

2013 AASRI Conference on Intelligent Systems and Control

Modeling the Lane Contention Algorithm for Control in Intelligent Vehicular Systems

Mayur Parulekar¹, Dr. V. Ramesh¹, Prasad Joshi¹, Viraj Padte¹, Dhaval Shroff¹

*Research & Innovation Center, D.J Sanghvi College, Mumbai, India
mvp@parulekar@gmail.com; virajpadte@gmail.com*

Abstract

Delegating the responsibility of safe driving on an autonomous vehicle needs the autonomous system to be extremely robust. In an attempt to design a robust guidance system for the autonomous vehicle we decided to go refer the most complex system designed which is “humans”. In humans we peculiarly observe that there exists an array of senses we possess that provide the required robustness and efficiency in chores that we perform. It clearly displays that in human beings the various senses we possess precisely complement each other to support a process. On this basis we propose an introduction to optimization of lane detection and tracking. Also the scalability of computation for smart vehicles such as auto park assist, cruise mode or auto lane keep assist continue to grow for driverless manoeuvre thus removing the possibility of manual or judgmental error in driving. We model a ground vehicle in a turning manoeuvre, such as a step lane-change where modeling process follows that of a vehicle undergoing translational as well as rotational motion. We shall be using this to design the control algorithm for executing a lane change maneuver under a safety set of rules using Dynamic Matrix Control which is a subset of the Model Predictive Control. This requires that we transform the vehicle lateral variable into the fixed coordinates. The application depends on recursively obtained sensor readings in a feedback loop mode for processing and deploying a corrective action that exploits non-linear functions of the state and finds control inputs such as state of system, position, acceleration, peer movement to recursively estimate and improve the quality of resulting estimation for collision avoidance and target localization.

© 2013 The Authors. Published by Elsevier B.V. Open access under [CC BY-NC-ND license](#).
Selection and/or peer review under responsibility of American Applied Science Research Institute

Keywords: Model predictive control, dynamic matrix control, labview, vehicular localization, lane contention

1. Introduction

Delegating the responsibility of safe driving on an autonomous vehicle needs the autonomous system to be extremely robust. In an attempt to design a robust guidance system for the autonomous vehicle we decided to go refer the most complex system designed which is “humans”. In humans we peculiarly observe that there exists an array of senses we possess that provide the required robustness and efficiency in chores that we perform. It clearly displays that in human beings the various senses we possess precisely complement each other to support a process. Let us take a brief example case shown in figure 1a-b.

