

Chapter 3

Path Optimization

Background information on optimization is discussed in this chapter, along with the inequality constraints that are used for the problem. Additionally, the MATLAB program for solving the optimization problem is discussed in detail.

3.1 Optimization Routine Formulation

In formulating the optimization problem, the objective of the simulation must be defined. As mentioned in Chapter 1, the objective of the optimization routine is to determine the optimal system input variables, as functions of time, which would satisfy the time minimization and tire force maximization criteria. System input variables for this problem are the inputs of the vehicle system, steering angle and the front tire longitudinal force.

In this research, the constrained optimization problem is constructed using the penalty function approach. The first-order representation of the equations of motion is augmented to include the penalty functions of the constraints. The definitions of these variables will be discussed later in this chapter.

By representing the penalty functions as augmented state variables, this allowed the use of the MATLAB Optimization Toolbox function “*fmins*” to evaluate the optimization problem. An initial set of system input variables are used to start the optimization simulation. For proceeding iterations, *fmins* determines the search direction towards the minimum of the cost function, where the system input variables are systematically varied at the end of each iteration, after being evaluated by the cost function. This is an iterative process that terminates when the convergence criterion is satisfied. The system input variables at the final iteration are then considered optimal for the particular cost function.

The algorithm of the function *fmins* will be discussed next.

3.2 MATLAB Optimization Toolbox Function

A good solver is an engineer's best friend if he or she does a lot of simulation or analytical work numerically. "Good" is defined as being easy to use, where the parameters of the simulation model can be easily accessed and changed without too much work. Since most simulations require much time for setting up the equations and making sure they are correct, it is helpful to have a solver that contains built-in functions so that time is not wasted in developing common differentiation routines or other necessary routines. MATLAB surely has that capability. Specifically, it has a toolbox that is particularly useful for this research, the optimization toolbox. Within the optimization toolbox itself there are many commands that could be used to solve this problem. The requirements for these commands vary depending on the algorithms.

"*fmins*" is used to minimize an unconstrained multivariable cost function. This optimization command uses the simplex direction search method, which can be applied to a quadratic cost function [23]. The simplex method uses a straightforward search algorithm that goes from one solution to another in the direction that decreases the value of the cost function until the minimum solution is found. In searching for the solution, the simplex method looks at the nearby extreme points moving in the direction that will result in the most reduction in the cost function value. This method is useful when the function is discontinuous, because it does not use any approximation of a gradient of the cost function. One limitation of *fmins* is the values x and $f(x)$ must be real [23].

There is a specific way *fmins* function is used in MATLAB. Let $f(x)$ be the cost function and x be a vector of variables that will minimize function $f(x)$. When *fmins* is used, the function $f(x)$ has to be provided as a function M-file which is called by *fmins*. The initial conditions to vector x must also be provided in order for *fmins* to execute successfully. There really is not a way to determine whether optimized vector x that minimizes the cost function produces the global minimum. The results of the simulation must be intuitively evaluated.

Constrained optimization commands from MATLAB were also evaluated and compared, it was found that the unconstrained optimization command was more

effective and robust in the aspects of evaluating the value of the cost function. Specifically, *fmins* was able to generate vehicle response that was acceptable and with a higher converging rate.

3.3 Optimal Vehicle Path Simulation

The program to determine the optimal vehicle path that will satisfy the maximization of tire forces objective is setup using the structure described earlier. There are three principle parts to the program, as shown in Figure 3.1. A main program, *veh_main*, is used to define the initial conditions of the variables to be optimized. In this program, *fmins* command is called and the options related to the command are set. The second part of the program, *veh_opt*, is a function M-file which is called by *fmins* from the main program. *veh_opt* contains the objective function that once evaluated, its value is passed back to the main program so that the system input variables can be adjusted for the next iteration of the simulation. *veh_opt* calls on an integration routine, *veh_int*, that evaluates the equations of motion of the vehicle model and tire model. This part of the program consumes the most time since there are many equations to be integrated over a period of time.

