CHALMERS

Generation of stochastic drive cycles

Master of Science Thesis

DAVID NYDAHL

MARTIN RYDSTEDT

Department of Signals and Systems
CHALMERS UNIVERSITY OF TECHNOLOGY
Göteborg, Sweden, 2009
Report No. EX028/2009

ABSTRACT

The purpose of this thesis project is to develop a tool that generates stochastic drive cycles used in simulation for truck development. Using these drive cycles it is possible to tune design parameters, using simulation, without risking cycle beating.

To allow the user to control the characteristic of the generated drive cycle, key parameters describing the characteristic of a drive cycle are identified. Special care is taken to include characteristics that affect the fuel consumption of a Hybrid Electrical Vehicle.

The drive cycle generator is developed in Matlab and Simulink and is based on Markov processes. The output of the tool is a drive cycle that includes; speed limit, topography, Curvature, impact of the surrounding traffic, planned and unplanned stops. The outputted drive cycle fits the present format of drive cycles at Volvo Powertrain.

Drive cycles can be generated according to the users' choice of values for the identified key parameters defining the characteristic. There is also a choice in the tool for the user to create drive cycles by the characteristics of some typical Swedish roads. It is possible to select a saved characteristic matching city roads, roads of the surroundings of a city, highway or roads from the northern parts of Sweden. Statistics to get these characteristics is collected from NVDB.

Tests comparing the average fuel consumption on drive cycles generated from the same key parameters show that during the first 100 kilometres the differences in average fuel consumption gets smaller. After 100 kilometres 4 of 5 drive cycles give an average fuel consumption that lies within ± 5% of their combined mean value.

Keywords: Cycle beating, drive cycles, drive cycle generation

Sammanfattning

Syftet med det här examensarbetet är att utveckla ett verktyg som genererar stokastiska körcykler för användning vid simulering under lastbilsutveckling. Användning av dessa körcykler gör det möjlig att justera designparameterar med hjälp av simulering utan att riskera så kallad cycle beating.

För att användaren skall kunna styra utseendet på de genererade körcyklerna, identifieras nyckelparametrar som beskriver karakteristikan hos en körcykel. Nyckelparametrarna väljs så att karakteristika som påverkar bränsleförbrukningen för hybridlastbilar inkluderas.

Körcykelgeneratorn utvecklades i Matlab och Simulink och baseras på markovprocesser. De genererade körcyklerna inkluderar; hastighetsbegränsningar, topografi, kurvatur, påverkan från omgivande trafik samt planerade och oplanerade stopp. Körcyklerna genereras enligt Volvo Powertrains format för körcykler.

Körcykler kan genereras efter användarens val av värden på de identifierade nyckelparametrarna. Det är även möjligt för användaren att generera vägar efter karakteristiken på några olika typer av svenska vägar. I körcykelgeneratorn finns sparad karakteristik för stadskörning, körning i en stads omgivningar, motorvägar och landsvägenskörning från norra Sverige. Statistik för dessa vägar har hämtas från NVDB (Nationell vägdatabas).

Tester där medelbränsleförbrukning för en hybridlastbil jämförs på olika körcykler genererade från samma nyckelparametrar visar att under de första 100 km minskar skillnaden i medelbränsleförbrukning mellan körcyklerna. Efter 100 km ger 4 av 5 körcykler en medelbränsleförbrukning som ligger inom ± 5% av deras gemensamma medel värde.

Nyckelord: Cycle beating, körcykler, körcykelgenerering.

PREFACE

This is a thesis project made by Martin to get a Master of Science in engineering physics at Uppsala University and David to get a Master of Science from the Automation and Mecatronics program at Chalmers University of Technology in Gothenburg. The project has been carried out at Volvo Powertrain, Hybrid Technology in Gothenburg.

Martin had the responsibility for the statistical calculation used in every subsystem and to develop the subsystems in Simulink. Responsible for the Java programming and extraction of the data from NVDB and the development of the graphical user interface which included putting all the subsystems together is David. The rest of the material in this thesis has been developed with a common responsibility.

We would like to thank our supervisor Mattias Åsbogård for the support during the project and also the helpful people at Hybrid Technology, Volvo Powertrain. Outside Volvo we want to thank Professor Jonas Sjöberg and PhD student Lars Johannesson, at Chalmers University of Technology, for valuable inputs and discussions. We also want to thank Professor Bengt Carlsson for good feedback on the report.

David & Martin
Göteborg, March 2009

Abbreviation list

ESS	Energy Storage System
GM model	General motors model
GGM model	Generalized general motors model
GPS	Global Positioning System
GTA	Global Truck Application
GUI	Graphical user interface
HEV	Hybrid electrical vehicle
NVDB	Nationell Vägdatabas (National road data base)
SSS theorem	Side-Side-Side Triangle Congruence Theorem

1 Introduction

The use of simulation in product development for the truck industries is becoming more and more important. Why that is so and what it is used for is described in the background chapter. Why the use of too few drive cycles might lead to misleading results from simulation and what can be done to prevent this is covered during the problem description. More details about what is included in this thesis project can be found in the purpose and aim and in the delimitations chapter.

1.1 Background

For the car and truck industry, computer simulation is an important tool in product development. Both to be able to test different configurations without having to build prototypes for every change and in order to efficiently test different control programs and parameters. Simulation can be used for a lot of different tests, including fuel consumption, emissions and load on the powertrain.

The early stages of a product development cycle are simulation intensive. These are the stages were decisions of what components to use in the power train are made. The powertrain includes engine, gearbox, driveshaft and final gear. The powertrain of a HEV (Hybrid Electrical Vehicle) also includes an electric motor and an ESS (Energy Storage System) more details in Chapter 1.1.1. To be used in simulation there is a model for each component, these models are connected to a bigger model describing the complete powertrain.

The control system of a HEV, that decides when to use the diesel engine and when to use the electric motor, has a lot of parameters. Trying to find the best possible value for these parameters, several different values are tested in simulations. This makes simulation even more important for development of a HEV.

When a simulation is run a drive cycle is used to define the road environment and how the vehicle shall behave. This includes acceleration and deceleration of the vehicle and slope of the road etc. Before a simulation is run a decision of which drive cycle to use has to be made. The drive cycle should match the driving conditions where the vehicle will be used in real application, as good as possible, to get relevant results. For example in very hilly areas stronger engines are needed than in flat areas and buses making a lot of stops can have a smaller battery than a truck used for driving long distances without stopping.

1.1.1 Hybrid Electric Vehicles

A HEV is a vehicle which combines a conventional propulsion system with a rechargeable ESS to achieve better fuel economy than a conventional vehicle. It includes both a diesel engine and an electric motor. The ESS is used as a buffer to let the diesel engine work at a more efficient work load and to store energy generated from breaking. The electric motor can also be used to help the diesel engine at high loads or to start it. This means that a smaller engine can be used, resulting in less energy loss and that the engine can be shut down at low speeds and short stops.

To lower the fuel consumption the battery is used to store overflow energy from braking. Sometimes the battery is charged to get a more optimal work point (rotation speed vs. load) for the engine. At high loads the energy buffer is used to, via the electric motor, help the engine. This occurs at uphill slopes and acceleration.

The fuel consumption of a HEV is affected more by different road characteristics than a traditional vehicle. If for example, downhill slopes and decelerations are too long the battery will not be big enough to store the surplus energy. Likewise if uphill slopes and accelerations are to long the stored energy will not be sufficient and the vehicle will have to use only the diesel engine. Since a smaller diesel engine is used to work with the electric motor, the performance of the truck will then be weak. Also, in these situations the fuel savings compared to a regular vehicle will not be as good as when driving over several small hills or using short accelerations / decelerations.

1.2 Problem description

When components and control parameters are selected a lot of simulations are run. If the same drive cycle is used for every simulation, the powertrain and parameter selection will be optimised for that exact drive cycle which does not mean that it is optimised for another drive cycle, even if it has about the same characteristics. This is called cycle beating.

For a traditional powertrain the decisions from the simulations are mainly which components to select. As the number of decisions increases with the rising number of software parameters, the risk of cycle beating increases. For a HEV the parameters that controls the power flow control program, are both numerous and important. This makes cycle beating a growing threat.

To avoid cycle beating several different drive cycles should be used. Since the drive cycles used must represent the environment where the vehicle will be used there might be a lack of representative drive cycles to use.

One solution to this problem is to have a tool that generates artificial drive cycles. Since the drive cycles used in simulation must be representative for the environment where the vehicle will be used, a way to define the characteristics of a drive cycle must be found. When identifying key parameters describing the driving environment, it must be made sure that what affects the fuel consumption and performance of a HEV is included. There are probably some things that affect a HEV much but are not important for traditional vehicles. For example the number of hills, see Chapter 1.1.1.

1.3 Purpose and aim

The goal for this project is to create a tool that generates stochastic drive cycles matching a given driving environment. The task of creating and the demands on this tool can be broken down to 4 points.

- Parameters defining the characteristic of the outputt drive cycle shall be identified. The parameters shall include HEV important characteristics.
- The generated drive cycles shall be stochastic in the way that they are different from other generated drive cycles matching the same parameters.
- The user shall be able to set the input parameters freely. There shall also be possible to choose between some setups matching real Swedish road and traffic environments.
- Generated drive cycles shall be compatible with the simulation environment used at Volvo Powertrain today.

1.4 Delimitations

In this project, focus will be on modelling the road and traffic environment. The behaviour of the driver and the vehicle is modelled elsewhere. This means that the outputted drive cycle does not contain information about driver and vehicle specific behaviour, but information describing the road and traffic environment. The road and traffic environment includes topography, curvature, and speed limit of the road. The speed limit is therefore not the actual speed of the vehicle in the simulation, but how fast the driver is allowed to drive. The driver and vehicle model will have to handle accelerations and decelerations of the vehicle.

In this way it is possible to use the same drive cycle for different types of vehicles and to use different drivers with different behaviour.

2 Thesis outline

Here follows a description of what is included in this thesis project and how those different parts are approached. In the road data base section there is a description of the data used for statistical calculations for the example roads.

2.1 Work outline

A drive cycle consists of several parts; speed limit, reference speed, altitude, curvature, stop times and position. The speed limit is the limit marked by signs while reference speed is to be considered as the limitations from road and traffic environment, for example tight curves or traffic jam.

The drive cycle generator consists of the following parts (subsystems), speed limit, topography, curvature, planned stops unplanned stops and traffic. Figure 2-1 shows an overview of how the subsystem collaborates with each other and what parts are included in the outputted drive cycle. In Chapter 7 details on how the subsystems are designed and implemented can be found. There is also a description of how they are connected to each other through the GUI (graphical user interface) and finally a short description of the format of the generated drive cycle.

In Chapter 4 key parameters defining the characteristics of a drive cycle are identified through reasoning about what represents energy and effect in a drive cycle. Verification of the key parameter selection is described in Chapter 8 and these results are discussed in Chapter 9.

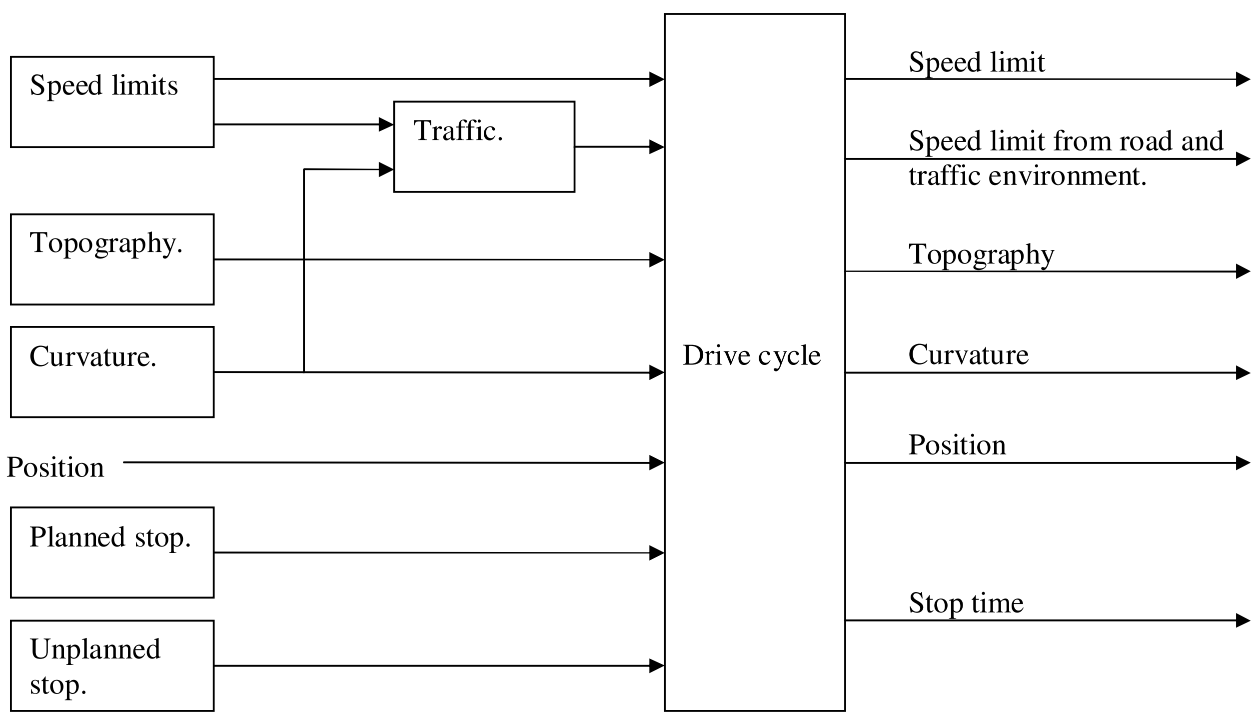

Figure 2-1. Outline of the drive cycle generator.

