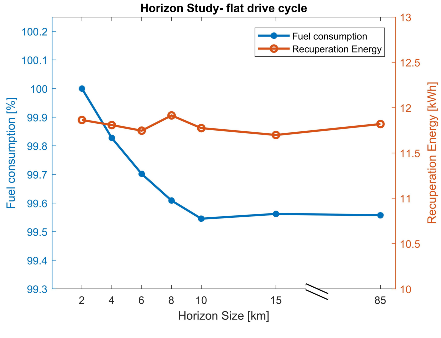


Figure 5.3: *Simulation results over numerical penalty values. Number of powertrain mode changes over flat drive cycle with full horizon.*

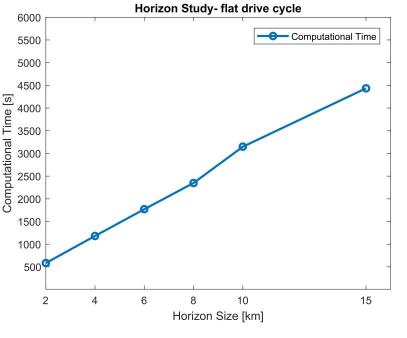
5.2 Horizon effect

In this section, how the size of the prediction control horizon affects the fuel consumption, total recuperation energy and computational burden is investigated. The forward-facing hybrid electric vehicle script model is simulated for different receding horizon lengths from $2km$ to $15km$ and then full horizon length. The effect of horizon length for the flat drive cycle is shown in Figure 5.4. The fuel consumption reduces, as the horizon length increases. The variation in fuel consumption is 0.52% from 2km receding horizon to 85km full horizon length. However, the change in total recuperation energy is negligible since the flat drive cycle has less long down-hill and short horizon length is sufficient to store all recuperation energy instances.

The computational burden increase heavily, specially when the horizon length exceeds 10 km which is shown in Figure 5.4 (b). Therefore, the choice of the horizon length needs to be chosen based on controller capability with optimal fuel consumption and computational time.



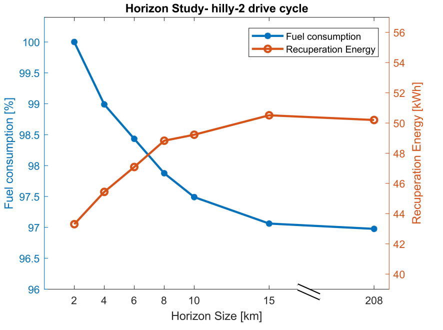
(a) Fuel consumption and total recuperation energy as a function of different receding horizon lengths (2km to 15km) and full horizon length (85km).



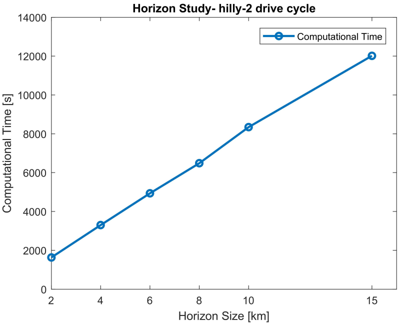
(b) Computational time as a function of different receding horizon lengths (2km to 15km). The computational time for full horizon is less compared to receding horizon since full horizon calculates optimal control only once at the beginning of simulation.

Figure 5.4: *Effect of different horizon length on fuel consumption, total recuperation energy and computational time for flat drive cycle. Fuel consumption value is compensated on basis of state of charge (40%) and velocity (85 km/h). Fuel consumption is normalized with respect to 2 km horizon result.*

The effect of different receding horizon length for hilly-2 drive cycle is shown in Figure 5.5. The fuel consumption decreases, as the horizon length increases. The change in fuel consumption is 3% from shortest receding horizon to full horizon length. Moreover, the reduction in fuel consumption over horizon length is related to the total recuperation energy along the mission. Figure 5.5 shows that the total recuperation energy increases, as the horizon length increases. The reason is that the longer horizon length is sufficient to look ahead down-hill instances. Therefore, the HEV controller depletes battery as much as possible up to its minimum level before down-hill instance. As a result, battery stores full recuperation energy and later use that energy along the driving mission in the longer horizon length. This reduces fuel consumption with increasing horizon length.



