CRASH PREDICTION AND COLLISION AVOIDANCE USING HIDDEN MARKOV MODEL

A Thesis

Submitted to the Faculty

of

Purdue University

by

Avinash Prabu

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

August 2019

Purdue University

Indianapolis, Indiana

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF THESIS APPROVAL

Dr. Lingxi Li, Chair

Department of Electrical and Computer Engineering

Dr. Yaobin Chen

Department of Electrical and Computer Engineering

Dr. Brian King

Department of Electrical and Computer Engineering

Approved by:

Dr. Brian King

Head of Graduate Program

Dedicated to my family, friends and my Professors, for their support and encouragement.

ACKNOWLEDGMENTS

Firstly, I would like to thank Dr. Lingxi Li for his continuous support and encouragement. He has been a wonderful advisor and an exceptional mentor, without whom, my academic journey will not be where it is now. I would also like to to thank Dr. Yaobin Chen, whose course on Control theory has had a great impact in shaping this thesis. I would like to extend my appreciation to Dr. Brian King, who has been a great resource of guidance throughout the course of my Master's degree. A special thanks to Dr. Renran Tian, who is my principle investigator at TASI, IUPUI, for his guidance and encouragement. I also wish to thank Sherrie Tucker, who has always been available to answer my questions, despite her busy schedule.

I would like to acknowledge my parents, Mr. & Mrs. Prabhu and my sister Chathurma for their continuous belief and endless love towards me. I also acknowledge my TASI team members and friends for their support.

TABLE OF CONTENTS

Page

LIST OF TABLES . . . vii

LIST OF FIGURES . . . viii

ABSTRACT . . . x

1 INTRODUCTION . . . 1

1.1 Literature Review . . . 2

1.2 Thesis Contributions . . . 5

1.3 Thesis Organization . . . 5

2 MARKOV CHAINS . . . 7

2.1 Description of a Markov Chain . . . 7

2.1.1 Model Specifications of a Markov Chain . . . 8

2.1.2 States, Transition Matrix and Initial Probability Distribution . . . 8

2.2 Types of Markov Chains . . . 14

2.2.1 Absorbing Markov Chains . . . 15

2.2.2 Ergodic and Regular Markov Chains . . . 17

3 HIDDEN MARKOV MODEL IMPLEMENTATION IN ACTIVE SAFETY . . . 23

3.1 Implementation and Analysis of HMM . . . 23

3.1.1 Elements of HMM . . . 26

3.1.2 HMM Explained Through a Simple Coin Toss Example . . . 27

3.2 Design of Active Safety System . . . 29

3.2.1 Markov Chain - Model Specification . . . 29

3.2.2 Model Design . . . 32

3.2.3 Description of Variables and Formulae Used . . . 33

3.3 Crash Prediction and Avoidance . . . 37

3.3.1 Flow-1 - Calculation for Probable Time of Crash . . . 37

Page

3.3.2 Flow 2 - Calculation of Probable Crash Probability 40
3.3.3 Flow 3 - Determining the Appropriate Active Safety System . . 42
3.4 Adaptive Cruise Control 43
3.5 Lane Keep Assist 44
4 SIMULATION AND RESULTS 46
4.1 Description of Data Obtained 47
4.2 Simulation 48
4.2.1 Determination of State Transition Matrix 49
4.2.2 Adaptive Cruise Control Simulation 56
4.3 Scenarios and Simulation Result 57
4.3.1 Scenario 1: Unsafe Operation with ACC 59
4.3.2 Scenario 2: Unsafe Operation with Lane Departure and Steering Assist 62
4.3.3 Scenario 3: Safe Operation 63
5 CONCLUSION AND FUTURE WORK 66
5.1 Conclusion 66
5.2 Future Work 67
REFERENCES 68

LIST OF TABLES

Table | Page

3.1 Observation Sequence of 1-coin Model . . . 24

3.2 Observation Sequence of 2-coin Model . . . 25

3.3 Observation Sequence of 3-coin Model . . . 25

3.4 Observation Probabilities of 3-coin Model . . . 26

3.5 First Layer . . . 30

3.6 Second Layer . . . 30

3.7 Observation Probabilities for Each Lane . . . 31

4.1 Speed Range Mapping . . . 54

LIST OF FIGURES

Figure Page

2.1 Random Walk of a Gentleman 11
2.2 Transition Graph of Random Walk 12
2.3 Transition Graph for a Markov Chain with Three States 13
2.4 Cat's Random Walk 15
2.5 Markov Chain of a Cat's Random Walk 16
3.1 1-coin Model 24
3.2 2-coin Model 25
3.3 3-coin Model 26
3.4 HMM Model 28
3.5 Road Layout 32
3.6 Lane Change Markov Chain 33
3.7 Speed Change Markov Chain 33
3.8 Lane Change Data 34
3.9 Probable Time of Crash 38
3.10 Lidar Principle 40
3.11 Probable Crash Probability 41
3.12 Flow 3 42
3.13 Distance Calculation 43
3.14 Adaptive Cruise Control 44
3.15 Lane Keep Assist 45
4.1 Data Obtained from USDOT 46
4.2 Filtered Data Used for Simulation 47
4.3 Instrumented Vehicle 47
4.4 Lane Change State Transition Matrix Flow 49

Figure		Page
4.5	Probability(Lane Change)	50
4.6	State Transition Matrix of Car 1	51
4.7	State Transition Matrix of Car 2	51
4.8	State Transition graph of Car 1	52
4.9	State Transition Graph of Car 2	52
4.10	Speed Change State Transition Matrix Flow	53
4.11	Speed Change State Transition Matrix of Car 1	55
4.12	Speed Change State Transition Matrix of Car 2	55
4.13	W, Z and M for Car 1	56
4.14	W, Z and M for Car 2	56
4.15	Adaptive Cruise Control	58
4.16	Plant Design for Cars	59
4.17	Lane probabilities of Car 1 at probable time of crash	60
4.18	Lane Probabilities of Car 2 at Probable Time of Crash	60
4.19	Crash Probability at Probable Time of Crash	60
4.20	Simulation of ACC	61
4.21	Lane Probabilities of Car 1 at Probable Time of Crash	62
4.22	Lane Probabilities of Car 2 at Probable Time of Crash	63
4.23	Crash Probability at Probable Time of Crash	63
4.24	Lane Probabilities of Car 1 at Probable Time of Crash	64
4.25	Lane Probabilities of Car 2 at Probable Time of Crash	64
4.26	Crash Probability at Probable Time of Crash	64

ABSTRACT

Prabu, Avinash. M.S.E.C.E., Purdue University, August 2019. Crash Prediction and Collision Avoidance using Hidden Markov Model. Major Professor: Lingxi Li.

Automotive technology has grown from strength to strength in the recent years. The main focus of research in the near past and the immediate future are autonomous vehicles. Autonomous vehicles range from level 1 to level 5, depending on the percentage of machine intervention while driving. To make a smooth transition from human driving and machine intervention, the prediction of human driving behavior is critical. This thesis is a subset of driving behavior prediction. The objective of this thesis is to predict the possibility of crash and implement an appropriate active safety system to prevent the same. The prediction of crash requires data of transition between lanes, and speed ranges. This is achieved through a variation of hidden Markov model. With the crash prediction and analysis of the Markov models, the required ADAS system is activated. The above concept is divided into sections and an algorithm was developed. The algorithm is then scripted into MATLAB for simulation. The results of the simulation is recorded and analyzed to prove the idea.

1. INTRODUCTION

The number of cars have been increasing exponentially every year. This implies the need for Advanced Driver Assisted Systems more than ever. It is the partial duty of the manufacturers to ensure that the occupants of the vehicle are safe and at the same time comply with the safety and driving regulations of the road.

Advanced Driver assisted systems are human-machine interface, that increase car and road safety. These systems help in reducing the human error and thereby reducing the probability of a crash. ADAS systems can be classified into active and passive systems. A passive system alerts the driver when there is a possibility of a dangerous situation. An example of a passive systems is the lane departure warning alert. On the other hand, active systems take the required action when a dangerous situation is met. Steering assist with land departure warning system is an example of an Active safety system. Adaptive cruise control, automatic emergency braking, parking assist, blind spot etc. are some examples of ADAS.

In recent times, driving behavior and driver characteristics have been used in automotive controls to assist the active safety systems. Study of driving behavior has been one of the main focus of research in the automotive industry. Some researchers have also used driving behavior profile as a composite measure of the risk of casualty crash. Studies have shown that driving behavior has been a contributing factor in 90 percent of road crashes. As a result of which, there is a definite benefit in monitoring the driving behavior and integrating it into active safety systems.

This thesis is focused on using Markov Model to predict the driving behavior and predicting the probability of occurrence of a collision thereby taking action to prevent the same. Markov Model is a stochastic process, which is used to determine the future states of a randomly changing systems. The future states depend only on the current state of the chain.

1.1 Literature Review

The authors in [1] suggest a method for assessing the safety of planned trajectories in autonomous vehicles. The future position of the trajectory is computed based on dynamic models. The dynamics of the participants on the road is considered for prediction of driving behavior. The proposed algorithm uses the dynamics of the vehicle and the dynamics of the participating vehicles on the road. In this paper, the author expects that the vehicle is instrumented sensors that perceive the environment such that the geometry of surrounding, position of obstacles and the dynamics of road sections are all available. The probability distribution of trajectory in accordance with the other vehicles on the road is calculated, thereby reaching a prediction framework for safety assessment in different traffic situations.

A wide range of research has been carried out to identify the driving intentions of a driver. In [2] the author uses an artificial neural network approach to predict the maneuvering intentions of a driver to improve the active safety features of the car. The researchers establish a Hidden Markov Model, which acts as a Bayesian network with two concurrent stochastic process. The states in the Markov model is used to describe the transfer of driving intentions. The stochastic process is used to describe the maneuvering behaviors of the driver. The ANN is then used to learn the driving conditions and the rules. The ANN and HMM work together to make the prediction of driving behavior accurate.

In [3] the Markov Models are used in estimation of traffic density in multi-lane roadways, taking lane change into account. Even though this paper focuses on different goals, it is useful to extract information on the lane change criteria, which is discussed. The authors use each lane as a state in the Markov Chain, which is a very efficient suggestion. The only drawback in this model is that the authors assume that a vehicle has a certain probability to stay in the same lane. This will lead to errors in calculating the probability and thereby affecting the prediction results.

Markov Models have also found their significance in learning of driving conditions such as vehicle speed, surrounding traffic speed, road geometry etc. In [4] the author mainly concentrates on energy efficiency in HEVs and Adaptive cruise control by capturing the driving conditions using Markov Models. The author uses real-time comparison of Markov Chains using Kullback-Liebler (KL) divergence. These are used to characterize each segment of the road with unique characteristics. An on-board learning technique has been used to update the Markov Chains. In addition, it is considered that the Markov Chain representing the road characteristics and the one learned on-board are convergent.

In [5], a connected cruise control model is achieved using a probabilistic model. The instrumented vehicle receives information about vehicles in front through wireless V2V communication. The car following dynamics of the preceding vehicle is then modeled using Markov chains. The connected cruise controller is achieved using a Markov decision making process. For the car following, the author uses an optimal velocity model, in which the optimal speed function not only depends on the headway of the each car but also the headway of the preceding vehicle. It also provides a non-complex mathematical form and a good physical intuition. The desired velocity is then calculated using non-linear range policy function. It is assumed that in an array of n vehicles, the tail vehicle is equipped with the CCC and receives motion information from the others. The human car following model and the CCC are considered in the linear region in discrete time to consider sampling and zero-order hold in digital controllers. The sampled dynamics of the vehicles are then written as a Markov chain. It is then used to formulate the optimization objective accordingly. Optimized feedback gains are used to design the discretized linear CCC. Simulations were carried out for various scenarios and the outcome showed a robust design. The drawback with this design is that, the lane change has not been considered, which is essential when calculating the number of headway vehicles.

In another research carried out by Ehsan Moradi-Pari and his team [6], model based communication has been used for V2V communication, based on small (braking, accelerating etc.) and large (free following, turning etc.) scale modeling of the vehicle dynamics. These are coupled into a Markov Chain and investigated for implementing Cooperative Adaptive Cruise Control. The Markov Chain is used for achieving Cruise control goals using a predictive method. The vehicles use DSRC radio for communication, which transmits the vehicles' speed, position and heading direction over a wireless broadcast link to all the neighboring cars. The movement model is transmitted over the DSRC and the receiving vehicle uses this model to develop a situational awareness model. With this information, a stochastic hybrid model is created with a set of discrete and continuous states. It is modeled in such a way that the each state uses ARX or pice-wise polynomial to track the designated functions and create the mathematical model. The authors consider a system model of n vehicles to implement the CACC and that all the vehicles are equipped with DSRC radio and the sensors capable of calculating the relative speed, distance and velocity from neighboring vehicles in its lane. The stochastic model is then developed from the discrete time state space model of the CACC. The algorithm uses a stochastic model predictive optimization approach to evaluate the optimal speed of the vehicle. Again, the lane change factor has been neglected in this paper.

The real time problem in automated cars is that they act reactively to the cars they follow. It leads to uncomfortable and sometimes unsafe situations (like stop and go scenarios). The authors in [7] use probabilistic anticipation of motion of the preceding vehicle and control the motion of the ego vehicle as a solution to the above problem. A Markov Chain predictor on the preceding vehicle behavior with an optimized MPC is used for optimizing the motion of the ego vehicle. This system is also coupled with motion predictor based on historical traffic data at different location and time for accuracy purposes. The kinematics of the vehicle motion is modeled with a first order lag between acceleration command and actual vehicle acceleration. A state space model is created and then an MPC is designed on the state space representation. Sparse data from probe vehicles are used to predict the travel times between segments of the road. Historical data is used to train the Markov Model to predict the future position of the preceding vehicle. Combining the predictors helped in improving the accuracy of prediction. Expectation-Maximization algorithm is used to relocate the segment travel time so that the likelihood function is maximized. The predictors are combined using Gaussian Mixture Model. The analysis shows that the two predictors work well at different speeds, giving the systems a wide range of operating points.

1.2 Thesis Contributions

This thesis discusses prediction of crash using hidden Markov model and prevention of the same through the existing active safety systems. The main contributions of the thesis are itemized below,

- Designed the first layer of Markov chain for lane change calculation.
- Designed the second layer of Markov chain for speed change evaluation.
- Created an algorithm for prediction of crash using the above two layers of Markov chain.
- Designed a methodology to decide the appropriate active safety system, depending on the crash probability.
- Developed a Simulink model to simulate adaptive cruise control model.
- Created and simulated various scenarios to prove the concept.

1.3 Thesis Organization

This thesis consists of five chapters. The first chapter is an introduction to the thesis concept. This chapter consists of literature reviews from various journals, discussing current trends and inspirations in automotive safety. The second chapter explains the particulars of Markov chains and the mathematics involved in it. The different classifications of Markov chains are also discussed in this chapter. Chapter 2 also discusses a few examples pertaining to Markov chains. Chapter 3 discusses hidden Markov model and the modifications made into the model to suit the concept in this thesis. This chapter also discusses the implementation of hidden Markov model in active safety. The two layers of the Markov model and the calculation of the probabilities are also explained. The algorithm implemented in this thesis is also discussed in this chapter. Chapter 4 consists of the simulation and results of various scenarios created to prove the concept of thesis. The conclusion and possible future scope of the concept are discussed in Chapter 5.

2. MARKOV CHAINS

Markov chains are a part of probability theory, which was first proposed by and named after the Russian Mathematician, Andrey Markov in 1907. Markov chains satisfy the Markov property. A process satisfies the Markov property if the future states depend only on the current state and not on the past states, predominantly known as the memoryless property. Markov chain is a discrete random process in a stochastic model. It is an array of random events where the state dependency is only between adjacent events, like in a chain, and thus the name Markov Chains. While the states in a Markov Chain can be continuous, the time parameter is always a discrete index set. Markov Chains find their application in statistical evaluation models and control theory. A few examples are agnostic and prognostic evaluation of plants, cruise control in cars, betting, weather prediction, queues in airport, population growths and exchange rate predictions [8].

2.1 Description of a Markov Chain

Discrete time Markov chain events happen at discrete time instants, that is, the state transitions happen at integer time intervals. Consider the time instants as $0, 1, 2.....k, ...$ and the stochastic sequence as $\{X_1, X_2, ...\}$. The stochastic sequence is characterized by the Markov property:

$$P[X_{k+1} = x_{k+1} | X_k = x_k, X_{k-1} = x_{k-1}, ...X_0 = x_0] = P[X_{k+1} = x_{k+1} | X_k = x_k] \tag{2.1}$$

The current state is x_k and the next future state is x_{k+1}. It is evident from Equation 2.1 that the future state is only dependent on the current state and independent on the past states.