One important demand is for the generated drive cycles to be stochastic, that is, to always get different drive cycles. This leads to the use of Markov processes in the speed limit, topography and curvature subsystems. The theory behind the Markov process is found in Chapter 3. Here the underlying theories for traffic modelling also are described. Two different traffic models are described, a spring model and the expanded GGM (generalized general motors) model. The last one is the one implemented because it is better documented and easier to parameterize.

The data used for the selection of typical Swedish roads are collected from NVDB (Nationell vägdatabas), see Chapter 2.2 for more information. The data were extracted with a search algorithm implemented in Java. Details are described in Chapter 3.3. Processing the data before calculating the Markov matrices for the speed limits, topography and curvature subsystems were made in different ways for all three cases. The final calculation of the Markov matrix were made in the same way. The calculations are presented in Chapter 6, together with the advanced mode calculations, which are a bit more complex than the standard mode calculation. First a speed limit or topography profile is created and then used for the Markov matrix calculation. In this chapter also a description of the calculations of the probabilities of unplanned stops can be found.

2.2 Road database

NVDB is a database from the Swedish Vägverket. Vägverket is an authority responsible for all public roads in Sweden. The database contains data for all Swedish roads, including road geometry and a number of utilities such as speed limits, maximum vehicle weights, bridges etcetera.

The geometric specification of roads in NVDB is represented by 3-dimensional coordinates along the road. The coordinates are chosen so that if a straight line is drawn between them any point along the line shall with a probability of 95% be within +/- 4 meters from the real road both in height and in side. A maximum of 2% of the data is allowed to disobey this demand.

From NVDB data covering different areas can be ordered via web interface. The area can be defined as a commune, a region or by coordinates. At the same time as the order is placed in the interface, parameters for which utilities to include are set.

3 Theory

Theory for Markov processes and a theory for modelling traffic flows are used in this master thesis and are presented in this chapter. In the end of this chapter, a description of the theory used for the search algorithm can be found.

3.1 Markov processes

3.1.1 Markov chain

An easy explanation of the Markov chain is that it is a process that calculates a future value independent of what the past values has been. The only parameter for the process to know is the present value. The definition of a Markov chain below shows that it is a stochastic process [2]

A stochastic process $X = \{X_n; n = 0,1,...\}$ *with discrete state space E is a Markov chain if the following holds for each* $j \in E$ *and n=0, 1, 2…*

$$\Pr\{X_{n+1} = j | X_0 = i_0, X_1 = i_1, ..., X_n = i_n\} = \Pr\{X_{n+1} = j | X_n = i_n\} \quad \textbf{(3-1)}$$

for any sets of states $i_0,...i_n$ *in the state space. The Markov chain is said to have stationary transition probabilities if*

$$\Pr\{X_1 = j \, | X_0 = i \} = \Pr\{X_{n+1} = j \, | X_n = i \} \quad \textbf{(3-2)}$$

This means that the transition probabilities are not changing over time.

Equation (3-1) represent the Markov property. i_n can be seen as the present state, and j is the future state. The equation shows that the probability of going to the next state, only knowing the present state. This is therefore a first order Markov process. For more details see [2].

A special case of a Markov chain is a **Markov chain of order *m*** (or a Markov chain with memory *m*) where *m* is finite, is defined as

$$\Pr(X_n = x_n \mid X_{n-1} = x_{n-1}, X_{n-2} = x_{n-2}, ..., X_1 = x_1) = \Pr(X_n = x_n \mid X_{n-1} = x_{n-1}, X_{n-2} = x_{n-2}, ..., X_{n-m} = x_{n-m})$$

for all n.

To see which order that will be needed, the autocorrelation could be calculated from the sampled data. For a first order Markov Process the data should not be significantly correlated passed the first lag. In Figure 3-1 an example of correlation for a random process can be seen, in this case only the first lag has significantly correlation and in this example a first order process will be sufficient.

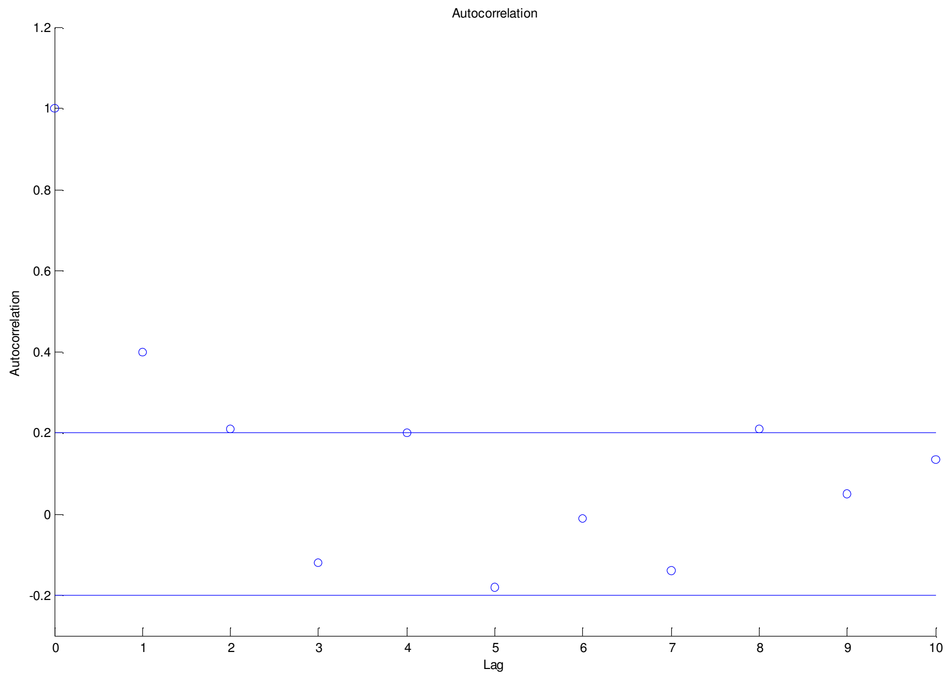

Figure 3-1. An example of autocorrelation for a random process.

3.1.2 Markov transition probabilities

To describe the transition probabilities, a Markov matrix $P(i, j)$ is used. If k states are used in the Markov chain, the matrix P will be

$$P = \begin{bmatrix} P_{11} & P_{12} & \dots & P_{1k} \\ P_{21} & P_{22} & \dots & P_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ P_{k1} & P_{k2} & \dots & P_{kk} \end{bmatrix} \qquad \textbf{(3-3)}$$

For example, P_{21} is the probability to go from state 2 to state 1, P_{kk} is the probability to stay in state k.

The probability will be calculated using the following equation

$$P_{ij} = \frac{N_{ij}}{\sum_j N_{ij}} \qquad \textbf{(3-4)}$$

where N_{ij} is the number of times a transition occurs from state i to state j.

3.2 Traffic flow theory

3.2.1 A spring model

One way of modelling traffic is to use Kinematics [4]. Two cars are separated from the vehicle flow, a lead car and a following car. The following car will adjust its speed to the lead car. To represent the acceleration and deceleration a "spring" will virtually exist in the interval between the cars. See Figure 3-2

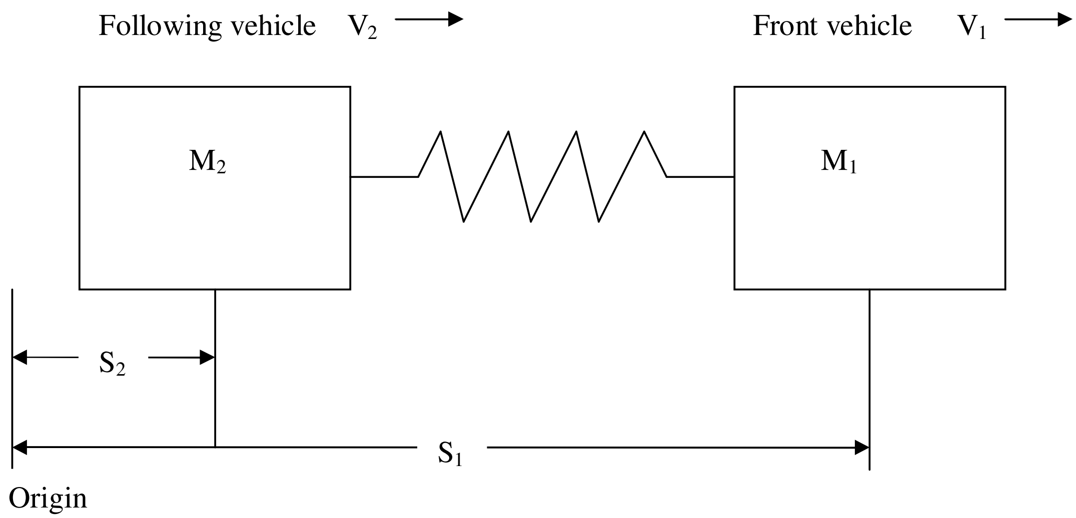

Figure 3-2. Overview of the Front and Following vehicle.

The distance between the cars when the spring is in equilibrium is the stopping distance for the following car. When the interval between the cars is smaller than the stopping distance, the spring has a repulsive force on the cars and an attraction force when the distance is longer than the safe distance. A separation of the forces for the 2 vehicle can be seen in Figure 3-3

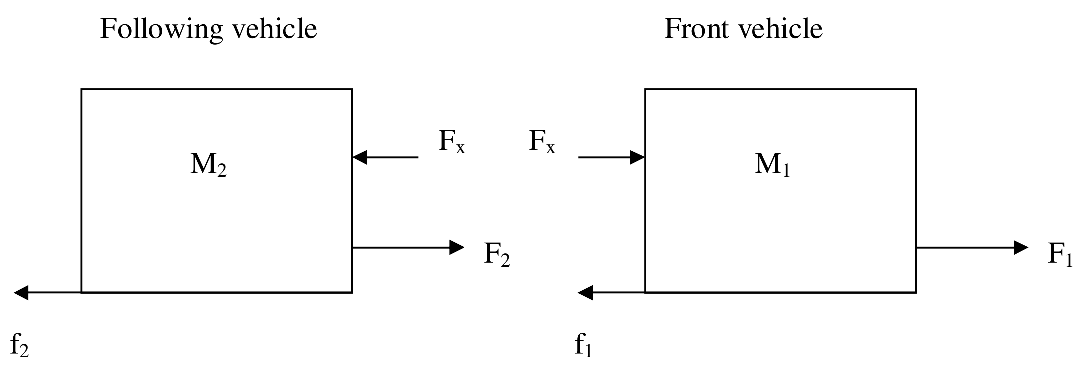

Figure 3-3. Forces acting on the vehicles.

If an assumption is made that the front car's movement is known Newton's second law can be applied on the following vehicle:

$$\sum F = m \cdot a \qquad \textbf{(3-5)}$$

In this case

$$F_2 - f_2 - F_x = M_2 \cdot a_2 \quad \textbf{(3-6)}$$

where F_2 is the second car's traction. f_2 is the friction force between the tires and the surface. M_2 is the weight of the following car. F_x is the spring force between the cars and finally a_2 is the acceleration for the second car.

The spring force can be obtained from the following equation,

$$F_x = -k\Delta X \quad \textbf{(3-7)}$$

where ΔX is the deformation amount of the spring,

$$\Delta X = X_0 - (S_1 - S_2) \quad \textbf{(3-8)}$$

X_0 is the equilibrium distance for the spring and S_1 and S_2 are the displacement for the cars and they could be obtained by the following equation,

$$S = v \cdot t + \frac{1}{2} a \cdot t^2 \quad \textbf{(3-9)}$$

There are one equation for each car to get S_1 and S_2.
If (3-9) , (3-8)) and (3-7)) are substituted in to (3-6) the following differential equation could be derived,

$$(M_2 + \frac{1}{2}kt^2)\frac{d^2 s_2}{dt^2} + kt\frac{ds_2}{dt} + f_2 + kX_0 - F_2 - k(v_1 t + \frac{1}{2}a_1 t^2) = 0 \quad \textbf{(3-10)}$$

The solution to the equation is

$$S_2 = \frac{1}{6}a_1 t^2 + v_1 t + \left(\frac{3F_2 - 3f_2 - M_2 a_1}{3k} - X_0\right)\ln\left(2M_2 + kt^2\right) + \sqrt{\frac{2M_2}{k}}\arctan\left(\sqrt{\frac{k}{2M_2}}\right)\cdot t + C_2 \quad \textbf{(3-11)}$$

where

$$C_2 = \frac{1}{3k}\left(-3F_2 + 3f_2 + M_2 a_1 + 3kX_0\right)\ln\left(2M_2\right) \quad \textbf{(3-12)}$$

3.2.2 Generalized GM (GGM) Model

Another way to model traffic flow was developed in 1958 by Chandler, Herman and Montroll at General Motors [5]. The model is a so called car-following model which are based on the assumption that every driver will follow the car directly in front of him, see Figure 3-4

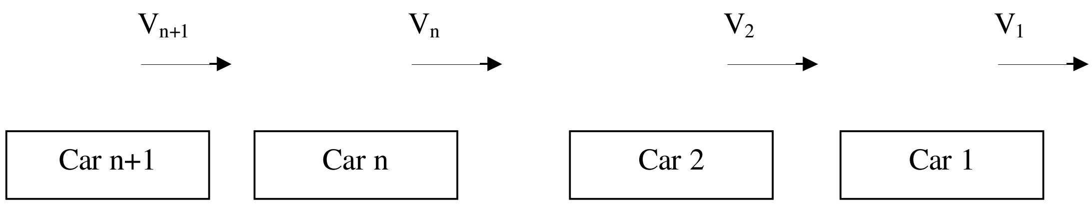

Figure 3-4. Overview of the vehicles in the model, Car 1 is the lead car, Car n+1 is the last Car.