(a) Fuel consumption and total recuperation energy as a function of different receding horizon length and full horizon length (208km).



(b) Computational time as a function of different receding horizon length. The computational time for full horizon is less compared to receding horizon since full horizon calculates optimal control only once at the beginning of simulation.

Figure 5.5: *Effect of different receding horizon length on fuel consumption, total recuperation energy and computational time for hilly-2 drive cycle. Fuel consumption value is compensated on basis of state of charge (40%) and velocity (85 km/h). Fuel consumption is normalized with respect to 2 km horizon result.*

Since hilly-2 drive cycle has many down-hill instances, the short horizon length is not sufficient to look ahead the down-hills and can not deplete the battery to its minimum level. So, the battery reaches its maximum level earlier and can not store full recuperative energy on the down-hill. Therefore, the choice of the horizon length needs to be select based on recuperation energy and computational time. Figure 5.5 (b) shows the computational time and it increases, as horizon length increases.

In all horizon lengths, the HEV controller drains the battery up to its minimum level before two long down-hills which is shown in Figure 5.6. The depletion rate of the battery increases, as the horizon length increases. Figure 5.6 shows the state of charge comparison between three different receding horizon and full horizon length. Before first long down-hill, the battery is not depleting to its minimum level for 2 km horizon. However, 15 km and full horizon length satisfy the optimality condition and depletes the battery to its minimum level. In similar way, 15 km and full horizon also has highest depletion rate of battery before second long down-hill by enabling pure electric drive. However, 15 km horizon length is computationally heavy and 8 km horizon length can cover almost full recuperation energy before first long down-hill. Therefore, 8 km receding horizon is suitable for the simulation.

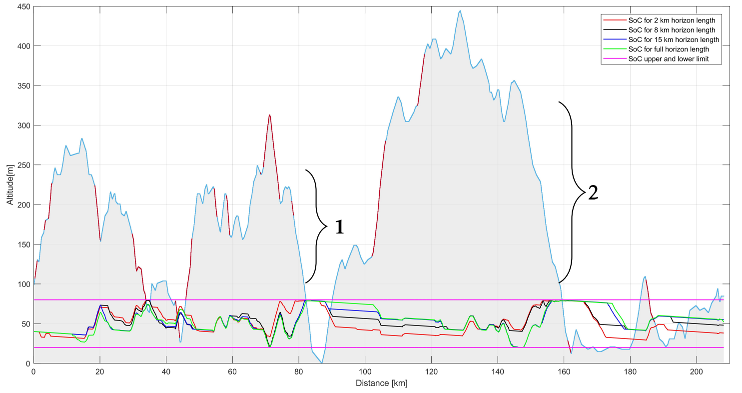


Figure 5.6: *Comparison of State of charge between two receding horizon (2 km and 15 km) and full horizon length over hilly-2 drive cycle. Bold number 1 and 2 represent the first and second long down-hill.*

5.3 Simulation results from different drive cycles

In this section, the important results of simulated HEV model on three different drive cycle namely predominantly flat, hilly-1 and hilly-2 will be studied with the set speed 85 *km/h*. The simulation has been carried out for full and receding horizon length (*8 km*). The fuel consumption for different drive cycles with different modes are summarized from Table 5.3 to 5.8. The allowed velocity interval is [75,90] *km/h* where the lower limit is extended when it is not reachable. The allowed state of charge interval is [20,80]%. The allowed gears are {10,11,12 }. The simulation has been carried out for two different allowed modes interval; first {1,2,3,4} and second {1,2}.

5.3.1 Drive cycle- predominantly flat

This drive cycle is 85 km long with flat road and frequent small down-hills. As can be seen in Figure 5.7 and 5.8, the state of charge is near to 40% for full horizon length and 45% for receding horizon length respectively at the end of drive cycle. However, the mode 4 is often activated for both full and receding horizon, since the usage of the engine and the motor are less due to the reduction of cost function, specially at the beginning of drive cycle (0 to 20 km). In addition, the engine is only activated on up-hill. Another prime factor is that the machine is switched on only when the long down hill comes, since the battery can store recuperative energy. In addition, the hybrid mode is activated when the usage of the engine and the motor is necessary to avoid the down-shifts during sharp up-hill at 30 km.

