Chapter 1

Introduction and Background

In today's competitive world, the need to develop a vehicle in the most efficient manner is of utmost importance. In particular, the need exists for robust and efficient optimisation algorithms for determining the optimal spring and damper characteristics of a vehicle for both ride comfort and handling. This optimisation is difficult to perform because of two reasons. First of all the objective and constraint functions used in the optimisation are determined via computationally expensive numerical simulations. Secondly, due to the need to include non-linear effects in the numerical model to accurately simulate reality, serious numerical noise may be present in the objective function. Both these factors, namely computational expense and the presence of noise, have seriously inhibited the general use of mathematical programming methods in the optimal design of mechanical systems. This research aims to provide the reader with an efficient methodology for optimising an off-road vehicle's suspension characteristics for ride comfort and handling.

1.1 Ride Comfort vs. Handling

Throughout the history of the modern motor vehicle, the suspension system design has been a compromise between ride comfort, handling and driver control. In newer passenger vehicles this compromise has been reduced by the addition of stiff anti-roll-bars, this allows for a soft suspension setup for vertical motion, associated with ride comfort, and a stiff suspension for roll motion, typically handling manœuvres. Off-road vehicles and sports utility vehicles (SUV's) inherently have soft suspension characteristics, for good off-road manœuvrability, with the spin-off being good ride comfort, however, they are very unstable when handling is considered. Stiff anti-roll-bars are generally infeasible as they result in limited wheel travel, affecting the off-road manœuvrability.

Els (2006) investigated this compromise between ride comfort and handling in off-road vehicles. A four state semi-active suspension system, to be known as $4S_4$, was developed and tested. The unique feature of this system is that it can switch not only between different damper characteristics but also different spring characteristics. Els developed a control algorithm for this unique system that has the ability to automatically switch from the ride comfort mode to the handling mode, using no physical input from the driver. A prototype vehicle was fitted with the $4S_4$ system. Large improvements were achieved in terms of handling over the baseline vehicle, with large improvements in ride comfort when in the ride comfort setting, over the handling mode setting. This system thus eliminates the traditional compromise between ride comfort and handling, as it operates in ride comfort mode when driving in a relatively straight line, but should the driver begin a handling type manœuvre the system switches to the handling suspension mode. The handling mode's suspension characteristics are optimised for optimal handling and the ride comfort mode's suspension characteristics for optimal ride comfort, thereby eliminating the compromise associated with traditional suspension systems. The work presented in this document, discusses the optimisation of the suspension settings of the $4S_4$ system.

1.2 Development of the $4S_4$

The suspension unit currently under development, has the unique feature that it incorporates two damper packs (fitted with bypass valves) and two gas accumulators, effectively giving two damper characteristics and two spring characteristics in a single suspension unit. This unit will be referred to as the '4-State Semi-Active Suspension System', or $4S_4$ (Theron and Els 2005).

The suspension consists of two settings, namely ride comfort and handling. The handling spring setting is achieved by the compression of a small gas volume, resulting in a stiff spring stiffness. The ride comfort spring setting is achieved by the compression of both the small gas volume and a larger gas volume resulting in a soft overall spring stiffness. In addition to the variable spring settings, the damping can be varied for each spring setting. The low damping setting, desirable for ride comfort, is achieved by the pressure drop, as a result of the flow through the by-pass valves to the spring accumulators. High damping is achieved, with the by-pass valves in the closed position, by the pressure drop, as a result of the oil flow through the damper pack for the desired spring.

Switching between the two spring and damper characteristics is achieved by solenoid valves as illustrated in Figure 1.1. Valve switching times vary between 50 and 100 milliseconds depending on system pressure. Spring and damper characteristics can be taken as design variables, to be optimised for both ride comfort and handling respectively. It is assumed that the suspension system will switch between the ride comfort and handling option, to suite the operating conditions, provided an intelligent control system can distinguish between the two different operating conditions, and switch the suspension system to the correct setting. Each operating setting is expected to have different optimum values for the spring and damper characteristics. This suspension has the ability to eliminate the traditional ride comfort vs. handling compromise.

