DIGITAL SIMULATION OF FREEWAY MERGING OPERATION

By

Donald R. Drew
Head of Design and Traffic Department
and Principal Investigator

Thomas C. Meserole
Data Processing Programmer

Johann H. Buhr
Assistant Research Engineer

Texas Transportation Institute
Texas A&M University
College Station, Texas

Sponsored by

U.S. Bureau of Public Roads
Contract No. CPR-11-2842
The opinions, findings and conclusions expressed in this publication are those of the authors and not necessarily those of the Bureau of Public Roads.
ABSTRACT

This publication can be considered as consisting of two parts. The accent of the first part is on the simulation of vehicular traffic in general. The reasons for and background of simulation are discussed and the various methods, techniques and procedures commonly used are treated in detail.

The second part of the paper describes a computer program developed for the simulation of a ramp-freeway junction. The computer logic includes lane changing maneuvers on the freeway as well as the entering maneuver. Some of the output of the program are presented as evidence of its capability of representing the merging process realistically.
CONTENTS

INTRODUCTION
   Definition of simulation ........................................ 1
   Why Simulate? .................................................. 1
   The Freeway Merging Process ................................... 3

SIMULATION OF VEHICULAR TRAFFIC
   Monte Carlo Methods ............................................ 5
   The Origin of Simulation ...................................... 7
   Random Numbers ................................................. 8
   Random Sequences Satisfying Desired Distributions ......... 8
   The Method of Inversion ....................................... 9
   Point Distribution Method .................................... 12
   Special Conversion Methods .................................. 15
   Scanning Techniques .......................................... 17
   The Simulation Program ....................................... 19
   Model Calibration ............................................. 25

FREeway MERGING SIMULATION
   Steps in Simulation ........................................... 27
   Basic Description of Operation .............................. 29
   Computer Considerations .................................... 33
   Chaining Technique .......................................... 34
   Logic Organization .......................................... 36
   Normal Flow .................................................. 36
   Lane Changing ................................................. 43
   Ramp Discipline .............................................. 46
   Gap Acceptance Subroutine ................................... 48
   Time-Space Output ............................................ 50

REFERENCES ......................................................... 53
INTRODUCTION

Definition of Simulation

A penny is tossed until it comes down heads. If this happens on the first toss the player receives one dollar from the bank. If heads appears for the first time on the second toss the player receives two dollars; on the third toss four dollars, etc., doubling each time. What should the player pay the bank for the privilege of playing this game if the game is fair?

If one has a coin handy, the simplest way to get some insight into how much a payer should pay is to play say a thousand games and determine the average winnings per game. This seemingly unscientific approach to the Gambler's Ruin or St. Petersburg Paradox as it is also called represents a simple illustration of simulation.

Simulation is essentially a working analogy. It involves the construction of a working model presenting similarity of properties or relationships to the real problem under study. Simulation is a technique which permits the study of a complex traffic system in the laboratory rather than in the field.

In a more general sense, simulation may be defined as a dynamic representation of some part of the real world achieved by building a computer model and moving it through time. The term computer model is used to denote a special kind of formal mathematical model, namely a model which is not intended to be solved analytically but rather to be simulated on an electronic computer. Thus, simulation consists of using a digital or analog computer to trace the time paths; the distinction being that the digital device counts and the analog device measures. This distinction is actually a fundamental one, being essentially the mathematical distinction between the discrete-variable (digital) and the continuous-variable (analog). The differences in capabilities between the digital and analog computers are manifest in the mathematical distinctions between summation and integration, or between difference equations and differential equations.

Why Simulate?

Simulation is resorted to when the systems under considerations cannot be analyzed using direct or formal analytical methods. There are a few additional reasons for simulation, most of which have been found to pertain to the simulation of traffic systems.
(1) The task of laying out and operating a simulation is a good way to systematically gather pertinent data. It makes for a broad education in traffic characteristics and operation.

(2) Simulation of complex traffic operations may provide an indication of which variables are important and how they relate. This may lend to eventual successful analytic formulations.

(3) In some problems information on the probability distribution of the outcome of a process is desired, rather than only means and variances such as obtained in queueing. Where traffic interaction is involved, the Monte Carlo technique is about the only tool which can give the complete distribution.