2.1.1 Model Specifications of a Markov Chain

Unlike other discrete time event models or stochastic times automaton, the Markov chains consider only the total probability of making a transition from one state to another. Let us consider x and x' to be the states and the total probability to go from x to x' be $p(x', x)$. The transition probabilities depend on the time instant at which it occurs. So the representation of the probability is,

$$p_k(x', x) = P[X_{k+1} = x' | X_k = x] \tag{2.2}$$

To represent a Markov Chain, we require the following,

1. State Space χ.

2. Initial state probability vector $p_0(x) = P[X_0 = x]$, $\forall x \in \chi$.

3. Probability of transition $p(x', x)$ where x' is the next state and x is the present state.

The state space χ is countable, so it is mapped to a set of countable integers (non-negative)

$$\chi = \{0, 1, 2....\}$$

2.1.2 States, Transition Matrix and Initial Probability Distribution

Let us consider the states to be $\{S_1, S_2, S_3, ..., S_k\}$. The chain starts in one of the states and traverses through the other states one by one. Each movement or transition is called a step.

State Transition Matrix

Let S_i be the current state and S_j be the next state. The state transition probability is modified as follows,

$$p_{ij}(k) = P[X_{k+1} = S_j | X_k = S_i] \tag{2.3}$$

where $S_i, S_j \in \chi$ and $k = 0, 1, 2, ...$ It can be observed that the total probability at any state S_i is 1.

$$\sum_{all j} p_{ij} = 1 \tag{2.4}$$

The process can stay in the same state i in the next step, which is denoted by p_{ii}.

The transition probabilities in a model are conveniently represented in matrix form. Let $\mathbf{P}$ be the state transition matrix and p_{ij} be the transition probabilities.

$$\mathbf{P} = [p_{ij}] \qquad i, j = 0, 1, 2, ... \tag{2.5}$$

Since the summation of all the probabilities that go out from any state is 1, the sum of all the elements in ith row of $\mathbf{P}$ matrix is 1, where $i = 0, 1, 2, ...$ It is to be noted that $\mathbf{P} = [p_{ij}(1)]$ in Equation 2.5.

Theorem 2.1.1 *Let P be the state transition matrix of a Markov Chain. The ij^{th} entry of $P_{ij}^{(n)}$ of matrix P^n gives the probability that the chain starts at state S_i and ends at state S_j after n states.*

Initial probability distribution

One of the main attributes of a Markov Chain is determining the probabilities of the chain being at a particular state at a specific time instant. These are called state probabilities and are defined as follows,

$$\pi_j(k) \equiv P[X_k = j] \tag{2.6}$$

So, the state probability vector is defined as,

$$\pi(k) = [\pi_0(k), \pi_1(k), ...] \tag{2.7}$$

The probability of the chain starting at a particular state is specified by the initial probability vector. It is defined as follows,

$$\pi(0) = [\pi_0(0), \pi_1(0), ...] \tag{2.8}$$

The initial probability vector provides the probability distribution of the initial state. A probability vector with k components is a row vector whose entries are non-negative values and also sum to 1. The ith entry of $\pi(0)$ $(1 \leq i \leq k)$ represents the probability that the chain starts in state S_i.

Theorem 2.1.2 *If $\boldsymbol{P}$ is the state transition matrix and $\pi(0)$ is the initial probability vector, the probability that the chain is in state S_i after n steps is the i^{th} entry of the vector,*

$$\pi(n) = \pi(0)P^n \tag{2.9}$$

Random walk example: Setting up a Markov Chain

Assuming we have a gentleman walking around four street corners, c_1, c_2, c_3 and c_4, in a random manner. At t=0, he start his walk at $c1$. At t=1, he flips a fair coin and moves either to c_2 or c_4, depending on the result of the coin toss. He continues to do this at every corner. Assuming that he does not move diagonally and that he moves clockwise is heads comes up and moves counter-clockwise if tails comes up, as shown in Figure 2.1

Assuming that the gentleman starts at c_1, we have,

$$P(X_0 = 1) = 1$$

In the next step, he will move to c_2 or c_4 with equal probability. Which leads to,

$$P(X_1 = 2) = 1/2$$
$$P(X_1 = 4) = 1/2$$

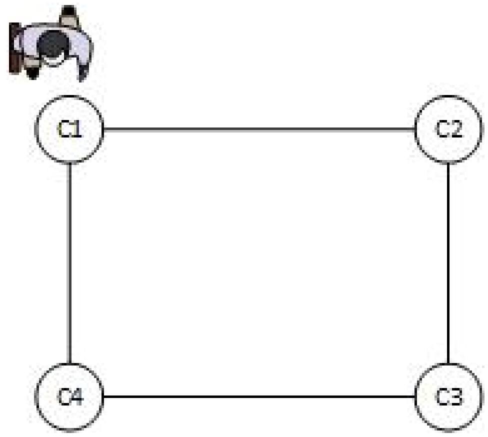

Fig. 2.1.: Random Walk of a Gentleman

At step n, the probability that the chain will go to any adjacent corners at step $n+1$ is $1/2$. This leads us to the state transition matrix P, which is,

$$P = \begin{bmatrix} 0 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 \\ 0 & 1/2 & 0 & 1/2 \\ 1/2 & 0 & 1/2 & 0 \end{bmatrix}$$

It can be noted that the matrix satisfies the below two conditions,

- $P_{ij} \geq 0$ for all $i, j = \{1, 2, 3, 4\}$
- $\sum_{j=1}^{4} P_{i,j} = 1$ for all $i = \{1, 2, 3, 4\}$

The above equations mean that the conditional probabilities are non-negative and sum up to 1.

In the above example, the gentleman starts at state $c1$. So the initial probability distribution is,

$$\pi(0) = \{1, 0, 0, 0\}$$

All the basic elements of a Markov Chain systems are obtained. With this information, we will be able to picture the Markov Chain using a transition graph. A

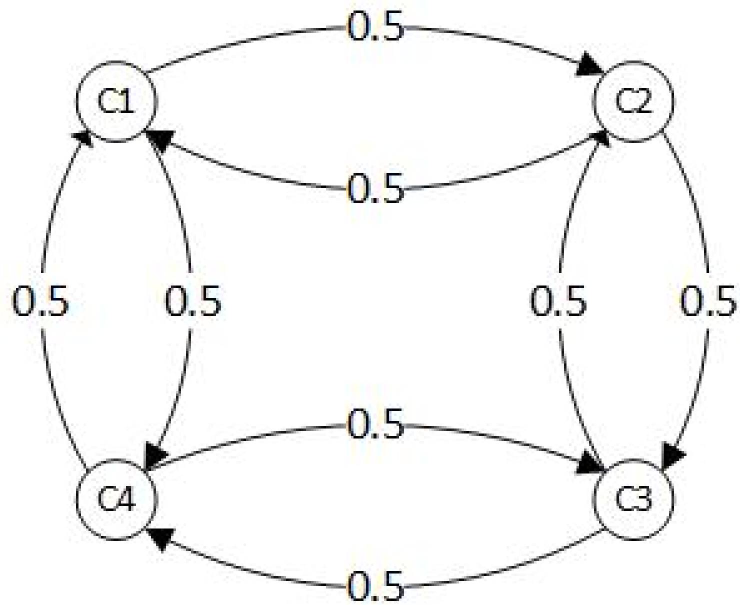

Fig. 2.2.: Transition Graph of Random Walk

transition graph has nodes, representing the states, and arrows which represent the transitions from one node to another. The transition probabilities are denoted above the arrows. Figure 2.2 shows the transition graph of the above example.

Example: Analysis of Markov Chain

Considering a Markov chain with 3 states, $\{S_1, S_2, S_3\}$. Assuming the chain does not stay in state S_2 for two consecutive steps, and the next step will be S_1 or S_2 in an equally likely manner. S_1 and S_2 have 50% probability to be in the same state at the next step and a 25% probability to move to S_3 in the next step. The chain for the model described above is depicted in Figure 2.3

The transition probabilities can be represented in matrix form as shown below,

$$P = \begin{bmatrix} 0.5 & 0.25 & 0.25 \\ 0.5 & 0 & 0.5 \\ 0.25 & 0.25 & 0.5 \end{bmatrix}$$

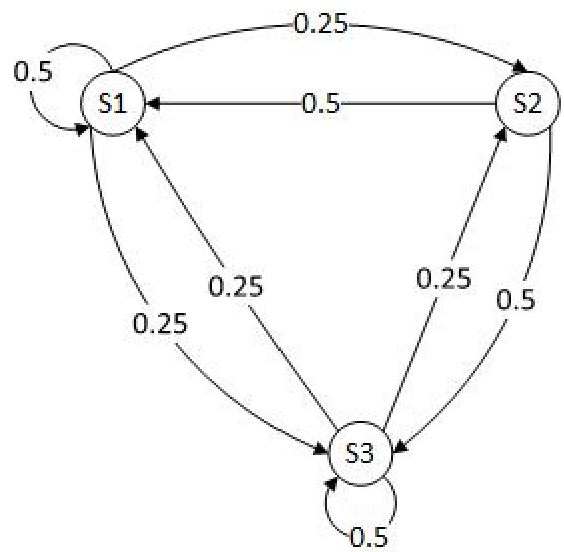

Fig. 2.3.: Transition Graph for a Markov Chain with Three States

Let us consider the problem of determining the probability of the chain being in a particular state two steps from now, given the current state. For instance, if the chain is in state $S1$ now, what is the probability that the chain is at state $S2$, two steps from now?

There are two ways to answer this problem. One way is to analyze the transition graph and sum up all the probabilities of possible events. Here, in this example, there are three such possibilities.

1. $S1 \to S1 \to S3$
2. $S1 \to S2 \to S3$
3. $S1 \to S3 \to S3$

The total probability of this transition is,

$$P_{13}^{(2)} = P_{11}P_{13} + P_{12}P_{23} + P_{13}P_{33} = (0.5)(0.25) + (0.25)(0.5) + (0.25)(0.5) = 0.375$$

The second way is by solving through the state transition matrix. This is explained by theorem 2.1.1. The first row, third element of matrix P^2 will give us the total probability of the required sequence.

$$P^2 = \begin{bmatrix} 0.4375 & 0.1875 & 0.375 \\ 0.375 & 0.25 & 0.375 \\ 0.375 & 0.1875 & 0.4375 \end{bmatrix}$$

$P_{13} = 0.375$ is the total probability that the chain will start in state $S1$ and end at state $S3$ in two steps.

Given the initial probability vector $\pi(0)$ and the state transition matrix P, we will be able to study the long term behavior of the given system. In the example above, let us assume there is equal chance of the chain starting at all the states. This helps in evaluating the initial probability vector, which is,

$$\pi(0) = \{1/3, 1/3, 1/3\}$$

From theorem 2.1.2, we can evaluate the probability of being at each state at step n. For example, if we need the probability of reaching the states at the third state, the calculation is as follows.

$$\pi(3) = \pi(0) * P^3 = \begin{bmatrix} 1/3 & 1/3 & 1/3 \end{bmatrix} \begin{bmatrix} 0.4063 & 0.2031 & 0.3906 \\ 0.4063 & 0.1875 & 0.4063 \\ 0.3906 & 0.2031 & 0.4063 \end{bmatrix}$$

$$= \begin{bmatrix} 0.4010 & 0.1979 & 0.4010 \end{bmatrix}$$

The probability of the chain reaching state $S1, S2$ and $S3$ is 0.401, 0.1979 and 0.4010 respectively.

2.2 Types of Markov Chains

Markov chains are classified into different types depending on the characteristic of their states. Three main types of Markov Chains are discussed in this section.

2.2.1 Absorbing Markov Chains

Definition 2.2.1 *A state i of a Markov Chain is said to be absorbing if it is impossible leave this state.*

This means that state i is absorbing, if and only if,

$$p_{ii} = 1 \tag{2.10}$$

$$p_{ij} = 0 \; for \; i \neq j \tag{2.11}$$

A Markov chain is absorbing only if at least one of its state is absorbing and it is possible to reach an absorbing state from every state, not necessarily in one step.

Definition 2.2.2 *In an absorbing Markov chain, the states which are not absorbing are called transient states.*

Example of an Absorbing Markov Chain: Cat's random walk

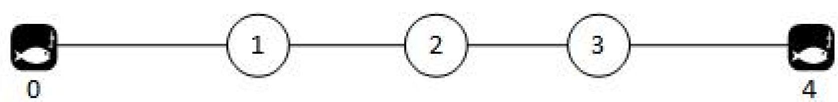

Fig. 2.4.: Cat's Random Walk

A cat walks along a street with four blocks (3 corners: 1, 2 and 3) and two fish markets at either ends (0 and 4) as shown in Figure 2.4. The cat follows a particular probability at each corner of the street. At corners 1, 2 and 3, it walks to the left or the right with equal probability. When it reaches either of the two fish markets, it stays there. The Markov Chain has five states {0, 1, 2, 3, 4}. The Markov chain is depicted in Figure 2.5

It is clear from the model that states 0 and 4 are absorbing. It is impossible to leave these states. States 1, 2 and 3 are transient states. The Chain is an absorbing Markov chain because it is possible to reach an absorbing state from every state.

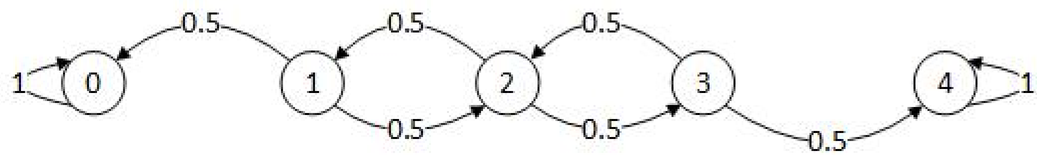

Fig. 2.5.: Markov Chain of a Cat's Random Walk

Given an absorbing Markov chain with k states, with a absorbing states and t transient states, the canonical form of the state transition matrix is obtained by rearranging the matrix such that the transient states form the first t rows and columns and the next a rows and columns are occupied by the absorbing states. The state transition matrix takes the following form,

$$P = \begin{bmatrix} Q & R \\ 0 & I \end{bmatrix} where\, Q \in \mathbb{R}^{t \times t},\ R \in \mathbb{R}^{t \times r}, I \in \mathbb{I}^{r \times r} \tag{2.12}$$

Now P is represented in canonical form and is an upper triangular matrix.

Theorem 2.2.1 *In an absorbing Markov Chain, the probability of ending up in an absorbing state is 1 i.e when $n \to \infty$, $Q^n \to \infty$*

Definition 2.2.3 *For an absorbing Markov Chain, the fundamental matrix is given by $N = (I - Q)^{-1}$*

The fundamental matrix helps in obtaining the expected number of times the chain visits a particular state. It also helps in obtaining the expected time to reach an absorbed state. This is explained in a better way with the help of the following theorems.

Theorem 2.2.2 *The ij^{th} entry of the fundamental matrix, N, gives the expected number of times the chain visits the transient state j if it starts at state i*

Theorem 2.2.3 *If a Markov chain starts in state i, and t is a column vector such that,*

$$t = NC = (I - Q)^{-1}C \tag{2.13}$$

where C is a column vector with all entries 1, the ith entry of t, i.e. $t(i)$, is the expected number of steps before the chain is absorbed, given that the chain starts in state i.

2.2.2 Ergodic and Regular Markov Chains

Definition 2.2.4 *An Ergodic markov chain in one in which it is possible to go from every state to all the other states, not necessarily in one step.*

Definition 2.2.5 *A regular Markov chain is one in which for some power of the state transition matrix, it has only positive elements.*

It can be inferred from the above definitions that not all ergodic Markov chains are regular but all regular Markov Chains are ergodic. So regular Markov Chains are a subset of ergodic Markov chains.

There are two important theorems that needs to be addressed, which can be observed in regular Markov chains.

Theorem 2.2.4 *In a regular Markov chain with a state transition matrix, P, as $n \to \infty$, P^n approaches a limiting matrix W, which has all rows as the same vector w. The vector w is a probability vector with all elements, strictly positive.*

$$\lim_{n\to\infty} P^n = W \tag{2.14}$$

Theorem 2.2.5 *If P is the state transition matrix of a regular Markov chain such that,*

$$\lim_{n\to\infty} P^n = W$$

and the limiting matrix, W has all its rows as the probability vector , w, then we have,

$$wP = w \tag{2.15}$$

and any vector v that satisfies, $vP = v$ is a constant multiple of w.

This means that the limiting probability vector is unique. The limiting probability vector, w, can be calculated using the following two equations.

1. $wP = w$.

2. $\sum_1^k w = 1$ where k is the total number of states.

By solving the above two equations for the unknown variables, w can be obtained. If we start a Markov chain with arbitrary initial probabilities, after n steps, $n \to \infty$, the probability of ending at state j is $w(i)$, which is the ith entry of w.

Theorems 2.2.4 and 2.2.5 hold good for ergodic Markov chains too. Since regular Markov chains are a subset of the ergodic Markov chains, all the properties of ergodic Markov chain also hold good for regular Markov chains.