1.3 The need for Optimisation

With the off-road vehicle's suspension system already a complex compromise between ride comfort and handling, the addition of additional complexity in the form of variable spring settings and damper settings, with associated control, the use of a few hit-and-miss hand calculations will not permit

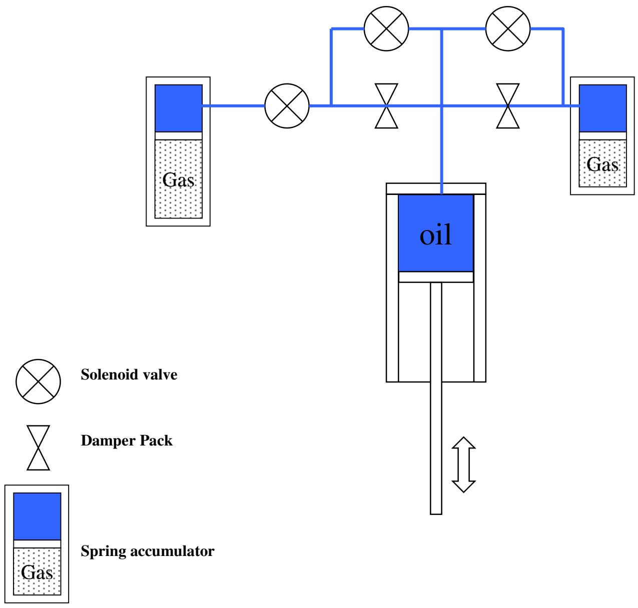

Figure 1.1: $4S_4$ Suspension Unit

the developed suspension system to live up to it's perceived qualities. To accurately model the vehicle for analysis purposes, of the new suspension system, requires the modelling of many highly non-linear components, like suspension characteristics, bushings, bump and rebound stops, and most importantly a very non-linear tyre. As a result of this complexity necessary to obtain accurate models, the design space that is to be investigated is dramatically large, non-linear and noisy. Where numerical noise in this thesis will be defined as: for small perturbations in the design variables sent to the full simulation model in MSC.ADAMS relatively large perturbations in the objective function values are noted. It has however also been suggested that this could be referred to as high sensitivity.

To accurately define the damper and spring characteristics for front and rear suspension setups requires at least 14 design variables. With such a large number of design variables, it is impossible to visualise the effect of each design variable on the ride comfort or handling, to select the optimal configuration. Additionally, the vehicle can travel at various speeds, over various terrains, and under various load conditions. The only way to take all these aspects into account is to make use of mathematical optimisation techniques. However, due to the sheer complexity of the problem to be solved, there are many aspects that need to be considered before mathematical optimisation will successfully determine the optimal suspension characteristics for the vehicle in question. The primary aim of this work is to propose a methodology for the efficient implementation of gradient-based mathematical optimisation for the optimisation of the off-road vehicle's suspension system.

1.4 Summary

In the author's masters degree dissertation (Thoresson 2003) the use of SQP and Dynamic-Q were investigated for vehicle suspension optimisation. It was found that the use of central finite differencing as opposed to forward finite differencing for the determination of gradient information for use within the Dynamic-Q optimisation algorithm, resulted in an improved optimisation convergence history. This is as a result of the central finite differences helping to reduce the undesirable effects of numerical noise on gradient determination. However, this came at the cost of additional expensive objective and constraint function evaluations per iteration. These additional expensive objective and constraint function evaluations result in a prohibitively expensive optimisation process when more design variables are considered.

The main aim of this work is the use gradient-based optimisation to efficiently optimise the off-road vehicle's suspension system for ride comfort and handling. In order to do this many steps have to be completed along the way.

This document describes the use of mathematical optimisation for vehicle suspension design, a summary into the investigation of the SQP and Dynamic-Q methods, followed by the advantages achieved when using central finite differences for gradient information, the development of accurate models to describe the vehicle dynamics, the validation of simplified models for gradient information, implementation of the simplified models for 2 and 4 variable optimisation, complications encountered with numerous design variables, a proposed automatic scaling of design variables, application of the process to 14 design variable optimisation of ride comfort and handling, and the optimisation of the compromise suspension setup.

The following original contributions to the application of gradient-based optimisation for vehicle suspension design are presented in this Thesis. Firstly the application of multi-fidelity optimisation to vehicle suspension design, in which a detailed simulation model is used for the evaluation of the objective and constraint functions and simplified models for the evaluation of the finite difference gradients. Secondly automatic scaling of the design variables with respect to the topology of the objective functions, to improve the convergence of the optimisation algorithm for the problems considered here. Thirdly the development of a robust steering driver model based on the nonlinear Pacejka Magic Formula for the description of the steering gain factors.

Chapter 2

Mathematical Optimisation

In this chapter, the use of mathematical optimisation for vehicle suspension characteristics is discussed. The general properties of gradient-based and stochastic algorithms are evaluated. The optimisation algorithms that were selected for the investigation of the problem at hand are defined, and a methodology for their implementation is defined.