(4) A simulation can be performed to check an uncertain analytic solution.

(5) Simulation is cheaper than many forms of experiment. Imagine the cost saving in simulating to find the optimum spacing of freeway interchanges.

(6) Simulation gives an intuitive feel for the traffic system being studied, and is therefore instructive.

(7) Simulation gives a control over time. Real time can be compressed and the results of a long amber phase can be observed in a few minutes of computer time. On the other hand, real time can be expanded and run slower than real time so that all the manifestations of the complex interactions of freeway movement can be comprehended.

(8) Simulation is safe. It provides a means for studying the effect of traffic control measures on existing highways. The effect of signals, speed limits, signs, and access control all can be studied in detail without confusing or alarming drivers. Simulation offers the ability to determine in advance the effect of increased traffic flow on existing facilities. Probable congestion points and accident locations can be anticipated and changes in the physical design of the highway can be affected before the need is demonstrated through accident and congestion experience.

The above summarizes some advantages of a simulation model. As a form of model, it should be compared with analysis on the one hand which involves the use of analytical, rigid, and probabilistic models and trial and error on the other, which involves devising some kind of trial solution and then taking it into actual traffic and trying it out. The relative merits of analysis, simulation, and trial and error, can best be discussed with the aid of a table prepared by Goode' (Table 1).

Refering to the table, the traffic problem has of course been attacked in the past with the tools of both analysis and trial. Simulation is actually a combination of both methods, but allows attack on the
most complicated of processes (which analysis does not) and on the other hand does not affect traffic until the solution has been reached (which trial methods do). Simulation, it will be noted in Table 1, is almost always midway between analysis and trial. But as the situation which is being studied becomes more complex (as in the traffic problem), the differences between methods in terms of cost, time, etc., become more pronounced, until finally neither of the extremes can be tolerated and simulation becomes the only feasible method.

<table>
<thead>
<tr>
<th>CRITERION</th>
<th>ANALYSIS</th>
<th>SIMULATION</th>
<th>TRIAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost</td>
<td>least</td>
<td>medium</td>
<td>most</td>
</tr>
<tr>
<td>Time</td>
<td>least</td>
<td>medium</td>
<td>most</td>
</tr>
<tr>
<td>Reproducibility</td>
<td>most</td>
<td>medium</td>
<td>least</td>
</tr>
<tr>
<td>Realism</td>
<td>least</td>
<td>medium</td>
<td>most</td>
</tr>
<tr>
<td>Generality of Results</td>
<td>most</td>
<td>medium</td>
<td>least</td>
</tr>
</tbody>
</table>

Simulation is a powerful tool and like all powerful tools it can be dangerous in the wrong hands. The increased emphasis on simulation studies and the corresponding lack of experience from the part of some people who attempt to apply the method can lead to a sort of pseudo-simulation. Pitfalls exists in simulation as in every human attempt to abstract and idealize. Some rules to following avoiding these pitfalls are (1) no assumption should be made before its effects are clearly defined, (2) no variables should be combined into a working system unless each one is properly explained and its relationships to the other variables are set and understood, and (3) remember that simplification is desirable, but oversimplification can be fatal.

For the most part it can be said that the goals achievable by simulation in the traffic process are clear-cut and offer a profound payoff. Simulation is an ideal technique for traffic research. The simulation model is not just another means for accomplishing what we can do today but is a tool for solving problems which cannot be solved today.

The Freeway Merging Process

One traffic problem that has generally defied analysis as defined
in the previous section is the process of entrance ramp vehicles merging into a freeway stream.

In the summer of 1965, the U.S. Bureau of Public Roads undertook research to furnish detailed criteria on the merging of ramp vehicles into the freeway system. A contract, "Gap Acceptance and Traffic Interaction in the Freeway Merging Process", was awarded the Texas Transportation Institute. The general aim of this research is the conception of a relationship between the many variables associated with the interaction of vehicles traversing a ramp and merging onto a freeway so as to determine the effects of the following on merging operation and level of service:

(1) Traffic characteristics such as gap availability, gap acceptance, speed and volume;
(2) Ramp geometrics such as length, curvature, angle of convergence and grades, and acceleration lane geometrics such as length, shape, delineation and location of lateral obstructions;
(3) System considerations such as interchange type, ramp configurations, frontage roads and upstream or downstream bottlenecks, and environmental elements such as metropolitan area size, location within the city, and lighting;
(4) Control devices such as freeway lane controls, yield or merge-ahead signs, traffic signal feeding the entrance ramp, and ramp merging stations.

