6
Referências Bibliográficas

1 D. Casanova, R. S. Sharp and P. Symonds, Minimum Time Manoeuvring: The Significance of Yaw Inertia, Vehicle System Dynamic, vol. 34, pp. 77-115, 2000.

2 H. S. Radt, W. F. Milliken, Non-dimensionalizing tyre data for vehicle simulation, Milliken Research Associates, Inc, New York, USA, 1983.

3 Charles C. MacAdam, Application of an optimal preview control for simulation of closed loop automobile driving, IEEE Transactions on systems, Man, and Cybernetics, vol. SMC-11, N° 6, June 1981.

4 H. Hatwal, E.C. Mikulcik, Na optimal control approach to the path tracking problem for an automobile, the University of Calgary, Calgary, Alberta, Canada, 1986.

5 A. Y. Maalej, D. A. guenther and J.R. Ellis, Experimental development of tyre force and moment models, Ohio State University, Columbus, USA, of Vehicle Design vol. 10, N° 1, 1898.

6 Dirk E. Smith and John M. Starkey, Effects of model complexity on the performance of automated vehicle steering controllers: controller development and evaluation, Vehicle System Dynamics, N° 23, 1994.

7 Dirk E. Smith and John M. Starkey, Effects of model complexity on the performance of automated vehicle steering controllers: model development and evaluation, Vehicle System Dynamics, N° 24, 1995.

8 Anthony B.Will and Stanislaw H. Zak, Modelling and control of an automated vehicle, Vehicle System Dynamics, N° 27, 1997.

9 E. Velenis and P. Tsiotras, Minimum Time vs Maximum Exit Velocity Path Optimization During Cornering, Georgia Institute of Technology School of aerospace Engineering, Atlanta, GA, USA, 2004.

10 E. Velenis and P. Tsiotras, Optimal velocity profile generation for given acceleration limits; the half-car model case, Georgia Institute of Technology School of aerospace Engineering, Atlanta, GA, USA, 2004.

11 Speranza Neto, M.; Spinola, A. L., “Analise do comportamento dinâmico de um veículo em uma trajetória pré-definida através de um modelo cinemático em malha fechada”, XIV Congresso e Exposição Internacionais de Tecnologia da Mobilidade (SAE Brasil), Society of Automotive Engineers, São Paulo, SP, 2005.

12 Speranza Neto, M., Spinola, A. L., Hey, F., “Análise do Comportamento de um Veículo em uma Trajetória Fechada Pré-Definida Através de um Modelo Dinâmico Linear em uma Malha de Controle”, submetido ao XV Congresso e

Exposição Internacionais de Tecnologia da Mobilidade (SAE Brasil), Society of Automotive Engineers, São Paulo, SP, 2006.

13 Spinola, A. L., “Modelagem e Controle Não Linear da Direção de um Veículo Terrestre” Dissertação de Mestrado, DEE/PUC-Rio, Rio de Janeiro, Dezembro de 2003.

14 http://servicios.hoy.es/datos/motor/michelin.html

15 http://www.renaultf1.com/en/car/chassis/

16 Georg Rill, Vehicle Dynamics, university of applied sciences, October 2004.

17 Josehp Kart, Aerodynamics of Race Cars, Department of Aerospace Engineering, San Diego State University, San Diego, 2006

18 Blanco, Ruy, Dinâmica dos veículos sobre rodas, curso automotivo para engenheiros, APAD & fundação CEFETBAHIA.

19 Jóvaj, M. S., Motores de Automovil, Editorial MIR Moscú, 1982.

20 William F. Milliken and Douglas L. Milliken, Race car vehicle dynamics, SAE international, USA 1995.

21 Ramanata Peeronon, Optimal Vehicle Path Generator using Optimization Methods, Virginia Polytechnic Institute and State University, Blacksburg - Virginia, April 1998

22 L. Valadares Tavares, F. Nunes Correia, Otimização linear e não linear, 2° edição, Fundação Calouste Gulbenkian, Lisboa 1999.