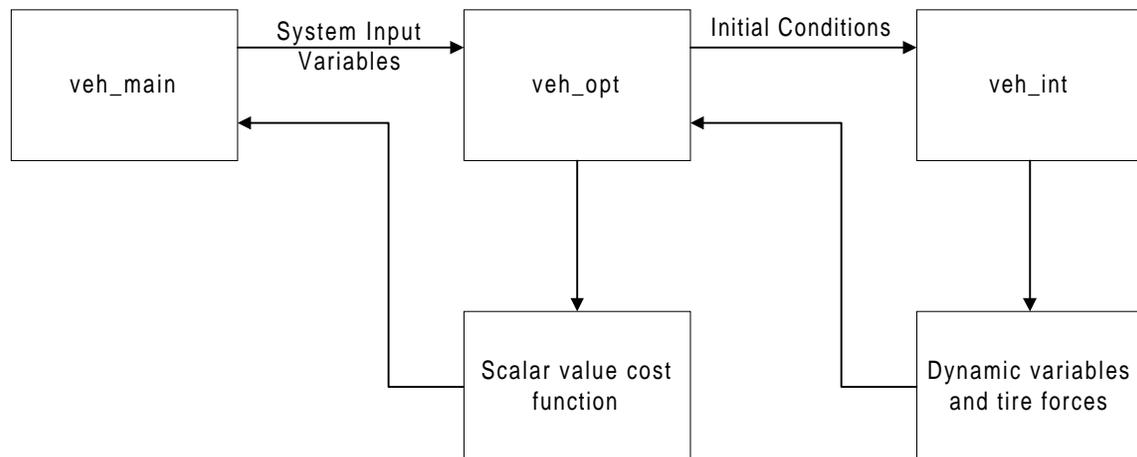


Figure 3.1 Optimal Vehicle Path Simulation Program Flow Chart

3.4 Computation Time

MATLAB is a capable solver for many engineering applications, however using its interpreter to do an optimization simulation can be very time consuming. Since it has no built-in compiler, it takes a long time for the optimization simulation to converge. However, there is a product that compiles MATLAB M-file codes into C/C++ codes. The advantage of going through this trouble is the speed gain. The reduction of computation time, once M-file code is converted to a stand-alone executable file, is about 5 to 10 times faster. This is a significant reduction in computation time at a relatively low cost.

3.5 Variable Definitions

This section describes all the variables associated with the simulation model, including dynamic variables, dynamic variable constraints, system input variables, system input variable constraints, and path constraints. Constraints are needed on certain variables because without them the values could actually reach magnitudes that are physically impossible. These constraints are discussed in details in the following sections.

3.5.1 Dynamic Variables

The dynamic variables are defined by the equations of motion of the vehicle model. These variables describe the motion of the vehicle and also its location. For this particular problem, there are 6 dynamic variables;

- yaw rate
- longitudinal velocity
- lateral velocity
- longitudinal position with respect to the inertia reference frame
- lateral position with respect to the inertia reference frame
- yaw angle of the vehicle

as shown in Eqs. (2.10) - (2.15).

3.5.1.1 Dynamic Variable Constraints

There are two constraints on the state variables; specifically the longitudinal position of the vehicle with respect to inertia reference frame and the longitudinal velocity of the vehicle.

The longitudinal position of the vehicle is constrained so the vehicle reaches the final desired location. The optimization routine uses this constraint as one of the objectives in the cost function. Eq. (3.1) describes the constraint of the longitudinal position.

$$x_4(tf) - x_{tf} = 0 \quad (3.1)$$

where

$$\begin{aligned} x_4(tf) &= \text{the longitudinal position of vehicle on X - Y plane at final time} \\ x_{tf} &= \text{the desired position of vehicle at final time} \end{aligned}$$

This term is included in the objective function to be driven to zero, making the argument statement (3.1) true. By doing so, it forces the vehicle to reach its desired final position.