The difference in fuel consumption for full and receding horizon has been found to be of very small magnitude due to less number of down-hills. For receding horizon, *8 km* length is sufficient to store all recuperative energy events. However, the fuel consumption with four powertrain modes has decreased in comparison with existing Volvo strategy. The one reason in the reduction of fuel consumption is due to the gear shifting in all modes which was not possible before in the Volvo's existing strategy.

Table 5.3: *Fuel consumption results for the full horizon and receding horizon with all four modes, expressed relative to the base value of 100% set for the existing strategy.*

Horizon length	FC [%]	Total mode change
Full Horizon	98.802	62
Receding Horizon	98.6446	73

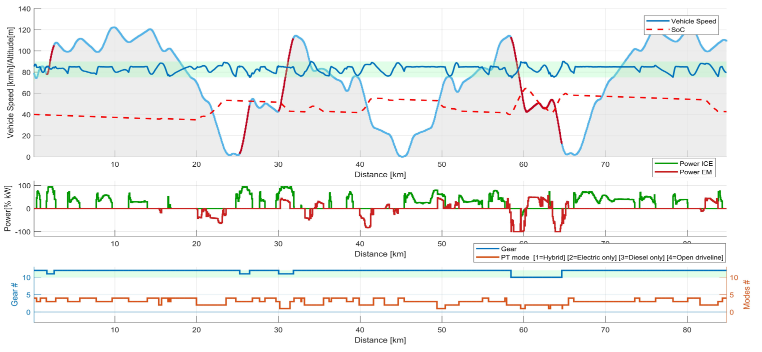


Figure 5.7: *Simulation result for full horizon length on flat drive cycle with all four modes.*

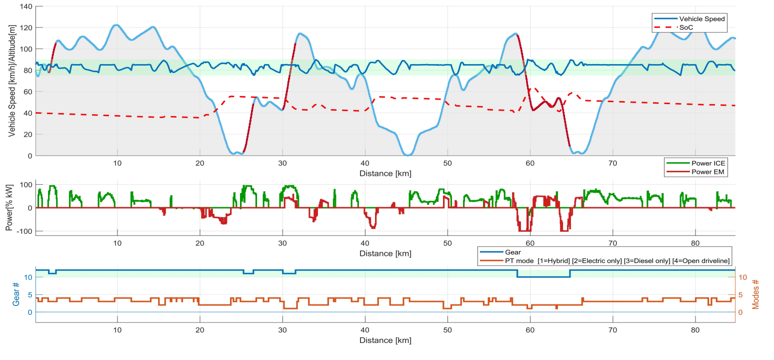


Figure 5.8: *Simulation result for receding horizon length (8 km) on flat drive cycle with all four modes.*

According to Table 5.4, the fuel consumption is 0.74% and 0.82% decreased for full and receding horizon with all four modes in the comparison with two modes (Hybrid mode and electric mode only). Since, the disengaging electric machine from driveline has benefit to reduce the cost function. However, Figure 5.9 and 5.10 represents the results for full and receding horizon with two modes. Declutch the electric machine option is not possible in these two modes, the use of machine is higher than results with all modes.

Table 5.4: *Fuel consumption results for the full horizon and receding horizon with all four modes, expressed relative to the base value of 100% set for two modes. (Hybrid mode and full electric mode)*

Horizon length	FC [%]	Total mode change
Full Horizon	99.26	40
Receding Horizon	99.18	40