The acceleration of vehicle n is described as follows,

$$a_n(t+T) = \lambda \Delta v(t) \quad \textbf{(3-13)}$$

where T is the reaction time of the driver and $\Delta v(t)$ is the difference in speed between the lead car (n-1) and the following car (n). λ is the sensitivity of the driver and is used to calibrate the model.

In 1961 the GM model (also known as chandler model) was extended by Gazis, Herman and Rothery[6]. The resulting model became the GGM (general GM) model, in Figure 3-5 the GGM model can be seen.

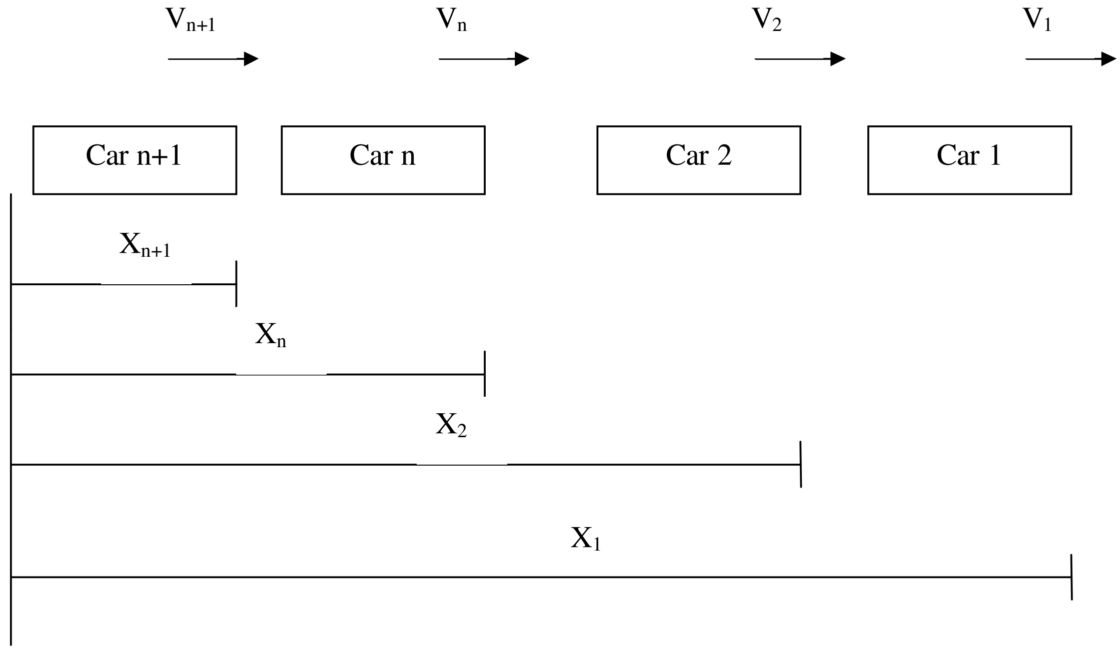

Figure 3-5. The expanded GM model.

The expanded model will take the distance between the cars in account when the acceleration for the following car is calculated, see equation (3-14)

$$a_{n+1}^{GGM}(t) = \frac{\alpha \cdot v_{n+1}^{m}(t) \cdot [v_n(t-T) - v_{n+1}(t-T)]}{[x_n(t-T) - x_{n+1}(t-T)]^l} \qquad \textbf{(3-14)}$$

where T still is the reaction time of the driver and $v_n(t)$ is the speed of car n. $x_n(t)$ is the position of the n:th car and α, m, l are parameters to calibrate the model. If m and l are set to 0 the original GM–model is obtained.

The GGM is one of the most used models for traffic simulation. But obviously there are some more models, some are described by Ranjitkar, Nakatsuji and Kawamua [9]. The calibrations parameters has been examined and studied in many models [10]. Many of the experiments made to calibrate the model have taken place in extreme start and stop conditions or at low speeds[10]. Also the following behaviour varies with traffic and flow conditions [11]. In Table 3-1 the most reliable estimates according to [10] can be seen.

Table 3-1. Most reliable estimates of parameters in the GGM model[1]

Source	m	l
Chandler et al (1958)	0	0
Herman and Potts (1959)	0	1
Hoefs (1972) dcn no brk/dcn brk/can)	1.5/0.2/0.6	0.9/0.9/3.2
Treiterer and Myers (1974) (dcn/can)	0.7/0.2	2.5/1.6
Ozaki (1993) dcn/can)	0.9/-0.2	1/0.2

[1] Key: dcn/acn: deceleration/acceleration; brk/no brk: declaration with and without the use of brakes.

3.2.3 An Expanded GGM model

According Chumsamutr and Fujioka [7] the GGM model itself is not enough to yield realistic car-following action, do (3-14) uses speed as a stimuli. For example will human drivers also react at stimulus such as spacing (when spacing to the leading vehicle is too long the driver will accelerate). They have introduced another equation (3-15) to handle such stimuli.

$$a_{n+1}^{\sup}(t) = \begin{cases} \psi \cdot f_a(v) \cdot e^{\beta(v_n - v_{n+1})} \left| s - s_d \right|^k, s - s_d \geq 0 \\ -\psi \cdot f_b(v) \cdot e^{\beta(v_{n+1} - v_n)} \left| s - s_d \right|^k, s - s_d < 0 \end{cases} \quad \textbf{(3-15)}$$

The parameters ψ, β and k represents different drivers, s is the spacing between the lead and follow car (the spacing is the same as $x_n(t-T) - x_{n+1}(t-T)$ in eq (3-14). s_d is the desired distance that the driver want between the cars. The function f_a and f_b is how the speed affects acceleration behaviour. Final eq (3-15) and eq (3-14) is combined and gives the total acceleration.

$$a_{n+1}(t) = a_{n+1}^{GGM}(t) + a_{n+1}^{\sup}(t) \quad \textbf{(3-16)}$$

This extended version only works when the spacing is closer than 90 meters, when spacing is larger than 90 meters the acceleration is calculated by the GGM model. The authors also suggest some more adjustment to make the car-following more stabile and to model real traffic more realistic. First they develop an algorithm to avoid collisions. When the following vehicle drives faster then the leading vehicle, a maximum acceleration is calculated through the following equations

$$E = \max\left(v_{n+1}^j \cdot T, v_{n+1}^j \cdot T + \frac{\left(v_{n+1}^j\right)^2 - \left(v_n^j\right)^2}{2 \cdot d_m} \right) \quad \textbf{(3-17)}$$

$$a_{n+1}^j = \frac{\left[x_n^{j+1} - x_{n+1}^j - v_{n+1}^j \cdot \Delta t - s_{\min} - E\right]}{0.5 \cdot \Delta t^2} \quad \textbf{(3-18)}$$

Explanation and description for x, v, T can be found in (3-14) in section 4.2.2.d_m is the maximum deceleration, Δt is the simulation time step and s_{min} is the minimum allowed spacing between the vehicles. The second adjustment is when vehicle speed goes below 5 km/h, the l, m and a parameter in the GGM are changed so that the model is represented by the Greenberg model [7]. When the Greenberg model is used, the stopping phase becomes smoother.

3.3 Search algorithm A*

In the standard mode of the drive cycle generator it is possible to select between some typical Swedish roads. The data describing these example roads are collected from NVDB. First data about the area around the desired road is ordered from the data base. Then an example road is defined by picking the endpoints of it. To find a path between two selected points the A* search algorithm is used. The speed limits, topography and curvature of the returned path are processed to get statistical data for the drive cycle generator.

A* is a best first search algorithm, that means that it explores the most promising path first. The heuristic used to determine which way to start exploring is the distance travelled so far plus an estimated cost for the remaining distance. For the algorithm to work correctly the heuristic must not be bigger than the cost of the best solution. As common in routing tasks the straight line distance to the goal is used as estimated cost for the remaining distance. To get the fastest route the distance is divided by the corresponding speed limit.

If a correct heuristic is chosen A* is optimal in the way that if there is a solution, A*will find it, and it will be the first found solution.

The time complexity of A* is in the worst case exponential but probably not more than polynomial. The memory complexity is a greater threat with its possibility of remembering an exponential number of nodes [8].

4 Identification of key parameters

To define the characteristics of a drive cycle, key parameters are selected describing speed limit and topography. The parameters are selected from reasoning about what represents energy and effect in a drive cycle.

4.1 Topography

In the topography profile the steepness of the slopes gives the effect the vehicle needs to deliver or take care of. And the height represents the energy that the position of the truck holds. When the truck changes height the energy change can be stored in or taken from the energy buffer of a HEV, until the buffer is full or empty.

From this reasoning the following parameters are selected to use as a hypothesis of what affects the fuel consumption and the performance of the powertrain.

- Mean of slopes
  - If the overall slope is negative or positive will obviously affect the fuel consumption.
- Variance of the slopes
  - Higher variance means more slopes and bigger potential to save fuel by having a hybrid powertrain. It also means steeper slopes which affects the performance of the powertrain.
- Distance between hill tops
  - Shorter distance between hill tops means more but lower hills. This increases the fuel saving potential of a HEV since fewer but bigger hills will increase the risk of the energy buffer to be full or empty.

4.2 Speed limits

In the speed profile the acceleration and deceleration affects the effect that is needed from or has to be taken care of by the vehicle, although breaking from a decrees in speed limit might be somewhat uncommon. While the speed represents the energy that the truck holds. When the truck changes speed the energy change can be stored in or taken from the energy buffer of a HEV, until the buffer is full or empty.

From this reasoning the following parameters are selected to use as a hypothesis of what affects the fuel consumption and the performance of the powertrain.

- Percent of distance at different speeds
  - The speed affects the fuel consumption.
- Average speed change amplitude
  - Higher amplitude in speed change leads to risk of the energy buffer to be full or empty. It also means bigger fuel consumption since acceleration consumes big effect.
- Distance between speed change
  - Shorter distance between speed changes means more acceleration and deceleration. This increases the fuel saving potential of a HEV since energy stored from deceleration can be used during acceleration.

5 Data collection

To get drive cycles corresponding to real world road, statistical data about Swedish roads are collected from NVDB. The data contains, among other things, information about curvature, topography and speed limits for roads. Some topography data are considered false since it consists of unrealistic steep slopes (in some cases above 99%). Because of these extreme slopes the maximum gradient and the rate of slopes above 10% in the drive cycle classification currently used at Volvo Powertrain, GTA (Global Truck Application), are ignored.

Due to some limitations in the data ordering system of NVDB, data for all of Sweden can not be downloaded. To include the differences of roads in different parts of Sweden a few different areas are used. Selections of areas to use are made and data is extracted from the following areas.
- Göteborg city with the eastern communes as far as Alingsås and Borås.
- The communes along the west coast of Sweden that E6 passes through.
- The communes along E45 from Östersund and north up to Kiruna.

5.1 Example roads

The following example roads are extracted from the data to represent different type of driving environments. These example roads are used as choice able typical Swedish roads in the tool.

5.1.1 E6 highway

This path starts on the highway outside of Malmö and follows the highway till it stops north of Munkedal. It passes through Göteborg which leads to some lower speed limit. And it goes over Hallandsås with the steepest highway slope in Sweden.

Data

Path length: 413 km

<u>Speed limit</u>

Average distance between changes:	4.49 km
Median value:	110 km/h
Mean value:	106.3 km/h
Variance:	126.04

<u>Topography</u>

Number of hills/km	1.91
Variance of the slope	$0.57 * 10^{-3}$
Accumulated rise/km	5.61
Accumulated drop height/km	5.71

Percent of distance on slopes:	< 3%:	92.02 %
	3% - 6%:	7.18 %
	6 % - 10%:	0.62 %
	> 10 %:	0.18 %
GTA classification:	Predominantly flat	

5.1.2 Göteborg city

A path running back and forth through Göteborg is selected to represent city driving. It is mostly heading through low speed areas and only occasionally on the bigger roads with higher speed limits.

Data

Path length: 116 km

<u>Speed limit</u>

Average distance between changes:	2.70 km
Median value:	50 km/h
Mean value:	54.2 km/h
Variance:	76.18

<u>Topography</u>

Number of hills/km	3.73
Variance of the slope	$1.3 * 10^{-3}$
Accumulated rise/km	10.11
Accumulated drop height/km	10.31

Percent of distance on slopes:	< 3%:	78.7 %
	3% - 6%:	15.4 %
	6 % - 10%:	4.66 %
	> 10 %:	1.24 %
GTA classification:	Hilly	

5.1.3 City close highway

The highway from Alingsås to Göteborg and further on to Borås is a road that passes through a lot of urban areas. This means lower and more frequent changes in, speed limits.

Data

Path length: 108 km

<u>Speed limit</u>

Average distance between changes:	2.46 km
Median value:	90 km/h
Mean value:	95.4 km/h
Variance:	178.9

<u>Topography</u>

Number of hills/km	1.81
Variance of the slope	$0.59 * 10^{-3}$
Accumulated rise/km	8.05
Accumulated drop height/km	7.42

Percent of distance on slopes:	< 3%:	84.4 %
	3% - 6%:	15.5 %
	6 % - 10%:	0.76 %
	> 10 %:	0.36 %

GTA classification: Predominantly flat

5.1.4 Landvetter Borås

This path is chosen since it is commonly used for simulation at Volvo today. It is a part of the Alingsås – Göteborg – Borås path.