The longitudinal velocity of the vehicle is also constrained. The vehicle is limited to only forward motion, therefore the speed of the vehicle cannot be negative thus the lower limit is zero. There is also an upper limit to the longitudinal velocity due to the physical limitations of engine, transmission, and tires. This constraint is set up differently than the previous one. A penalty function is used to describe the two-sided inequality as,

$$0 \leq x_3 \leq 120 \quad (3.2)$$

$$x_3 = \text{longitudinal velocity}$$

An augmented state variable is used to describe the penalty function according to Eq. (3.3).

$$\text{if } (x_3 - 0) \geq 0, \text{ and} \quad (3.3a)$$

$$\text{if } (120 - x_3) \geq 0, \quad (3.3b)$$

$$\text{then } x_7 = 0 \quad (3.3c)$$

$$\text{else } x_7 = (120 - x_3)^2 \quad (3.3d)$$

$$\text{else } x_7 = x_3^2 \quad (3.3e)$$

where

x_3 = longitudinal velocity

x_7 = augmented state describing longitudinal velocity constraint

The longitudinal velocity constraint is included in the objective function to be driven to zero, making the argument statements (3.3a) and (3.3b) true, thus forcing the vehicle speed to be between 0 and 120 miles per hour.

3.5.2 System Input Variables

The system input variables are the inputs to the dynamic system. For this particular problem, the inputs to the system are the steering angle of the front wheel and the front tire longitudinal force required for the vehicle to move forward, as represented by,

$$u_1 = \delta \quad (3.4)$$

$$u_2 = P_f \quad (3.5)$$

where

δ = steering angle

P_f = longitudinal force

3.5.2.1 System Input Variables Constraints

Constraints are put on the system input variables to limit these values generated by the optimization routine to acceptable operating ranges. The full lock steering angle for this vehicle is approximately 45 degrees for both left and right turns, as shown in Eq. (3.6). There is no rear wheel steering associated with this model. Eq. (3.7) introduces an augmented state variable as a penalty function such that its value is equal to zero if the argument in Eq. (3.7a) was not violated. Otherwise, the value of the penalty function is equal to Eq. (3.7c). Since the steering limits are equal in both left and right turns, this allows only one constraint equation to be used, by squaring the steering angle variable. The term is normalized so that the magnitude of steering angle constraint equation, x_9 , is scaled down to be between zero and one. This normalized ratio is subtracted from one, so the term can be minimized directly in the cost function.

$$-45^\circ \leq u_1 \leq 45^\circ \quad (3.6)$$

$$\text{if } \left(1 - \frac{u_1^2}{(45)^2}\right) \geq 0, \text{ then} \quad (3.7a)$$

$$x_9 = 0 \quad (3.7b)$$

$$\text{else } x_9 = \left(1 - \frac{u_1^2}{(45)^2}\right)^2 \quad (3.7c)$$

where

x_9 = augmented state variable describing steering angle constraint

This term is included in the objective function to be driven to zero, making the inequality constraint Eq. (3.6) true, and forcing the vehicle steering angle to be between -45 degrees and +45 degrees.

According to Gillespie [22], the traction limits for an independent suspension vehicle can be calculated as a function of the friction coefficient, weight distribution, wheelbase and the center of gravity location of the vehicle. Eq. (2.8) and Eq. (2.9) can

be used to determine the traction limit of a front wheel drive vehicle. Therefore, the longitudinal force must operate within the following range,

$$-\left(\mu \frac{W}{l}(a + \mu h)\right) \leq u_2 \leq \left(\frac{\mu \frac{Wb}{l}}{1 + \mu \frac{h}{l}}\right) \quad (3.8)$$

where

- μ = friction coefficient
- a = center of gravity location from front axle
- b = center of gravity location from rear axle
- W = vehicle weight
- l = vehicle wheelbase
- h = center of gravity location from ground

For this particular problem, however, the limits of the longitudinal force are simplified and defined as,

$$-5000 \leq u_2 \leq 5000 \quad (3.9)$$

The inequality argument in Eq. (3.9) allows both negative and positive longitudinal force limits to have the same magnitude. Only one augmented state variable is necessary to describe this constraint instead of two, as indicated below,

$$\text{if } \left(1 - \frac{u_2^2}{(5000)^2}\right) \geq 0, \text{ then} \quad (3.10a)$$

$$x_{11} = 0 \quad (3.10b)$$

$$\text{else } x_{11} = \left(1 - \frac{u_2^2}{(5000)^2}\right)^2 \quad (3.10c)$$