The following two quantities can be obtained from an ergodic Markov chain.

1. The mean time to return to a state.

2. The mean time to go from one state to another.

It is to be noted that the mean time gives only the approximate time that a chain will take to reach a state. These two quantities prove to be very efficient in analyzing a model's behavior.

Mean Recurrence Time

Assuming the chain starts at state S_i, the expected number of steps to return back to state S_i, for the first time is called the mean recurrence time for state S_i. The mean recurrence time is denoted by r_i.

Theorem 2.2.6 *For an ergodic Markov chain, the mean recurrence time for state S_i is given by,*

$$r_i = 1/w_i \tag{2.16}$$

where, w_i is the ith component of the limiting probability vector, w.

Consider the example below with four states. Assuming the limiting probability vector is,

$$w = \begin{bmatrix} 1/12 & 1/4 & 1/3 & 1/3 \end{bmatrix}$$

Then, the mean recurrence vector becomes,

$$r = \begin{bmatrix} 12 & 4 & 3 & 3 \end{bmatrix}$$

For state 1, it takes an average of 12 steps to return back to state 1. The mean recurrence matrix, D, is the matrix with its diagonal entries as r_i and all the other entries as 0.

Fundamental Matrix of an Ergodic Markov Chain

Let P be the state transition matrix of an ergodic Markov chain. Let W be the limiting matrix and I be the identity matrix. The fundamental matrix is then given by,

$$Z = (I - P + W)^{-1} \tag{2.17}$$

It can also be proven that the matrix $(I - P + W)$ has an inverse.

Let x be a column vector such that,

$$(I - P + W)x = 0$$

Pre-multiplying the above equation with w,

$$(wI - wP + wW)x = 0 \tag{2.18}$$

We know that, $wP = w$ and $wW = w$. So 2.18 becomes,

$$wx = 0$$

Since w is a limiting probability vector with all its elements strictly positive, we can conclude that $x = 0$. This implies, $(I - P + W)$ has an inverse and thus, the fundamental matrix of an ergodic markov chain is,

$$Z = (I - P + W)^{-1}$$

Mean First Passage Matrix

Definition 2.2.6 *If an ergodic Markov chain starts in state S_i, the expected number of steps to reach state S_j, for the first time is defined as the mean first passage time from S_i to S_j.*

The mean first passage matrix is denoted by M, with all its diagonal entries as 0 and it's non-diagonal entries as m_{ij}. m_{ij} is the mean first passage time from state i to j.

Theorem 2.2.7 *The mean first passage matrix, M, for an ergodic Markov chain is obtained from the fundamental matrix, $Z = (I - P + W)^{-1}$ and the limiting probability vector w.*

$$m_{ij} = \frac{(Z_{jj} - Z_{ij})}{w_j} \tag{2.19}$$

As defined earlier, the diagonal entries of M are 0 and the non-diagonal entries are the mean first recurrence times, m_{ij}.

Example

We can use the second example presented in 2.1.2 with three states. Recall that the state transition matrix is given by,

$$P = \begin{bmatrix} 0.5 & 0.25 & 0.25 \\ 0.5 & 0 & 0.5 \\ 0.25 & 0.25 & 0.5 \end{bmatrix}$$

Let the limiting probability vector w be, $w = \begin{bmatrix} w_1 & w_2 & w_3 \end{bmatrix}$

We have to solve for $wP = w$ and $\sum w_i = 1$. This will give us the following equations,

$$0.5w_1 + 0.5w_2 + 0.25w_3 = w1 \tag{2.20}$$

$$0.25w_1 + 0.25w_3 = w2 \tag{2.21}$$

$$0.25w_1 + 0.5w_2 + 0.5w_3 = w3 \tag{2.22}$$

$$w_1 + w_2 + w_3 = 1 \tag{2.23}$$

Solving the above equations, we obtain,

$$w1 = 0.4$$

$$w2 = 0.2$$

$$w3 = 0.4$$

Now we can form the limiting matrix, which is given by,

$$W = \begin{bmatrix} 0.4 & 0.2 & 0.4 \\ 0.4 & 0.2 & 0.4 \\ 0.4 & 0.2 & 0.4 \end{bmatrix}$$

$$I - P + W = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.5 & 0.25 & 0.25 \\ 0.5 & 0 & 0.5 \\ 0.25 & 0.25 & 0.5 \end{bmatrix} + \begin{bmatrix} 0.4 & 0.2 & 0.4 \\ 0.4 & 0.2 & 0.4 \\ 0.4 & 0.2 & 0.4 \end{bmatrix}$$

$$I - P + W = \begin{bmatrix} 0.9 & -0.05 & 0.15 \\ -0.1 & 1.2 & -0.1 \\ 0.15 & -0.05 & 0.9 \end{bmatrix}$$

$$Z = (I - P + W)^{-1} = \begin{bmatrix} 86/75 & 1/25 & -14/75 \\ 2/25 & 21/25 & 2/25 \\ -14/25 & 1/25 & 86/75 \end{bmatrix}$$

The mean first passage time for the above system is given by,

$$m_{ij} = \frac{(Z_{jj} - Z_{ij})}{w_j}$$

So,

$$\begin{aligned}
m_{12} &= \frac{(Z_{22} - Z_{12})}{w_2} = \frac{21/25 - 1/25}{1/5} = 4 \\
m_{13} &= \frac{10}{3} \\
m_{21} &= \frac{8}{3} \\
m_{23} &= \frac{8}{3} \\
m_{31} &= \frac{10}{3} \\
m_{32} &= 4
\end{aligned}$$

The mean first passage matrix is given by,

$$M = \begin{bmatrix} 0 & 4 & 10/3 \\ 8/3 & 0 & 8/3 \\ 10/3 & 4 & 0 \end{bmatrix}$$

3. HIDDEN MARKOV MODEL IMPLEMENTATION IN ACTIVE SAFETY

In Chapter 2, the formation and analysis of a Markov model was discussed. An extension to Markov chains is a complex stochastic process, known as the hidden Markov model (HMM). Hidden Markov model is a statistical model, in which the system to be designed follows the Markov property, or otherwise the memoryless property. In Markov chain, the state is directly visible to the observer, whereas in Hidden Markov Model, the states are not directly visible but the observation sequence, or output, can be observed. The output of the model is directly dependent on the states and thus a pattern for sequence of states can be observed. This is otherwise called as the pattern theory. The term hidden refers to the unknown states in the model. In most practical applications, the states are usually known. The model is still called a hidden Markov process, even if the states are known [9].

3.1 Implementation and Analysis of HMM

A coin toss experiment model is presented to grasp the understanding of a HMM. Assuming that a coin (or multiple coin) toss experiment is being performed with a barrier such that only the result (Heads or Tails) of the experiment is known to the observer. The process of the experiment is also unknown, that is information about which coin produces a corresponding result is unknown. Given the result of the above experiment, the first step is determining the observed and the hidden states. It is clear that the hidden states are the number of coins and the observation sequence consists of heads or tails. In this experiment heads will be denoted by H and tails will be denoted by T. To model the Markov model, the best choice is to assume the number of states [10].

Let us assume that only one coin is tossed. This makes it an observable Markov chain with only one state. It becomes a degenerate hidden Markov model. Here each state will correspond to the result of the experiment, that is H or T. This model is depicted in Figure 3.1. The observation sequence of the model is observed in Figure 3.1.

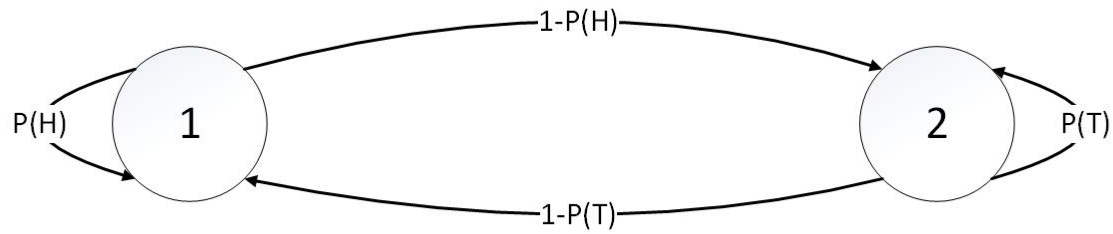

Fig. 3.1.: 1-coin Model

Table 3.1.: Observation Sequence of 1-coin Model

Observation	H	H	T	T	H	T	H	H	T	T	H
Sequence	1	1	2	2	1	2	1	1	2	2	1

Now, let us assume two coins are tossed. Here, there are two states in the model. The states are determined by the probability distribution of H and T and the transition between the states are given by a state transition matrix. This model is depicted in Figure 3.2. The observation sequence is given in Table 3.2. This example gives an idea of how HMMs are modeled. Here, the two coins are denoted by $S1$ and $S2$ and the observation is given by H or T. In Table 3.2, the second row gives the sequence of the coin used in the experiment.

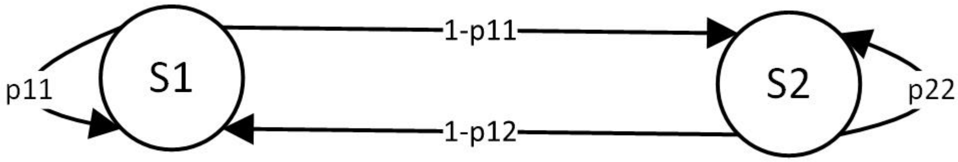

Fig. 3.2.: 2-coin Model

Table 3.2.: Observation Sequence of 2-coin Model

Observation	H	H	T	T	H	T	H	H	T	T	H
Sequence	2	1	1	2	2	2	1	2	2	1	2

From the observation sequence, the state transition matrix can be evaluated. The state transition matrix is given by,

$$P = \begin{bmatrix} P11 & 1 - P11 \\ 1 - P22 & P22 \end{bmatrix}$$

A three coin assumption will give us a better picture for identifying the elements of HMM. This is depicted in Figure 3.3 and its corresponding observation sequence is given in Table 3.3.

Table 3.3.: Observation Sequence of 3-coin Model

Observation	H	H	T	T	H	T	H	H	T	T	H
Sequence	3	1	2	3	3	1	1	2	3	1	3

The transition of the states can be modeled from the probability of the observed sequence, given in Table 3.4.

With the above example, the elements of HMM can be defined easily.

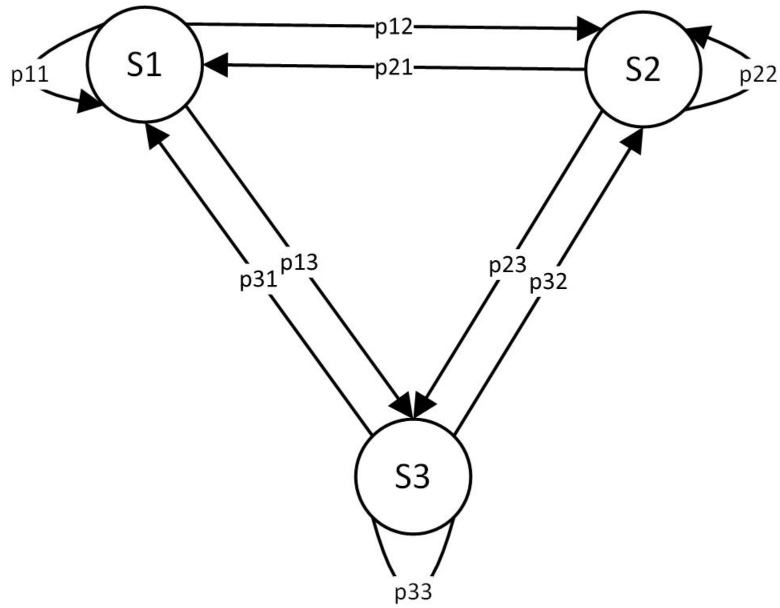

Fig. 3.3.: 3-coin Model

Table 3.4.: Observation Probabilities of 3-coin Model

	S1	**S2**	**S3**
P(H)	P1	P2	P3
P(T)	1-P1	1-P2	1-P3

3.1.1 Elements of HMM

In order to model a hidden Markov model, it is required to define certain parameters, discussed below.

1. $N \rightarrow$ the number of states. Even though the states are hidden, in most practical applications, there are relations between observed sequence and the number of states. In some cases the number of states are completely known.

2. $M \rightarrow$ the number of observations in a state.

3. $P = p_{ij} \rightarrow$ the state transition matrix.

4. $B_i \rightarrow$ observation sequence probability at state S_i, where $i = 1, 2, 3....N$.

5. $\pi \rightarrow$ the initial probability vector.

Given N, M, P, B and π, the HMM can be modeled. The steps are briefly explained below:

1. Initial state is chosen according to the initial probability vector, π.

2. set step time, t=1.

3. The the initial observation is chosen according to the observation probability, B_i, in the chosen initial state.

4. The chain is transitioned to a new state depending on the state transition matrix.

5. increment t, go to step 3.

As mentioned earlier, in practical applications, the states are known and with all the elements of HMM, modeling and simulation of a HMM can be achieved.

3.1.2 HMM Explained Through a Simple Coin Toss Example

Problem statement: Two biased coins are tossed at random. There are only two observations, H or T. The initial probability vector is also given. The elements are given as follows.

$$N = 2, M \rightarrow H, T$$

$$P = \begin{bmatrix} 0.6 & 0.4 \\ 0.4 & 0.6 \end{bmatrix}$$

Observation Probability:

	$S1$	$S2$
P(H)	0.3	0.7
P(T)	0.7	0.3

$$\pi = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}$$

Solution: All the required quantities for modeling a HMM are given. The Markov chain for the hidden states are summarized in Figure 3.4.

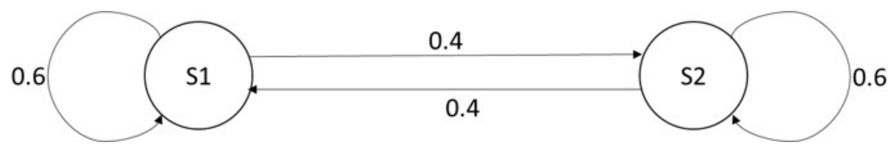

Fig. 3.4.: HMM Model

Let the required observation sequence be $Q = HTHT$, observed from $S_1S_2S_2S_1$. Let λ be all the possible state transitions.

$$\begin{aligned}\lambda \rightarrow\ & S_1S_1S_1S_1, S_1S_1S_1S_2, S_1S_1S_2S_1, S_1S_1S_2S_2\\ & S_1S_2S_1S_1, S_1S_2S_1S_2, S_1S_2S_2S_1, S_1S_2S_2S_2\\ & S_2S_1S_1S_1, S_2S_1S_1S_2, S_2S_1S_2S_1, S_2S_1S_2S_2\\ & S_2S_2S_1S_1, S_2S_2S_1S_2, S_2S_2S_2S_1, S_2S_2S_2S_2\end{aligned}$$

$$b_1(H) = 0.3, \quad b_1(T) = 0.7$$

$$b_2(H) = 0.7. \quad b_2(T) = 0.3$$

Probability of Q from $S_1S_2S_2S_1$ is,

$$P[HTHT, 1221] = b_1(H)b_2(T)b_2(H)b_1(T) = (0.3)(0.3)(0.7)(0.7) = 0.0441$$

Probability of $S_1S_2S_2S_1$ happening is,

$$P[1221, \lambda] = \pi_1 P_{12}P_{22}P_{21} = (0.5)(0.4)(0.6)(0.4) = 0.0048$$

Probability of the sequence, $Q = HTHT$, happening from $S_1S_2S_2S_1$ is,

$$P[HTHT, 1221.\lambda] = P[HTHT, 1221]P[1221, \lambda] = 0.0021$$

Through this we can find the probability of an observation sequence happening through a particular event.

3.2 Design of Active Safety System

Hidden Markov model can be used to design a predictive algorithm to enhance active safety in cars. Prediction of driving behavior [11] has been one of the main areas of research in active safety systems. The idea here is to design a model that can predict the driving behavior of a particular vehicle and thereby, predicting it's location at a particular time. This will help in predicting crash between vehicles by establishing an algorithm that can compare the HMMs of the vehicles, through proper Vehicle-to-Vehicle communication. This information can be used to trigger the appropriate active safety system. The two main active systems used in this thesis are adaptive cruise control and lane departure system. A brief description of these systems are presented at the end of the chapter. There are some basic assumption to design the HMM. These assumptions are explained below:

- All vehicles are equipped with active safety elements. Preferably level 3 autonomous vehicles.
- All vehicles have capabilities of communicating with each other.
- Vehicle dynamics data can be observed and used (Lane number and speed).
- Historical data of vehicle dynamics are also available.