Data

Path length: 45.6 km

Speed limit

Average distance between changes:	1.83 km
Median value:	90 km/h
Mean value:	95.2 km/h
Variance:	95.19

Topography

Number of hills/km	1.12
Variance of the slope	$0.38*10^{-3}$
Accumulated rise/km	5.95
Accumulated drop height/km	7.64

Percent of distance on slopes:	< 3%:	89.8 %
	3% - 6%:	9.7 %
	6 % - 10%:	0.39 %
	> 10 %:	0.14 %
GTA classification:	Predominantly flat	

5.1.5 E45 Inlandsvägen

Inlandsvägen is a path through the northern inland area of Sweden. From Östersund, situated in the middle of Sweden, to Karesuando, at the border to Finland almost as far north as you can get in Sweden.

Data

Path length: 1026 km

Speed limit

Average distance between changes:	4.89 km
Median value:	100 km/h
Mean value:	95.3 km/h
Variance:	237.18

Topography

Number of hills/km	1.52
Variance of the slope	$0.40*10^{-3}$
Accumulated rise/km	6.32
Accumulated drop height/km	6.33

Percent of distance on slopes:	< 3%:	89.4 %
	3% - 6%:	9.20 %
	6 % - 10%:	1.31 %

	> 10 %:	0.13 %
GTA classification:	Hilly	

6 Calculations

In this chapter it is described how the collected data is used and which calculations are done. In Chapter 6.1 the decision of what order of Markov processes to use are motivated. In Chapter 6.2 it is described how the Markov matrixes are created. And in Chapter 6.3 and 6.4 the different calculations from example roads and user set parameters can be found.

6.1 Decision of the Markov processes order to be used

To see what order of the Markov process that is sufficient, autocorrelation is preformed on the data collected from NVDB. In Figure 6- the autocorrelation result for the road slope angle in Göteborg city path is shown. The results show that a first order Markov process is enough. For the other example roads from Chapter 5.1, the results are about the same, see Figure 6-, it will be sufficient with a first order Markov process in those cases as well. When a first order Markov process is used a Markov matrix is needed.

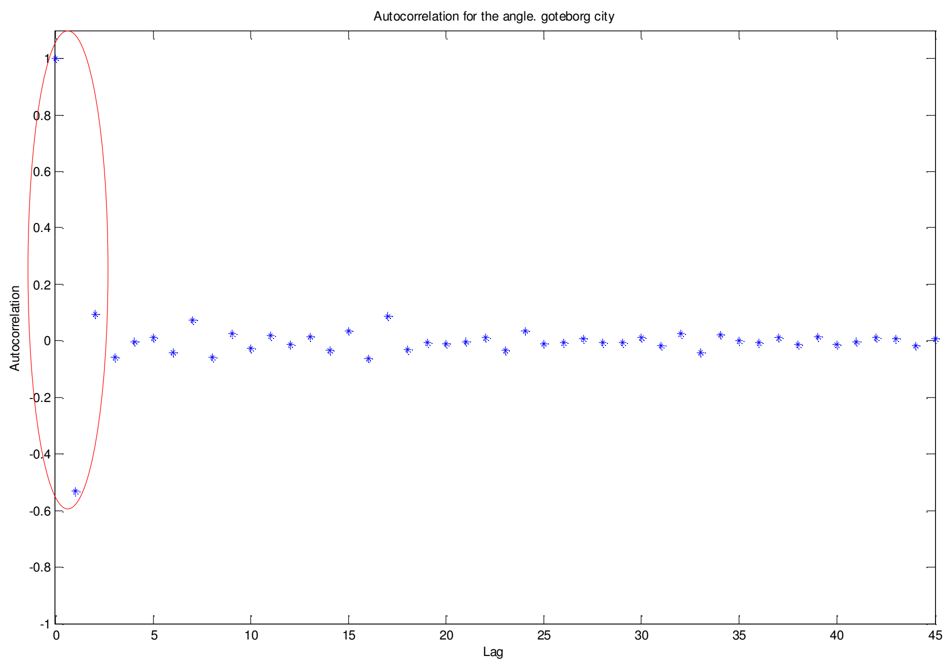

Figure 6-1. The autocorrelation of the road slope angle. Gothenburg city.

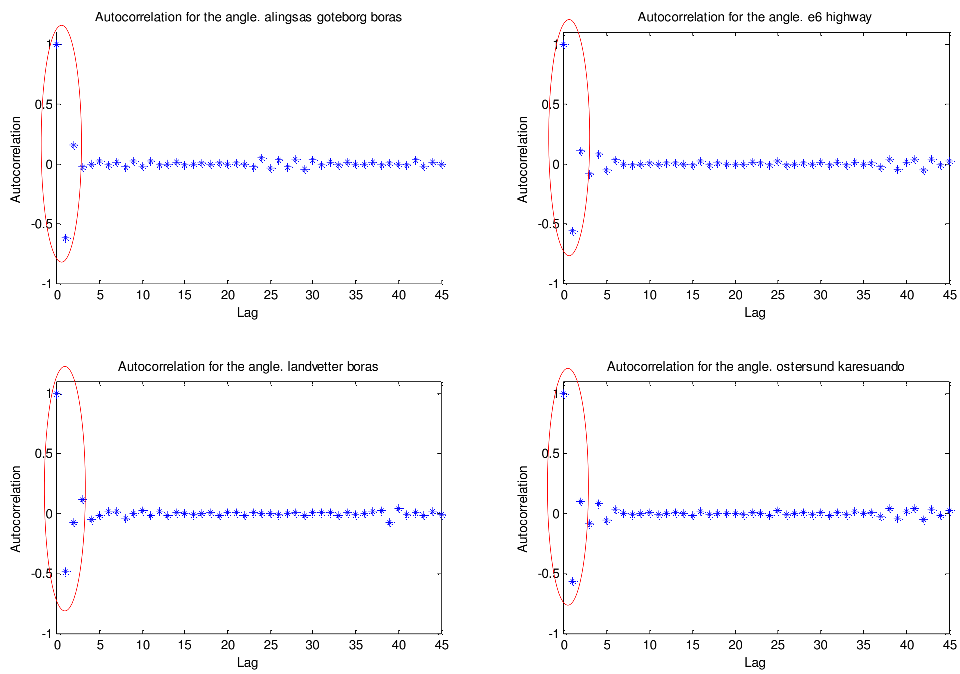

Figure 6-2. The autocorrelation of the road slope angle, for some example roads.

6.2 Creation of Markov matrices

The path containing the data that the Markov matrix is to be calculated from is walked through, and at each meter the present slope and speed limit are known. The slope and speed limit at the next step is also known. The matrix is calculated using equations (3-3) and (3-4)

6.2.1 Topography

When calculating the Markov matrix for the topography generation, the slopes are the states. Every state is an interval of slopes, the states are equally large and the spans from -0.159 radians to +0.159 radians (+/-10% slope). The number of states to use is decided in Chapter 6.2.1.1.

6.2.1.1 Determine number of states to use

To determine the number of states, two things need to be considered. How good the correspondence between the simulated topography and the real topography is in terms of the key parameters identified earlier in Chapter 4. And how long time the drive cycle generation takes. The number of states used determines the size of the Markov matrix. For N states the matrix gets the size N x N.

In Figure 6-3 and Figure 6-4 it can be seen that a good number of intervals is 400 since the generated road matches the real road well, in terms of the key parameters, when this number of intervals is used.

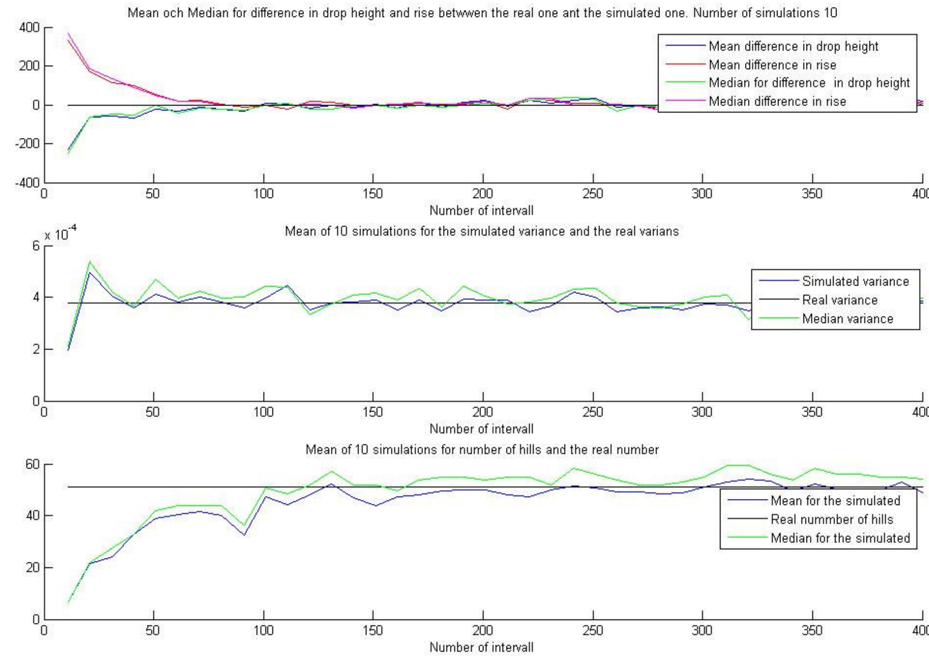

Figure 6-3. Correspondence between a real (Landvetter - Borås) and a generated topography, using different number of intervals. In the first graph the difference in accumulated drop height and accumulated rise between the real topography and the generated topography are shown. In the middle graph the variance are shown and at the bottom graph the number of hills are shown.

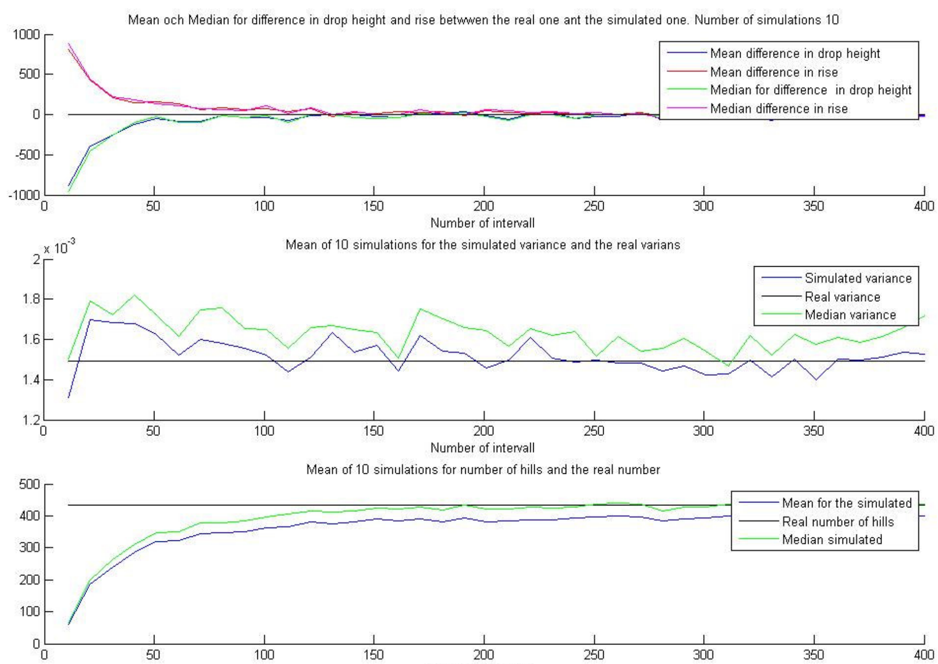

Figure 6-4. Correspondence between a real (Göteborg city) and a generated topography, using different number of intervals. In the first graph the different in accumulated drop height and accumulated rise between the real topography and the generated topography are shown. In the middle graph the variance are shown and at the bottom graph the number of hills are shown.

In Figure 6-4 the time for generating the topography for different number of intervals is shown for a drive cycle. It is obvious that the generation time rises when the number of states gets larger. Time consumption is important since the tool is supposed to be able to generate drive cycles while the user waits at the computer.

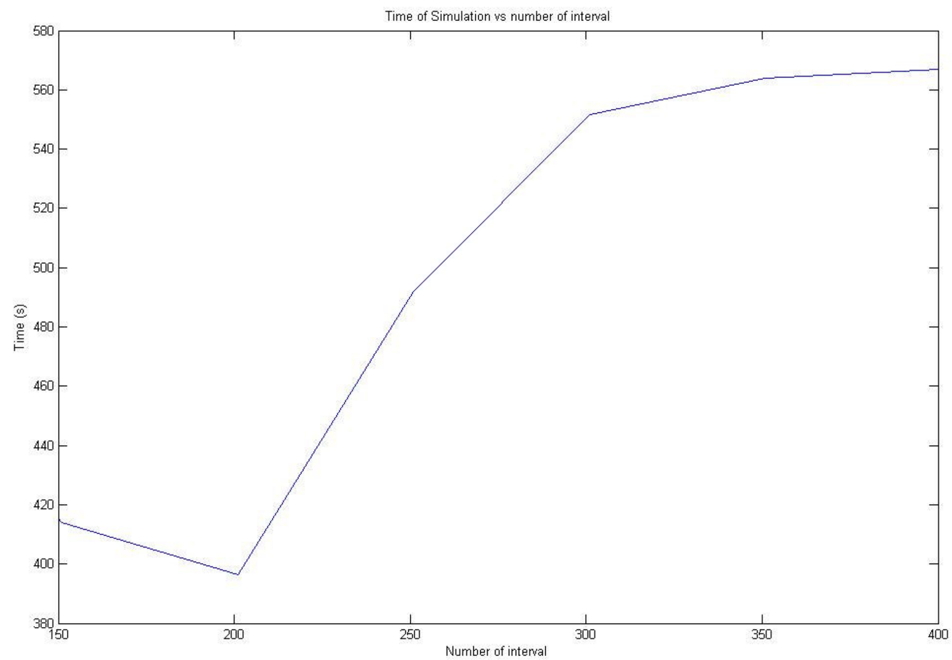

Figure 6-4. Time consumption for generation of a drive cycle, with different number of intervals for the slopes.