where

x_{11} = augmented state variable describing the longitudinal force constraint

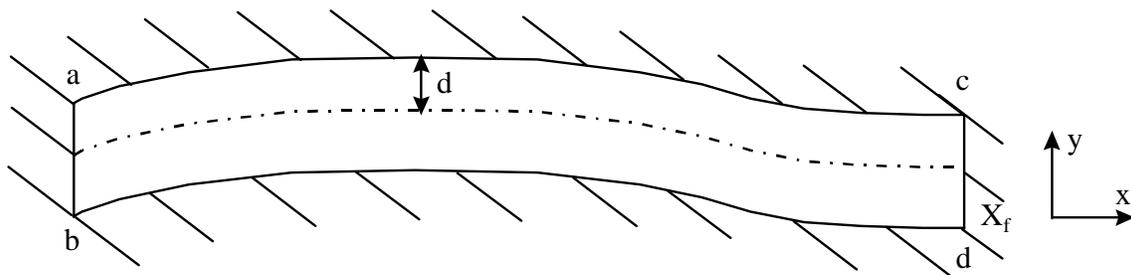
This not only simplifies the problem, but also reduces computation time in the integration routine. The longitudinal force constraint is described using the penalty function. The value of this state is zero if the longitudinal force limit is not exceeded. Otherwise, the value of the state is set to a finite number as shown in Eq. (3.10c). In effect, this number increases the scalar value of the cost function, telling the optimization routine that it is moving in the wrong direction.

This term is included in the objective function to be driven to zero, making the inequality constraint (3.10a) true, and forcing the vehicle longitudinal force to be between -5000 N and +5000 N.

3.5.3 Path Constraints

Path constraints are used to restrict the vehicle to only travel within the specific road during the simulation. Since practice, a test track has some geometric dimensions (width, length, and curves) to it, this issue needs to be addressed.

Path constraints are implemented by putting boundaries on the X and Y coordinates of the vehicle model, as shown in Figure 3.2. The vehicle now has a region that it can travel. If a boundary of the region is exceeded, a heavy penalty is assigned to the cost function.



d = track width from center line to edge

Figure 3.2. Path Constraint Definition for a Generic Road

The path of the vehicle can be described by the X,Y coordinates of the vehicle. X is the longitudinal position of the vehicle with respect to a fixed reference point. Y is the

lateral position of the vehicle with respect to the same reference point. The initial location of the vehicle is supplied in the simulation, somewhere along the line ab on Figure 3.2. The vehicle is constrained to only move forward by the longitudinal velocity constraint, i.e., Eq. (3.2a). The vehicle is free to deviate to the left or right directions. The vehicle is expected to arrive at a final longitudinal position by the end of the time duration, according to Eq. (3.1). As for the final lateral position of the vehicle, it is a bit more tricky. Referring to Figure 3.2, the vehicle has to reach the final longitudinal position X_f . Laterally, however, the vehicle could end up anywhere along the line cd , because the exact location of the final position of the vehicle is not known. In fact the final position depends on the system input variables to which the optimization algorithm converges.

To define the path constraint for the time during the starting and ending points, a function needs to be derived in terms of X and Y . Let's assume that the width of the road is approximate 4 meters. So by looking at Figure 3.2, d is equal to 2 meters. This distance is defined with respect to the centerline of the road. To account for curvature in the road, there are ways of fitting a high order polynomial to describe that curvature. For this study, however, a simplified approach was used. The strip of road is broken down into sections, where each section is described by a linear equation with a slope, m , as shown in Figure 3.3.

In Figure 3.3, q represents the current position of the vehicle along the path, and p corresponds to the vertical projection of the vehicle location onto the slope of that road section. The distance between q and p is defined by the lateral deviation, s , of the vehicle, described as,

$$s = y_{\text{prev}} + dy - y_i \quad (3.11)$$

where

- s = vehicle lateral deviation
- y_{prev} = previous y - intercept
- dy = lateral distance from y - intercept to p
- y_i = current lateral position