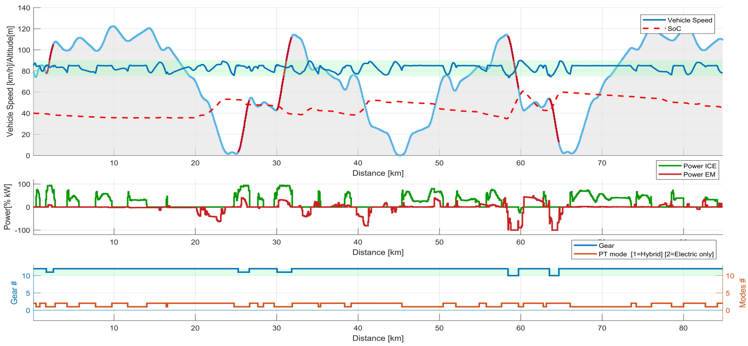


Figure 5.9: *Simulation result for full horizon length on flat drive cycle with two modes. (Hybrid mode and full electric mode)*

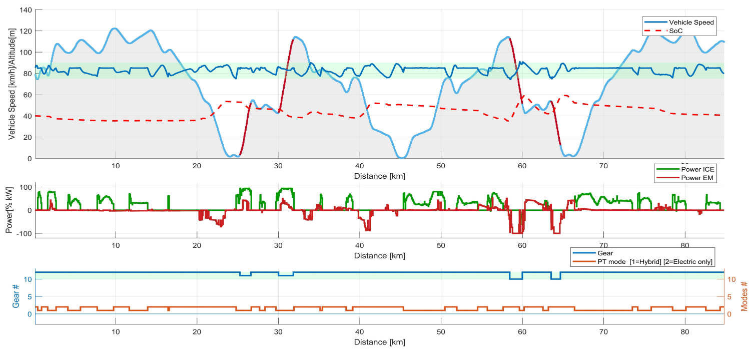


Figure 5.10: *Simulation result for receding horizon length (8 km) on flat drive cycle with two modes. (Hybrid mode and full electric mode)*

5.3.2 Drive cycle- hilly-1 road

The hilly-1 drive cycle has frequent sharp up and down hills and the total distance is 58 km. The simulation results can be seen in Figure 5.11 and 5.12 for both full and receding horizon with all four modes. Only diesel mode and open driveline mode are less activated due to sharp up and down-hills. The hybrid mode is mainly activated since the machine torque provides additional torque requirement to the driveline in order to avoid down-shifts. However, the fuel consumption is reduced related with Volvo's existing strategy for both full and receding horizon length.

Fuel consumption is decreased up to 0.5% for both full and receding horizon with all four modes in comparison with two modes (hybrid mode and electric only) and shown in Table 5.6. The reason for decrement in the fuel consumption is that the machine-off command is available when all four modes are activated. The simulation results with only two modes can be seen in Figure 5.13 and 5.14.

Table 5.5: *Fuel consumption results for the full and receding horizon with all four modes, expressed relative to the base value of 100% set for the existing strategy.*

Horizon length	FC [%]	Total mode change
Full Horizon	98.44	49
Receding Horizon	99.16	55

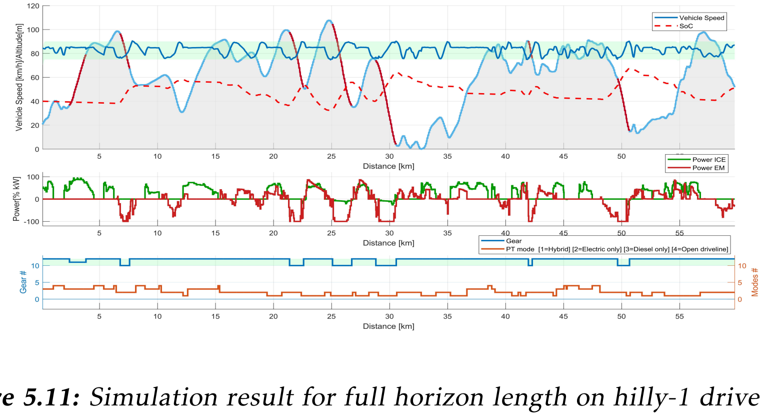


Figure 5.11: *Simulation result for full horizon length on hilly-1 drive cycle with all four modes.*