23 Antonio Galvão Novaes, Métodos de Otimização aplicados aos transportes, editoria Edgard Blücher Ltda., Brazil 1978.

24 http://www.gta.ufrj.br/~marcio/genetic.html

25 The MathWorks, Otimization Toolbox for user with Matlab, version 3, 2005.

Apêndice

<u>Apêndice A</u>.- Tratamento no programa *Matlab/Simulink* para a obtenção dos resultados do modelo dinâmico não linear.

Existem muitas variáveis que podem ser obtidas como resultado da simulação deste modelo, e para facilitar a observação, os gráficos são separados em grupos. A Tabela A.1 a seguir indica os comandos a serem executados para obtenção de gráficos com os resultados depois da simulação.

Tabela A.1. Comandos do programa Matlab para visualizar todos os resultados do modelo dinâmico não linear do veículo.

Comando	**Descrição**
plotEnt	Mostrará a função entrada da volante (no tempo), e os ângulos de esterçamento das rodas dianteiras.
plotDeriva	Mostrará os ângulos de Deriva.
plotFlat	Mostará as forças laterais em cada pneu.
plotFnormal	Mostrará as forças normais em cada pneu.
plotAerodinamica	Mostrará as forças aerodinâmicas, longitudinal e vertical.
plotFxyz	Mostrará as forças no plano ***xy***, a forças total no eixo ***x*** e no eixo ***y***, o momento total no eixo ***z*** **.**
plotAcelera	Mostrará as acelerações em cada eixo do referencial local do veículo.
plotVelocidade	Mostrará as velocidades, tanto velocidades angulares quanto velocidade longitudinal e lateral.
plotSaida	Mostrará os resultados finais do modelo, os três ângulos de rotação ***yaw***, ***roll*** e ***pitch***, e as duas posições ***x*** e ***y*** no tempo, e tambem mostrará o deslocamento espacial do veículo.
plotTodo	Mostrará todos os resultados dos comandos anteriores.

<u>Apêndice B</u>.- Obtenção da trajetória ótima no Toolbox de Matlab.

Descreve-se a seguir o emprego da ferramenta computacional adotada para a solução do problema de otimização descrito no Capitulo 3. O programa *Matlab* possui rotinas pré-programadas para otimizar funções com restrições. Este pacote de otimização tem diversos algoritmos apropriados para diferentes casos. Apresenta-se aqui apenas aqueles de interesse do problema a seguir resolvido.

A rotina empregada para minimizar uma função escalar de muitas variáveis com restrições não lineares [26] é a “***fmincon***”. Para iniciar a procura do resultado ótimo esta *function* precisa de uma estimativa inicial que deve ser estimada pelo usuário. A "***fmincon***" utiliza um método de programação não linear, que não será aqui tratado. A função objetivo adotada pode ser de muitas variáveis, representada por um vetor, mas que gera um escalar como resposta. As restrições podem ser lineares ou não lineares. A escolha recaiu sobre esta *function* pois o problema de trajetória ótima é tipicamente não linear, tanto no que diz respeito a função objetivo quanto às restrições.

Para utilizar a *function* "***fmincon***" deve-se definir uma outra *function* em *Matlab* que contém a função objetivo: "***myfun.m***"; assim com uma que contém as restrições do problema de otimização: "***confuneq.m***". As restrições são divididas em dois tipos: restrições de igualdade, representada através de uma matriz $C_{eq}(A)$ onde ***A*** é o vetor que tem por componentes todas as variáveis de otimização; e as restrições de desigualdade, representada por uma matriz $C(A)$. Para tratar do problema de otimização considerado, criou-se um programa *Matlab* dedicado "***Otimizar.m***", no qual define-se a função objetivo e também as restrições utilizadas na rotina "***fmincon***", e determina-se a trajetória ótima considerando o tempo mínimo.