The underlying purpose of this research is the application of the above information to the following:

(1) In design and operation - the furnishing of more-detailed information on the effect that geometric variables and traffic characteristics have on merging traffic;

(2) In simulation - the development of usable distributions of traffic variables for simulation programs.

This report deals with the latter application. In the next section of this report, the application of simulation to traffic is discussed very generally, whereas in the last section the application of simulation to freeway merging is treated specifically.
Monte Carlo Methods

Often an equation arises in the formulation of a model which cannot be solved by standard numerical techniques. It may then be more efficient to construct an analogous stochastic model of the problem. Thus, essentially, an experiment is set up to duplicate the features of the problem under study. The calculation process is entirely numerical and is carried out by supplying "random numbers" into the system and obtaining numerical answers.

One of the simplest and most powerful applications of this idea is the evaluation of a multi-dimensional integral. Consider the simplest case of evaluating the area of a bounded area. Surround the area with a square, normalizing to make its side of unit length. Taking a point in the area at random, the probability that it lies in area A is simply A. If a large number of points are taken at random, it follows that the proportion of these lying in the area is an estimate of A (see Figure 1). The general equation for evaluating a definite integral using the "point distribution method", as this method is called, is

\[ \int_{a}^{b} f(x) \, dx = (b-a) \max_{y} \left[ P \left( Y < f(X) \right) \right] \]  \hspace{1cm} (1)

where

\[ X = (b-a) \, RN + a \]  \hspace{1cm} (2)

and

\[ Y = y_{\max} \cdot RN \]  \hspace{1cm} (3)

Thus, RN being a random number between zero and one, the point distribution method consists of choosing two random numbers, the first between a and b and the second between 0 and some number which is greater than or equal to the maximum of f(x) in (a, b), or \( y_{\max} \). Thus if the first is regarded as the abscissa and the second as the ordinate, the probability that the point will fall below the curve is equal to the ratio of the area below the curve to the area of the indicated rectangle (see Figure 1). This provides the basis for obtaining the value of the definite integral using Equation 1, which is of course
NOTE: \( A = \frac{4}{4+6} = 0.4 \) BASED ON 10 PAIRS OF RANDOM NUMBERS

MONTE CARLO EVALUATION OF AN INTEGRAL

FIGURE 1
the area under the curve. This method yields an accuracy proportional to \( n^{1/d} \) where \( n \) is the number of points and \( d \) is the number of dimensions. The whole idea can be generalized to higher dimensions and the result remains true.

The Origin of Simulation

The true origin of simulation lies in the theory of mathematical statistics. In its infancy, the subject of statistics consisted of the collection and display, in numerical and graphical form, of facts and figures from the fields of economics and science. One of the most useful forms of display was the histogram or frequency chart and the transformation of statistics began when it was realized that the occurrence of such diagrams could be explained by invoking the theory of probability.

Since a probability distribution is by its nature, in most cases, composed of an infinite number of items, whereas frequency charts by their nature are composed of a finite number of items, the latter had to be thought of as samples from an underlying theoretical probability distribution. Tocher explains how the problem then raised itself as to how to describe a probability distribution given only a sample from it. Because of the seemingly immense mathematical difficulties associated with this, such steps as were taken required experimental verification to give early workers confidence. Thus was born the sampling experiment. A close approximation to a probability distribution was created, samples were taken, combined and transformed in suitable ways and the resulting frequency chart of sampled values compared with the predictions of theory.

It is not hard to visualize situations arising where some method of sampling is indicated, but where the actual taking of a physical sample is either impossible or too expensive. In such situations, useful information can often be obtained from some type of simulation sampling. Typically, simulated sampling involves replacing the actual universe of items by its theoretical counterpart, a universe described by some assumed probability distribution, and then sampling from this theoretical population by means of random numbers. The advent of automatic digital computers to perform the tedious calculations associated with these sampling experiments has revitalized this as a possible approach to the solution of problems still beyond the reach of analysis. The method of taking such a sample is called simulation; the decision problems which rely heavily on such sampling methods are often referred to by the catch-all label of "Monte Carlo Methods".
Random Numbers

We have seen that random numbers are required in the sampling experiments associated with simulation. Humans are too full of associations to think up truly random numbers - no one would pick three 4s in a row although such a sequence might be part of a random series.