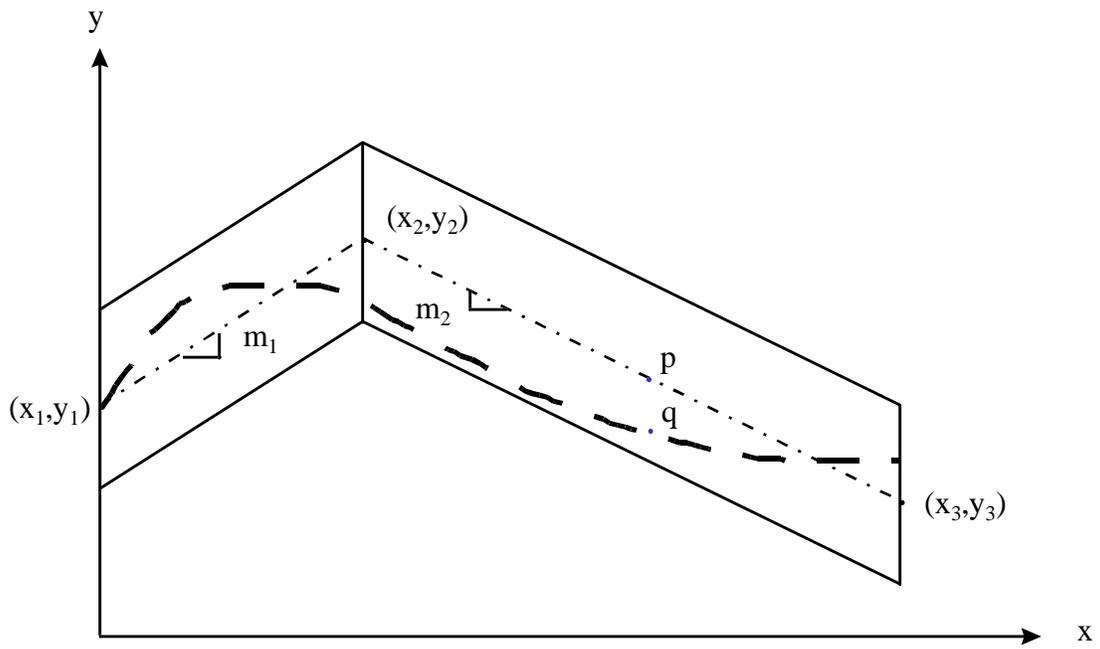


Figure 3.3 Path Constraint Function

In order to determine whether the vehicle has exceeded the track width, d , at any point along the track, the lateral deviation of the vehicle, s , is compared to d as shown in Figure 3.4.