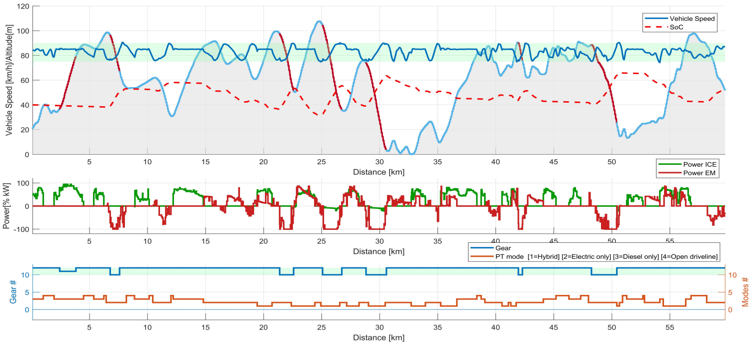


Figure 5.12: *Simulation result for receding horizon length (8 km) on hilly-1 drive cycle with all four modes.*

Table 5.6: *Fuel consumption results for the full horizon and receding horizon with four modes, expressed relative to the base value of 100% set for two modes. (Hybrid mode and full electric mode)*

Horizon length	FC [%]	Total mode change
Full Horizon	99.51	47
Receding Horizon	99.60	47

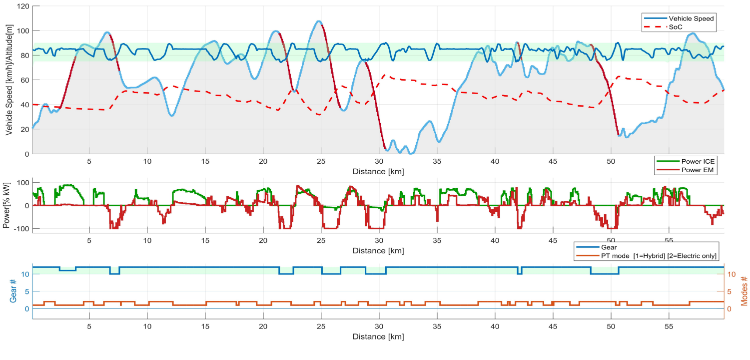


Figure 5.13: *Simulation result for full horizon length on hilly-1 drive cycle with two modes. (Hybrid mode and full electric mode)*

Figure 5.14: *Simulation result for receding horizon length (8 km) on hilly-1 drive cycle with two modes. (Hybrid mode and full electric mode)*

5.3.3 Drive cycle- hilly-2 road

The hilly-2 drive cycle length is 215 km with long up and down hills. The simulation results for full and receding horizon length with all modes is shown in Figure 5.15 and 5.16 respectively. There are two long down hills, first from 76 km to 88 km and second from 145 km to 160 km. Therefore, the battery's state of charge is reached at 80% twice and it could not store more recuperative energy. On the other-hand, the open driveline mode is less activated in hill-2 drive cycle same as previous drive cycle since the traction power is required in order to drive on the hills.

For hilly-2 drive cycle, 0.7% increment in fuel consumption for full horizon compare to receding horizon since full horizon covers all long down-hills and stores all recuperative energy instances. In addition, the fuel consumption is reduced in comparison with Volvo's strategy.

Table 5.7: *Fuel consumption results for the full horizon and receding horizon with all four modes, expressed relative to the base value of 100% set for the existing strategy.*

Horizon length	FC [%]	Total mode change
Full Horizon	99.45	131
Receding Horizon	99.26	155

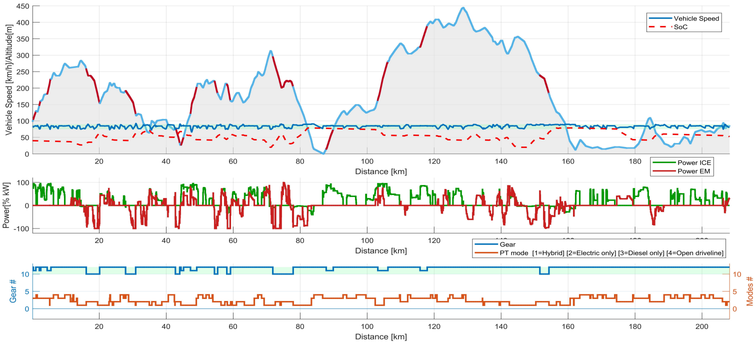


Figure 5.15: *Simulation result for full horizon length on hilly-2 drive cycle with all four modes.*