3.2.1 Markov Chain - Model Specification

In this model two layers of Markov Chains are used to develop the prediction flow. The usage of Hidden Markov model in this thesis, slightly deviates from the

Table 3.5.: First Layer

States
Lane 1
Lane 2
Lane 3
Lane 4
Lane 5
Lane 6

Table 3.6.: Second Layer

Observed States	**Symbol**
0-10 m/s	a
10-20 m/s	b
20-30 m/s	c
30-40 m/s	d
40-50 m/s	e
50-60 m/s	f

conventional hidden Markov models. Here, all the states are known and the state transition matrices of both the layers can be computed with the available historical data.

The first layer of the Markov chain (states) is for lane change. The initial assumption here is that, there are six lanes on each side of the road. Depending on the number of lanes, the number of rows in a state transition matrix changes. The dimension of the state transition matrix depends on the number of lanes. The second layer of Markov chain (Observed states) consists of speed ranges. The historical data of speed dynamics has been divided into six speed ranges. The state transition matrix will consist of six rows and six columns. The speed probabilities are calculated for each lane to increase the efficiency of crash prediction.

The states of both the layers of the HMM are listed in Tables 3.5 and 3.6. The lanes are numbered from one to six from left to right of the road correspondingly. The speed dynamics are converted to m/s for flexibility. The speed variable has been divided into six states with 10 m/s range. It can also be seen that end value of an observed state is the same as the start of a next state. This is to provide durability in the design. The end values are sorted depending on the sequence of the previous speed data. Each observation state is related to a symbol. Table 3.7 shows how each observation probability is classified into lanes. The initial probability vector depends

Table 3.7.: Observation Probabilities for Each Lane

1	**2**	**3**	**4**	**5**	**6**
$b_1(a)$	$b_2(a)$	$b_3(a)$	$b_4(a)$	$b_5(a)$	$b_6(a)$
$b_1(b)$	$b_2(b)$	$b_3(b)$	$b_4(b)$	$b_5(b)$	$b_6(b)$
$b_1(c)$	$b_2(c)$	$b_3(c)$	$b_4(c)$	$b_5(c)$	$b_6(c)$
$b_1(d)$	$b_2(d)$	$b_3(d)$	$b_4(d)$	$b_5(d)$	$b_6(d)$
$b_1(e)$	$b_2(e)$	$b_3(e)$	$b_4(e)$	$b_5(e)$	$b_6(e)$
$b_1(f)$	$b_2(f)$	$b_3(f)$	$b_4(f)$	$b_5(f)$	$b_6(f)$

on the current lane of the vehicle. The elements of the modeled HMM are summarized below,

- Number of hidden states, N = 6. This denoted the number of lanes.
- Number of observations in a state, M = 6. This denotes the speed change range.
- State transition matrix, $P = p_{ij}$. Here i and j represent the lane number. $i, j = \{1, 2, 3, 4, 5, 6\}$.
- Observation probability, b. This is summarized in Table 3.7.

The state transition matrices for lane change and speed change are summarized in Equations 3.1 and 3.2 respectively.

$$P = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} & p_{15} & p_{16} \\ p_{21} & p_{22} & p_{23} & p_{24} & p_{25} & p_{26} \\ p_{31} & p_{32} & p_{33} & p_{34} & p_{35} & p_{36} \\ p_{41} & p_{42} & p_{43} & p_{44} & p_{45} & p_{46} \\ p_{51} & p_{52} & p_{53} & p_{54} & p_{55} & p_{56} \\ p_{61} & p_{62} & p_{63} & p_{64} & p_{65} & p_{66} \end{bmatrix} \tag{3.1}$$

$$S = \begin{bmatrix} s_{aa} & s_{ab} & s_{ac} & s_{ad} & s_{ae} & s_{af} \\ s_{ba} & s_{bb} & s_{bc} & s_{bd} & s_{be} & s_{bf} \\ s_{ca} & s_{cb} & s_{cc} & s_{cd} & s_{ce} & s_{cf} \\ s_{da} & s_{db} & s_{dc} & s_{dd} & s_{de} & s_{df} \\ s_{ea} & s_{eb} & s_{ec} & s_{ed} & s_{ee} & s_{ef} \\ s_{fa} & s_{fb} & s_{fc} & s_{fd} & s_{fe} & s_{ff} \end{bmatrix} \tag{3.2}$$

3.2.2 Model Design

The road is assumed to have six lanes. Each lane is associated with a state in the first layer of Markov chain. The road layout considered is illustrated in Figure 3.5.

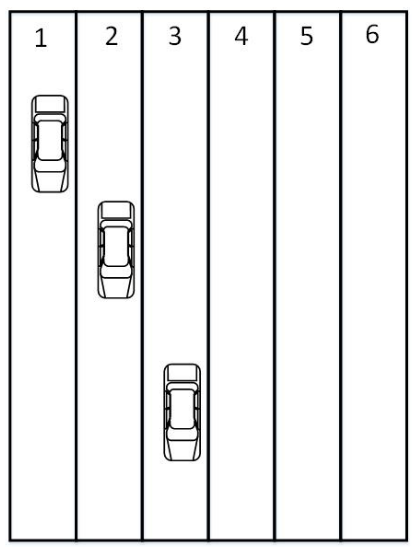

Fig. 3.5.: Road Layout

The Markov chain for lane change is depicted in Figure 3.6. L1, L2, L3, L4, L5 and L6 are states discussed above. Practically examining the model, it is found to be an ergodic Markov chain. It is possible to go from every state to all the other states (not necessarily in one step). It means that overtime, a car can traverse to every lane on the road at least at one point of time.

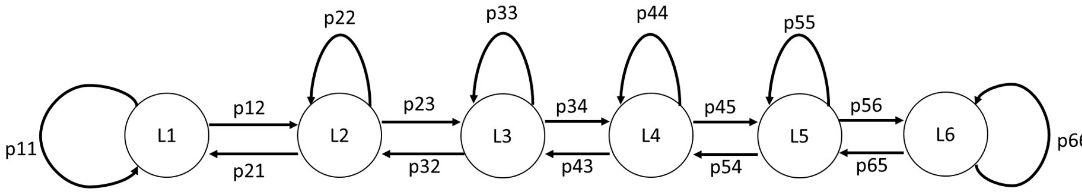

Fig. 3.6.: Lane Change Markov Chain

Figure 3.7 depicts the second layer of the model, speed change Markov chain. The observed states are a, b, c, d, e and f which correspond to the speed ranges. Just like the lane change chain, this is also an ergodic Markov chain. In practical applications, these chains are regular. At some power of the state transition matrix, all the elements of the matrix are strictly positive.

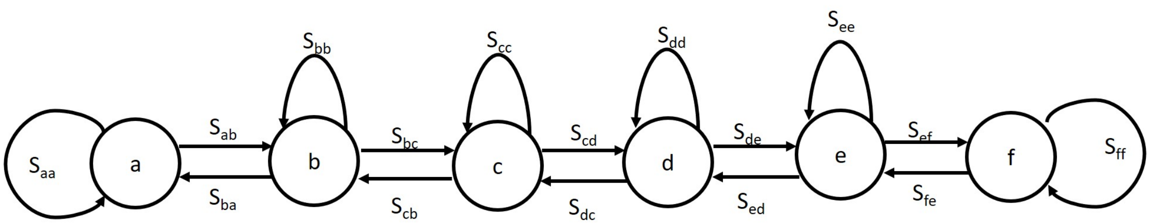

Fig. 3.7.: Speed Change Markov Chain

3.2.3 Description of Variables and Formulae Used

There are five main important elements used in the design of the model. These are listed below,

- $P \rightarrow$ State transition matrix for lane change.
- $S \rightarrow$ State transition matrix for speed change.
- $\pi \rightarrow$ Initial probability vector for lanes.
- $B \rightarrow$ Observation probability for each state.

- $W \rightarrow$ Limiting matrix.
- $Z \rightarrow$ Fundamental matrix.
- $M \rightarrow$ Mean first passage matrix.

Lane Change State Transition Matrix, P

The state transition matrix is obtained from historical lane data of a vehicle. An example of historical data is depicted in Figure 3.8. The formula used for calculating

Car ID	Speed (m/s)	Run ID	Lane	Time
411	55.3397242	1	2	3:08:45
411	54.1625052	1	2	3:08:46
411	53.9462742	1	2	3:08:47
411	53.9795643	1	2	3:08:48
411	54.1787349	1	2	3:08:49
411	54.3926968	1	2	3:08:50
411	53.4974602	1	2	3:08:51
411	53.7246165	1	2	3:08:52
411	54.5598748	1	3	3:08:53
411	55.5720394	1	3	3:08:54
411	55.549364	1	3	3:08:55
411	55.9143163	1	3	3:08:56

Fig. 3.8.: Lane Change Data

each element of the state transition matrix is discussed in Equation 3.4.

$$\text{Probability from } i \text{ to } j = \frac{\text{No. of rows with current lane } = i \text{, and next lane } = j}{\text{No. of rows with current lane } = i} \tag{3.3}$$

$$p_{ij} = \frac{\sum_{k=1}^{n} \eta(X_k = i | X_{k+1} = j)}{\eta(X_k = i)} \tag{3.4}$$

$$P = \{p_{ij}\} \tag{3.5}$$

Where, $i, j = \{1, 2, 3, 4, 5, 6\}$

$i \rightarrow$ current lane, $j \rightarrow$ next lane

$\eta \rightarrow$ number of occurrences

$n \rightarrow$ total number of data rows

Speed Change State Transition Matrix, S

The speed change data is also obtained from historical data as shown in Figure 3.8. The formula for calculating the speed change probabilities varies slightly from lane change probabilities, since it considers a range of data. The formula for calculating the same is given in Equation 3.7.

Probability speed range (l to m) to (m to n)=

$$\frac{\text{No. of rows with current speed in range } (l \text{ to } m)\text{, and next speed in range } (m \text{ to } n)}{\text{No. of rows with range } (l \text{ to } m)} \tag{3.6}$$

$$s_{(l-m)-(m-n)} = \frac{\sum_{k=1}^{n} \eta(l < Y_k < m | m < Y_{k+1} < n)}{\eta(l < Y_k < m)} \tag{3.7}$$

$$S = \{s_{(l-m)-(m-n)}\} \tag{3.8}$$

Where,

(l to m), (m to n) $= \{a, b, c, d, e\}$

$\eta \rightarrow$ number of occurrences

$n \rightarrow$ total number of data rows

Speed Probabilities for Each Lane, B

It is important to note the probabilities of speed range for each lane. This helps in analyzing if a driver has been at the approximately same speed range at a particular lane for the required period of time. The equations concerning B are discussed below.

$$\text{Probability of range } (l \text{ to } m) \text{ at lane } i = \frac{\text{No. of rows with speed range}(l \text{ to } m)}{\text{No. of rows with lane } = i} \tag{3.9}$$

$$B_{(l-m)} = \frac{\sum_{k=1}^{n} \eta(l < Y_k < m | X_k = i)}{\eta(X_k = i)} \tag{3.10}$$

Where,

$i = \{1, 2, 3, 4, 5, 6\}$

$(l \text{ to } m) = \{a, b, c, d, e\}$

$\eta \rightarrow$ number of occurrences

$n \rightarrow$ total number of data rows

Limiting Matrix (W), Fundamental Matrix (Z) and Mean First Passage Matrix (M)

The mean first passage matrix finds its use in determining which active safety system needs to be triggered. This concept will be explained in the upcoming sections. To find the Mean first passage matrix, we need the limiting matrix and fundamental matrix. The formula for W, Z and M are shown in Equations 3.11, 3.13 and 3.16.

$$wP = w \tag{3.11}$$

$$\sum_{1}^{6} w_i = 1, \quad i = \{1, 2, 3, 4, 5, 6\} \tag{3.12}$$

$$Z = (I - P + W)^{-1} \tag{3.13}$$

$$m_{ij} = \frac{Z_{jj} - Z_{ij}}{w_j}, \quad i, j = \{1, 2, 3, 4, 5, 6\}, i \neq j \tag{3.14}$$

$$m_{ij} = 0, i = j \tag{3.15}$$

$$M = m_{ij} \tag{3.16}$$

Probability of Position at Time=t

The probability of the vehicle being at a particular lane can be found using the initial probability vector and the state transition matrix. The formula for the same is given in Equation 3.17.

$$\pi_t = \pi_0 P^t \tag{3.17}$$

π_t has six elements, each of which gives the lane probability at time t.

3.3 Crash Prediction and Avoidance

The prediction of crash has three parts, each of which has been explained using flow charts in the upcoming subsection. The first part is determining the time at which the cars will be at the same y-coordinate. The second section of the flow is to determine if the two cars will be at the same x-coordinate at the previously calculated time and evaluate the crash probability. The third part is to trigger the appropriate active safety system to prevent the crash.

3.3.1 Flow-1 - Calculation for Probable Time of Crash

The state transition matrix for lanes of the vehicles are computed using Matlab. The Matlab code was developed using Equation 3.4. It is assumed that car 1 is in front and car 2 is following car 1 (in the same lane or in any of the other lanes). P1 and P2 denote the state transition matrix of car 1 and car 2 respectively. Then the lane number of both the cars are identified using lane detection system. Let the lane number of car 1 be i and lane number of car 2 be j. This is used to determine the initial probability vector of both the vehicles.

The next step is to determine the current speed of both the vehicles (V1 and V2). This data can be obtained from the speed sensor in the vehicles. The speed is obtained in m/s to provide flexibility in calculation. The relative velocity (v) of car

2 is calculated with respect to car 1, since car 2 is following car 1. The formula for calculating relative velocity is given in Equation 3.18.

$$V = V2 - V1 \tag{3.18}$$

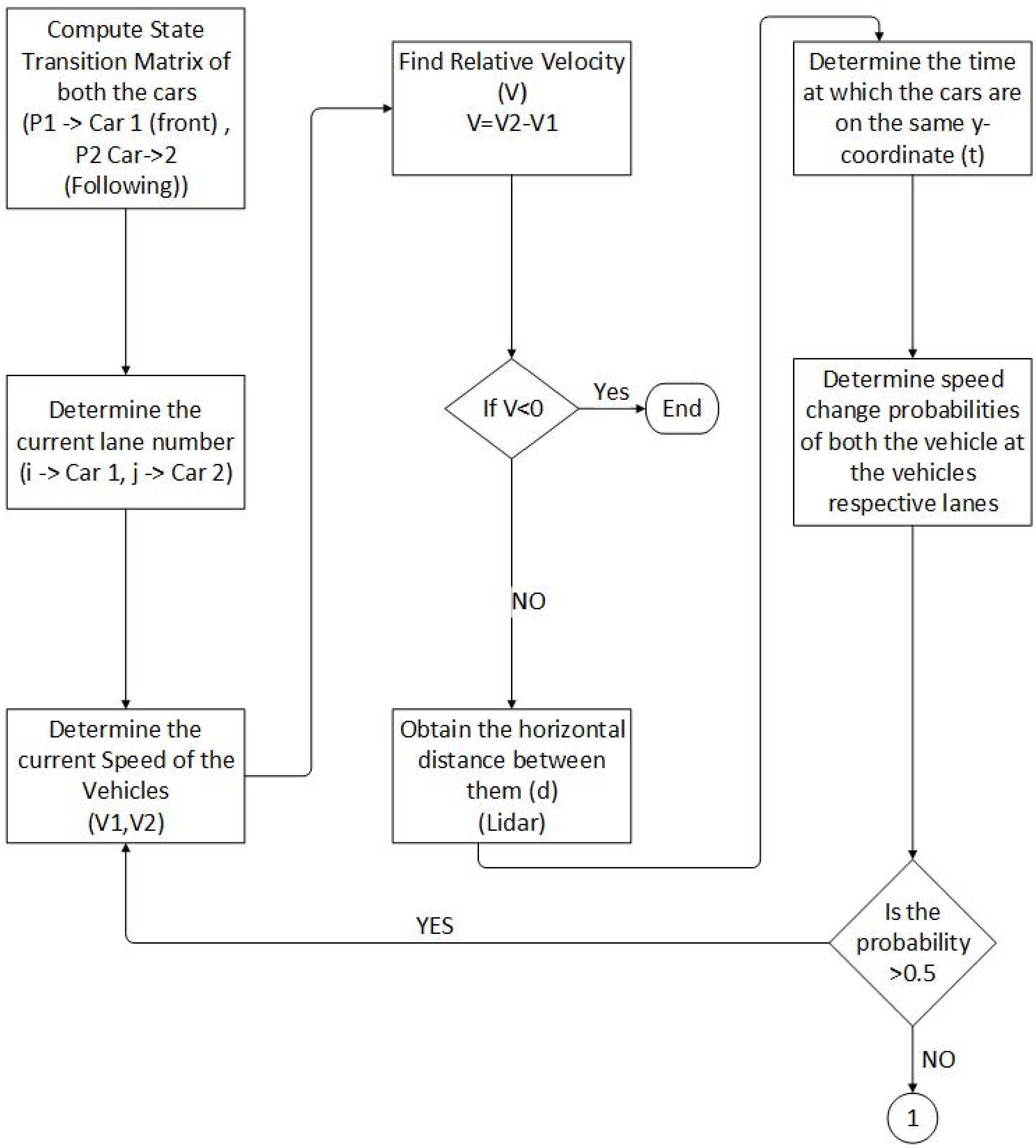

Fig. 3.9.: Probable Time of Crash

If the relative velocity of car 2 with respect to car 1 is less than 0, it means that the speed of car 2 is less than the speed of car 1. So, car 2 will not be able to catch up to car 1 in the next few steps. So the program terminates and starts from the beginning. If the relative velocity is positive, there is a chance that both the cars will be in the same y-coordinate at some point of time. To determine the time that car 2 will take to reach car 1, the distance between the cars is required. A lidar is used to determine the y-distance between the cars. The technique used is summarized below.