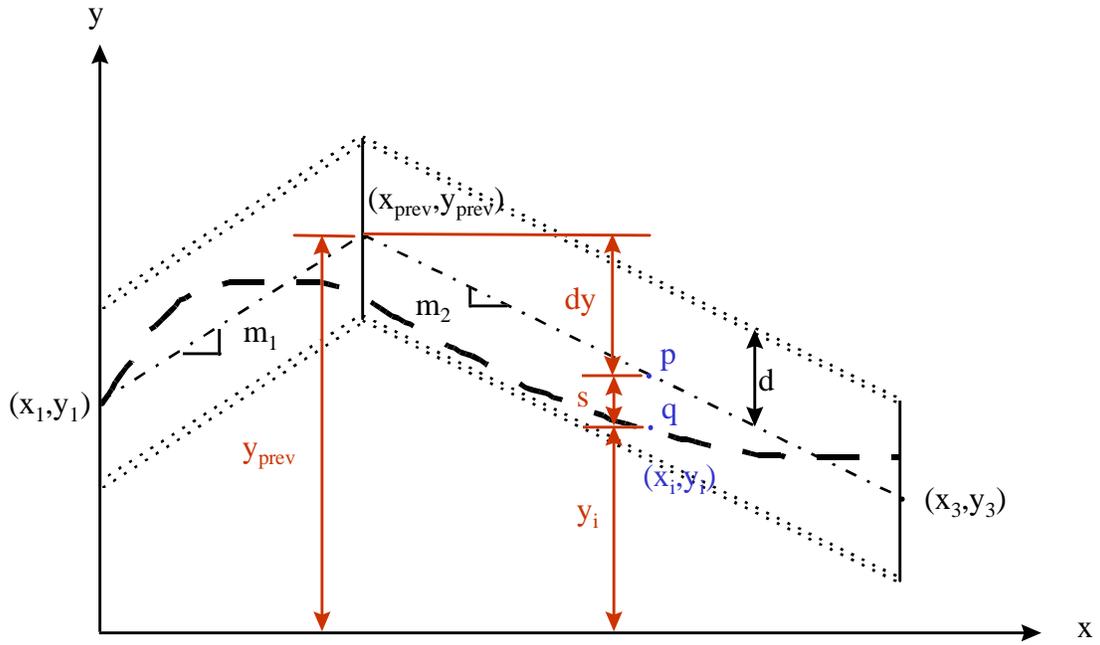


Figure 3.4 Path Constraint Function Derivation

The distance, dy is calculated using the following equation;

$$dy = m_i(x_i - x_{prev}) \quad (3.12)$$

where

m_i = slope of current road section

x_i = current longitudinal position

x_{prev} = previous x - intercept

The difference between the track width and the lateral deviation of the vehicle from the road center line is used to determine whether the vehicle satisfies the path constraint.

This constraint function is set up as an augmented state variable, which is described by

$$\text{if } (d^2 - s^2) \geq 0 \quad (3.13a)$$

$$x_8 = 0 \quad (3.13b)$$

$$\text{else } x_8 = (d^2 - s^2) \quad (3.13c)$$

where

d = track width from road center line

s = vehicle lateral deviation

x_8 = path constraint variable

The above equations imply that this state acts similar to a penalty function. If the value of Eq. (3.13a) condition were true, the state variable x_8 is set to zero. If not, x_8 is set to a scalar value which the optimization routine will drive to as close to zero as possible.

3.6 Cost Function

The cost function defines the objective of the problem. This is an important part of setting up the optimization algorithm because it directly relates to the issues of convergence and accuracy. Care must be taken when defining this function. One problem associated with multivariable cost function is that the magnitude of each variable can differ from one end of the spectrum to another. In order to correct this problem, each variable of the cost function needs to be scaled by weighting factors such that it becomes comparable to other terms in the cost function. Weighting factors also serve as sensitivity multipliers where they determine how sensitive the scalar value of the cost function is relative to other variables. The two objectives of this optimization problem are described by using two separate cost functions, which are discussed next.

3.6.1 Time Minimization Cost Function

The cost function of time minimization, described by

$$\begin{aligned} J = & \lambda_1 (x_4(t_f) - x_{4f})^2 + \lambda_2 \sum (x_7(t))^2 + \lambda_3 \sum (x_9(t))^2 \\ & + \lambda_4 \sum (x_{11}(t))^2 + \lambda_5 \sum (x_8(t))^2 + \lambda_6 t_f^2 \end{aligned} \quad (3.14)$$

where

- λ_1 = weighting factor for vehicle final position constraint
- λ_2 = weighting factor for vehicle longitudinal speed constraint
- λ_3 = weighting factor for vehicle steering angle constraint
- λ_4 = weighting factor for vehicle longitudinal force constraint
- λ_5 = weighting factor for vehicle path constraint
- λ_6 = weighting factor for vehicle final time minimization

consists of six terms where the objective is to minimize the scalar sum of these six terms. The first term defines the final position condition of the vehicle, Eq. (3.1). The second term defines the speed constraint of the vehicle, Eq. (3.2). The third term defines the steering angle constraint, as described by Eq. (3.6). The fourth term defines the longitudinal force constraint, according to Eq. (3.9). The fifth term defines the path constraint condition, Eq. (3.13). These terms are summed, according to Eq. (3.14), to determine if the constraint has been violated anywhere along the path. The last term of the cost function defines the goal of minimizing travel time.

The cost function is evaluated at the end of the integration routine rather than at every point along the path because it is less time consuming. The last term of the cost function describes the time minimization objective.