The idea of using tables of random numbers was introduced by Tippett\textsuperscript{3} who constructed a table of 10,400 random digits by taking the terminal digits of entries in a censustable. The RAND Corporation used an electronic roulette wheel to prepare the million-digit book of random number tables (hence the name Monte Carlo). Actually a wide variety of natural phenomena have been used to produce randomness although some controversy exists about the validity of such procedures. Certain philosophers have questioned the randomness of any digits committed to tabular form, regardless of how they were obtained. For practical purposes, these arguments are irrelevant. One is forced to accept any phenomenon as random of which the behavior is not predictable by an obvious deterministic laws and of which the numbers satisfy several standard tests of randomness to insure for example, that each decimal digit occurs with equal frequency without any serial correlation.

From the standpoint of checking out computer programs, there are advantages in having a reproducible sequence of numbers instead of purely random numbers. Programs for digital computers have been written which will output a sequence of numbers that satisfies the various statistical tests of randomness that have been devised. Random numbers such as these, which are generated in a non-random fashion, are called "pseudo-random numbers". In automatic computation, the storage of the large volume of random numbers used becomes a serious problem. It is this storage problem that has led to the abandonment of the use of tables of random numbers in computers. A satisfactory computer program for generating pseudo-random numbers would (1) require little storage space in the computer, (2) be relatively fast in operation, and (3) the sequence of numbers generated must satisfy the tests of randomness. The "power residue method" satisfies these three requirements and is the method most often used today.

Random Sequences Satisfying Desired Distributions

The numbers resulting from the power residue method form a uniformly distributed pseudo-random sequence of numbers, that is,
one in which the probability of a number falling in a given interval is proportional to the width of the interval, and does not depend on the location of the interval. Random numbers so generated may be interpreted as random integers or as random fractions. The latter is more appropriate since, as shall be seen in the following sections, the basic problem in simulation consists in sampling from statistical distributions for which any associated probability must, by definition, be a fraction of unity.

There are two principal methods employed for the conversion of random fractions to "random deviates" satisfying a desired frequency distribution: the inversion method and the point distribution method. The theory behind these two methods and applications of each to common distributions used in traffic simulation will be discussed in some detail.

The Method of Inversion

The distribution of a variable can be described by a density function \( f(t) \) in the continuous case or by a set of frequencies in the discrete case, but both cases can be described by means of the cumulative distribution function \( P(t < T) \) which specifies the probability of obtaining the given value or less from a distribution. Therefore, one possibility consists of the analytic inversion of this cumulative distribution function and the calculation of the value of this function for the value of a selected uniform random fraction.

In symbolical terms, the above means

\[
P(t < T) = \int_{0}^{T} f(t) \, dt \quad (4)
\]

Equation (4) must be solved for the random deviate \( T \) by inversion. Since the left side of the equation is equivalent to a uniformly distributed random fraction between 0 and 1 (see Figure 2), one obtains

\[
R = \int_{0}^{T} f(t) \, dt \quad (5)
\]

where \( R \) is a pseudo-random fraction generated as explained in a preceding section. It should be evident that the success of this method
depends on first being able to integrate the density function \( f(t) \) and secondly, being able to take the inverse of (5) after integration. There are some simple but quite important cases in which this is possible.

Many phenomena characterized by random arrivals, as in traffic situations, may be described by use of the negative exponential distribution, which has a probability density function of the form

\[
f(t) = qe^{-qt}
\]

(6)

where \( t \) is the time between arrivals or headway, usually expressed in seconds, and \( q \) is the rate of arrivals or flow, usually expressed in vehicles per second. Integration yields

\[
P(t < T) = 1 - e^{-qt}
\]

or in more general terms

\[
P(h < t) = 1 - e^{-qt}
\]

(7)

where \( h \) is the headway between successive arrivals and \( t \) is time.