2.1 The Use of Mathematical Optimisation

The use of mathematical optimisation techniques for the improvement of the engineering design process, is rapidly gaining acceptance. There is great debate in the optimisation world as to whether gradient-based approximation techniques or stochastic-based methods, like genetic algorithms, are more efficient and suited to engineering design. Stochastic techniques generally require a large starting population, in order to achieve a sufficiently feasible solution. This makes the stochastic methods computationally expensive, when expensive numerical models, of the physical system are to be optimised. Most researchers have to utilise costly multiple processing systems, as the desktop computer can take days or even weeks to arrive at a solution. On the other hand, gradient-based optimisation techniques tend to be heavily dependent on the initial starting point, and require accurate gradient information for the iterative approximation of the design space. The determination of this gradient information, is costly when many design variables are considered. The gradient calculation is also severely affected by numerical noise that is normally inherent in complex numerical simulation models, e.g. full vehicle models. Research, with reference to vehicle suspension optimisation, is now briefly discussed.

Dahlberg (1977, 1979), investigated the optimisation of a vehicle's suspension system for ride comfort and working space, subject to a random road input. A 1-degree of freedom (dof) model, was optimised using the Sequential Unconstrained Minimisation Technique (SUMT) (Fiacco and McCormick 1968). This was then expanded to a linear 2-dof model, to investigate the speed dependence of the optimal suspension settings. It was found that for a small suspension working space, the optimal spring and damper settings are heavily dependent on vehicle speed, while for a large working space the optimum is not really dependant on vehicle speed. It is suggested that active suspension systems be considered when small suspension working spaces are available.

Eberhard et al. (1995) successfully used a gradient based optimisation method (a sequential quadratic programming, or SQP, algorithm) to optimise a simple pitch-plane vehicle model's non-linear damper characteristics for ride comfort. The non-linear damper characteristic is modelled with piecewise Hermite splines. The Hermite splines, however, require difficult to handle constraints in order to ensure feasibility of the optimised damper characteristic. Nevertheless, satisfactory results were obtained. Boggs and Tolle (2000) provide an introduction to the SQP method and discuss recent developments for large scale non-linear applications.

Etman et al. (2002) designed a stroke dependent damper, for the front axle of a truck, using Sequential Linear Programming (SLP), a gradient based optimisation algorithm. They use a 2-dof quarter car model, for the initial investigation of the desired non-linear damper characteristics. Ride comfort is optimised using discrete road obstacles. The non-linear damper characteristics are modelled using an empirical piecewise quadratic approximation. Finally a full vehicle model is used for the ride comfort optimisation, for one discrete road obstacle. Bump-stop contact is ignored, to remove numerical noise and lessen computational expense. Difficulties were experienced due to poor finite difference approximations of the gradients, and with multiple feasible optima being found.

Naudé and Snyman (2003a, 2003b) and Naudé (2001) make use of a pitch-plane vehicle model to optimise the piecewise linear damper characteristics of an off-road military vehicle, for ride comfort. The 'Leap-Frog' (LFOPC) optimisation algorithm (Snyman 2000) was used, and although taking many iterations to reach the optimum, the optimisation was completed within a few seconds, because the vehicle model code was specially written for the vehicle being investigated.

Baumal et al. (1998) compared the efficiency of a Genetic Algorithm (GA) to a gradient-based optimisation method (gradient projection method) for a pitch-plane vehicle model, that was computationally efficient. The GA converged to an optimum that was only a 4% improvement over the gradient based method, but, required thousands more objective function evaluations.

Eberhard et al. (1999) investigate the use of a stochastic optimiser (simulated annealing) and a gradient-based (deterministic) optimiser (a SQP algorithm) for the optimisation of a full linear vehicle model's ride comfort. The four design variables considered are the linear spring and damper coefficients, the distance of the body center of gravity (cg) between the axles and the track width of the wheels. They conclude that deterministic optimisation approaches offer rapidly converging algorithms that often get stuck in local minima, when optimising multi-body dynamic systems. Nevertheless, the global optimum may be obtained by these methods if used within a multi-start strategy. They also find that simulated annealing is useful in avoiding local minima. It does, however, require substantially more function evaluations in order to locate the global optimum. Thus both methods are successful in locating the global optimum. They consequently suggest a hybrid combination of stochastic and deterministic algorithms for optimisation. They state, however, that the switching strategy is and will continue to be a challenging task.

Eriksson and Friberg (2000) optimised the linear spring and damper characteristics of the engine mounting system on a city bus, for ride comfort. Use was made of a linear finite element method (FEM) model to simulate the response of the bus to a given road input, with three passenger positions used for the ride comfort objective function. A 7 % improvement in ride comfort was achieved and it was found that the local minima, to which the gradient based algorithm (form of SQP algorithm, with gradients determined by forward finite differencing) converged to, were heavily dependent on the initial starting point. Eriksson and Arora (2002) investigated the use of three continuous global optimisation methods for the ride comfort optimisation of the city bus. It was found that the modified zooming method in terms of number of objective function evaluations (464) is most efficient in locating the global optimum.