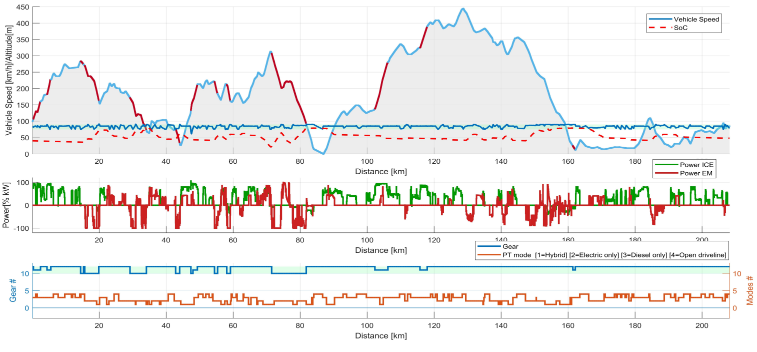


Figure 5.16: *Simulation result for receding horizon length (8 km) on hilly-2 drive cycle with all four modes.*

Figure 5.17 and 5.18 represent the results with enabling two powertrain modes, i.e. hybrid mode and electric mode only. The reduction in fuel consumption is 0.2% and 0.51% for full and receding horizon with all modes in comparison with two modes. This is predominantly because electric machine dis-engaging reduces cost function.

Table 5.8: *Fuel consumption results for the full horizon and receding horizon with four modes, expressed relative to the base value of 100% set for two modes. (Hybrid mode and full electric mode)*

Horizon length	FC [%]	Total mode change
Full Horizon	99.82	89
Receding Horizon	99.49	91

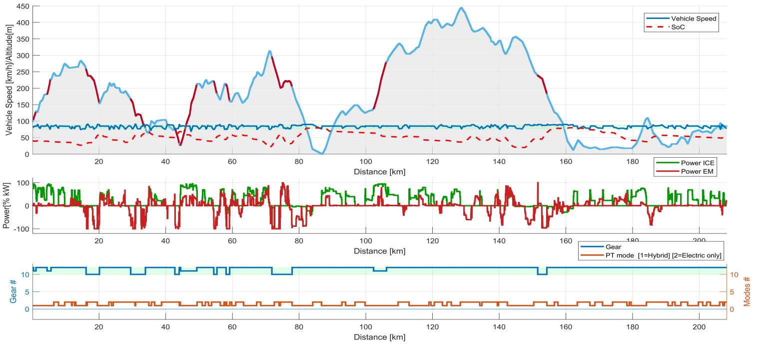


Figure 5.17: *Simulation result for full horizon length on hilly-2 drive cycle with two modes. (Hybrid mode and full electric mode)*

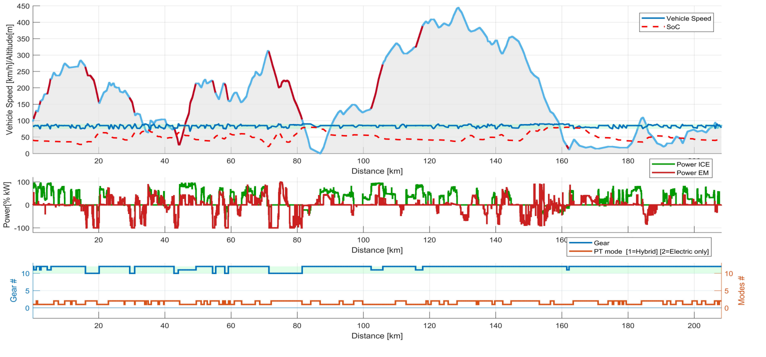


Figure 5.18: *Simulation result for receding horizon length (8 km) on hilly-2 drive cycle with two modes. (Hybrid mode and full electric mode)*

In summary, the fuel consumption is reduced for full horizon in the comparison with receding horizon length over all drive cycle. In similar way, the fuel consumption is decreased related to Volvo's strategy. According to Table 5.4 and 5.8, the fuel consumption is decreased more on flat drive cycles than hilly-1 and hilly-2 cycle when all four modes are activated in comparison with only two modes (hybrid mode and full electric mode). The reason of decrement is that enabling all powertrain modes on flat drive cycle has great benefits over two modes since activating all modes have functionality to de-clutch the machine from the driveline often than hilly-2 drive cycle.