Equations (6) and (7) are shown in Figure 2. If \( R_1 \) is a uniformly distributed random fraction, \( 0 < R_1 < 1 \), generated on the computer as explained, then \( (1-R_1) \) must also be a uniformly distributed random fraction, \( 0 < (1-R_1)^<1 \), and

\[
1 - R_1 = P(h < t)
\]

(8)

since \( P(h < t) \) is also a uniformly distributed random variable, \( 0 < P(h < t) < 1 \). Substituting (8) in (7) and solving gives

\[
R_1 = e^{-qt}
\]

(9)

Solving (9) for \( t \), a random deviate satisfying the negative exponential distribution, is called inversion. This is accomplished by taking the logarithm of each side,
METHOD OF INVERSION ILLUSTRATED FOR THE NEGATIVE EXPONENTIAL DISTRIBUTION

FIGURE 2
\[ \ln R_1 = -qt \]

and the desired random deviate becomes

\[ t = -q \ln R_1 \]

Equation (10) tells us that in order to generate a negative exponential random deviate, one simply takes the negative logarithm of a generated pseudo-random fraction.

Because vehicles possess length, it is impossible to have two arrivals in the same lane at the same instant. If the minimum possible headway between successive arrivals is taken as \( \tau \), then the negative exponential distribution should be shifted by an amount \( \tau \) such that the probability of a headway between successive vehicles of less than \( \tau \) is zero, and the cumulative distribution function becomes

\[ P(h \leq \tau) = 0 \]

\[ P(\tau < h < t) = 1 - e^{-\frac{t-\tau}{\bar{t}-\tau}} \]  \hspace{1cm} (11)

Inversion of (11) gives

\[ t = \tau - (\bar{t}-\tau) \ln R_1 \]

where \( \bar{t} \) is the average headway.

The probability density functions of many useful distributions such as the normal distribution, gamma distribution and beta distribution are difficult to integrate. Moreover, there are cases in which it is possible to integrate the probability density function and represent the cumulative distribution function analytically, but the inversion is either impossible or impractical analytically. In such cases, the point distribution method may be used.

**Point Distribution Method**

Consider a bounded probability function (if it is not bounded, a point \( t_{\text{max}} \) may be chosen in the abscissa which is sufficiently large so that the probability of \( t \) falling to the right of \( t_{\text{max}} \) is negligible) such that
The steps in this method consists of (1) generating two random numbers, \( T_1 \) and \( T_2 \), meeting the conditions

\[
T_1 = (t_{\text{max}} - t_{\text{min}}) R_1 + t_{\text{min}}
\]

and (2) checking to see if \( T_1 \) is the desired random deviate by the test,

\[
f(T_1) < T_2
\]

If (14) is satisfied then \( T_1 \) is accepted as conforming to the desired distribution; if (14) is not satisfied, \( T_1 \) is rejected and the two steps are repeated.

This method will be illustrated using the Erlang distribution, which has considerable conceptual appeal for describing the distribution of intervals between arrivals both in queueing situations in general and for the traffic phenomenon in particular. Since the form of density function is

\[
f(t) = \frac{(aq)^a}{(a-1)!} t^{a-1} e^{-aqt}
\]

where \( a \) is a positive integer, the analytic form of the cumulative distribution can only be obtained by successive integration by parts, making use of the inversion technique impractical. Equation (15) has been plotted in Figure 3 for the case \( a = 2 \). Two random fractions \( R_1 \) and \( R_2 \) are generated and \( T_1 \) and \( T_2 \) calculated using (13) in which \( t_{\text{min}} = 0 \) and \( f(t)_{\text{max}} \) is the mode. Then \( T_1 \) is substituted in (15) and
WHERE

\[ T_1 = t_{\text{max}} \cdot R_1 \]

\[ T_2 = f(t)_{\text{max}} \cdot R_2 \]
the value of \( f(T_1) \) obtained is compared to \( T_2 \) according to (14). It should be apparent from Figure 3 that the number of \( T_1 \)'s accepted in an interval \( \Delta t \) in the abscissa is proportional to the area under the curve in that interval, which is precisely the desired distribution.

Special Conversion Methods

The methods of converting random fractions to random deviates discussed above were of general application. However, a few special theoretical distributions have certain characteristics that lend themselves to other methods of conversion.