Gobbi et. al. (1999, 1999) use a back-propagated Artificial Neural Network (ANN) of the full vehicle simulation model, coupled with a genetic algorithm for the optimisation of ride and handling of a sedan vehicle. Suspension non-linearities are modelled as piecewise linear approximations. The full simulation model has been verified against test data. The ANN was used for function evaluations within the genetic algorithm optimisation process. However, this methodology requires an extensive number of function evaluations, of the expensive full simulation model, to sufficiently train a representative ANN, making it infeasible for stand-alone workstations.

Schuller et al. (2002) optimised the comfort and handling of a BMW sedan using a simplified vehicle model composed of transfer functions. Because of the nature of the vehicle model the suspension design parameters were only allowed to have a small variance of 15% over the current vehicle design. This process thus aims to refine an already feasible design for the next model launch. The numerical model solves faster than real-time, making the use of genetic algorithms feasible. Only open loop handling manœuvres were considered for the optimisation process.

Andersson and Eriksson (2004) optimised the non-linear damper and spring characteristics of a full city bus vehicle model, that was validated against test data. The model consists of non-linear bushings, bump-stops, springs, dampers and a non-linear 'Magic Formula' tyre model. The ride comfort of the bus was optimised for three discrete road obstacles, with a 23 % improvement achieved. The handling was optimised using a single lane change manœuvre at 40 and 80 km/h, with a 6 % improvement achieved. The handling objective function is defined as a combination of the yaw rate gain and yaw rate time lag, with an inequality constraint limiting the maximum body roll angle to less than 1.3 *degrees*. The built-in MSC.ADAMS SQP method was used, and the optima were reached after approximately 145 function evaluations. An attempt was made at the combined optimisation of handling and ride comfort, and it was found that the result is heavily dependent on the weights assigned to the various performance objectives.

Gonsalves and Ambrósio (2005) make use of a vehicle model consisting of a flexible vehicle body and linear spring and damper characteristics, to perform optimisation of the suspension characteristics for ride comfort and handling of a sports car. The ride comfort objective function consists of the ride index, which is the summed contributions of the vibration dose values for different positions in the vehicle. The handling objective function consists of the time taken to reach steady state lateral acceleration and the overshoot of the roll angle for an open loop manœuvre. The optimisation algorithm used is the Modified Method of Feasible Directions of Vanderplaats (1992), with improvement in ride comfort and handling achieved.

Els et al. (2006) compared the efficiency of the Dynamic-Q optimisation algorithm to the SQP method for vehicle suspension optimisation. They found that the use of central finite differencing for the determination of gradient information improved the convergence of the Dynamic-Q optimisation algorithm towards a feasible optimum within fewer objective function evaluations, when compared to SQP or Dynamic-Q with forward finite differencing. The objective functions exhibited severe noise. It appeared, however, that using central finite differencing with relatively large steps in computing gradient information, was successful in smoothing out the effect of the noise in the optimisation.

Bandler et al. (2004) and Koziel et al. (2005) introduced to the engineering optimisation world the theory of 'Space Mapping', which makes use of a coarse simple model (surrogate model) and a detailed fine model for the optimisation process. The Space Mapping technique involves the matching and updating of the coarse model to more accurately describe the fine model. This has been successfully applied to the structural optimisation of a vehicle for crash safety, by Redhe and Nilsson (2004). In their research the coarse model was constructed using linear Response Surface Methodology (RSM) with the optimisation converging within 14 iterations, and using a total of 26 expensive function evaluations. However, the RSM model must be trained.

Space Mapping is also refered to as multi-fidelity optimisation, which is also defined as the use of a high-fidelity model (fine model in space mapping speak), and a medium or low fidelity model (coarse model), for optimisation. Balabanov and Venter (2004) made use of a greatly simplified finite element method (FEM) model of a full FEM model for the determination of gradient information for structural optimisation, with success. Gobbi et al. (2005) suggest the use of neural networks, or piecewise quadratic function approximations of the full simulation model, when optimising a vehicle's dynamics. van Keulen and Toropov (2006) investigate the use of the Multipoint Approximation Method (MAM) for a FEM structural problem that exhibits numerical noise. The basic idea is to replace the noisy optimisation problem with a succession of noise-free approximations at each iteration. This noise-free approximation is then optimised, and the optimum used for the next iteration point. van Keulen and Toropov (2006) also suggest the use of mechanistic approximations, to be used for the optimisation process, where the simplified numerical model is based on a prior knowledge of the physical system.