Lidar Operation

Light detection and ranging (Lidar) is range measurement device which uses pulsed laser to determine the distance between the two vehicles. Lidar has a 360 degree visibility with extremely accurate distance measurement. Once the lidar sends a pulse of laser, it determines the time it takes for the light to return back to the sensor. This concept is illustrated in Figure 3.10. The diagonal distance between the cars are determined by the lidar. The formula embedded in the device is presented in Equation 3.19. The equation to calculate the y-distance (d) is explained in Equation 3.20.

$$Hyp = \frac{(Ltime)(c)}{2} \tag{3.19}$$

$Ltime$ is the time elapsed by the lidar to return back.

$$d = \sqrt{Hyp^2 + LW^2} \tag{3.20}$$

Probable Time of Crash

The probable time,t is calculated using the formula in Equation 3.21.

$$t = d/V \tag{3.21}$$

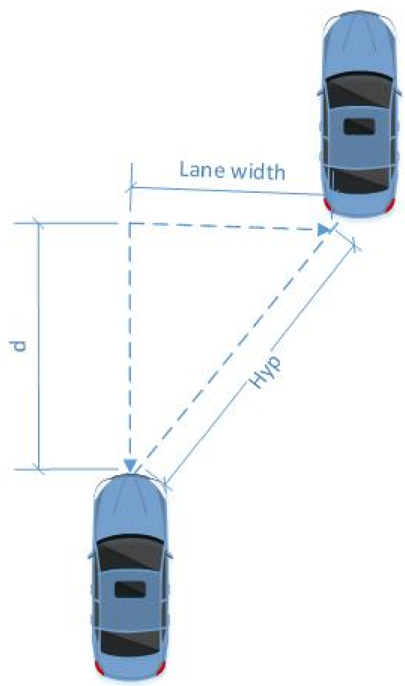

Fig. 3.10.: Lidar Principle

Now that the probable time of the crash has been calculated. To verify that the cars will maintain the same speed, the state transition matrix for speed change is calculated. If the probability of speed change is higher than 0.5, then the loop goes back to determining the current speed. Otherwise, the flow goes into the next step.

3.3.2 Flow 2 - Calculation of Probable Crash Probability

The state transition matrix of both the cars are calculated at time t ($P1^t$ and $P1^t$). Then the probability vector is determined to know the probabilities of the vehicles at different lanes at time t. The probability vector is calculated using the initial probability vector $\pi 1_0$ and $\pi 2_0$. The calculation is summarized in Equations 4.3 and 4.4.

$$\pi 1_t = (\pi 1_0)(P1^t), \text{ Car 1} \tag{3.22}$$

$$\pi 2_t = (\pi 2_0)(P2^t), \text{ Car 2} \tag{3.23}$$

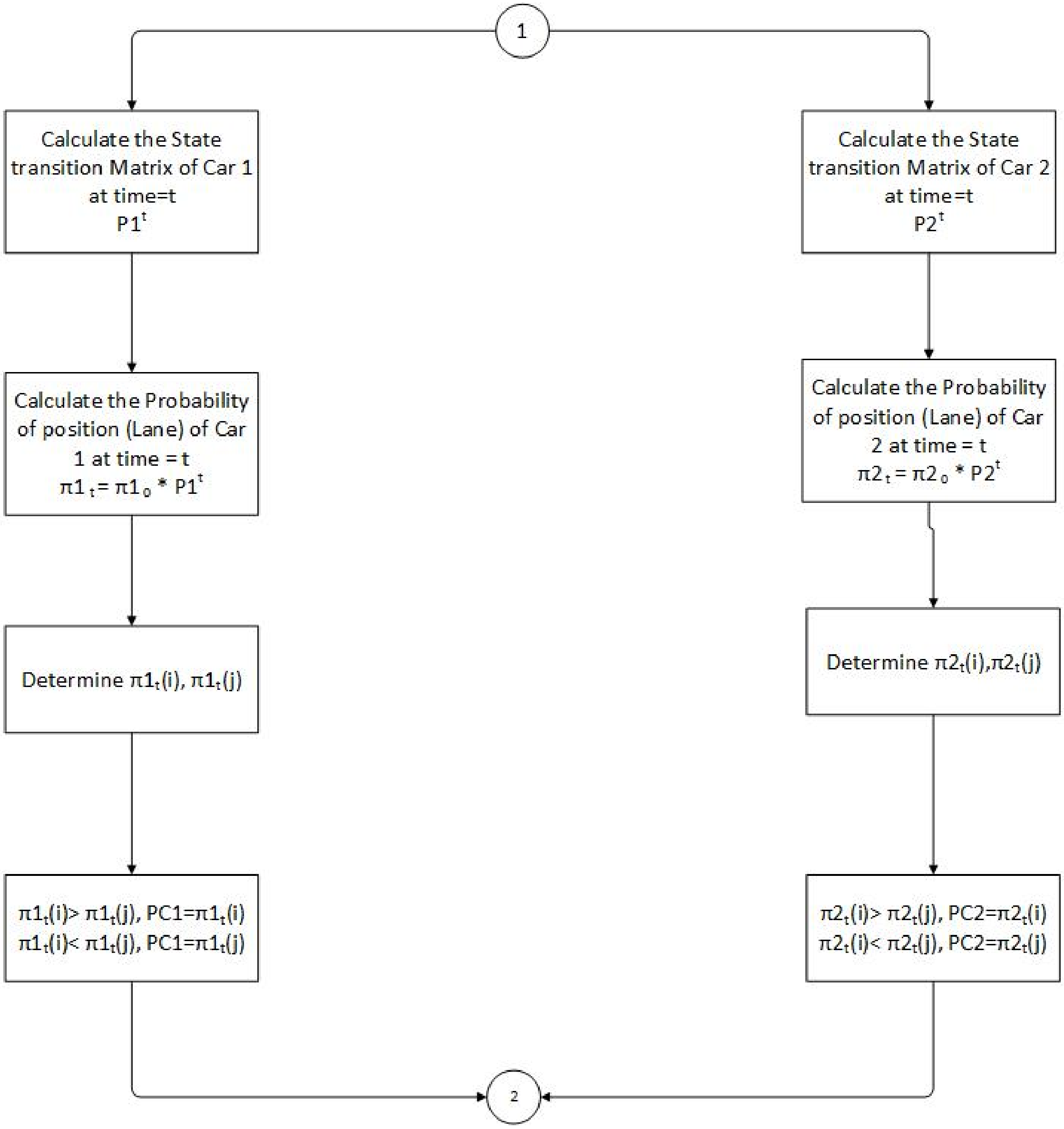

Fig. 3.11.: Probable Crash Probability

The j^{th} element of $\pi 1_t$ and i^{th} element of $\pi 2_t$ give the lane probability of car 1 being at lane j and the lane probability of car 2 being at lane i. Simultaneously, the probability of each car staying at the same lane is also calculated. The crash probabilities are then compared and then the highest probability is chosen as the probable crash probability.

3.3.3 Flow 3 - Determining the Appropriate Active Safety System

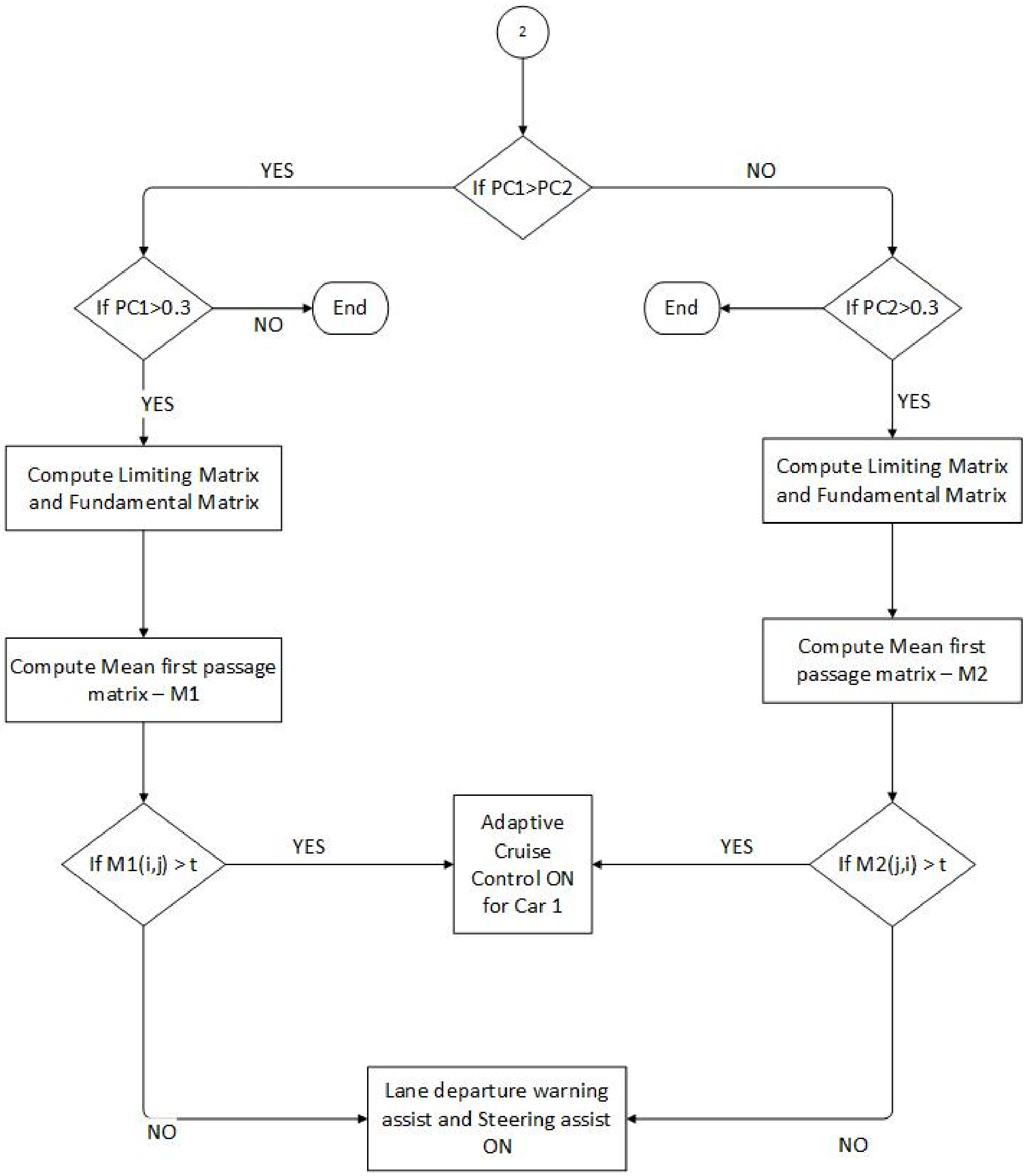

Fig. 3.12.: Flow 3

After the two probable crash probabilities are calculated, the next step is to determine which car has the higher probability to change its lane. The two probabilities are compared and the highest probability is chosen. If the chosen probability is lesser than 30%, then the program terminates and starts from the beginning. Else, the mean first passage matrix is calculated. The mean first passage matrix helps in determining whether the car changes the lane before time t. If the $(i,j)^{th}$ element of the mean first passage matrix is greater than t, then the car changes the lane before time, t. This will lead to a rear end crash. So, adaptive cruise control is on for the following car. If the $(i,j)^{th}$ element of the mean first passage matrix is less than t, the chance of sideways collision at the blind spot is possible. So the lane departure warning and steering assist is activated for both the cars. The flow is depicted in Figure 3.12.

3.4 Adaptive Cruise Control

Adaptive cruise control is an active safety system which controls the speed of the vehicle depending on the dynamics of the vehicle in front. It can be viewed as an add-on to the cruise control system. The ego vehicle is usually equipped with a front facing radar or a Lidar, which calculates the distance and the relative velocity from the lead vehicle. With the safe distance required, relative velocity and the actual distance, the time needed for the ego vehicle to decelerate to the lead vehicle's speed can be calculated. This is used to adjust the speed of the ego vehicle [12].

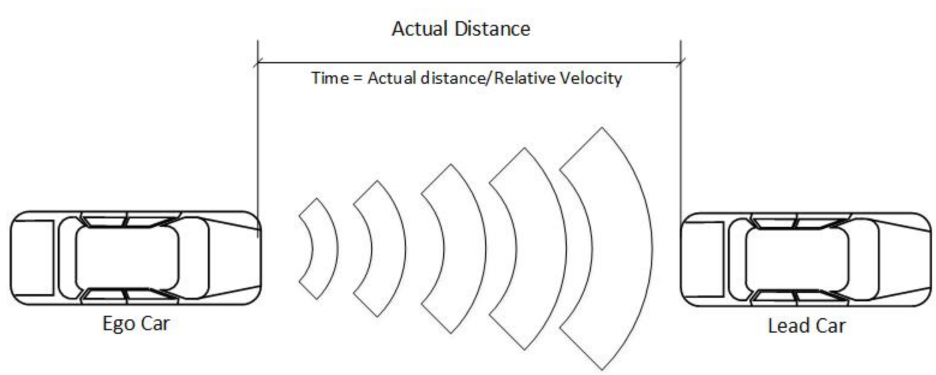

Fig. 3.13.: Distance Calculation

The ACC module communicates with various electronic control units to control the speed of the vehicle. The engine control module controls the engine throttle by obtaining information from the ACC module. The brake control module decelerates the vehicle by using electronic enhancements in brakes (example: ABS brake system). The instrument cluster acts as a mediator, which receives information from the cruise switches and sends the information to the ACC and engine control module.

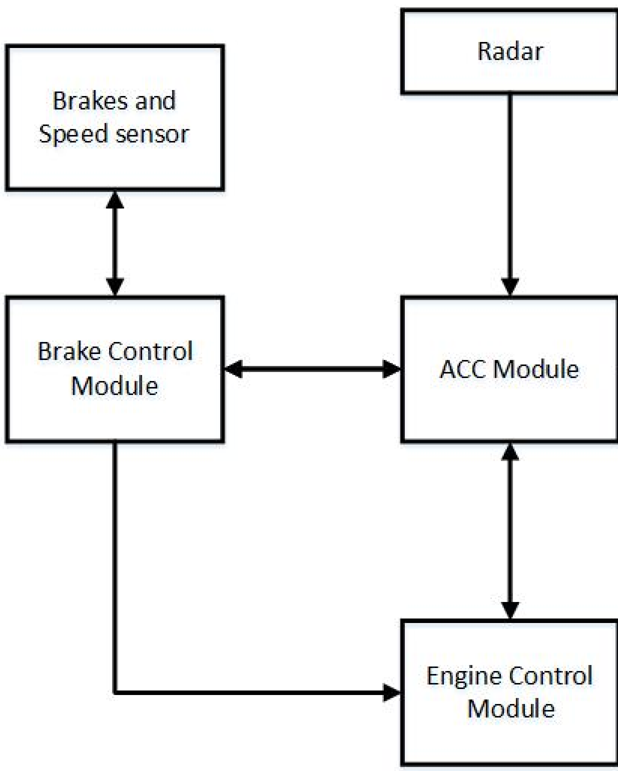

Fig. 3.14.: Adaptive Cruise Control

3.5 Lane Keep Assist

Lane keep assist is an active safety mechanism, which controls the lateral position of the vehicle in a particular lane. The lane keep assist module receives its data from cameras. The lane keep assist module calculates the lateral distance from the left and the right boundaries from the lanes, depending on the image received from the cameras. If the car moves closer to the either of the boundaries, the car is steered accordingly, using the steering assist. The lane keep assist system first alerts the driver about possible lane departure. If no action is taken, then the steering is actively controlled to keep the vehicle in the same lane. The required torque for controlling the steering motor is calculated depending on the extent of departure [13], [14].