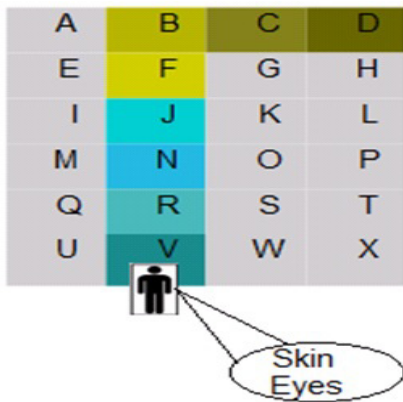


Figure 1.a- Human Sensory Perception

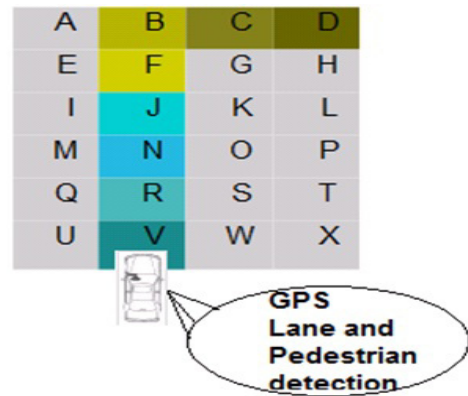


Figure 1b- Car Sensory Perception

Suppose a human has to navigate from a blue room to a red room in a complex of rooms where the additional information for navigation is that the red room is extremely hot while the blue room is cold. As seen the path the human follows to reach the right destination is V-R-N-J-F-B-C-D path where he uses his eyes for identifying the color. Also the temperature increases as the human approaches the hotter room. This additional sense increases speed of approach and also facilitates choosing the direction and hence the path for reaching undergoes optimization. This indicates that the navigation efficiency increases as an array(in this case two) of sensors which are eyes and skin are used for color detection and temperature sensing respectively. This can now be equated to autonomous navigation system. Suppose a vehicle needs to navigate for place A to place B. Primarily a GPS will be used for navigation by utilizing its co-ordinate information for path determination and tracking. To complement this navigation process we propose to add the element of pedestrian and lane tracking mechanisms for making navigation robust and fast. This is illustrated in figure 1b.

2. Mathematical Modelling for Lane Contention Using SCILAB

It is necessary to do a mathematical modelling for predicting the controller action based on changing inputs. The Model Predictive Control [1] algorithm is used below to take these into account where the inputs used are acceleration (positive and negative), mass, inertia and steering, peer position of vehicles in the neighbourhood set whereas the outputs are the positional co-ordinates of the vehicles. Consider the parameters

R: No of Measured Process Variable; S: No of Control Outputs; P: Prediction Horizon; M: Control Horizon.

A: Dynamic Matrix = $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$; where a_{11} = Matrix of Step-Response coefficients and

Δu = Vector/Matrix of control Outputs of S variables for the next M sampling instants.

In Dynamic Matrix Control which is a subset of Model Predictive Control we wish to have N_p future outputs match some optimum trajectory by finding the best values of N_c future changes in the manipulated variables. This is same as fitting N_p data points with an equation with N_c co-efficients where N_p should be greater than N_c , obviously. Suppose there are N_p data points that give the values of measured variable y at N_p known values of two variables X_1 and X_2 shown in table 1 below then we wish to find the values of two parameters M_1 and M_2 such that the measured variable y is given by the equation from table 1 and where \hat{y} is the predicted value of y given by

$$\hat{y} = M_1 X_1 + M_2 X_2 \quad (1)$$

Table 1

DATA POINTS	X_1 VALUES	X_2 VALUES	Y VALUES
1	X_{11}	X_{21}	Y_1
2	X_{12}	X_{22}	Y_2
3	X_{13}	X_{23}	Y_3
\vdots	\vdots	\vdots	\vdots
N_p	X_{1N_p}	X_{2N_p}	Y_{N_p}

We can apply this to each of the data points as

$$\hat{y}_i = M_1 X_{1i} + M_2 X_{2i} \quad (2)$$

The Performance Index is the error in prediction, given by;

$$J = \sum_{i=1}^{N_p} (y_i - \hat{y}_i)^2 \text{ OR } J = \sum_{i=1}^{N_p} (y_i - M_1 X_{1i} - M_2 X_{2i})^2 \quad (3)$$

Differentiating J w.r.t M_1 and M_2 and setting the derivatives to zero, we get;

$$\frac{\partial J}{\partial M_1} = 2 \sum_{i=1}^{N_p} (y_i - M_1 X_{1i} - M_2 X_{2i})(-X_{1i}) = 0 \quad (4)$$

$$\frac{\partial J}{\partial M_2} = 2 \sum_{i=1}^{N_p} (y_i - M_1 X_{1i} - M_2 X_{2i})(-X_{2i}) = 0 \text{ OR } M_1 \sum_{i=1}^{N_p} X_{1i}^2 + M_2 \sum_{i=1}^{N_p} X_{1i} X_{2i} = \sum_{i=1}^{N_p} y_i X_{1i} \quad (5)$$

$$\text{And } M_1 \sum_{i=1}^{N_p} X_{1i} X_{2i} + M_2 \sum_{i=1}^{N_p} X_{2i}^2 = \sum_{i=1}^{N_p} y_i X_{2i} \quad (6)$$