Papalambros (2002) suggests constructing surrogate models for optimisation, by making use of the computationally expensive simulation model for 'computational experiments'. With this experimental data curve-fitting techniques are applied to represent the original functions with acceptable accuracy. The problem with this method, however, is that the correct underlying form of the fit needs to be chosen, and higher accuracy requires increased sample points, resulting in increased computational cost.

The concept of Automatic Differentiation (AD) is a novel way of obtaining gradient information with one function evaluation (Tolsma and Barton 1998, Bartholomew-Biggs et al. 2000). This methodology was evaluated by Bischof et al. (2005) for the shape optimisation of an airfoil, with the objective function being evaluated by a software chain. Although AD provides more accurate gradient information than forward finite differences, the evaluation of the objective function was approximately 16 times slower than the original code for eight ($n = 8$) design variables. Using forward finite differences would have used the original code $n + 1$ times, equating to a cost of nine times the cost of one function evaluation of the original code. The other downside of AD is that access to the original source code is necessary, and it is normally not available when commercial simulation software, such as MSC.ADAMS is used.

Snyman (2005a) introduced a new implementation of the conjugate gradient method (Euler-trapezium optimiser for constrained problems, ETOPC) that overcomes the problem of severe numerical noise superimposed on a smooth underlying objective function. Snyman introduces a novel gradient-only line search, that requires two gradient vector evaluations per search direction, and no explicit function evaluations. It is also found that the computation of the gradients by central finite differences with relatively large perturbations, allowed for smoothing out of the inherent numerical noise.

The principal aim of this work is to promote the use of gradient-based optimisation algorithms for vehicle suspension optimisation. In order to do this, the complications associated with computational cost and inherent numerical noise have to be investigated. For this reason this work investigates the use of the Sequential Quadratic Programming (SQP) method and the locally developed Dynamic-Q method, for the optimisation of the suspension problem.

2.2 The SQP Method

The Sequential Quadratic Programming (SQP) optimisation algorithm is well known and is considered the industry-standard gradient-based method for constrained optimisation problems if the number of variables is not too large. The version used here is found in Matlab's Optimisation Toolbox (Mathworks 2000a). SQP makes use of successive quadratic approximations of the objective and constraint functions at each iteration step. In constructing these approximations second order differential information is required, in the form of the Hessian matrix. The Hessian matrix is approximated by making use of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) approximation. The BFGS method relies on forward finite differences to approximate the gradient of the objective function. The Hessian matrix does, however, require updating if the problem behaves poorly, requiring an extra $n + 1$ function evaluations per iteration. SQP makes use of line searches to find the solution of the approximate subproblem, this solution is then the next iteration point.

2.3 The Dynamic-Q Method

Complications associated with computational cost and inherent numerical noise have to be investigated in this study, for this reason the locally developed Dynamic-Q optimisation algorithm is used. Having direct access to the code allows more freedom to investigate the effects of different optimisation concepts. Dynamic-Q has also proved to be a feasible algorithm for vehicle suspension optimisation by Els and Uys (2003). The Dynamic-Q method has been developed to address the general optimisation problem:

$$\underset{w.r.t.x}{minimize} \quad f(\mathbf{x}), \quad \mathbf{x} = [x_1, x_2, .., x_n]^T \in R^n \tag{2.1}$$

subject to the inequality constraints:

$$g_j(\mathbf{x}) \leq 0, \qquad j = 1, 2, .., m \tag{2.2}$$

and the equality constraints:

$$h_j(\mathbf{x}) = 0, \qquad j = 1, 2, .., r \tag{2.3}$$

where $f(\mathbf{x})$, $g_j(\mathbf{x})$ and $h_j(\mathbf{x})$ are scalar functions of $\mathbf{x}$. In this formulation $\mathbf{x}$ is the vector of design variables, $f(\mathbf{x})$ is the objective function, $g_j(\mathbf{x})$ the inequality constraint functions, and $h_j(\mathbf{x})$ the equality constraint functions.

The Dynamic-Q algorithm is defined as: 'Applying a *Dynamic* trajectory optimisation algorithm to successive spherical *Quadratic* approximations of the actual optimisation problem' (Snyman and Hay 2002). This algorithm has the major advantage that it only needs to do relatively few function evaluations of the original expensive objective function to construct a simple quadratic approximate function. This new approximate sub-problem's objective and constraint functions can then be evaluated cheaply and the optimum point of the approximate sub-problem may be found economically, using the robust dynamic trajectory method LFOPC (Snyman 2000). At this new approximate optimum point, a new quadratic approximate sub-problem of the objective and constraint functions is constructed, that is again optimised. This procedure is iteratively repeated until convergence is obtained. This method is very efficient for optimising objective and constraint functions that require an expensive computer simulation for their evaluation. In standard form Dynamic-Q makes use of forward finite differences to obtain gradient information required for the generation of the approximations. The details of the method can be found in the publications by Snyman and Hay (2002), and Els and Uys (2003) where it was applied to a similar vehicle as in this study, and formed the building block for this work. A basic outline of the algorithm is set out below.