Neste item explica-se o procedimento para a obtenção da trajetória ótima em um caso geral utilizando o programa *Matlab*. Os passos a seguir são:

1. A sintaxe com que o programa trabalha que tem a seguinte forma:

   [x,fval] = fmincon (@myfun,A0,[],[],[],[],[],[], @confuneq)

   onde ***myfun*** é uma função definida em *Matlab* que contém a função objetivo, ***confuneq*** é uma função definida em *Matlab* que contém as restrições, e ***AO*** é o vetor que contém os valores iniciais das variáveis de otimização. Esta *function* devolve o vetor ***X*** com os valores de todas as variáveis que são resultados do problema de otimização, e o valor da função objetivo ***fval***.
2. A função objetivo deve ser definida em ***myfun***. Neste caso se quer minimizar o ***tempo***, que depende do tipo de movimento do veículo, conforme analisado anteriormente. Em aceleração ou desaceleração ou velocidade constante, estas funções estão definidas no Capitulo 3. Caso se tenha um problema mais complexo, no qual existem os três tipos de movimento, simplesmente se trabalha por trechos e se faz a soma dos tempos de cada trecho, ou seja,

$$T = T_{aceleração} + T_{desaceleração} + T_{constante}$$

Deve-se lembrar que, ao se passar de um trecho a outro, se iguala as condições finais as condições iniciais do outro.

3. Os valores iniciais que o programa precisa para fazer as iterações encontram-se na função ***Otimizar***, definidos da seguinte forma

$$A0 = [A0_{(1)},A0_{(2)},A0_{(3)}, ...]$$

Lembra-se que na função ***Otimizar*** chama a *function "**fmincon**"*

4. As restrições são definidas em ***confuneq***. O número de restrições depende da complexidade da função objetivo, assim como o número de valores iniciais, que estão associados ao tipo de movimento que se quer analisar. Em um caso geral, no qual estão presentes todos os tipos de movimentos antes mencionados, tem que se considerar também o tipo de pista que o veículo vai percorrer. Isto é, se o veículo faz uma curva simples ou uma curva dupla, ou simplesmente uma reta. Se o veículo faz alguma curva, deve-se levar em conta a aceleração lateral e os seus efeitos. Deve-se ter atenção para não ultrapassar o número máximo de restrições admitidas pelo programa.
   Uma das restrições está relacionada com as acelerações, que são descritas por desigualdades matemáticas, como mostrado no Capitulo 3, de modo que a aceleração total fique sempre dentro do Círculo de Aderência. Estas desigualdades devem ser inseridas no programa na forma de um vetor, onde cada componente está associada a uma restrição, que por definição. devem sempre ser negativos. Assim tem-se

$$C = [C_{(1)},C_{(2)},C_{(3)}, ...]$$

e as restrições de aceleração ficam

$$C_{(i)} = \text{Aceleração Total} - \text{Aceleração Máxima}$$

onde a Aceleração Total é dada para todos os pontos do movimento, ou nos pontos críticos onde se tenha certeza que as acelerações totais sejam a maiores do movimento; e a Aceleração Máxima é aquela obtida do Circulo de Aderência. O número de restrições de desigualdade para a aceleração depende do tipo de movimento.

5. Outra restrição de desigualdade é dada pela velocidade do veículo. Pode-se considerara esta restrição para todos os pontos do movimento, ou para os pontos críticos nos quais se tenha certeza que neles existirão as maiores velocidades de todo o movimento. As relações relativas às velocidades e as restrições associadas forma mostradas no Capítulo 3, levando a

$$C_{(i)} = \text{Velocidade} - \text{Velocidade Máxima}$$

6. A restrição de maior complexidade é aquela que delimita a trajetória do veículo dentro dos limites da pista. Para definir tais limites existem duas funções matemáticas: uma para cada lado da pista, os chamados limite inferior e limite superior. Estas funções devem ser avaliadas no programa *Simulink*. Depois de se realizar a simulação da trajetória, são obtidos os dois vetores posição vertical (no eixo "***Y***" do referencial global), um para cada limite da pista, que dependem da posição horizontal (no eixo "***X***" do referencial global). Da mesma forma se obtém da simulação todos os pontos da trajetória percorrida pelo veículo (no eixo "***Y***"), as acelerações empregadas e velocidades alcançadas. Após este levantamento, tais vetores são comparados e se dividem em dois grupos de desigualdades, de acordo com as equações seguintes. Este mesmo procedimento é empregado para a determinação das restrições associadas ao traçado, levando a

$$C_{(i)} = \text{Trajetória - Limite Superior}$$

$$C_{(i)} = \text{Limite Inferior - Trajetória}$$

Estas desigualdades devem ser obtidas em cada ponto "***i***" da posição horizontal (eixo "***X***" no referencial global). Deve-se observar que o número de pontos empregados para descrever a trajetória depende da complexidade da forma da pista. A discretização será feita para cada tipo de traçado em particular.