6
Conclusion

This chapter provides the main conclusion from the preceding chapters and then the recommendation for a future work. In the simulated model, two types of drive cycles, i.e. flat road drive cycle and hilly road drive cycle can be used and the behavior of the vehicle plant model can be studied. The HEV controller is capable of running the vehicle plant model in hybrid mode, pure electric mode, pure diesel mode and open drive-line mode. The status of the battery, electric machine, engine, gearbox, clutch and drive shaft can be studied at each moment.

- **HEV controller:-** Dynamic programming is successfully implemented as controller and provides the outputs to the vehicle plant model with respect to the vehicle position. The EMS block provides an optimal trajectory of the state dynamics with optimal control inputs based on receding horizon length or full horizon length.
- **Fuel consumption:-** The difference in the fuel consumption between receding and full horizon is less in flat drive cycle which is 0.09%, and more in hilly-2 drive cycle which is 0.7%. It concludes that full horizon length is more efficient than receding horizon in terms of fuel consumption and recuperation energy. However, it is difficult to predict whole drive cycle accurately in real-world condition. On the other hand, 1.2% fuel consumption is reduced with compared to Volvo's strategy on flat drive cycle and 0.55% on hilly-2 drive cycle respectively for full horizon length. Moreover, the fuel consumption reduces 1.4% and 0.74% on flat and hilly-2 drive cycle respectively for 8 km receding horizon.
- **Powertrain modes:-** Powertrain modes are discrete states and control inputs in dynamic programming. Both discrete functions are successfully implemented and penalty values are optimized. Having powertrain modes

as states give flexibility to the gear shifting which reduces the fuel consumption.

- **Horizon Study:-** After horizon study on the fuel consumption and total recuperation energy, it is concluded that the depletion rate of battery before two long down-hill is high in 15 km and full horizon length. But due to computational burden, 8 km horizon length is sufficient. However, the problem which is mentioned in section 1.5 is solved by implementation of powertrain modes in the dynamic programming. After comparison of state of charge from Figure 1.7 and 5.7, it is clearly observed that the battery depletes more in dynamic programming before the down-hill comes, compared to the existing method.

7 Future Work

There is more work has to be done within the HEV simulations, and here one improvement is given for the future work.

- In order to increase accuracy for the dynamic programming, the sensitivity analysis of control and state grid can improve the results.
- In dynamic programming, the continuous states and control inputs are based on discrete decision processes and therefore continuous inputs and states has to be approximated by a discrete-value system. This requires a perfect grid selection and number of elements in it. One way of improving dynamic programming is to implement continuous optimization for torque-split on different control layer.

Bibliography

[1] Regulation (eu) 2019/1242 of the european parliament and of the council of 20 june 2019 setting co2 emission performance standards for new heavy-duty vehicles and amending regulations (ec) no 595/2009 and (eu) 2018/956 of the european parliament and of the council and council directive 96/53/ec. 198:202–240, 7 2019. doi: OJL198,25.7.2019.

[2] Volvo AB. Hybrid electric vehicle control- volvo technical report.

[3] Daniel Ambühl and Lino Guzzella. Predictive reference signal generator for hybrid electric vehicles. *Vehicular Technology, IEEE Transactions on*, 58: 4730 – 4740, 12 2009. doi: 10.1109/TVT.2009.2027709.

[4] Richard Bellman. Dynamic programming. *Princeton University Press*, 6, 1972.

[5] H.A. Borhan, A. Vahidi, A.M. Phillips, M.L. Kuang, and I.V. Kolmanovsky. Predictive energy management of a power-split hybrid electric vehicle. pages 3970–3976, 2009. doi: 10.1109/ACC.2009.5160451. cited By 167.