These two equations become:

$$X^T X M = X^T Y \quad (7)$$

$$\text{Where, } X = \begin{bmatrix} X_{11} X_{21} \\ X_{21} X_{22} \\ \vdots \\ X_{1N_p} X_{2N_p} \end{bmatrix} \quad M = \begin{bmatrix} M_1 \\ M_2 \end{bmatrix} \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix}$$

$$\text{and } M = (X^T X)^{-1} X^T Y \quad (8)$$

Suppose our model consists of N_c parameters for N_c variables then

$$\tilde{y}_i = m_1 x_{1i} + m_2 x_{2i} + \dots + m_{N_c} x_{N_{ci}} \quad (9)$$

In this case, our parameter vector becomes:

$$m = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_{N_c} \end{bmatrix} \quad \text{an } N_c \times 1 \text{ vector} \quad (10)$$

Similarly, the coefficient matrix becomes: $X = \begin{bmatrix} x_{11} & x_{21} & \dots & x_{N_c 1} \\ x_{12} & x_{22} & \dots & x_{N_c 2} \\ \vdots & \vdots & \dots & \vdots \\ x_{1 N_p} & x_{2 N_p} & \dots & x_{N_c N_p} \end{bmatrix}$ $N_c \times N_p$ matrix (11)

Now, we may define the Performance Index as:

$$J = \sum_{i=1}^{N_p} (y_i - \tilde{y}_i)^2 + X^2 \sum_{i=1}^{N_c} (m_k)^2 \quad (12)$$

The partial derivative of J with respect to m_k is:

$$\frac{dJ}{dm_k} = 2 \sum_{i=1}^{N_p} \{ (y_i - \sum_{k=1}^{N_c} m_k x_{ki}) (-x_{ki}) \} + 2\lambda^2 m_k^2 = 0 \quad (13)$$

For the first parameter m_1 , this equation can be written as:

$$m_1 \{ \sum_{i=1}^{N_p} x_{1i}^2 + \lambda^2 \} + m_2 \{ \sum_{i=1}^{N_p} x_{1i} x_{2i} \} + \dots + m_{N_c} \{ \sum_{i=1}^{N_p} x_{1i} x_{N_{ci}} \} = \sum_{i=1}^{N_p} (y_i x_{1i}) \quad (14)$$

All these equations can be written in the matrix form as:

$$[X^T X + \lambda^2 I] = X^T Y \text{ OR } m = [X^T X + \lambda^2 I]^{-1} X^T Y \quad (15)$$

2.1 Step Response Module

Suppose Δm_1 is the step change in the input at $t=0$ and y_1 is the response at $t=T_s$

Then,

$$\text{At } t=0 \quad \tilde{y}_0 = 0$$

$$\text{At } t=T_s \quad \tilde{y}_1 = b_1 \Delta m_1$$

$$\text{At } t=2T_s \quad \tilde{y}_2 = b_2 \Delta m_1 + b_1 \Delta m_2$$

$$\text{At } t=3T_s \quad \tilde{y}_3 = b_3 \Delta m_1 + b_2 \Delta m_2 + b_1 \Delta m_3$$

$$\text{At } t=4T_s \quad \tilde{y}_4 = b_4 \Delta m_1 + b_3 \Delta m_2 + b_2 \Delta m_3 + b_1 \Delta m_4$$

$$\text{At } t=5T_s \quad \tilde{y}_5 = b_5 \Delta m_1 + b_4 \Delta m_2 + b_3 \Delta m_3 + b_2 \Delta m_4 + b_1 \Delta m_5$$

Similarly, the output for N_p instants with N_c changes in the input variable can be written as shown below

$$\begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \vdots \\ \tilde{y}_{N_p} \end{bmatrix} = \begin{bmatrix} b_1 & 0 & 0 & \dots & 0 \\ b_2 & b_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ b_{N_p} & b_{N_p-1} & b_{N_p-2} & \dots & b_{N_p-N_c+1} \end{bmatrix} \begin{bmatrix} \Delta m_1 \\ \Delta m_2 \\ \vdots \\ \Delta m_{N_c} \end{bmatrix}$$