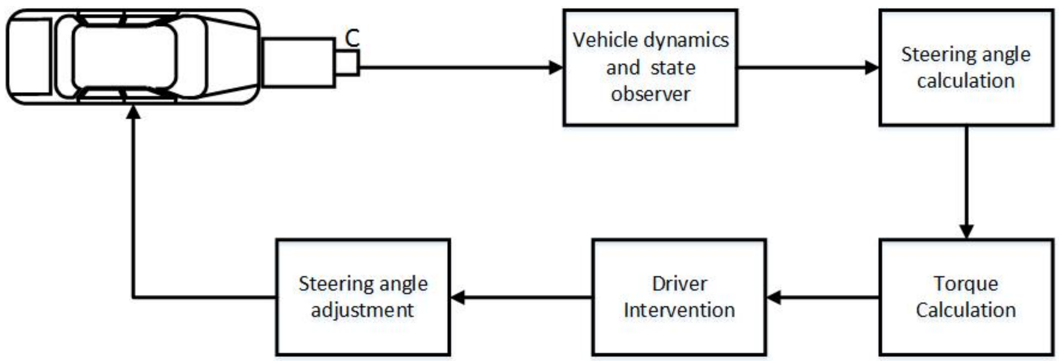

Fig. 3.15.: Lane Keep Assist

The vehicle dynamics considered are longitudinal velocity, lateral velocity, yaw rate and heading angle. The state observer determined are, steering torque, steering wheel angle, lateral vehicle movement relative to lane boundaries and road configurations (road slope and curvature).

4. SIMULATION AND RESULTS

The simulation of the whole system was carried out on MATLAB/SIMULINK, using the algorithm discussed in Chapter 3. The database used for simulation was obtained from U.S. Department of Transportations (USDOT) Intelligent Transportation Systems (ITS) [15]. The database of two cars were obtained and simulated into different scenarios to prove the validation of the system designed. The cars were chosen in such a way that the longitude and the latitude of the whole run matches approximately, which means that both the cars are on the same road. The database was filtered out to obtain the car ID, speed, run ID, Lane and time. The data was limited to five days of historical data to prove the concept. The filtered data was then imported to MATLAB and simulated to obtain the required result. Figures 4.1 and 4.2 illustrate the actual data and the data used for simulation.

GlobalTime	CarID	Latitude	Longitude	Speed	Acceleration	Heading	Run	DistanceAlongCircuit	LaneID	DateTime
1.46E+12	411	37.83003	-122.28066	0.018223	-0.0537127	285.8412	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.016058	-0.0740164	285.8383	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.016058	0.1287131	285.8383	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.016058	0.036482	285.8366	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.015444	-0.1436423	285.8378	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.015444	-0.0600816	285.8364	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.013279	0.0461584	285.8332	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.013279	0.0237433	285.8318	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.013279	-0.1671919	285.8325	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.013279	-0.047296	285.8318	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.012665	0.090657	285.8326	1	0	1	3:05:53 PM
1.46E+12	411	37.83003	-122.28066	0.012665	0.0346903	285.832	1	0	1	3:05:54 PM
1.46E+12	411	37.83003	-122.28066	0.012665	-0.0888562	285.8313	1	0	1	3:05:54 PM
1.46E+12	411	37.83003	-122.28066	0.0105	-0.0939766	285.8305	1	0	1	3:05:54 PM
1.46E+12	411	37.83003	-122.28066	0.0105	0.1337345	285.8296	1	0	1	3:05:54 PM
1.46E+12	411	37.83003	-122.28066	0.0105	0.02882	285.8293	1	0	1	3:05:54 PM
1.46E+12	411	37.83003	-122.28066	0.012051	-0.1341105	285.8276	1	0	1	3:05:54 PM
1.46E+12	411	37.83003	-122.28066	0.009886	-0.0151245	285.8273	1	0	1	3:05:54 PM
1.46E+12	411	37.83003	-122.28066	0.009886	0.0898067	285.8254	1	0	1	3:05:54 PM
1.46E+12	411	37.83003	-122.28066	0.009886	-0.0619264	285.8241	1	0	1	3:05:54 PM

Fig. 4.1.: Data Obtained from USDOT

Car ID	Speed (m/s)	Run ID	Lane	Time
411	55.3397242	1	2	3:08:45
411	54.1625052	1	2	3:08:46
411	53.9462742	1	2	3:08:47
411	53.9795643	1	2	3:08:48
411	54.1787349	1	2	3:08:49
411	54.3926968	1	2	3:08:50
411	53.4974602	1	2	3:08:51
411	53.7246165	1	2	3:08:52
411	54.5598748	1	3	3:08:53
411	55.5720394	1	3	3:08:54
411	55.549364	1	3	3:08:55
411	55.9143163	1	3	3:08:56

Fig. 4.2.: Filtered Data Used for Simulation

4.1 Description of Data Obtained

The vehicles were instrumented with a high precision GPS, two video cameras, two radars and an OBD port reader. Figure 4.3 gives an illustration of the devices and sensors used in data collection.

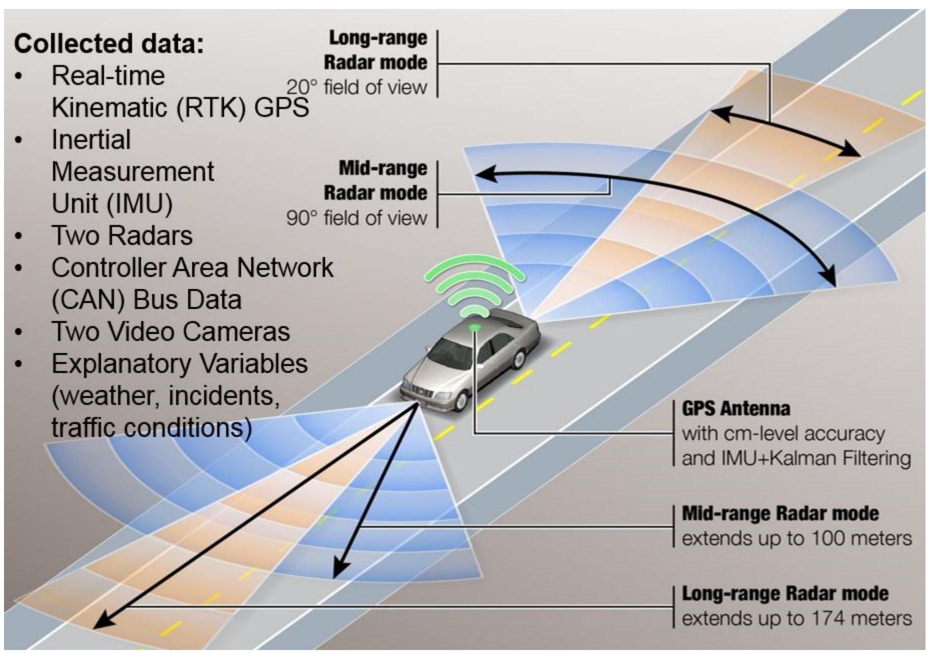

Fig. 4.3.: Instrumented Vehicle

The real-time kinematic (RTK) GPS was used, which provides an accuracy of 4cm when at least four satellites were visible. The RTK GPS device was installed with a precision inertial measurement unit, which measured the rate of acceleration and rotation with great accuracy. A time-step of 50ms was record the position of the vehicle. However, for the purpose of simulation, the data obtained was filtered to each second, keeping in mind the reaction time of the driver. The obtained data was from three different drivers through a period of seven days each. The cameras were used to detect the lane IDs. Radar 1, is a mid-range radar, which extends its distance measurement to up to 100m, while the long range radar extends up to 174m.

4.2 Simulation

Matlab's control toolbox and econometric toolbox have been used to simulate the Markov chains for simulating the crash prediction probabilities. Matlab's MPC toolbox was used to simulate the adaptive cruise control part of the system. The econometric toolbox helps in modeling and analysis of Markov chain with the help of random walks, keeping in mind the Markov property. The structure of the state transition matrix helps in determining the evolutionary trajectory graph of the specified Markov chain. The model predictive control toolbox helps in designing a controller by adjusting the dynamics of the plant designed. In this thesis, the MPC toolbox helps in designing the required ACC system, which controls the car's acceleration, depending on the lead car. The plant to be controlled here is the ego vehicle's motion, which is described in state space format.

Three different scenarios were identified and the required modifications were done on speed and the distance between the cars for the purpose of simulation. In all the three scenarios, the state transition matrix at the start time is the same, since it depends only on the historical data. Since the exact distance between the two cars is not available with the data obtained, the relative distance is assumed to be a non-negative value. Another limitation in regards to the relative distance is, it is always set below 100m.

4.2.1 Determination of State Transition Matrix

The lane data and speed data were first converted from table columns to a column matrix to simplify the processing of data. The formation of the lane change and speed change matrices are illustrated through flow charts represented below.

Lane Change State transition matrix

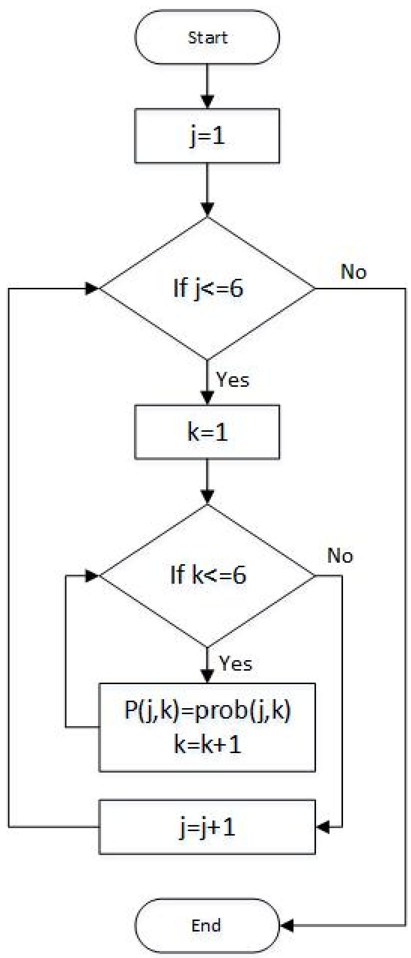

Fig. 4.4.: Lane Change State Transition Matrix Flow

The lane data is stored as a column vector, *glane*. The total length of the vector(n) and index of the matrix(x,y) are obtained from the main program. The total

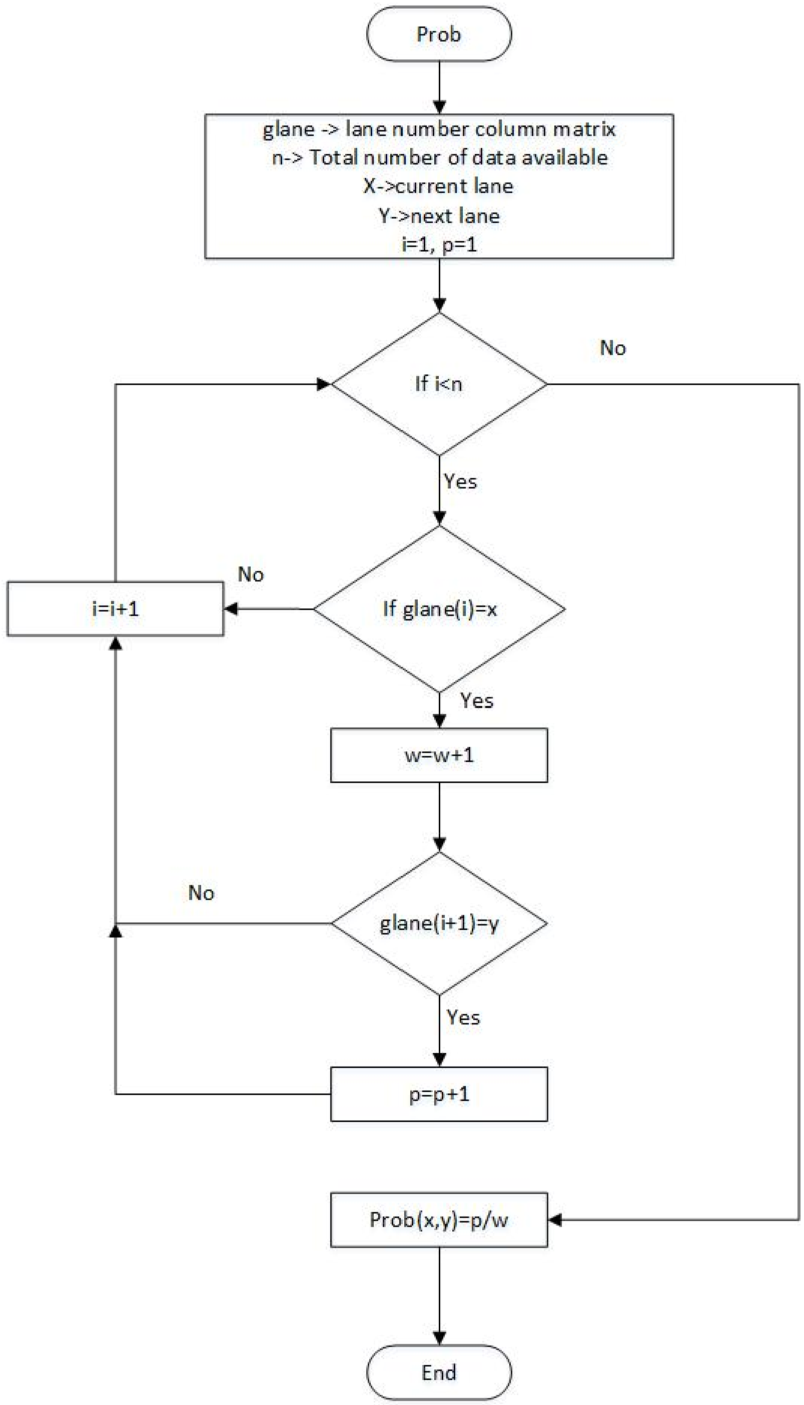

Fig. 4.5.: Probability(Lane Change)

number of occurrences of current lane(x) is calculated. Then the total number of times the next lane(y) appears after x is also calculated. This is stored in the appropriate position of the lane change state transition matrix, P. The lane number exactly corresponds to the index of the matrix. So, no mapping is required while scripting the algorithm.

```
>> Car1
State transition matrix at t=0
```

	Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
Lane1	0.9854	0.014599	0	0	0	0
Lane2	0.0055021	0.99312	0.0013755	0	0	0
Lane3	0	0.071429	0.78571	0.14286	0	0
Lane4	0	0	0.076923	0.88462	0.038462	0
Lane5	0	0	0	0.003125	0.8875	0.10938
Lane6	0	0	0	0	0.74468	0.25532

Fig. 4.6.: State Transition Matrix of Car 1

```
>> Car2
State transition matrix at t=0
```

	Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
Lane1	0.93684	0.063158	0	0	0	0
Lane2	0.019324	0.97262	0.0080515	0	0	0
Lane3	0	0.030488	0.95122	0.018293	0	0
Lane4	0	0	0.028571	0.93333	0.038095	0
Lane5	0	0	0	0.0095923	0.96643	0.023981
Lane6	0	0	0	0	0.10638	0.89362

Fig. 4.7.: State Transition Matrix of Car 2

Figures 4.6 and 4.7 depict the result of state transition matrices of car1 and car2 at time 0. It can be observed that each row the state transition matrix equals 1, which proves that the state transition matrix is a right stochastic matrix. The state transition matrix also gives us an idea of how often a driver changes lane.