[6] Boli Chen, Simos Evangelou, and Roberto Lot. Hybrid electric vehicle two-step fuel efficiency optimization with decoupled energy management and speed control. *IEEE Transactions on Vehicular Technology*, 10 2019. doi: 10.1109/TVT.2019.2948192.

[7] Jianhua Guo, Wei Zhang, Cui Liu, and Liang Chu. Adaptive model predictive control strategy of hybrid electric bus based on soc programming. pages 796–799, 05 2016. doi: 10.1109/ITNEC.2016.7560470.

[8] Lino Guzzella and Antonio Sciarretta. *Vehicle Propulsion Systems: Introduction to Modeling and Optimization*. 01 2007. doi: 10.1007/978-3-642-35913-2.

[9] G. Heppeler, Marcus Sonntag, and O. Sawodny. Fuel efficiency analysis for simultaneous optimization of the velocity trajectory and the energy management in hybrid electric vehicles. *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 19:6612–6617, 01 2014.

[10] Lars Johannesson Mårdh, Nikolce Murgovski, Erik Jonasson, Jonas Hellgren, and Bo Egardt. Predictive energy management of hybrid long-haul trucks. *Control Engineering Practice*, 41, 08 2015. doi: 10.1016/j.conengprac.2015.04.014.

[11] Ulf Jönsson. Optimal control. *Royal Institute of Technology*, 2010.

[12] Viktor Larsson, Lars Johannesson Mårdh, and Bo Egardt. Analytic solutions to the dynamic programming subproblem in hybrid vehicle energy management. *Vehicular Technology, IEEE Transactions on*, 64:1458–1467, 04 2015. doi: 10.1109/TVT.2014.2329864.

[13] C.-C. Lin, J.-M. Kang, J.W. Grizzle, and H. Peng. Energy management strategy for a parallel hybrid electric truck. volume 4, pages 2878–2883, 2001. cited By 180.

[14] S. Onori, L. Serrao, and G. Rizzoni. Adaptive equivalent consumption minimization strategy for hybrid electric vehicles. volume 1, pages 499–505, 2010. doi: 10.1115/DSCC2010-4211. cited By 85.

[15] G. Paganelli, Sebastien Delprat, Thierry-Marie Guerra, J. Rimaux, and Jean-Jacques Santin. Equivalent consumption minimization strategy for parallel hybrid powertrains. *IEEE Vehicular Technology Conference*, 4:2076 – 2081 vol.4, 02 2002. doi: 10.1109/VTC.2002.1002989.

[16] Abdoulaye Pam, Alain Bouscayrol, Philippe Fiani, and Frederic Noth. Rule-based energy management strategy for a parallel hybrid electric vehicle deduced from dynamic programming. pages 1–6, 12 2017. doi: 10.1109/VPPC.2017.8331055.

[17] Rongjun Zhang and Yaobin Chen. Control of hybrid dynamical systems for electric vehicles. In *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, volume 4, pages 2884–2889 vol.4, 2001.

[18] M. Salman, Niels J. Schouten, and Naim A. Kheir. Control strategies for parallel hybrid vehicles. volume 1, pages 524–528, 2000. cited By 90.

[19] A. Sciarretta and L. Guzzella. Control of hybrid electric vehicles. *IEEE Control Systems Magazine*, 27(2):60–70, 2007.

[20] A. Sciarretta, M. Back, and L. Guzzella. Optimal control of parallel hybrid electric vehicles. *IEEE Transactions on Control Systems Technology*, 12(3): 352–363, 2004.

[21] Jinzhu Wu, Bingzhao Gao, Qing Zheng, and Hong Chen. Optimal equivalence factor calculation based on dynamic programming for hybrid electric vehicle. pages 6640–6645, 10 2017. doi: 10.1109/CAC.2017.8243973.

[22] Fengqi Zhang, Lihua Wang, Serdar Coskun, Hui Pang, Yahui Cui, and Junqiang Xi. Energy management strategies for hybrid electric vehicles: Review, classification, comparison, and outlook. *Energies*, 13:3352, 06 2020. doi: 10.3390/en13133352.