A sequence of approximate sub-problems ***P**[i] i = 0,1,2,...* are generated by constructing successive spherically quadratic approximations to the objective and constraint functions, at successive points $\mathbf{x}_i$. The approximation to the objective function, for example, is as follows:

$$\tilde{f}(\mathbf{x}) = f(\mathbf{x}_i)(\mathbf{x}-\mathbf{x}_i) + \nabla^T f(\mathbf{x}_i)(\mathbf{x}-\mathbf{x}_i) + \frac{1}{2}(\mathbf{x}-\mathbf{x}_i)^T \mathbf{A}(\mathbf{x}-\mathbf{x}_i) \qquad (2.4)$$

The Hessian matrix $\mathbf{A}$ takes on a simple diagonal matrix form:

$$\mathbf{A} = a\mathbf{I}; \qquad (2.5)$$

This form of Hessian matrix indicates that the approximate subproblems are spherically quadratic in nature. The curvature a takes on a value of zero for the first subproblem $i = 0$. Thereafter it is defined by:

$$a = \frac{2[f(\mathbf{x}_{i-1}) - f(\mathbf{x}_i) - \nabla^T f(\mathbf{x}_i)(\mathbf{x}_{i-1}-\mathbf{x}_i)]}{\|\mathbf{x}_{i-1}-\mathbf{x}_i\|^2} \qquad (2.6)$$

The approximate constraint functions are constructed in a similar manner. If the gradient vectors ∇f, ∇g, and ∇h are not known analytically they may be approximated by first order finite differences, traditionally forward finite differences are used.

Additional side constraints of the form $\hat{k}_i \leq x_i \leq \check{k}_i$ are normally imposed on the design variables. Because these constraints do not exhibit curvature properties they are treated as linear inequality constraints. These constraints thus take on the form:

$$\hat{g}_{il}(\mathbf{x}) = \hat{k}_i - x_i \leq 0, \qquad l = 1, ..., r \leq n, \qquad (2.7)$$

$$\check{g}_{iu}(\mathbf{x}) = x_i - \check{k}_i \leq 0, \qquad u = 1, ..., s \leq n, \qquad (2.8)$$

To obtain stable and controlled convergence of the solutions of successive approximate sub-problems, a move limit is set which takes on the form of an inequality:

$$g_\delta(\mathbf{x}) = \left\|\mathbf{x}-\mathbf{x}^{i-1}\right\|^2 - \delta^2 \leq 0 \qquad (2.9)$$

where δ corresponds to the specified maximum magnitude of the move limit. The approximate subproblem at $\mathbf{x}_{i-1}$ can now be solved using the dynamic trajectory 'Leap-Frog' optimisation algorithm for constrained optimisation LFOPC. This solution is taken as $\mathbf{x}_i$, the point at which the next approximate sub-problem is constructed. This process is continued until convergence is obtained. The process is illustrated in a simplified form in Figure 2.1, where f represents the approximated subproblem at each iteration step, and x_n the x value obtained at each iteration step. The x_1 value was limited by the allowable move limit.