This can be written as:

$$\tilde{y} = B \Delta m \quad (16)$$

Where B is $N_p \times N_c$ is the coefficient matrix; \tilde{y} is $N_p \times 1$ is the output vector and Δm is $N_c \times 1$ input vector

We define a new $N_p \times N_c$ matrix A as:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1N_c} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2N_c} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ a_{N_p 1} & a_{N_p 2} & a_{N_p 3} & \dots & a_{N_p N_c} \end{bmatrix}$$

whose elements are related to those of B as follows:

$$a_{ik} = b_{i+1-k} \text{ with } b_i = 0 \text{ for } i \leq 0$$

Then, the value of the output can be written as:

$$y_i = \sum_{k=1}^{N_c} b_{i+1-k} \Delta m_k = \sum_{k=1}^{N_c} a_{ik} \Delta m_k \quad (17)$$

If there have been N_p changes in the manipulated variable during the previous N_p steps and if no other changes were made, the output would change in future because of the old changes in the input.

$$\tilde{y}_{0L,i} = \sum_{k=0}^{-N_p+1} b_{i+1-k} (\Delta m_k)^{\text{old}} \quad (18)$$

However, at the current instant, the 0^{th} , we can measure the actual output of the process. If the model were perfect and if there were no disturbances, the actual and predicted values would be equal. The predicted value at the 0^{th} instant is given by:

$$\tilde{y}_{0L,0} = \sum_{k=0}^{-N_p+1} b_{1-k} (\Delta m_k)^{\text{old}} \quad (19)$$

The difference between y_0^m and $\tilde{y}_{0L,0}$ is added to the predicted value at the i^{th} sampling period so that the response will be improved.

$$\begin{aligned} \tilde{y}_{0L,i} &= \sum_{k=0}^{-N_p+1} b_{i+1-k} (\Delta m_k)^{\text{old}} + y_0^m - \tilde{y}_{0L,0} \\ \tilde{y}_{0L,i} &= \sum_{k=0}^{-N_p+1} b_{i+1-k} (\Delta m_k)^{\text{old}} + y_0^m - \sum_{k=0}^{-N_p+1} b_{1-k} (\Delta m_k)^{\text{old}} \end{aligned} \quad (20)$$

The last expression is used in the DMC calculations where the closed loop response will be given by:

$$y_{CL,i} = y_{0L,i} + \sum_{k=1}^{N_c} a_{ik} (\Delta m_k)^{\text{new}} \quad (21)$$

Thus the DMC algorithm finds the best values for the manipulated variables, $(\Delta m)^{\text{new}}$ by minimizing a performance index J. By least squares formulation, we can write:

$$(\Delta m)^{\text{new}} = [A^T A + \lambda^2 I]^{-1} A^T Y$$

We then proceed as follows

1-Calculate the N_p values of $y_{0L,i}$ from the past values of the manipulated variables and the present measured value of the controlled variable Y^m .

2-Calculate the N_c values of the future changes in the manipulated variables from the above equation.

3- Implement the first change Δm_i .

4- Measure the controlled variable Y^m at the next instant and repeat

We can now represent the vehicle modeling equations in state space format based on the Will & Zak Model

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} -\frac{2Caf+2Car}{mvx} & 0 & -vx - \frac{2Caf l_1 - 2Car l_2}{mvx} & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{2l_1 C\alpha - 2l_2 Car}{lvvx} & 0 & -\frac{2l_1 * l_1 C\alpha f + 2l_2 * l_2 Car}{lvvx} & 0 \\ -1 & -vx & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} \frac{2Caf}{m} & \frac{2Car}{m} \\ 0 & 0 \\ \frac{2l_1 C\alpha f}{lv} & -\frac{2l_2 Car}{lv} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta f \\ \delta r \end{bmatrix} \quad (22)$$