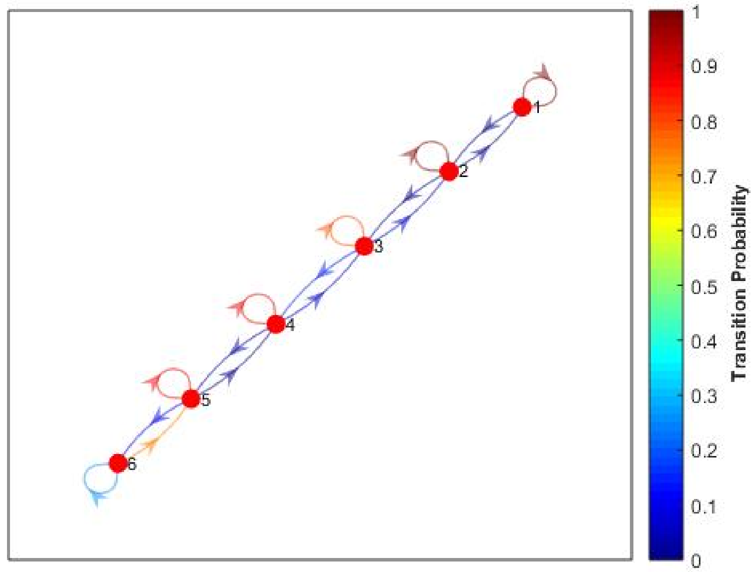

Fig. 4.8.: State Transition graph of Car 1

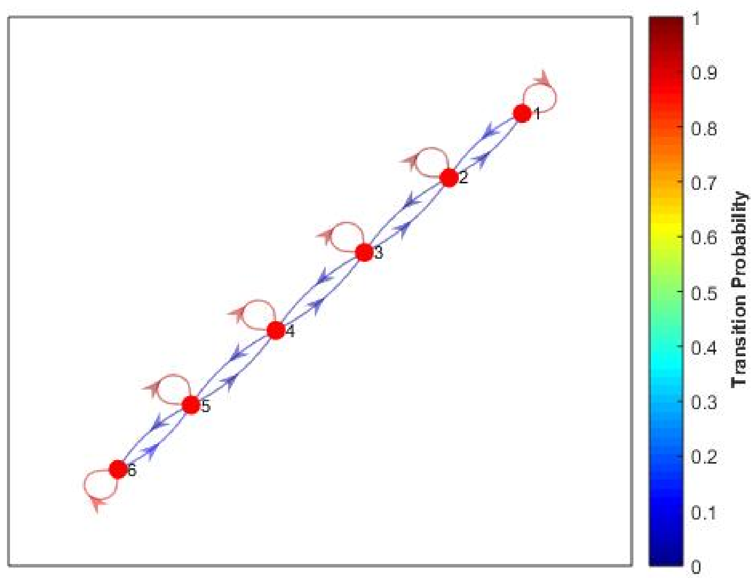

Fig. 4.9.: State Transition Graph of Car 2

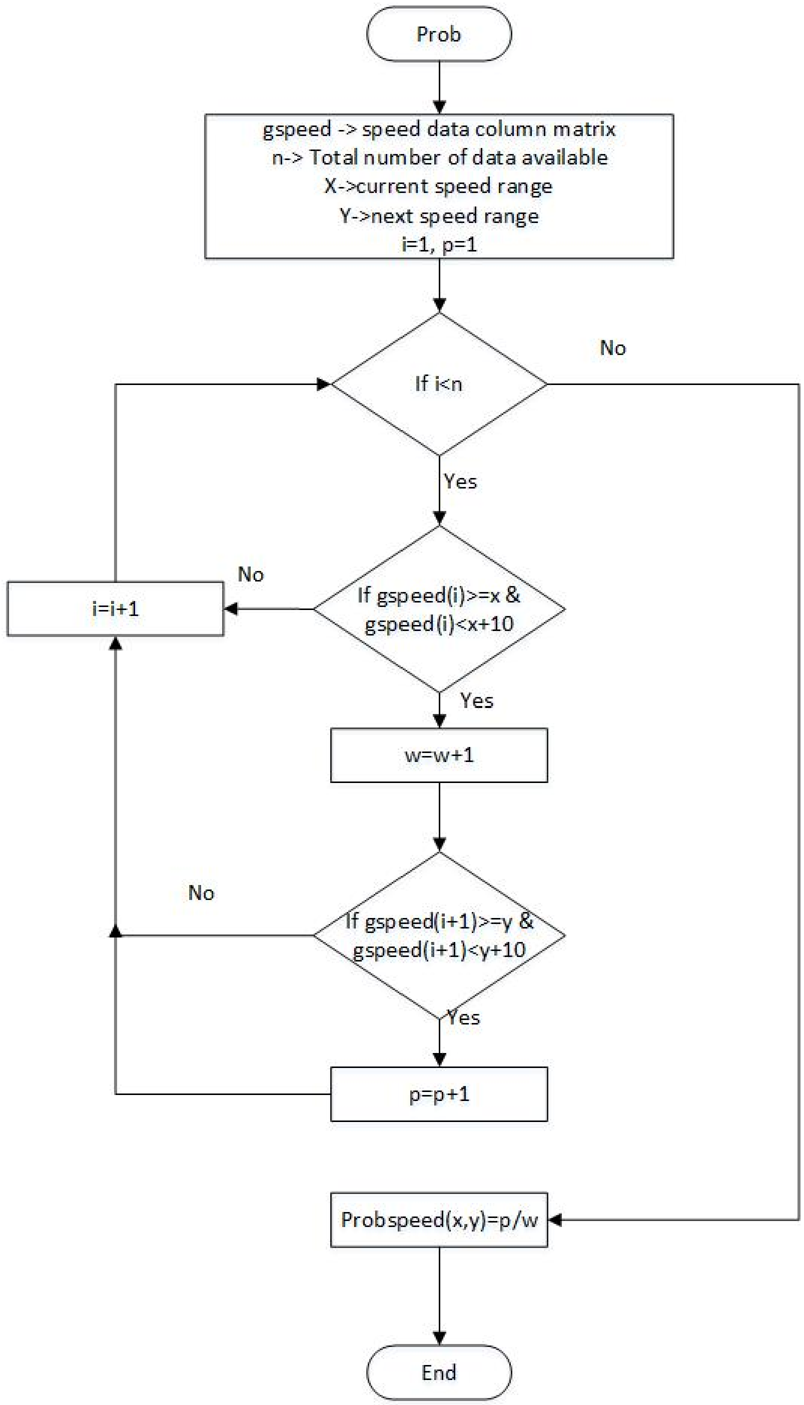

Fig. 4.10.: Speed Change State Transition Matrix Flow

Speed Change State Transition Matrix

The speed change matrix evaluation follows the same algorithm as the lane change matrix as shown in Figure 4.10. Here, the matrix index is not equal to the speed ranges used. This requires mapping of each row/column number to the extremes of each speed range. The formula for mapping the speed range is explained in Equation 4.1.

$$x = (i - 1) * 10 \tag{4.1}$$

x denotes the lower extreme of the speed range and i denoted the index in the matrix. Table 4.1 below gives us the mapping of matrix index to speed range extremities. For example, S(3,4) will give us the speed change probability from $20 - 30m/s$ to $30 - 40m/s$. The mapping of these speed ranges, is the only difference between the speed change and lane change state transition matrices.

Table 4.1.: Speed Range Mapping

i	**Speed range Extremities**
1	0 m/s
2	10 m/s
3	20 m/s
4	30 m/s
5	40 m/s
6	50 m/s

The results of the simulation are provided in Figure 4.11 and 4.12. It can be observed that in both the matrices, the sum of each row approximately equals 1, proving the stochastic behavior of the matrices. Combining both the matrices (P and S), the driving behavior and the intent of the driver at a particular situation can be determined. From the lane change probabilities of Car 1, it can be seen that the driver does not change lanes that often when the car is at lane 1 to 5. The driver changes his lane more often when he is at lane 6. While looking at the speed change matrix of car 1, it can be noted that the driver does not change speeds that often. So, it can be concluded that the driver of car 1 is a safe driver but has tendency to immediately change the lane when the car is at lane 6.

Speed Change Probabilities of Car 1

	S0toS10	S10toS20	S20toS30	S30toS40	S40toS50	S50toS60
S0toS10	0.99812	0.039474	0	0	0	0
S10toS20	0	0.91903	0.076923	0	0	0
S20toS30	0	0	0.93811	0.035831	0	0
S30toS40	0	0	0	0.95339	0.025424	0
S40toS50	0	0	0	0	0.84211	0.026316
S50toS60	0	0	0	0	0	0.97959

Fig. 4.11.: Speed Change State Transition Matrix of Car 1

Speed Change Probabilities of Car 2

	S0toS10	S10toS20	S20toS30	S30toS40	S40toS50	S50toS60
S0toS10	0.99855	0.053468	0	0	0	0
S10toS20	0	0.89181	0.049708	0	0	0
S20toS30	0	0	0.92609	0.052174	0	0
S30toS40	0	0	0	0.95122	0.01626	0
S40toS50	0	0	0	0	0.92727	0.054545
S50toS60	0	0	0	0	0	0.88889

Fig. 4.12.: Speed Change State Transition Matrix of Car 2

Mean First Passage Matrix

As discussed earlier, the mean first passage matrix requires the fundamental matrix and the limiting matrix. The fundamental matrix is calculated directly from the formula given in Equation 3.13. The limiting matrix is solved using the Equation 3.11. To solve the system of equations, Matlab's ODE solver was used to obtain the limiting matrix, W. With these two matrices, the mean first passage matrix was obtained for the cars.

```
Limiting matrix for Car 1
    0.1946    0.5163    0.0099    0.0185    0.2273    0.0334
    0.1946    0.5163    0.0099    0.0185    0.2273    0.0334
    0.1946    0.5163    0.0099    0.0185    0.2273    0.0334
    0.1946    0.5163    0.0099    0.0185    0.2273    0.0334
    0.1946    0.5163    0.0099    0.0185    0.2273    0.0334
    0.1946    0.5163    0.0099    0.0185    0.2273    0.0334

fundamental matrix for Car 1
   96.8205  110.5114   -1.9187   -7.1919 -171.9249  -25.2963
   41.6508  145.8803   -1.2376   -5.9270 -156.3568  -23.0097
  -37.5523  -64.2681    8.7155   12.5574   71.1432   10.4043
  -75.7917 -165.7280    6.7617   21.9288  186.4842   27.3450
 -147.2107 -355.2230    3.1125   15.1518  423.0751   62.0943
 -147.4721 -355.9164    3.0992   15.1270  422.7699   63.3924

Mean first passage matrix Car 1
   1.0e+03 *

         0    0.0685    1.0695    1.5770    2.6180    2.6569
    0.2835         0    1.0010    1.5085    2.5495    2.5884
    0.6905    0.4070         0    0.5075    1.5485    1.5874
    0.8870    0.6035    0.1965         0    1.0410    1.0799
    1.2540    0.9705    0.5635    0.3670         0    0.0389
    1.2553    0.9718    0.5648    0.3683    0.0013         0
```

Fig. 4.13.: W, Z and M for Car 1

```
Limiting matrix for car 2
    0.1194    0.3903    0.1031    0.0660    0.2621    0.0591
    0.1194    0.3903    0.1031    0.0660    0.2621    0.0591
    0.1194    0.3903    0.1031    0.0660    0.2621    0.0591
    0.1194    0.3903    0.1031    0.0660    0.2621    0.0591
    0.1194    0.3903    0.1031    0.0660    0.2621    0.0591
    0.1194    0.3903    0.1031    0.0660    0.2621    0.0591

fundamental matrix for car 2
   35.9907   72.0628    2.9506  -11.6621  -79.7984  -18.5435
   22.0482   78.2428    4.5827  -10.6172  -75.6485  -17.6080
    3.4184   17.3528   21.3023    0.0874  -33.1360   -8.0249
  -21.1029  -62.7930    0.1365   21.5362   52.0463   11.1769
  -36.3590 -112.6564  -13.0319   13.1052  122.8131   27.1291
  -37.4816 -116.3255  -14.0009   12.4848  120.3493   35.9737

Mean first passage matrix for car 2
         0   15.8333  178.0333  503.0333  773.0333  922.7333
  116.7500         0  162.2000  487.2000  757.2000  906.9000
  272.7500  156.0000         0  325.0000  595.0000  744.7000
  478.0833  361.3333  205.3333         0  270.0000  419.7000
  605.8333  489.0833  333.0833  127.7500         0  149.7000
  615.2333  498.4833  342.4833  137.1500    9.4000         0
```

Fig. 4.14.: W, Z and M for Car 2

4.2.2 Adaptive Cruise Control Simulation

The Matlab's MPC toolbox provides a Simulink block that uses model predictive control approach to determine the acceleration or deceleration required for the ego vehicle to maintain a safe distance from the target vehicle. There are two main objective to the ACC block used which are explained below:

1. if the relative distance is greater than the safe distance, the ego vehicle travels in the driver set speed.

2. if the relative distance is less than the safe distance, the control goal is to reduce the speed of the ego vehicle so that the distance between the cars is maintained at a safe distance.

The acceleration of the Lead vehicle follows a sine wave, to approximate the realistic driving conditions. The adaptive cruise control system block generates an acceleration output, which is induced into the ego vehicle, thereby modifying the speed of the ego vehicle. The plant model of the ego vehicle and the lead vehicle is given in Equation 4.2 [16].

$$G = \frac{1}{s(0.5s + 1)} \tag{4.2}$$

The inputs to the ACC system block are,

- Speed of ego vehicle.

- Relative distance between ego vehicle and lead vehicle(radar).

- Relative velocity between ego vehicle and lead vehicle(radar).

Figure 4.15 is the screenshot of the Simulink file used for simulating ACC in different scenarios. The design of plant for the cars is given in Figure 4.16.

4.3 Scenarios and Simulation Result

As explained earlier, three different scenarios have been developed and simulated to prove the concept of HMM in active safety systems. The scenarios differ from each other on the basis of speed of the vehicles, relative distance between them and the

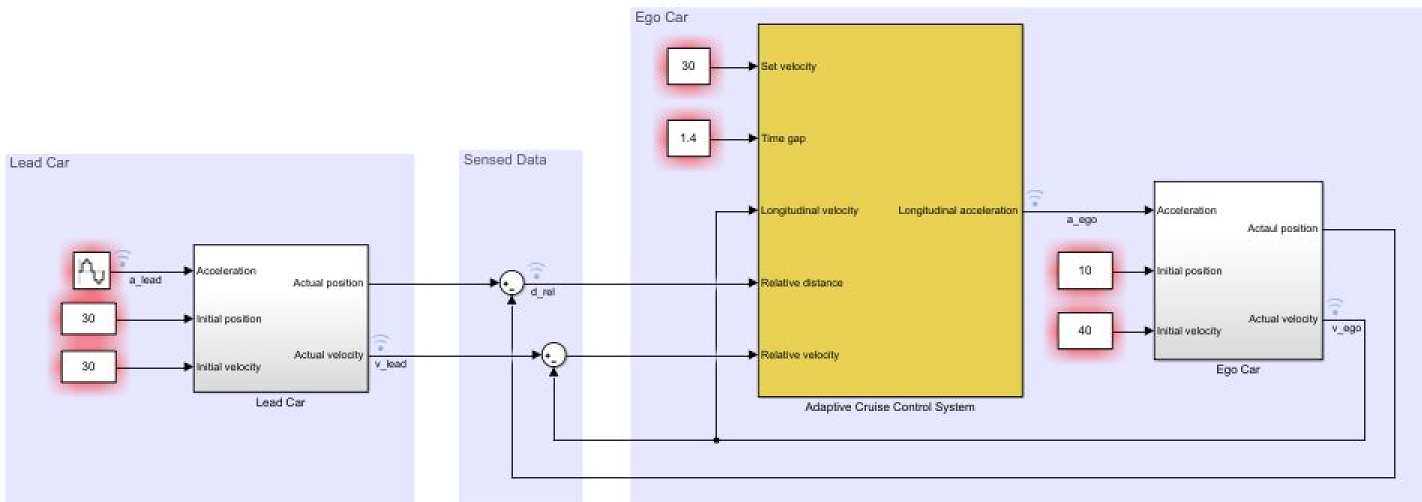

Fig. 4.15.: Adaptive Cruise Control

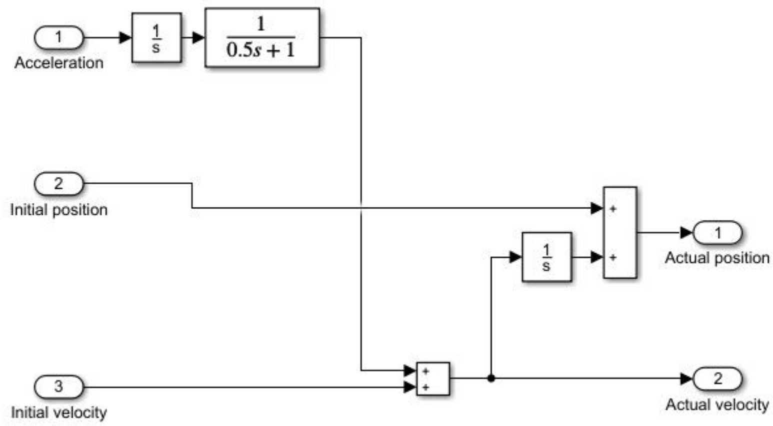

Fig. 4.16.: Plant Design for Cars

lane at which they are traveling in. The objective of the scenarios is to see how the system reacts to safe and unsafe conditions. Every scenario has some assumptions, which will be discussed in the following sections.

4.3.1 Scenario 1: Unsafe Operation with ACC

The assumptions of the scenario are as follows:

1. There are six lanes on one side of the road.
2. Car 1 is at lane 6, traveling in front of car 2 at 30 m/s with a constant acceleration of 0.6 m/s^2.
3. Car 2 is at lane 5, traveling at 40m/s.
4. The relative distance between the cars is 40m.

Here, car 1 is the lead car and car 2 is the ego car. The initial probability vector of car 1 and car 2 is given in Equations 4.3 and 4.4.

$$\pi_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.3}$$

$$\pi_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{4.4}$$

The results of the simulation are illustrated in Figures 4.17-4.20.