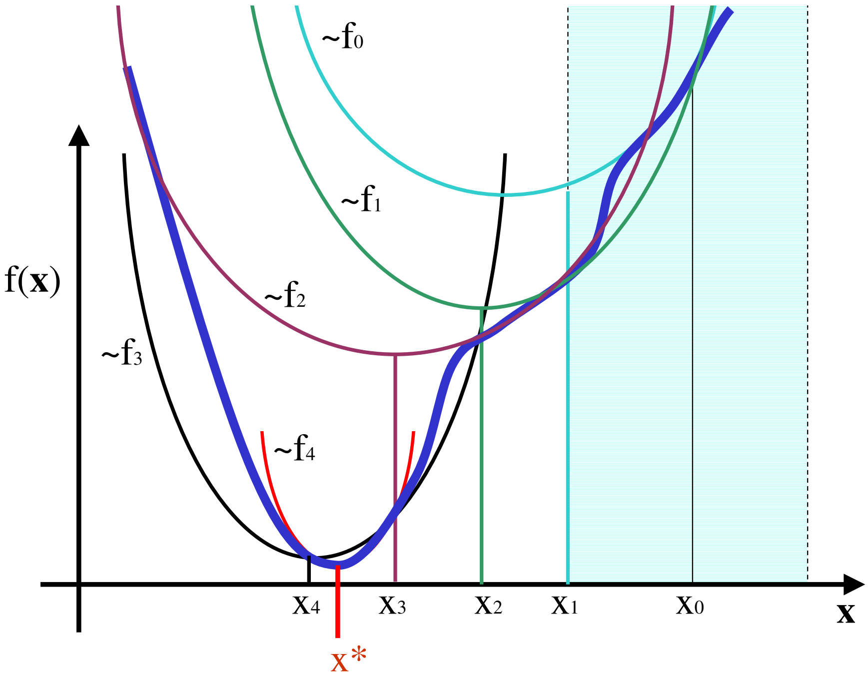

Figure 2.1: Simplified illustration on how Dynamic-Q progresses with optimisation iterations

2.4 Gradient Approximation Methods

Most gradient-based optimisation algorithms require the determination of the first and/or second order gradient information of the objective and constraint functions with respect to the design variables. In most engineering optimisation problems this gradient information is not analytically available.

The only information available to the designer is the values of objective and constraint functions obtained via expensive simulations. This paragraph investigates the use of forward and central finite differences in the Dynamic-Q optimisation algorithm, for the determination of the first order gradient information.

2.4.1 Forward Finite Difference (ffd)

This is the simplest and most economic method for approximating the gradients of the objective and constraint functions, required by gradient-based mathematical optimisation algorithms. This method approximates the first order gradient information of a multi-variable function $F(\mathbf{x})$, by evaluating the change in the function $F(\mathbf{x})$ for a small change dx_k in each of the design variables x_k, $k = 1, 2, ..., n$, as illustrated in Figure 2.2. Thus, in order to carry out the full gradient vector evaluation, a total number of $n+1$ function evaluations are required for each iteration, where n is the total number of design variables. The forward finite difference approximation to the k^{th} component of the gradient at $\mathbf{x}$ is defined as follows:

$$\frac{\partial F}{\partial x_k} = \frac{F(x_1, x_2, ..., x_k + dx_k, ..., x_n) - F(\mathbf{x})}{dx_k} \tag{2.10}$$

for $k = 1, 2, ..., n$. Noisy objective functions, however, severely limit the accuracy of the forward finite difference gradient approximation, as is apparent from Figure 2.2. This can be partly overcome by using larger step sizes dx_k or by considering instead, central finite differences.

2.4.2 Central Finite Difference (cfd)

Central finite differences make use of a function evaluation on either side of the current iteration point $\mathbf{x}$, resulting in a better approximation to the gradient of the underlying smooth function in the presence of noise. Although this method requires $2n + 1$ function evaluations per gradient vector evaluation, it may result in fewer optimisation iterations to obtain a minimum.