The values of the weighting factors depend on the geometry of the track. For example, if the track has many turns, a larger weighting factor is placed on the path constraint to ensure that the vehicle stays within the boundary of the track. In this study, relatively large weighting factors are placed on the final position constraint, the path constraint, and the longitudinal force constraint. These variables are more susceptible to become erroneous since minimizing the time could cause the final position of the vehicle to fall short of the desired position, the path constraint to be violated by going off the road, and the longitudinal force to exceed the maximum tire force. A small weighting factor is placed on the steering angle constraint because intuitively a path that would require the minimum time should be a straight line, using small steering angles. Table 3.1 lists the values of the weight factors used in the minimum time cost function.

**Table 3.1 Weighting Factors for Minimum
Time Cost Function**

Weighting Factor	Value
λ_1	20
λ_2	0
λ_3	1
λ_4	10
λ_5	25
λ_6	5

The values of these weighting factors were selected by trial and error approach. Since all the terms in the objective function are normalized to have values between zero and one, weighting factor values were selected to place more emphasis on certain terms so that they would be more sensitive to any input changes. The final values shown in Table 3.1 demonstrate one combination of many possible values of the weighting factors. These values, however, show the most acceptable vehicle response and the most acceptable optimization converging rate.

3.6.2 Tire Force Maximization Cost Function

The second objective of this research is to determine an optimal path which maximizes the tire forces. Let's recall the friction circle concept in Fig. 2.13, where the circle signifies the boundary of tire force operating range. The radius of friction circle is equal to the maximum allowable tire force, which is defined as the product of the friction coefficient and the vertical load. If the resultant of lateral and longitudinal forces is equal to this radius, then the tire force is maximized. Therefore, it is desirable for the value of the resultant to equal the radius of the friction circle, or as close to it as possible. An augmented state variable, defined by

$$x_{10} = 1 - \frac{F_x^2 + F_y^2}{(\mu N)^2} \quad (3.15)$$

where

- x_{10} = augmented state variable describing maximization of resultant tire force
- F_x = longitudinal force
- F_y = lateral force
- μ = friction coefficient
- N = vertical load of tire

is introduced to maximize the resultant tire force.

This term is included in the cost function,

$$J = \lambda_1 (x_4(t_f) - x_{4f})^2 + \lambda_2 \sum (x_7(t))^2 + \lambda_3 \sum (x_9(t))^2 + \lambda_4 \sum (x_{11}(t))^2 + \lambda_5 \sum (x_8(t))^2 + \lambda_6 \sum (x_{10}(t))^2 \quad (3.16)$$

where

- λ_1 = weighting factor for vehicle final position constraint
- λ_2 = weighting factor for vehicle longitudinal speed constraint
- λ_3 = weighting factor for vehicle steering angle constraint
- λ_4 = weighting factor for vehicle longitudinal force constraint
- λ_5 = weighting factor for vehicle path constraint
- λ_6 = weighting factor for vehicle tire force maximization

making the ratio of the resultant force to the maximum allowable force to equal one or as close to one as possible. By doing so, the resultant tire force is maximized. This method of cost function variable maximization is implemented rather than using the conventional method of placing a negative sign in front of the term to be maximized. The cost function also contains other terms to be minimized, the conventional method may cause conflicting objective. Making all the terms to carry the same sign will reduce the chances of the optimization routine not converging. The tire force maximization cost function is shown in Eq. (3.16)

The term x_{10} , may be impossible to be driven to zero at all times because it may be impossible to operate the vehicle such that tire forces lie on the friction circle at all times.

Table 3.2 lists the values of the weighting factors used in this cost function. Larger emphasis is placed on the path constraint because significant tire forces could easily propel the vehicle outside the track. Final position constraint and the tire force maximization objective are equally weighted they have relatively the same sensitivity to system input changes.

Table 3.2 Weighting Factors for Maximum Tire Force Cost Function

Weighting Factor	Value
λ_1	10
λ_2	0
λ_3	1
λ_4	1
λ_5	20
λ_6	10

The values of the weighting factors were selected by using a trial and error approach. Since all the terms in the objective function are normalized to have values between zero and one, weighting factor values were selected to place more emphasis on certain terms so that they would be more sensitive to any input changes. The final values shown in Table 3.2 demonstrate one combination of many possible values of the weighting factors. These values, however, show the most acceptable vehicle response and the most acceptable optimization converging rate.