6.2.2 Speed limit

When calculating the Markov matrix for the speed limit generation, the speed limits on Swedish roads are used as the states. Existing speed limits are 5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110 and 120 km/h.

6.2.3 Curvature

When calculating the Markov matrix for the curvature generation the curve angles of the road are the states. Every state is an interval of angles, the states are equally large and the spans from 0 radians to $\pi / 2$ radians. Since the curvature only is used to calculate decrease in speed it does not matter in which direction the curve is. The size of the intervals used as states are the same as for the topography calculations.

6.3 Markov matrices from user set parameters

To get a Markov matrix from user set parameters an example path, fulfilling those parameters is created. Those example paths are then used to calculate Markov matrixes as described in Chapter 6.2. How the example path is created is described separately for the topography and the speed limits in Chapters 6.3.1 and 6.3.2.

6.3.1 Topography

Parameters used to describe the topography profile are mean and variance of the slope and distance between hill tops. The example topography profile, used to calculate the Markov matrix, is created by placing a large number (1000) of hills in a row. Every hill has an uphill and a downhill slope whose combined length is equal to the "distance between hill tops" parameter entered by the user. The hills are created one at a time by randomly picking one slope for every meter of the hill. The slopes are selected following a Beta distribution created to fulfil the mean and variance given by the user. As the slopes are picked they are sorted by if they are uphill or downhill and then the downhill slopes are placed after the uphill slopes and this way they represent one hill. The hills are placed after each other and this topography profile is used to calculate a Markov matrix.

6.3.2 Speed limits

Input parameters for the speed profile are average distance between speed limit changes, mean amplitude of speed limit change and percentage of the distance for every speed limit. The example speed profile, used to calculate the Markov matrix, is generated in two steps. First the order of the changes is decided and then the length of every interval. The length of the example path is a large number (400) of speed limit changes multiplied with the "distance between speed limit changes" parameter given by the user.

When the order of the speed limits is decided, the limits that shall occur are considered as a cyclic graph were the limits are the nodes and the amplitude of the changes are the cost of the links, see Figure 6-5 for an example. A randomly ordered depth first search is done until a solution is found. The solution searched has a total cost equally to the "mean change amplitude" parameter multiplied by the number of changes.

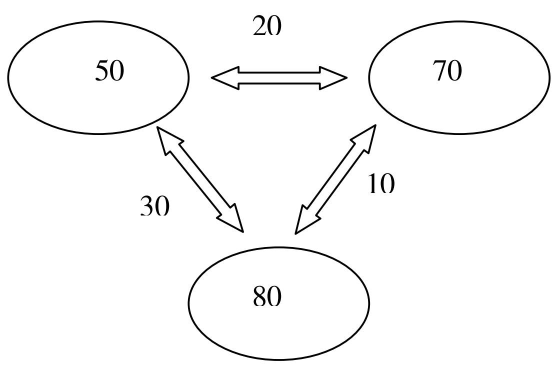

Figure 6-5. Example of graph for choosing order of changes between 50, 70 and 80 km/h

The length to spend at every speed limit is divided by the number of times this limit occurs. Since this profile only is used for the generation of the Markov matrix it does not matter that every interval with the same speed limit has the same length.

6.4 Markov matrices and statistics from example roads

In this chapter the preparations made to calculate the Markov matrices for the example roads are presented for topography in Chapter 6.4.1, speed limits in Chapter 6.4.2 and curvature in Chapter 6.4.3. The statistical calculations for unplanned stops is shown in Chapter 6.4.4.

6.4.1 Topography

Some coordinates in the z direction are missing in the data extracted from NVDB, at those points a linear interpolation is made to get an assumed value for those points.
In the following description it is shown how the topography matrix is calculated.

- First an interpolation is made to get the z coordinates that are missing.
- The second step is to calculate the slope at each meter of the path. To do this the z coordinate and the distance between the coordinates are used.
- Now when the distance between coordinates and the slope at each coordinate are known, the slope will be placed at each meter between the 2 coordinates.

This series of slopes is used to calculate a Markov matrix with Equation 3-3 and Equation 3-4.

6.4.2 Speed limit

The probability matrix calculation for the speed limit resembles the topography case at many points. But in this case no interpolation is needed since every coordinate has a speed limit. The speed limits are "placed" at each meter directly, and the whole distance is walked through. The possible states here are [5 10 20 30 40 50 60 70 80 90 100 110 120], and the Markov matrix is calculated using Equation 3-3 and Equation 3-4.

6.4.3 Curvature

The Curvature is not calculated for the different road types (highway, city roads etc) but instead for different speed limits. It is assumed that the speed limits and the curvature correlates. The matrices are calculated for the following speed intervals [0 59], [60 89] and [90 120]. To calculate the angel of the Curvature first the circumradius is calculated for 3 points using the SSS (Side-Side-Side) theorem. See Figure 6-6.

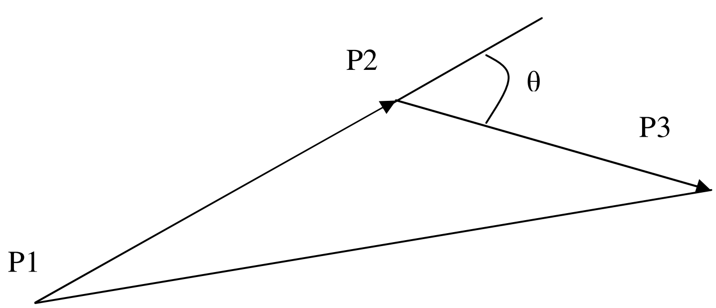

Figure 6-6. Relationship between coordinates and angle.

First the semiparameter

$$S = \frac{1}{2}(|P2 - P1| + |P3 - P2| + |P3 - P1|) \quad \textbf{(6-1)}$$

is calculated and then the area of the Triangle P1 P2 P3 is calculated with Heron's formula

$$K = \sqrt{S \cdot (S - |P2 - P1|)(S - |P3 - P2|)(S - |P3 - P1|)} \quad \textbf{(6-2).}$$

The circumradius is then

$$R = \frac{(|P2 - P1|)(|P3 - P2|)(|P3 - P1|)}{4K} \quad \textbf{(6-3).}$$

To get the angle θ, the following calculation is made

$$\theta = a\sin\left(\frac{|P3 - P1|}{2 * R}\right) \quad \textbf{(6-4).}$$

The angle is placed at each meter of the distance between P2 and P3. Except for when the angel is bigger than 45 degrees, then the angel is placed only at the first meter of the distance, and the rest of the distance gets the angle $\theta=0$. This is done since this sharp curves are considered to be a cross road and not a turning road. Finally the Markov matrix is calculated using Equation 3-3 and Equation 3-4.

6.4.4 Unplanned stops

The probability of an unplanned stop to occur is assumed to only be dependent of the current speed limit. To calculate examples of probabilities for unplanned stops at different speed limits, data from bus line 17 in Göteborg is used. The data comes from 21 drives at each direction of the bus route, measured with a GPS (Global Positioning System) during another thesis work at Chalmers. All of the measurements are not used because some of the GPS data is bad. The only existing speed limits on this route are 50 and 70 km/h. There are no data for how often an unplanned stop occurs at other speed limits.

The mean and variance of the distance between unplanned stops are calculated for 50 km/h and 70km/h separately, and so are the stop times.

7 The drive cycle generator

The drive cycle generator consists of several parts responsible for different components of the drive cycle. These parts, the format of the outputted drive cycle and the GUI are presented in this chapter.