7. Uma restrição de igualdade é adotada para estabelecer que o ponto final da trajetória percorrida pelo veículo coincida com o ponto final da função objetivo. Este ponto final é obtido da simulação, e deve ser igual à distância total, estando associado ao valor final da função aceleração. Assim,

$$Ceq_{(i)} = \text{Dist. final Aceleração e Tempo} - \text{Dist. final da Trajetória}$$

No programa *Simulink* se define uma função ***Stop*** de parada, onde se especifica que quando o veículo chegar ao final da pista a simulação pára. Este ponto final da pista deve estar muito bem definido pela geometria da pista.

8. Outras restrições de igualdade são definidas para estabelecer a continuidade entre os diferentes tipos de movimento que existem em cada trecho da trajetória total do veículo. Os casos de movimento independentes uns dos outros são:
   - Velocidade constante em linha reta.
   - Velocidade constante em curva.
   - Velocidade variável em linha reta, aceleração.
   - Velocidade variável em linha reta, desaceleração.
   - Velocidade variável em curva, entrada à curva.
   - Velocidade variável em curva, saída dá curva.

   Para conectar estes movimentos é preciso igualar as condições iniciais de um com as condições finais do outro. O número de restrições de igualdade depende da quantidade de trechos e/ou dos tipos de movimentos diferentes a serem concatenados. Para um trecho ***n*** qualquer se tem

$$Ceq_{(i)} = \text{Condição inicial}_{(n+1)} - \text{Condição final}_{(n)}$$

Algumas condições iniciais e finais já são igualadas pela continuidade das funções ou mesmo pelo programa de simulação, mas pode ser preciso especificar, através de restrições de igualdade, por exemplo, os pontos de aplicação de um tipo de aceleração (final e inicial) dentro da distância percorrida total.

9. Apenas os pontos inicial e final da pista estão especificadas no problema (pela pista), os outros pontos da distância percorrida são variáveis resultantes do problema de otimização e seguem uma seqüência lógica crescente, isto é, não podem ser maiores que o ponto final, nem menores que o ponto inicial da pista. Quando o movimento está se desenvolvendo, os pontos associados à distância sempre estão crescendo, estando já conectados quando as condições inicias e finais de cada trecho ou movimento foram igualadas. Não se pode permite trechos "soltos", pois todos estão conectados. Pode ser necessário, entretanto, dentro de um mesmo trecho, deixar pontos flutuantes, e o programa de otimização irá estabelecer o ponto ótimo de aplicação de uma determinada condição. Por exemplo, na entrada da curva, antes que o veículo inicie a aceleração lateral, ele já está desacelerando, e, portanto, o ponto de aplicação da aceleração lateral está flutuando dentro de uma faixa. Outro caso parecido acontece na saída da curva, e ainda outro ocorre na aceleração em linha reta, quando o veículo chega a sua velocidade máxima e se movimenta, a partir daí, com velocidade constante. Assim, deve-se especificar como uma restrição de desigualdade, as seguintes condições

$$C_{(i)} = \text{Inicio do trecho} - \text{Aplicação da condição}$$

$$C_{(i)} = \text{Aplicação da condição} - \text{Final do trecho}$$

10. A partir das restrições impostas, o programa pode obter os valores ótimos do problema, fornecendo como resposta o valor da função objetivo, o tempo mínimo, e os valores de todas as variáveis de otimização, as características de aceleração em cada trecho, ou seja

$$fval = T = T_{aceleração} + T_{desaceleração} + T_{constante}$$

$$x = [A_{(1)}, A_{(2)}, A_{(3)}, \ldots]$$

Os resultados obtidos pelo procedimento apresentado, em cada tipo de movimento, são mostrados no Capítulo 4 da dissertação, empregando o *Simulink* para reprodução do movimento através do modelo de massa pontual e do modelo dinâmico não linear do veículo, para fins de comparação.