One such method of conversion is based on the central-limit concept. The term central-limit theorem is applied to certain theorems which show that the sum of independent variates from any arbitrary distribution is in the limit normally distributed. The rapidity with which this limit is satisfying the normal distribution. For example, one observation from a uniform distribution is naturally uniformly distributed; the sum of two observations from a uniform distribution is distributed according to a triangular distribution; and the sum of three observations from the same uniform distribution is parabolic and very close to the desired normal (see Figure 4). Thus if \( n \) random fractions \( R_i \) are summed, the resulting variate \( t \) is normally distributed,

\[
t = \left( \mu - \frac{n}{\sum_{i=1}^{n} R_i} \right) R_i + \sigma
\]  

(16)

with mean \( \mu = n/2 \) and standard deviation \( \sigma = \sqrt{n/12} \). Half the range between \( t_{min} \) and \( t_{max} \) is obtained by substituting the forms for \( \mu \) and \( \sigma \) in (16) and choosing \( \sum_{i=1}^{n} R_i = 0 \) or \( n \) from the equation

\[
t = \sqrt{3n}
\]  

(17)

In practice, the number of random fractions, \( n \), chosen, will vary from 6 to 12 depending on the range of the phenomenon simulated and the precision desired.

A second special method of converting random fractions to random deviates is based on the theory of convolutions. It has been shown that
DISTRIBUTION OF SUMS OF n OBSERVATIONS FROM A UNIFORM DISTRIBUTION

FIGURE 4
the Erlang distribution, as expressed in (15) represents the a-fold convolution of the negative exponential distribution. This relationship suggests that a random deviate, t, belonging to the Erlang distribution can be generated by a generalization of (6),

\[ t = -q^{-1} \ln (R_1 \cdot R_2 \cdot \ldots R_a) \] (18)

Tocher\(^2\) discusses several special sampling methods for many common probability distributions. His book "The Art of Simulation" is perhaps the best book published on the subject and is highly recommended.

**Scanning Techniques**

Whatever the choice of size and scale for a simulation model, one has some options as to basic structure. There are two extreme possibilities which may be termed the "periodic-scan" and the "event-scan" methods, both of which permit the simulation of randomness in the course of events.\(^4\) In the periodic-scan method, one divides the duration of the simulated phenomenon into a number of successive time intervals displaced by time periods. In the event-scan method, on the other hand, after a given event has occurred, one determines and "stores" a set of "imminent" forthcoming events and times at which they will occur, and selects the earliest. The occurrence of this next significant event may alter the possibility or timing of other events that had been listed, so that a new set of events and times may then be calculated. Thus an event-scan program is essentially asking, "what happens next", whereas the periodic-scan program asks, "What will the situation be one time unit from now?"

The event-scan technique is much faster and can result in an increase of computing speed by a factor of about ten, but usually requires greater program complexity. The periodic-scan technique is very straightforward and is usually much easier to program. In practice, the periodic-scan and event-scan methods may be implemented in many ways and partially combined, in order to produce a program that is artfully suited to the problem.

An important aspect of the periodic-scan concerns the generation of Poisson arrivals. A technique often used consists of comparing a random fraction R to the flow per second of the traffic stream for the scanning period \( \Delta t \), written
Thus for a traffic lane volume of 900 vehicles per hour and a scanning period of 1 second, a random fraction less than .25 will generate an arrival. The proof that arrivals generated in this fashion are Poisson follows from the derivation of the Poisson distribution itself.

Mathematically, the conditions for randomness may be stated as follows:

1. \( P_0(\Delta t) = 1 - q \cdot \Delta t \)  \[ (1) \] (Probability of 0 arrivals at \( \Delta t \))
2. \( 1 - P_0(\Delta t) = q \cdot \Delta t \)  \[ (2) \] (Probability of one arrival at \( \Delta t \))
3. \( P_n(\Delta t) = (q \cdot \Delta t)^n \)  \[ (3) \] (Probability > arrival in \( \Delta t \))

where \( q \) is the rate of flow. Let \( P_n(t) \) be the chance of generating \( n \) arrivals in a time interval \( t \) according to (19). In an interval of length \( t + \Delta t \) \( n \) arrivals may be generated in two ways, either \( n \) arrivals in the interval \( t \) and none at \( \Delta t \), or \( n-1 \) arrivals in \( t \) and 1 arrival in \( \Delta t \). Thus,

\[
P_n(t + \Delta t) = P_n(t) (1 - q \cdot \Delta t) + P_{n-1}(t) q \cdot \Delta t \tag{20}
\]