7.1 Drive cycle format

After the generation, drive cycles are stored in Matlab m-files matching the present format used at Volvo Powertrain, for an example see Figure 7-1. When the m-file is run in Matlab, a struct-variable that is used during the simulation is created.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%   row       pos       vset       alt       accset      t_stop      curv       Temp     event_code  event_data      t_ref      v_ref
%  -------------------------------------------------------------------------------------------------------------------------------------
%   [1]        [m]      [m/s]       [m]      [m/(s*s)]     [sec]      [1/m]     [deg C]       [1]         [1]          [sec]     [m/s]
%
Road_Matrix = [
    1.0000 ,   0.0000 ,   0.0000 , 224.3000 , 999.0000 ,   3.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2189.0000 ,   0.0000
    2.0000 ,   0.0010 ,   0.0000 , 224.3000 , 999.0000 ,   3.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2190.0000 ,   0.0000
    3.0000 ,   0.0020 ,   0.0000 , 224.3000 , 999.0000 ,   3.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2191.0000 ,   0.0000
    4.0000 ,   0.9870 ,  14.3980 , 224.3000 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2192.0000 ,   5.3984
    5.0000 ,   3.9870 ,  14.3980 , 224.0000 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2193.0000 ,  10.0800
    6.0000 ,   6.9870 ,  14.3980 , 223.7000 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2194.0000 ,  13.1990
    7.0000 ,  10.9870 ,  14.3980 , 223.4000 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2195.0000 ,  14.3980
    8.0000 ,  14.9870 ,  13.3260 , 223.1000 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2196.0000 ,  13.8980
    9.0000 ,  18.9870 ,  12.3470 , 222.9100 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2197.0000 ,  12.5000
   10.0000 ,  21.9870 ,  11.5200 , 222.9100 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2198.0000 ,  11.3980
   11.0000 ,  24.9870 ,  10.9190 , 222.9100 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2199.0000 ,  10.6040
   12.0000 ,  27.9870 ,  10.6990 , 222.9100 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2200.0000 ,  10.0980
   13.0000 ,  30.9870 ,  10.6990 , 222.9100 , 999.0000 , 999.0000 , 999.0000 ,  18.0000 , 999.0000 , 999.0000 , 2201.0000 ,   9.7969
```

Figure 7-1. An example from a drive cycle file used at Volvo Powertrain.

7.1.1 Description of the columns in the Road_Matrix

Here follows an explanation of what the different components of the drive cycle represents in the drive cycle format used at Volvo Powertrain, Volvo common road data format.

row

- Row index, starting at 1.

pos

-Position measured from starting point.

vset

- The vehicle speed that the driver aims for at the current position specified in the pos column. Typically, this can be a legal speed limit, a suitable speed due to the prevailing road, road surface or traffic situation

alt

-Position based altitude (above sea level).

accset

-Position based setpoint for acceleration/deceleration.
- No values generated by our drive cycle generator.

t_stop

-Time that the vehicle stops for instance at a red light or a bus stop.
The stop time is at all times the time of the last stop.

curv

-Distance based curvature. This is a 'scientific' property of the measured road. One divided by the curve radius. A zero value thus means straight forward. A positive value means turning to the right.

Temp

-Distance based temperature.

- No values generated by our drive cycle generator.

event_code

-A three digit number, or code, representing a specific event. E.g. a stop at a bus station, opening one or several doors, compactation etc.

-Used by our drive cycle generator to signal upcoming stop.

event_data

-A variable containing numbers adding information, and closely connected, to correlating elements in "event_code". "event_code" can be defined alone without "event_data", but not the other way around.

- Used by our drive cycle generator to give distance to a signalled stop.

t_ref

-Time during measurement. In a reference vehicle, "t_ref" typically is the sampled time.

- No values generated by our drive cycle generator, since data is generated and not measured. This means that time based simulation can't be run from our drive cycles.

v_ref

-Vehicle speed during measurement. In a reference vehicle, "v_ref" typically is the sampled vehicle speed.

- No values generated by our drive cycle generator, since data is generated and not measured.

[12]

7.2 Subsystems

How the subsystems of the generation tool are connected to each other can be seen in Figure 7-2. The subsystems are described in respectively chapter below.

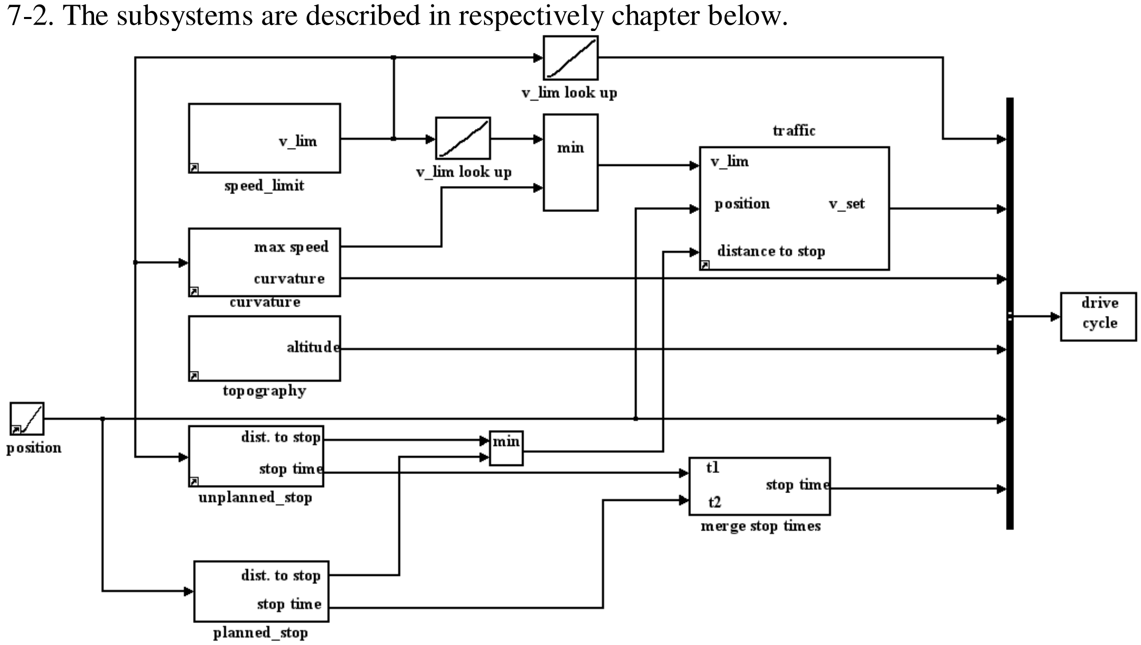

Figure 7-2. Outline of how the subsystems are connected.

7.2.1 Topography

The topography system is built in Simulink. In Figure 7-3 the Simulink model for the topography generaritor can be seen. The lookup table (2-D) contains the currently used Markov matrix, and the output from the table is the matrix row representing the probabilities of change from the present state. In the function block the index of the next state is calculated and then sent to the lookup table that returns the next sloop. This is made for each meter and the accumulated height is the output. For small angles (< 0.159 rad.) the arctangents value for the angle is equal to the angle. This combined with a step size of one meter makes it possible to add the angles and get the height.

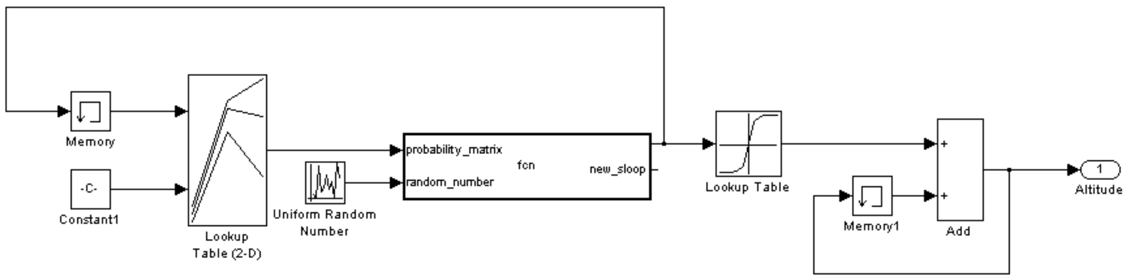

Figure 7-3 The Simulink model generating the topography.

7.2.2 Speed limit

The Simulink model for the speed limit generation is shown in Figure 7-2. The lookup table (2-D) contains the currently used Markov matrix for speed limits, and the output from the table is the matrix row representing the present state. This row contains the probabilities for respectively state to be the next state. In the function block the next speed is selected, by comparing a random number to the values in the row containing the probabilities for next state. This comparison is made with a loop, when the random number is larger than the value in the row the number of iterations done is the index for the next speed limit.

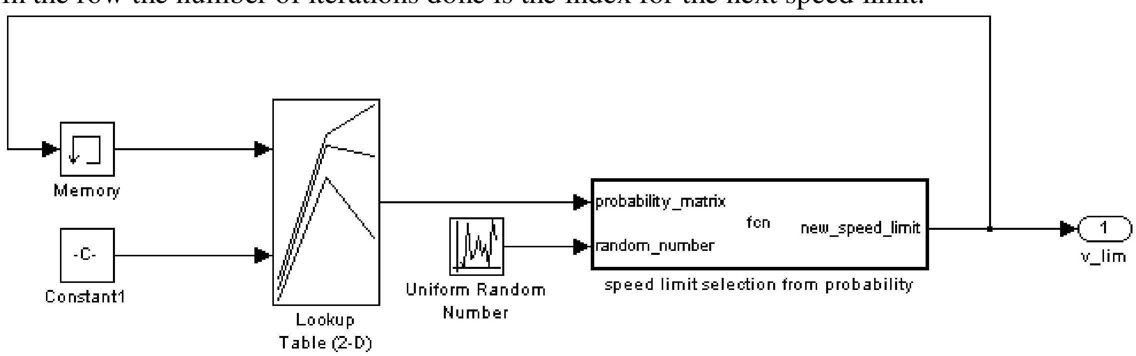

Figure 7-4 The Simulink model generating the speed limit.

7.2.3 Planned stops

The Simulink model handling planned stops can be seen in Figure 7-5. The position and the length of the planned stops are given by the user and stored in a constant-block. The embedded function compares the current position to the first value in the list of stop positions supplied from the constant-block. When the position of a planned stop is reached the outputted "last stop nr" is changed and the stop time is, via the lookup table containing the stop times, updated to the new value. The "stop time" output is at all times the stop time of the last stop. Before the first stop it is set to zero. The "dist to planned stop" output is intended to be used in the speed limit generation, this is not fully implemented but the ideas can be read about in Chapter 11.

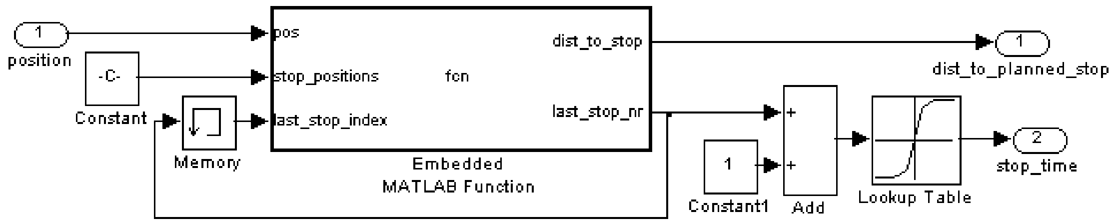

Figure 7-5 The Simulink model adding planned stops to the drive cycle.

7.2.4 Unplanned stops

Input to the unplanned stop generator model (seen in Figure 7-6) is the current speed limit index. The "lookup distance to stop" table contains one row for every speed limit. Every row contains a cumulative Beta distribution with the probability of a stop to occur within different distances. The distances present in a row spans from one meter to the entered mean plus 4 times the entered variance of the distance between stops. The "lookup stop time" table works the same way but with time [s] instead of distance [m]. The "embedded function" block selects a distance to stop and a stop time by comparing random numbers to the Beta distributions. Comparison of the current speed limit with the last in "relational operator" is done to signal to the "embedded function" block when the speed limit is changed. While the speed limit is the same as in the last step the stop time is kept and the distance to stop is decreased with one. In the cases were the speed limit is changed, the count down to the next step is reset and a new distance and a new stop time are calculated matching the new speed limit.

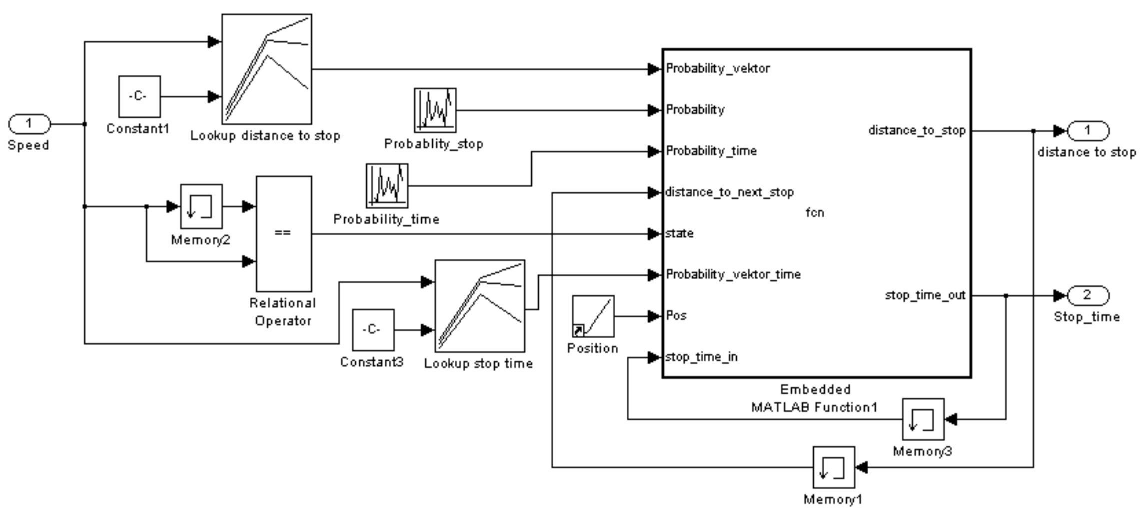

Figure 7-6 The Simulink model generating unplanned stops.

7.2.5 Traffic

The expanded GGM model used for simulating the surrounding traffic is implemented in Simulink (se Figure 7-7). The inputs to the model are how many vehicles there is on the road (traffic intensity) a large number of cars represents traffic stocking, the current position, the speed limit and the distance to the next unplanned stop. The first car follows the speed limit as good as it can. Depending of the speed and the spacing between the cars, some parameters in the model will dynamically change see Chapter 3.2.3. The calculations made in the embedded function are derived from Chapter 3.2.3. The speeds of the vehicles are calculated at each meter. The speed of the last car will be the output from this system and represent v_set in the drive cycle model. The stop time will not be taken into account when the traffic is simulated, it will be handled by the driver.