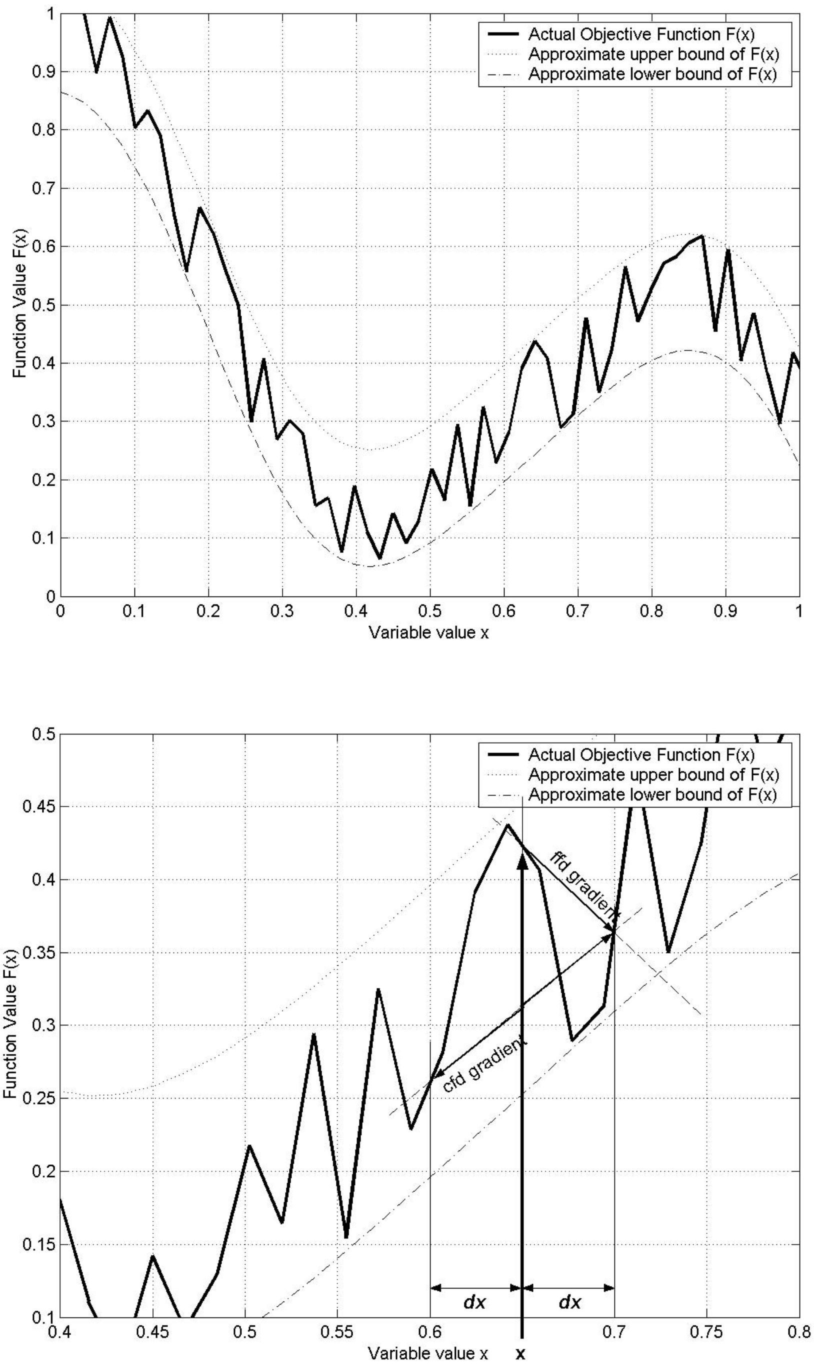

Figure 2.2: Finite difference gradient approximation methods

The central finite difference procedure is defined as follows:

$$\frac{\partial F}{\partial x_k} = \frac{F(x_1, x_2, ..., x_k + dx_k, ..., x_n) - F(x_1, x_2, ..., x_k - dx_k, ..., x_n)}{2dx_k} \tag{2.11}$$

for $k = 1, 2, ..., n$. In this way the gradient is evaluated by looking at information behind and ahead of the current iteration point, while the forward finite difference only looks ahead of the current iteration point. This results in a more accurate approximation to the function gradient, when noise is present, as illustrated for the case depicted in Figure 2.2. The effects of noise cannot be completely eliminated by this method, but it certainly yields gradient approximations that are superior to that given by forward finite differences.

2.4.3 Higher Order Gradient Information

The Sequential Quadratic Programming (SQP) method (Mathworks 2000a, Vanderplaats 1999) and other Quasi-Newton optimisation algorithms such as the Davidon-Fletcher-Powell (DFP) method uses, in addition to first order gradient approximations, also second order curvature information. This information is very costly to obtain, as it corresponds to a partial derivative of a partial derivative. This information is stored in a n x n square matrix, commonly known as the Hessian matrix. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) approximation to the Hessian matrix is used in Matlab's implementation of SQP. The Hessian matrix is approximated and updated at iteration $k+1$, $k = 0, 1, 2, ...$ by:

$$H_{k+1} = H_k + \frac{\mathbf{q}_k \mathbf{q}_k^T}{\mathbf{q}_k^T \mathbf{s}_k} - \frac{H_k^T \mathbf{s}_k^T \mathbf{s}_k H_k}{\mathbf{s}_k^T H_k \mathbf{s}_k} \tag{2.12}$$

where

$$\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k \tag{2.13}$$

and

$$\mathbf{q}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) \tag{2.14}$$

and

$$\nabla f(\mathbf{x}_k) = [\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, ..., \frac{\partial f}{\partial x_n}] \tag{2.15}$$

At the start of the optimisation procedure, (i.e. at iteration $k = 0$) most algorithms set H_0 equal to any positive definite symmetric matrix, normally the identity matrix I. Thereafter the approximation is updated at every iteration via equations 2.12 - 2.14.

2.5 Conclusions

This chapter looked at vehicle suspension optimisation research, and defined the optimisation methods to be used for the rest of this work.

The primary aim of this work is the promotion of gradient-based optimisation algorithms for vehicle suspension optimisation, due to the minimal number of function evaluations they require over stochastic based methods to arrive at a feasible optimum. The decision was thus taken that the SQP method, with it's strong industry presence, and the locally developed Dynamic-Q method will be used.

The successful implementation of gradient-based methods, is strongly dependent on good gradient information. Finite differencing is, however, necessary for the determination of gradient information when the objective and constraint functions are determined via numerical simulations. Forward and central finite differencing will be investigated for it's efficiency in determining gradient information.