Probabale time of crash = 4

State transition matrix of Car 1 at t= 4s

	Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
Lane1	0.94334	0.056544	0.00010279	1.0467e-05	1.1033e-07	0
Lane2	0.021311	0.97375	0.0039768	0.00093365	2.6837e-05	8.2663e-07
Lane3	0.0020118	0.20651	0.4259	0.33939	0.024493	0.0016906
Lane4	0.00011031	0.026106	0.18275	0.66093	0.11763	0.012475
Lane5	9.4472e-08	6.0971e-05	0.0010716	0.009557	0.86231	0.127
Lane6	0	1.2786e-05	0.00050358	0.0069011	0.86469	0.12789

Probability of the position of car 1 at t=4s

Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
0	1.2786e-05	0.00050358	0.0069011	0.86469	0.12789

Fig. 4.17.: Lane probabilities of Car 1 at probable time of crash

State transition matrix of car 2 at t= 4s

	Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
Lane1	0.7769	0.22028	0.0027755	3.5293e-05	3.5437e-07	0
Lane2	0.067398	0.90305	0.02873	0.00080197	2.1454e-05	1.3455e-07
Lane3	0.0032155	0.10879	0.82285	0.061301	0.0037789	6.2577e-05
Lane4	6.3863e-05	0.0047431	0.095746	0.76356	0.13113	0.0047566
Lane5	1.6146e-07	3.1949e-05	0.0014862	0.033019	0.88792	0.077541
Lane6	0	8.889e-07	0.00010918	0.0053133	0.34398	0.65059

Probability of the position of car 2 at t=4s

Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
1.6146e-07	3.1949e-05	0.0014862	0.033019	0.88792	0.077541

Fig. 4.18.: Lane Probabilities of Car 2 at Probable Time of Crash

```
Probability of crash at lane 6 is 0.991655 percent
Probability of crash at lane 5 is 76.778132 percent
Adaptive Cruise Control on for Car 2
```

Fig. 4.19.: Crash Probability at Probable Time of Crash

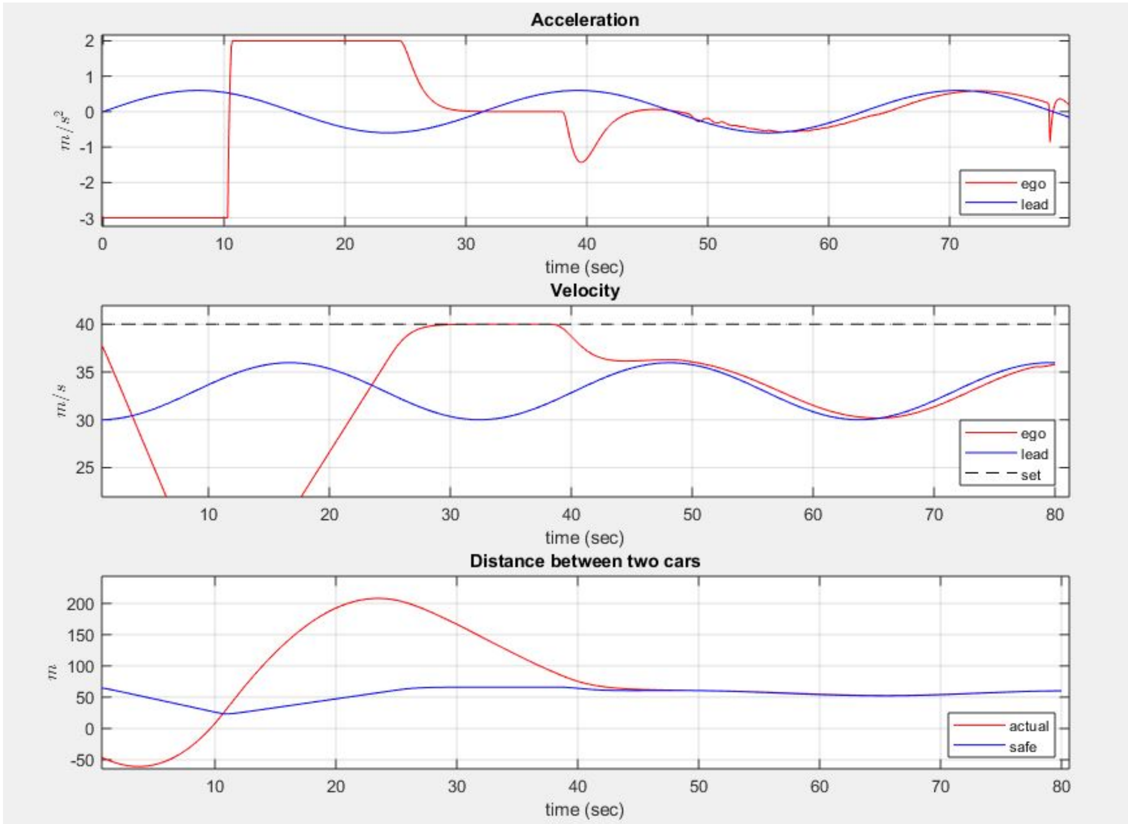

Fig. 4.20.: Simulation of ACC

With the relative distance and the relative velocity obtained from the radar, the probable time of crash was calculated. The state transition matrix of the two cars at this time was calculated. Using the initial probability matrix, the probability of lanes of the cars at probable time of crash was also calculated. From the matrices obtained from the Matlab program, it is clear that Car 1 and car 2 have a higher probability to be at lane 5 at the probable time of crash. Multiplying these probabilities will give us the crash probability at the probable time of crash, which is 0.768 or 76.8%. Comparing the probable time of crash with (6,5)th element of the mean first passage time matrix, it is possible that car 1 changes it's lane much earlier than the time of crash. So this indicates that ACC for car 2 has to be turned on to prevent rear end crash.

4.3.2 Scenario 2: Unsafe Operation with Lane Departure and Steering Assist

The assumptions of the scenario are as follows:

1. There are six lanes on one side of the road.
2. Car 1 is at lane 6, traveling at 50 m/s with a constant acceleration of 0.6 m/s^2.
3. Car 2 is at lane 5, traveling in front of car 1 at 40m/s.
4. The relative distance between the cars is 9m.

This scenario could be one in which car 1 is entering the highway at lane 6, since ACC would have come into effect if car 1 had stayed in the highway much before the relative distance became 9m. Car 2, on the other hand, is traveling at 40 m/s in front of car 1. The initial probability vector stays the same for both the vehicles as in scenario 1. The results of the simulation are illustrated in Figures 4.21-4.23.

Probabale time of crash = 1.200000e+00
State transition matrix of Car 1 at t= 1.200000e+00s

	Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
Lane1	0.98252	0.01748	2.5825e-06	-1.0921e-07	2.2943e-09	-3.1675e-10
Lane2	0.0065882	0.99177	0.0016111	2.6026e-05	-3.1926e-07	3.303e-08
Lane3	5.0543e-05	0.083665	0.75028	0.16528	0.00077012	-4.3102e-05
Lane4	-1.1509e-06	0.00072773	0.088995	0.86471	0.044822	0.00074963
Lane5	1.9645e-09	-7.2533e-07	3.3693e-05	0.0036418	0.88104	0.11528
Lane6	-1.8466e-09	5.1091e-07	-1.2839e-05	0.00041469	0.7849	0.2147

Probability of the position of car 1 at t=1.200000e+00s

Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
-1.8466e-09	5.1091e-07	-1.2839e-05	0.00041469	0.7849	0.2147

Fig. 4.21.: Lane Probabilities of Car 1 at Probable Time of Crash

After calculating the state transition matrices of the cars at the probable time of crash, the probability of the cars being at a each lane is calculated. It can be seen from the results that there is a high probability that the cars will be at lane 5 at the same time. Since the reaction time (relative time for car 1 to catch up to car 2) is

State transition matrix of car 2 at t= 1.200000e+00s

	Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
Lane1	0.92485	0.075087	6.3408e-05	-3.2765e-07	5.8677e-09	-8.617e-11
Lane2	0.022973	0.96742	0.0095865	1.8384e-05	-1.9502e-07	2.2869e-09
Lane3	7.346e-05	0.0363	0.94185	0.021691	8.7149e-05	-6.031e-07
Lane4	-5.9288e-07	0.00010873	0.033879	0.92065	0.045242	0.00011614
Lane5	2.6736e-09	-2.9043e-07	3.4275e-05	0.011392	0.96022	0.028359
Lane6	-1.7417e-10	1.5108e-08	-1.0522e-06	0.00012973	0.1258	0.87407

Probability of the position of car 2 at t=1.200000e+00s

Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
2.6736e-09	-2.9043e-07	3.4275e-05	0.011392	0.96022	0.028359

Fig. 4.22.: Lane Probabilities of Car 2 at Probable Time of Crash

```
Probability of crash at lane 6 is 0.608853 percent
Probability of crash at lane 5 is 75.367410 percent
Lane departure indication and steering assist on
```

Fig. 4.23.: Crash Probability at Probable Time of Crash

very low, the adaptive cruise control will cause discomfort while reducing the speed of car 1. As a result, the time of crash is compared with the mean first passage time of car 1 from lane 6 to lane 5 and if it is lesser, then lane departure warning and steering assist are on for car 1, which will restrict car 1 to go from lane 6 to lane5 and thus, avoiding the crash.

4.3.3 Scenario 3: Safe Operation

The assumptions of the scenario are as follows:

1. There are six lanes on one side of the road.
2. Car 1 is at lane 1, traveling in front of car 2 at 50 m/s.
3. Car 2 is at lane 2, traveling at 60m/s.
4. The relative distance between the cars is 30m.

The results of the simulation are illustrated in Figures 4.24-4.26.

```
Probabale time of crash = 3
State transition matrix of Car 1 at t= 3s
```

	Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
Lane1	0.95708	0.042863	5.5507e-05	2.8686e-06	0	0
Lane2	0.016155	0.98002	0.0032945	0.00052337	7.5578e-06	0
Lane3	0.0010864	0.17108	0.5123	0.30088	0.014054	0.00060096
Lane4	3.0231e-05	0.014634	0.16201	0.72065	0.094148	0.0085289
Lane5	0	1.717e-05	0.00061486	0.0076495	0.86473	0.12699
Lane6	0	0	0.00017901	0.0047181	0.86458	0.13052

```
Probability of the position of car 1 at t=3s
```

Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
0.95708	0.042863	5.5507e-05	2.8686e-06	0	0

Fig. 4.24.: Lane Probabilities of Car 1 at Probable Time of Crash

```
State transition matrix of car 2 at t= 3s
```

	Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
Lane1	0.82571	0.17282	0.0014547	9.3022e-06	0	0
Lane2	0.052876	0.92433	0.022367	0.00042082	5.6108e-06	0
Lane3	0.0016853	0.084695	0.86287	0.048748	0.0019867	1.6711e-05
Lane4	1.6832e-05	0.0024888	0.076139	0.81555	0.10326	0.0025519
Lane5	0	8.3557e-06	0.00078136	0.026	0.91088	0.062328
Lane6	0	0	2.9156e-05	0.0028505	0.2765	0.72062

```
Probability of the position of car 2 at t=3s
```

Lane1	Lane2	Lane3	Lane4	Lane5	Lane6
0.052876	0.92433	0.022367	0.00042082	5.6108e-06	0

Fig. 4.25.: Lane Probabilities of Car 2 at Probable Time of Crash

```
Probability of crash at lane 1 is 5.060651 percent
Probability of crash at lane 2 is 3.961952 percent
Safe operation and probability of crash is low
```

Fig. 4.26.: Crash Probability at Probable Time of Crash

In this scenario, the cars are in adjacent lanes traveling 30m apart from each other. The state transition matrices of the cars are evaluated and the probability of positions of the cars are evaluated at the probable time of crash. From the results, it is seen that the probability that the cars will change lane or be at the same lane is very negligible. So the cars will continue to operate without any intervention of the active safety systems.

5. CONCLUSION AND FUTURE WORK

5.1 Conclusion

The idea of this thesis was to develop an algorithm to predict crash probabilities and take action depending on the dynamics of the vehicle. The thesis is divided into five chapters. The first chapter gives a brief explanation on the existing ADAS technology and the impact it has created. It also explains the need for prediction of driving behavior and how it will help in improving active safety systems. This chapter also gives a brief description of Markov chains and its use in prediction algorithms. A number of papers were cited and reviewed, which served as inspirations for this thesis.

The second chapter deals with explaining the characteristics and the mathematics involved in Markov chains. This chapter describes the elements of a Markov chains and the model specification, which include the states, state transition matrix and the probability distribution. The classification of Markov chains and the characteristics of the types of Markov chains are discussed in detail. The elements for analyzing a Markov chain has been discussed and the derivation of the same was also done. The concepts were also explained through a simple example.

Chapter 3 focuses on hidden Markov model and the implementation of the same in active safety. A variant of the same was also derived and explained to suit the goal of the thesis. The elements of the HMM used in this this thesis was also explained, keeping in mind the road design. The active safety system was designed and an algorithm to predict crash was also developed. The description and derivation of the formulae used was also discussed. A brief introduction of adaptive cruise control and lane keep assist systems were also done.

Simulation of the system and the algorithm explained in Chapter 3 was described in Chapter 4. The results of the same was also illustrated. Chapter 4 also explains the minimum system requirement needed to implement the system. This chapter gives a brief description of the toolboxes used in Matlab to program and simulate the algorithm. A few sections of this chapter emphasizes on the flow which helps in obtaining the various state transition matrices. Three scenarios were created to test the system and the results of the same were illustrated.

The final chapter provides the summary and future work of this thesis.

5.2 Future Work

The probable future work of this project are itemized below,

- The mechanical dynamics of the vehicle (mass, center of mass) can be considered while designing the movement of the vehicle.

- The efficiency of braking and the time required to accelerate/decelerate from one speed to another can be an extension to this thesis.

- Addressing the change in number of lanes from one highway to another can be considered and the necessary changes to the state transition matrices can be implemented.

- Improving the efficiency of the system in urban traffic conditions by having a separate state transition matrix for different condition can also be implemented.

- The Markov model can be updated during vehicle operation to improve the efficiency of prediction.

- The same system can be extended to level 5 autonomous vehicles.

REFERENCES

REFERENCES

[1] M. Althoff, O. Stursberg, and M. Buss, "Safety assessment of driving behavior in multi-lane traffic for autonomous vehicles," *IEEE Intelligent Vehicles Symposium*, pp. 893–900, 2009.

[2] C. Zong, C. Wang, D. Yang, and H. Yang, "Driving intention identification and maneuvering behavior prediction of drivers on cornering," *International Conference on Mechatronics and Automation*, pp. 4055–4060, 2009.

[3] K. Singh and B. Li, "Estimation of traffic densities for multilane roadways using a markov model approach," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4369–4376, 2012.

[4] A. Hoekstra, D. Filev, S. Szwabowski, K. McDonough, and I. Kolmanovsky, "Evolving markov chain models of driving conditions using onboard learning," *IEEE International Conference on Cybernetics (CYBCO)*, pp. 1–6, 2013.

[5] J. Ge, G. Orosz, and R. Murray, "Connected cruise control design using probabilistic model checking," *American Control Conference (ACC)*, pp. 4964–4970, 2017.

[6] E. Moradi-Pari, H. Mahjoub, H. Kazemi, Y. Fallah, and A. Tahmasbi-Sarvestani, "Utilizing model-based communication and control for cooperative automated vehicle applications," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 1, pp. 38–51, 2017.

[7] N. Wan, C. Zhang, and A. Vahidi, "Probabilistic anticipation and control in autonomous car following," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 1, pp. 30–38, 2019.

[8] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Springer, 2008.

[9] D. Jurafsky and J. H. Martin, *HMM in Speech and Language Processing*, 2018.

[10] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[11] A. B. Ellison, S. P. Greaves, and M. C. Bliemer, "Driver behaviour proles for road safety analysis," *Elsevier Accident Analysis and Prevention 76*, pp. 118–132, 2015.

[12] W. Pananurak, S. Thanok, and M. Parnichkun, "Adaptive cruise control for an intelligent vehicle," *2008 IEEE International Conference on Robotics and Biomimetics*, 2009.

[13] J. Pohl and J. Ekmark. A lane keeping assist system for passenger cars - design aspects of the user interface. Last Date Accessed: 03/30/2019. [Online]. Available: https://bit.ly/2FUwG98

[14] D. Hoehener, G. Huang, and D. D. Vecchio, "Design of a lane departure driver-assist system under safety specifications," *IEEE 55th Conference on Decision and Control (CDC)*, pp. 2468–2474, 2016.

[15] U. D. of Transportations (USDOT) Intelligent Transportation Systems (ITS) Joint Program Office (JPO). Atdm trajectory level validation instrumented vehicle trajectory data. Last Date Accessed: 04/20/2019. [Online]. Available: https://www.its-rde.net/

[16] A. Bemporad, N. L. Ricker, and M. Morari. (2016) Model predictive control toolbox user's guide. Last Date Accessed: 04/05/2019. [Online]. Available: https://instruct.uwo.ca/engin-sc/391b/downloads/mpc.pdf