To handle unplanned stop, a calculation of how many meters it will take for the first car to stop, using full deceleration from the current speed, is made. The distance it will take is then compared to the distance to next stop. If the distance the first car needs to stop is equal or smaller than the distance to the stop, the first car will start to break with full deceleration. The other cars will follow the first car and there will be a chain reaction. An example of a simulation with 3 cars is shown in Figure 7-8.

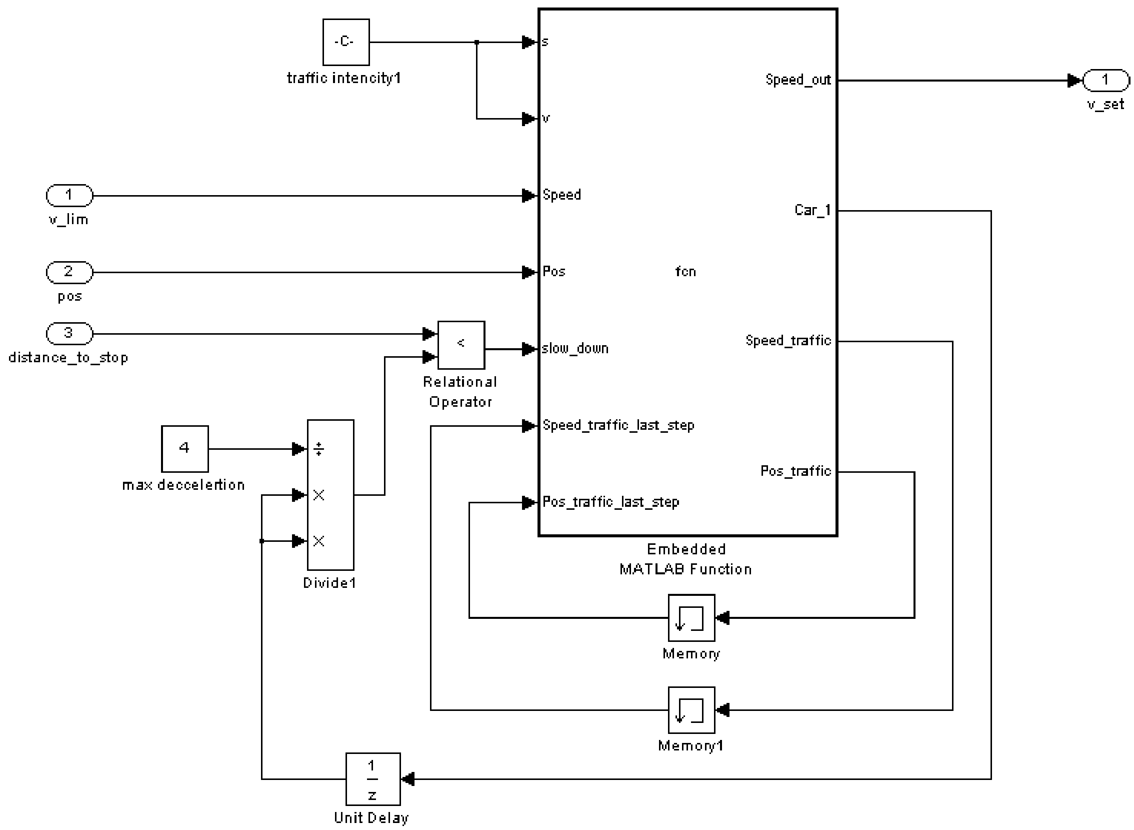

Figure 7-7. The Simulink model generating influence on v_set from traffic

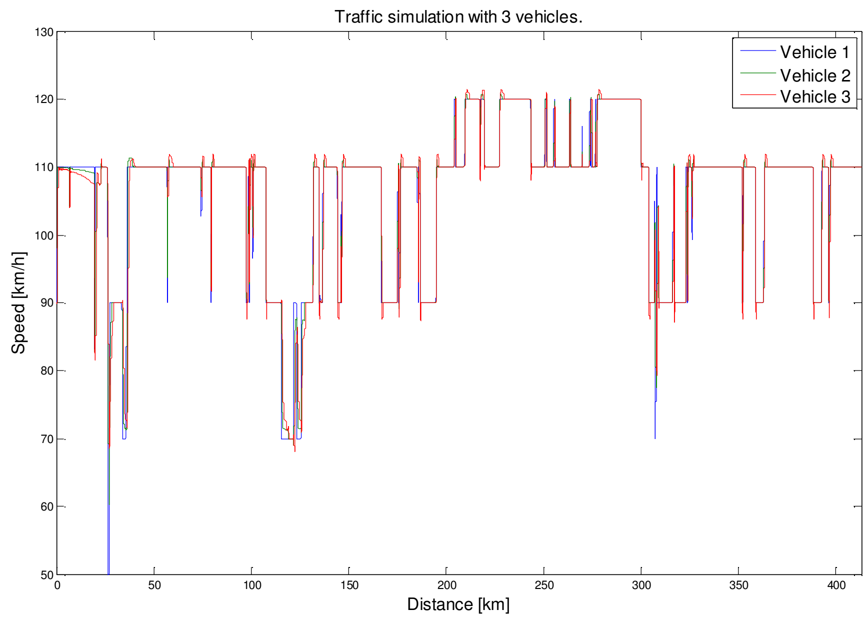

Figure 7-8. Example of how traffic behaves at speed limit changes.

7.2.6 Curvature

The model generating the curvature can be seen in Figure 7-9. The input to the model is the current speed limit index and the output is the curvature and the maximum speed for the traffic model to use in the curve. In the first lookup table three different matrixes are stored and they are shifted depending of what the current speed is. The output from the table is a row containing the probabilities of what the angle of the curve is at the next position. In the embedded function the index of the curve at the next position is selected. In the second lookup table the angle is selected. The angle is transformed to the desired format one divided by the curve radius. The curve radius is then used to calculate the maximum speed for the traffic model to use in the current curve.

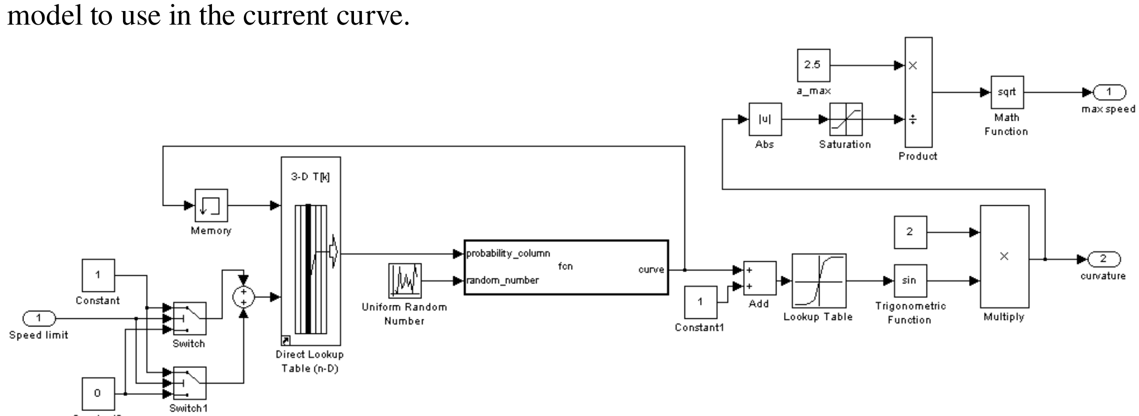

Figure 7-9. Simulink model generating the curvature.

7.3 The graphical user interface

The graphical user interface consists of two modes. In the advanced mode (Figure 7-10) the user sets all the key parameters identified in Chapter 4, plus values for planned and unplanned stops and the length of the drive cycle. When a generation is started the user gets the opportunity to save the combination of parameters.

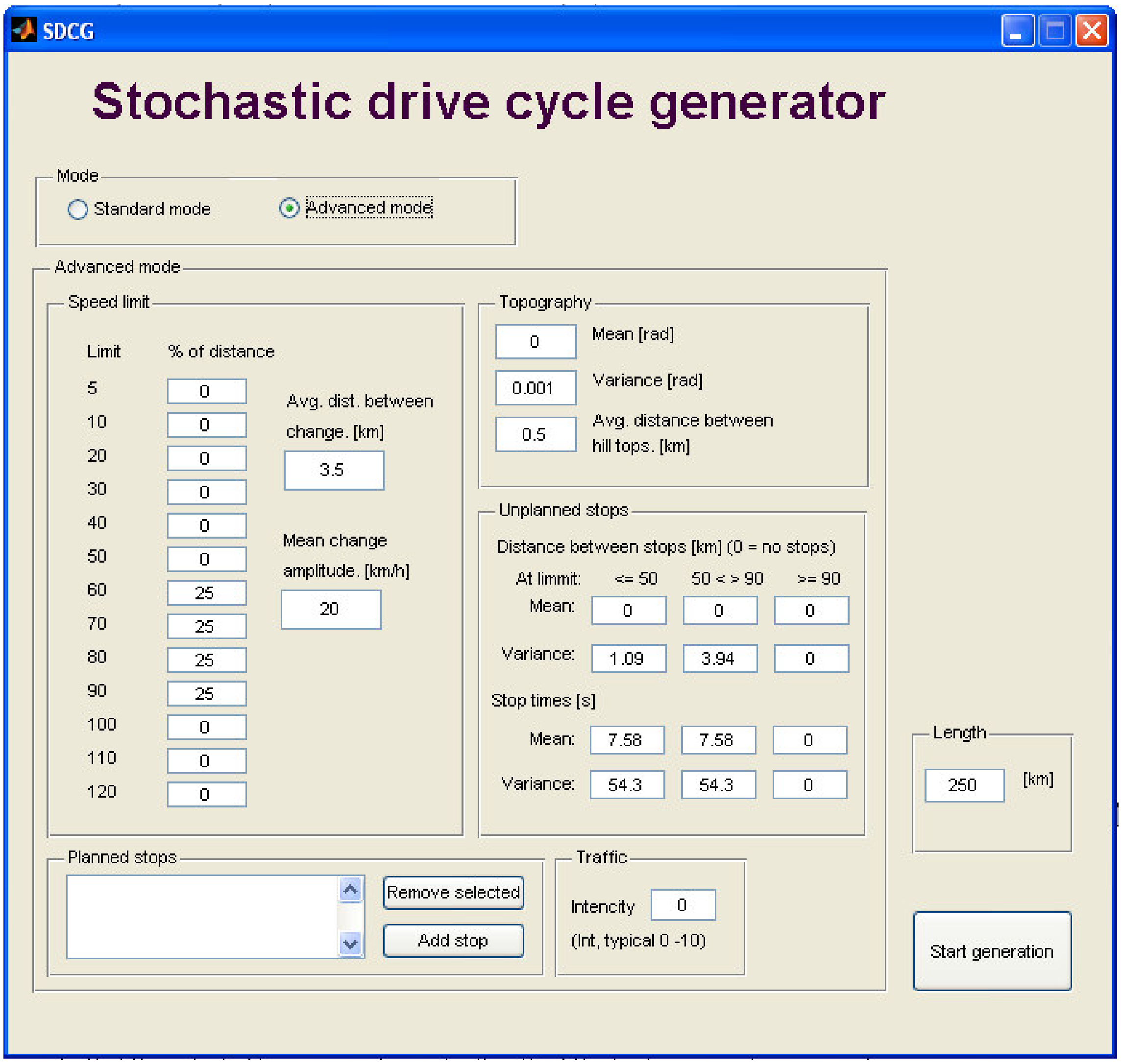

Figure 7-10. The advanced mode of the GUI.

In the standard mode (Figure 7-11) the user can chose between saved combinations of parameters. The example roads from Chapter 4 and combinations of parameters saved from the advanced mode is available in the list. The length of the drive cycle is also asked for.

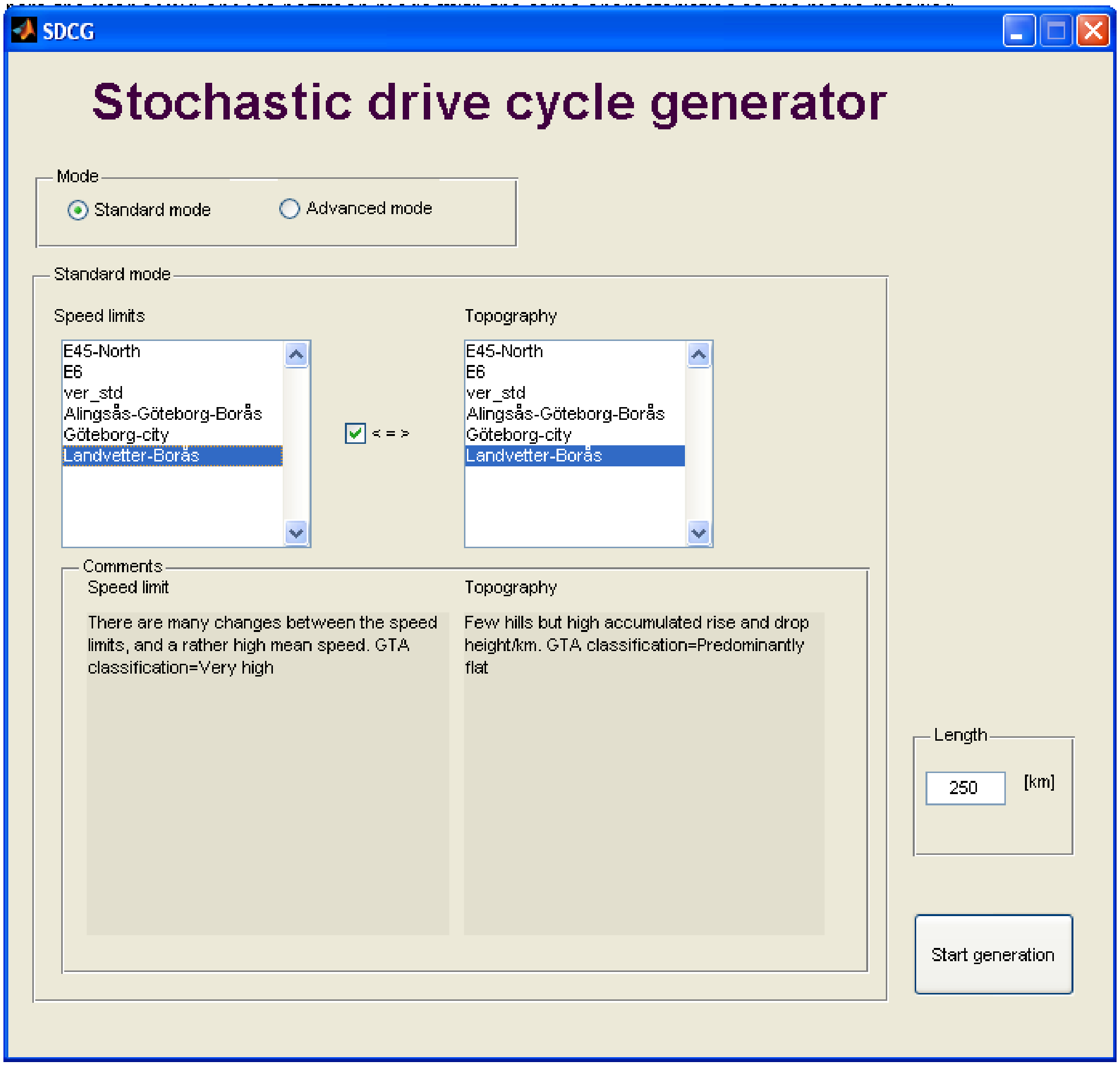

Figure 7-11. The standard mode of the GUI.

When the start button is pressed the generation of a drive cycle is started using the entered and selected values. After the generation the user is asked where and under what name to save the generated drive cycle.

8 Verification of tool performance and key parameter identification

Verification is done to test how good drive cycles generated from the same key parameters correlates and if and how the selected key parameters affect the energy consumption. How the verification is done is described in Chapter 8.1 and the results of the verification can be found in Chapter 8.2.

8.1 Verification plan

Here follows descriptions of how the verifications are done. The tool performance and the key parameter identification verification are described in different chapters.

For the verification a simple HEV model is used. Parameters for the model (weight, buffer size, frontal area etc.) are selected to realistically match a Volvo long haul truck

8.1.1 Tool performance

To test how good different drive cycles generated from the same key parameters corresponds, five drive cycles, of 300 km, are created. The five drive cycles are used in simulation with the HEV model and at every kilometer, the mean energy consumption of the diesel engine so far, is saved. The results from the five drive cycles are compared to see how good they correspond.

8.1.2 Key parameter identification

To verify that the selected key parameters actually affect the fuel consumption, the energy buffer model is run against a batch of drive cycles were one key parameter is changed at a time. Since the key parameters only refers to the topography and the speed limit these are the only drive cycle components used in the model. The simulations show the average energy consumption during a drive cycle, and if this value is changed when a key parameter is changed, the key parameter is interesting, since it affects the fuel consumption.

8.1.2.1 Topography

To test the impact from change of topography key parameters, one of the parameters are changed at a time while the rest are kept at their standard value. The speed limit is kept the same during all simulations run to test the topography key parameters. The simulated average energy consumption can be seen in Table 8-1.

Key parameter values used to verify the selection of topography key parameters:

	Standard:	Varied with:
1) Mean slopes:	0	-0.002, 0.002
2) Variance of slopes:	0.001	0.0005, 0.0015
3) Distance between hill tops:	0.5	0.8, 0.2

<u>Speed limit values during the topography verification.</u>

Existing speed limits:	70 km/h
Amplitude of change:	0
Distance between changes:	inf

8.1.2.2 Speed limit

To test the impact from change of speed limit key parameters, one of the parameters is changed at a time while the rest are kept at their standard value. The topography is kept flat during all simulations run to test the speed limit key parameters. The simulated average energy consumption can be seen in Table 8-2.

Key parameter values used to verify the selection of speed limit key parameters:

	Standard:	Varied with:
1) Existing speed limits:	[60 70 80 90]	[60 70 80 90] [60 70 80 90]
% of distance:	[25 25 25 25]	[40 30 20 10] [10 20 30 40]
2) Amplitude of changes:	20	18, 22
3) Distance between changes:	3.5	5, 2

Topography values during the speed limit verification.

Mean slopes:	0
Variance of slopes:	0
Distance between hill tops:	inf

8.2 Verification results

The results of the verifications are found in the following chapters.

8.2.1 Tool performance

In Figure 8-1 the mean energy consumption so far at each kilometre for five drive cycles is shown. The fact that the results gets more and more alike shows that drive cycles generated from the same key parameter values demands about the same average energy consumption. The reason that there still is a gap between the results is that the drive cycles are generated stochastically. If infinitely long drive cycles were created the average energy consumption would be the same.

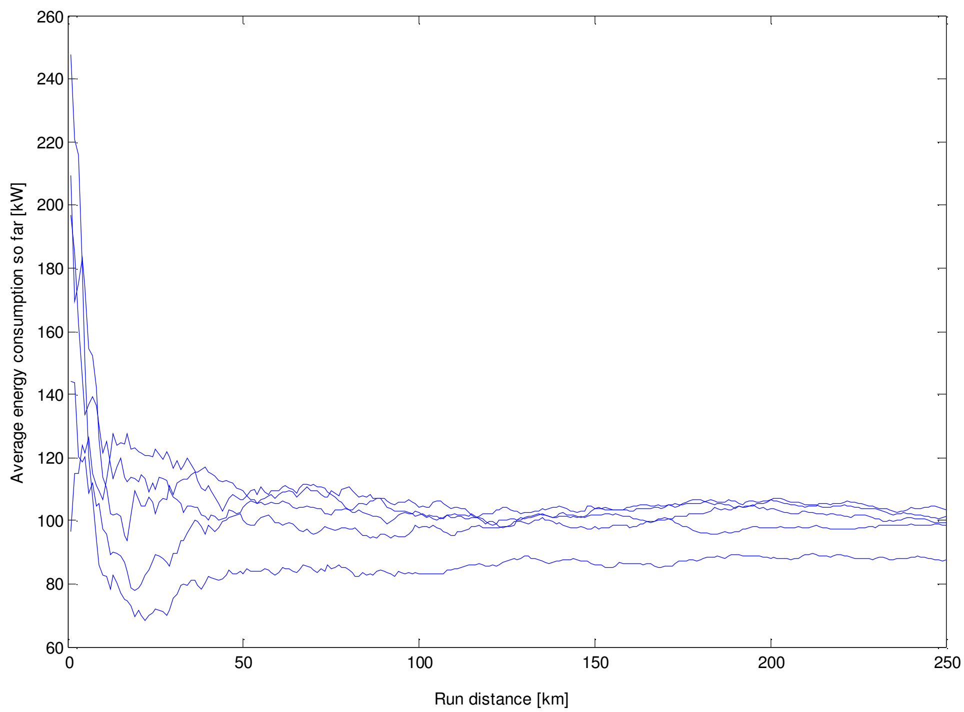

Figure 8-1. Comparison on how the average energy consumption develops during five drive cycles generated from the same key parameters.

8.2.2 Key Parameter identification

Key parameter identification is done for topography and speed limits. The verification is done separately and so is the result presentation.

8.2.2.1 Topography

The simulations show that all changes in key parameter values affect the average energy consumption. The affects also corresponds to the hypothesis from Chapter 4.1, except for the distance between hill tops parameter. The explanation for the strange behaviour when the distance between hill tops is changed might be that the parameter settings and the battery size is very good for a drive cycle with 500 meters between the hill tops.

Table 8-1. Affect on average energy consumption from change of key parameters.

Changed value	Changed to	Average energy consumption [kW]
All standard	-	82.5
Mean slope	-0.002	61.9
Mean slope	0.002	89.3
Variance slope	0.0005	76.4
Variance slope	0.0015	105.3
Distance between hill tops	0.8	91.7
Distance between hill tops	0.2	88.2

8.2.2.2 Speed

The simulations show that all changes in key parameter values affect the average energy consumption. The affects also corresponds to the hypothesis from Chapter 4.2

Table 8-2. Affect on average energy consumption from change of key parameters.

Changed value	Changed to	Average energy consumption [kW]
All standard	-	88.1
Distribution between	[40 30 20 10] %	68.7
[60 70 80 90] km/h	[10 20 30 40] %	95.7
Speed change amplitude	18	84.8
Speed change amplitude	22	91.3
Distance between change	5	84.1
Distance between change	2	92.4

9 Results

There are two important results of this thesis project, a tool for generation of drive cycles described in Chapter 9.2 and a hypothesis of what key parameters that is important in a drive cycles characteristics presented in Chapter 9.1.

9.1 The key parameters

Identification of key parameters describing the characteristics of the speed limits and the topography profile of a drive cycle is done. It is shown (in Chapter 8) that the identified parameters affect the performance and fuel consumption of a HEV.

The identified key parameters for the topography profile are:

- Mean value of slopes.
- Variance of slopes.