This may be written in the form

\[
\frac{P_n(t + \Delta t) - P_n(t)}{\Delta t} = q P_{n-1}(t) - q P_n(t), \tag{21}
\]

and then taking the limit as \( \Delta t \to 0 \), one obtains

\[
P_n(t) = q P_{n-1}(t) - q P_n(t) \quad (n = 1, 2, 3, \ldots). \tag{22}
\]

For the case, \( n = 0 \),

\[
P_0(t + \Delta t) = P_0(t) (1 - q \cdot \Delta t) \tag{23}
\]
giving

\[ P_0'(t) = -q P_0(t) \]  \hfill (24)

The solution of this differential equation is

\[ P_0(t) = e^{-qt} \]  \hfill (25)

and by induction it can be shown that the solution of the general equation (22) is

\[ P_n(t) = (qt)^n e^{-qt}/n! \]  \hfill (26)

which is the desired Poisson Distribution.

The Simulation Program

The simulation logic for stepping the vehicles through the system, once the inputs are known, may be divided into three classifications: (1) flow logic for unimpeded vehicles, (2) car-following logic for platooned vehicles, and (3) maneuvering logic for vehicles executing maneuvers involving more than a single stream of traffic. In actuality, all drivers of vehicles within the roadway system are continually and simultaneously making decisions and modifying their behavior. In the course of a simulation, the classification of must vehicles - unimpeded, following or maneuvering - will change many times. The computer, however, can make only one simple logical choice at a time. To control all the occurrences at any given instant, it must process all decisions sequentially. In other words, it must process each decision for every vehicle, for each vehicle in every lane, and for each lane within the system. It must do this in accordance with a prescribed sequence for each instant of time to be considered.

The success of the digital computer lies in the fact that extensive computations can be described as repetitions of cycles. Thus, the computer program consists of blocks of instructions which effect the computations required, each terminated by tests which direct the computer to the appropriate next block. The functional diagram used for describing the simulation problem for the computer programmer is
called the block diagram. Figure 5 depicts a block diagram for illustrating the relationship between the input generation, unimpeded flow logic, car following logic and maneuvering logic.

The simulation of a complex traffic system demands a good understanding of computer programming. After the various traffic manifestations of the problem have been built into the model, a program must be written to enable the computer to carry out the simulation. Ideally, a traffic simulation should represent a cooperative effort between a traffic theorist and a computer technologist. Based on this, a good simulation program should fulfill the following requirements:

1. It must provide an easy inexpensive method of highway simulation.
2. It must be general enough that any highway configuration can be simulated by the input of the proper geometric parameters.
3. The input to such a system must be easily understood and capable of execution by non-computer oriented personnel.
4. It must furnish output which is easily readable and which contains all parameters needed by the traffic engineer for application in the design or modification of highway systems.
5. It must be written in a modular fashion such that any of the moduli can be changed without affecting the rest of the program. (i.e., The car-following process should be completely independent of the input generation process, etc.)
6. It must be written such that new moduli, such as traffic hazards, curves, etc. can be added without extensive programming changes.
7. It must be machine independent, written in one of the higher level languages such as Fortran IV in such a manner that a novice programmer can modify it.

The major components of a simulation program are illustrated in Figure 6. The input data, input generation and program logic have already been discussed. Moving on, there are two basic outputs. One of these normally consists of a one or two page report giving the input geometry and overall performance of the system under study. A second form of output is a time-space diagram which is desirable in "debugging" and evaluating the program visually. Of course any type of plot such as percent gap acceptance, "contour" maps, "profile" maps, etc. can be incorporated.

The internal bookkeeping procedures have not been discussed. These are vitally important to the efficient and successful operation of the program. To implement any practical simulation program the method of bookkeeping needed must keep core storage at a
INPUT

VEHICLE X IN SYSTEM

NO  YES

CONSTRAINED VEHICLE?

YES  NO

CALL IN FREE FLOW LOGIC

AVAILABLE GAP ON LEFT

NO  YES

PASS ON LEFT

AVAILABLE GAP ON RIGHT?