Based on the above equations we can simulate lane contention for a vehicle using SCILAB

$$\dot{x}_1 = -v_x x_3 + \frac{1}{m}(F_{yf} + F_{yr}); \dot{x}_2 = x_3; \dot{x}_3 = \frac{2}{I}(l_1 F_{yf} - l_2 F_{yr}); \dot{x}_4 = -v_x x_2 - x_1 \quad (23)$$

Where, δ = Steering angle of front wheel; θ = Heading angle; $F_{yf/r}$ = Lateral force on front tire/rear tire; m = mass of vehicle = 1280 kg; I = 2500 kg.m²; I = Moment of Inertia of vehicle = 2500 kg.m²; l_1 & l_2 = distance from centre of gravity to front axle & rear axle = 1.2m and 1.22m respectively.

3. Results on SCILAB and Simulation of System using Java Applets

Using the data available and equations from the SAE we demonstrate our system behavior using SCILAB [2] where we train the car to initially understand and create a reference based on step input seen in figure 2.

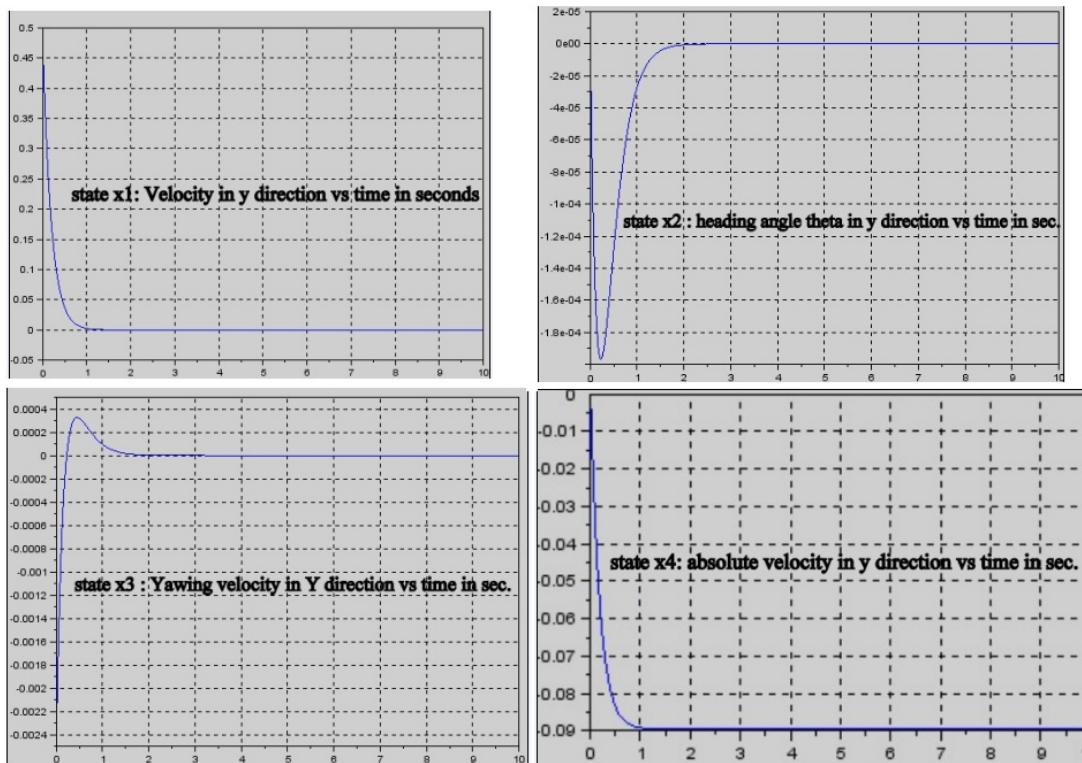


Fig.2- x1: velocity in y direction; x2: heading angle θ ; x3: yawing velocity; x4: absolute velocity in y-direction

In figure 3 each car is associated with the following state vectors- position (x, y), velocity (v), acceleration (a), and current lane (l). It will update its information based on current speed & heading of its neighbour which will be a random, stochastic and indeterministic function of time and thus do a lane contention. It will then use this information and based on its own safety & control calculations make a decision whether to continue in the same lane or undertake an overtaking manoeuvre as shown in 3(a-c) below.