- Average distance between hill tops.

The identified key parameters for the speed limit profile are:

- Mean speed.
- Variance of speed.
- Average distance between speed changes.
- Mean change amplitude.

When the mean and variance for the speed limits are entered in the GUI it is entered as percentage of the distance for every speed limit.

9.2 The tool

The tool generates stochastic drive cycles according to user set values of the key parameters describing the characteristics. The parameters can be entered in a Graphical user interface, see Figure 7-10. A combination of parameter values can be saved to use for generation of more drive cycles.

Parameter value combinations saved from the advanced mode and combinations describing the example roads from Chapter 5.1 can be selected from the standard mode of the GUI, see Figure 7-11.

The outputted drive cycles includes speed limit, topography, Curvature, impact of the surrounding traffic, planned and unplanned stops. The output of the tool fits the Volvo Common Road format already used for drive cycles at Volvo Powertrain.

Drive cycles generated from the same key parameters give the same characteristics, but since the drive cycles are generated stochastically there is no guarantee for the key parameters to be fulfilled, the longer drive cycle the better mach to the key parameters. Verification shows that drive cycles of at least 100 kilometres (se Figure 8-1) should be used to make sure that the desired characteristic is included.

10 Discussion

The result of the key parameter identification is discussed in Chapter 10.1 and the discussion about the tool is found in Chapter 10.2. Some drawbacks of the approach in this thesis project are discussed in Chapter 10.3.

10.1 Key parameters

Verification shows that the key parameters identified have influence on the fuel consumption of a HEV (see Chapter 8). However it is not shown that those key parameters are the only ones affecting the fuel consumption.

One candidate for a key parameter that never got examined is; between which speed limits occur the speed limit changes. The energy consumption is probably bigger when the speed is changed between high speeds than when it is changed between low speeds. This should be examined before any big conclusions are drawn from the use of this tool.

10.2 Stochastic drive cycle generator

The stochastic drive cycle generator can create unlimited number of drive cycles according to the key parameters given by the user. This can prevent the upcoming of cycle beating during optimization and component selection. Another good aspect is that it gives the possibility to create narrowly defined environments following the identified parameters. It is also easy to work towards a broad environment by creating and using several different drive cycles.

In the generation of drive cycles from the advanced mode there is no control of characteristics not defined by key parameters. Generation from standard mode, on the other hand, uses saved Markov matrixes for speed limit and topography, thus the characteristics will be the same every generation.

There can always be improvements to increase the speed performance of the tool, more comments on this in Chapter 11.3.

10.3 Drawbacks

Since the drive cycles are generated stochastically there is no guarantee for the key parameters to be fulfilled, the longer drive cycle the better mach to the key parameters. This makes simulation on short distances (less the 100 kilometres) only usable for comparison on the same drive cycle.

The time needed to generate a drive cycle seams to be exponential to the length of the drive cycle. If several drive cycles are needed it might not be possible to generate a long drive cycle and divide it into several small cycles.

11 Future work

Ideas for some extensions and improvements arise during the work with this project. Some of the ideas are for improvement of the tool or additional key parameters to examine and some for how to use the drive cycle format.

11.1 Key parameters

A candidate for being a key parameter is where the speed changes occur. If for example the mean change amplitude is 20 km/h how much impact will it have on simulation results if the changes occur between 30 and 50 or between 90 and 110 km/h. The energy consumption is probably bigger changing between higher speeds. This should be examined before any big conclusions are drawn from use of this tool.

When generating Markov matrixes for topography in advanced mode, the order of the slopes is considered to be irrelevant. That is as long as the values of the key parameters are preserved. In reality two slopes next to each other probably are about the same. Our hypothesis is that this is not important for simulation of fuel consumption and powertrain performance. It might be a good idea to verify this theory.

11.2 Extensions

One extension that could be of big value is to create a function that reads GPS-data or an existing drive cycle and calculates Markov matrixes and key parameters to be used in the tool. In the standard mode of the tool the pre-defined roads are only from Sweden, this function could make it easy to add roads from all over the world.

The distance in front of the vehicle that the driver can overview affects the behaviour of the driver, especially the braking behaviour. If an obstacle is discovered late the braking will be more powerful than if the driving is better planned. The vision range of the driver is limited by the surroundings; therefore it would be a good idea to include the changes in vision range in the drive cycles. The changes could be signalled from the drive cycle using the "event _code" and "event_data" fields.

When approaching a planned stop it is likely that the speed limit is not very high. It is more common for bus stops to be where the speed limit is 50 or 70 km/h than where it is 110 km/h. The functionality for implement this is included in the tool but it is not parameterised due to lack of data. If it is examined how the speed limit correlate with planned stops this could be parameterised or left for the user to set the parameters.

11.3 Improvements

When it comes to unplanned stops and stop time for unplanned stop, more data is needed to get better accuracy especially for speed limits over 70 km/h, since there is no data for those speed limits included in the tool.

It should be possible to optimize the code in the program to speed up the generation of the drive cycles. The topography generation, having a lot of states, is probably the bottleneck. It seams that the time used for drive cycle generation is exponential to the generated distance.

Some kind of feed back of the progress during drive cycle generation would probably calm the user when generating long cycles, since this might take long time.

References

[1] Lin, L. Niemeier, D.A. (2003). Est*imating Regional Air Quality Vehicle Emission Inventories: Constructing Robust Driving cycles.* Transportation science, vol 37, No. 3, pp.300-346, ISSN 1526-5447

[2] Feldman, R.M. Valdez-Flores, C (1996). *Applied probability & Stochastic processes.* Boston: PWS publishing company. ISBN 053493921

[3] *Lee T.C, Judge G.G and Zellner. A Estimating the Parameters of Markov Probability Model from Aggregate time Series Data.* Amsterdam, 1970, Contribution to economic analysis, 0573-8555; 65

[4] Qipeng, Y and Xiaoqiang, L (2000). New Application of Dynamics Traffic Flow Research – Dynamics-Kinematics Model of Mixed Vehicle Flow. *Proceedings of the Conference on Traffic and Transportation Studies, ICTTS.* Jul 31-Aug 2 2000. Beijing, China *pp 541-546*

[5] Chandler, R.E. Herman, R. Montroll, E.W (1958). *Traffic dynamics: studies in car following*. Operations Research, vol 6, No 2. pp 165-184. ISSN 0030364X

[6] Gazis, D.C. Herman, R. Rothery, R.W (1961). *Nonlinear follow the leader models of traffic.* Operations Research, vol 9, No 3. pp 545-567. ISSN 0030364X

[7] Chumsamutr, R. Fujioka, T. (2003). *Development of Car-Following Model with Parameter Identification by Genetic Algorithm.* JSME international journal. Series C, Mechanical systems, machine elements and manufacturing, vol 46. No 1 pp188-196 . ISSN 1344-7653

[8] Dechter, Rina; Judea Pearl (1985). *Generalized best-first search strategies and the optimality of A*.* Journal of the ACM **32** (3): 505–536

[9] Ranjitkar, P. Nakatsuji, T. Kawamua, A. (2005). *Car-following models: an experiment based benchmarking.* Journal of the Eastern Asia Society for Transportation Studies, vol 6. No 119 pp1582-1596 ISSN 1881-1124

[10] Brackstone, M. Mcdonald, M. (1999). *Car-following: a historical review.* Transportation Research Part F, vol 2. No 4 pp181-196 ISSN 1369-8478

[11] Rockwell, T.H. Treiterer, J. (1966). *Sensing and communication between vehicles. The Ohio State University, Systems research Group, Final Report No. EFS 227-2. Columbus, Ohio.*

[12] Volvo common road format. Power Point used at Volvo Powertrain, defining the format to use in drive cycles for simulation. Volvo AB 405 08 GÖTEBORG, reception +46 31 665170