Fig3a- Position 1 of cars with random velocity



Fig.3b- Position 2 of cars

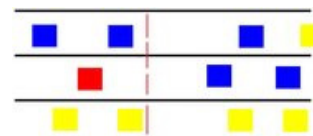


Fig.3c- Previous and next frame shown above

In terms of static obstacle avoidance the system under target can behave as shown in fig.4a below. This is extended for dynamic obstacles using the car where the processing is done taking into instant each prediction horizon t and control horizon using a LiDAR[3] and proximity sensors mounted on our test car fitted with LabVIEW [4] and Ackerman steering shown in fig (4b-c) which does mapping of the target environment to decide the best lane to be in which is reported in our prior work of vehicular localisation [5-6]

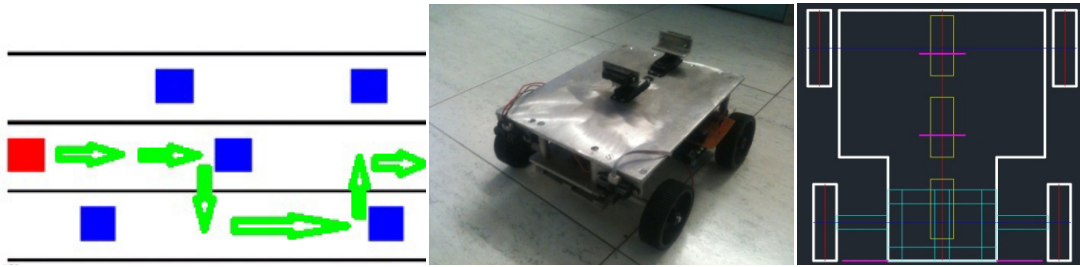


Fig. 4a- static obstacle avoidance manoeuvre Fig.4b- Test car in Lab Fig.4c. CAD layout

4. Conclusions

We have introduced our work with stress on need of more sensing requirements for robust autonomous self-driven and self-aware vehicles. With this, we introduce the concept of lane contention and the need for mathematical modelling based on the reason that software computation being faster can reduce the constraints on hardware. The modelling is used to train the system to initially learn the behaviour of the car from prior references and, with change in the dynamics of the vehicle optimize itself to provide same performance irrespective of hardware weaning thus introducing optimization at run time. We have then validated our work with control of a formation of vehicles using simulation based on results obtained from Scilab computation.

5. References

- [1] B.WayneBequette, " Non-Linear predictive Control using Multi-rate Sampling", The Canadian Journal of Chemical Engineering Vol.69, Issue 1,pg 136-143 Feb 1991
- [2] www.scilab.org
- [3] M.; Sakai, U.; Ogata, J., "Real-time Pedestrian Detection Using LIDAR and Convolutional Neural Networks," Intelligent Vehicles Symposium, 2006 IEEE , vol., no., pp.213,218, 0-0 0
- [4] www.ni.com/labview
- [5] Parulekar, M.; Shroff, D.; Padte, V.; Nangalia, H.; Metawala, A., "Vehicular localization and Intelligent Transportation Systems," 12th International Conference on Hybrid Intelligent Systems (HIS), pp.306,311
- [6] M. Parulekar et al, "QoS Improvement Using NLMPC for Congestion Control and Co-operative Information Processing," ICIP 2012. ISBN 978-3-642-31686-9