NO  YES

PASS ON RIGHT

CALL IN CAR FOLLOWING LOGIC

X = X + 1

OUTPUT

TRAFFIC STREAM LOGIC

FIGURE 5
START

PROBLEM DEFINITION

MODEL FORMULATION

INPUT

SIMULATION RUN

OUTPUT

EVALUATION

ALL DESIGNS CONSIDERED

CHANGE DESIGN

YES NO

SELECT BEST DESIGN

SIMULATION FOR SYSTEM DESIGN

FIGURE 6
minimum and lend itself to fast sequential processing.

Several procedures can and have been used to represent the flow of traffic within the computer. The memorandum notation uses an entire word to represent a vehicle. Various parts of the word are used for such individual characteristics as its time of entry into the system and its desired velocity. Each vehicle's characteristics are identifiable as it moves through the system, making it possible to compute the delays associated with an individual vehicle. Distance along the roadway, using this method, is quantified using a unit block which is one lane wide and has a length equivalent to some fractional part of the length of an average vehicle. Thus, a vehicle may occupy only a limited number of discrete positions. Each vehicle can be advanced by changing the record to show its position one time unit later (periodic scan). This is done by multiplying the vehicle's speed by the time increment and adding the product to the present position. Thus, vehicles move through the system in much the same manner that players move their pieces through a monopoly game.

In another procedure, the entire roadway system is represented by a three-dimensional mathematical array (Figure 7). The length dimension corresponds to relative position along the roadway; that is, vehicle data is stored in adjoining array elements in the same order as the vehicles occupied by a particular lane. The vertical dimension of the array accommodates all the information characteristics of each particular vehicle, and the width dimension represents the several traffic lanes. In contrast with the memorandum notation, this mathematical notation procedure allows each vehicle to be associated with its own position indicator. Therefore, a vehicle's position is essentially continuous and speed and acceleration are no longer step functions. In addition to the circular array, this procedure utilizes two special registers for each traffic lane—the index position of the lead vehicle and the number of vehicles in the lane.

Sandefur has developed a method of simulation bookkeeping known as "list processing" or "chaining". The assumption made in chaining is that every information set is a matrix C made up of information vectors X. In each of the information vectors there are "pointers" indicating the preceding vector X_{i-1}, and the following vector X_{i+1}. Take for example a (4, 8) information matrix C, made up of the 4 vectors X_{i}, where x_{i, 1} and x_{i, 2} are the chain pointers and x_{i, j} j = 3, 4..., 8 is the information associated with the vector X_{i}. The matrix would appear in the following manner:
VELOCITY
POSITION
TURN
TIME OF ARRIVAL

DIRECTION OF ____.,

TRAFFIC FLOW

DECELERATION RATE
VELOCITY
POSITION
TURN
TIME OF ARRIVAL

DIRECTION OF TRAFFIC FLOW

INDEX OF LEAD VEHICLE
NUMBER OF VEHICLES IN LANE

COMPUTER REPRESENTATION OF TRAFFIC

FIGURE 7
Machine Location | Last | Next
--- | --- | ---
\(\bar{X}_1\) | Loc(A) | \(-\) | \(I_{1,j}\) where \(j = 3, 4, \ldots, 8\)
\(\bar{X}_2\) | Loc(C) | Loc(A) | \(I_{2,j}\) where \(j = 3, 4, \ldots, 8\)
\(\bar{X}_3\) | Loc(Z) | Loc(C) | \(I_{3,j}\) where \(j = 3, 4, \ldots, 8\)
\(\bar{X}_4\) | Loc(B) | Loc(Z) | \(I_{4,j}\) where \(j = 3, 4, \ldots, 8\)

where Loc (A) is the computer location (address) of the vector \(\bar{X}_i\). In the above manner, information vectors can be added to or deleted from a chain without data movement resulting in a decrease in machine time. It also decreases storage requirements because many chains can share the same location pool.

**Model Calibration**

If programmed properly, the realism of the computer output is a function only of the realism of the system model and the inputs to the model. A simulation model is essentially a hypothesis and therefore must be tested before it can be accepted by fact. Such tests include its feasibility, realism and validity.

Since the analyst is generally interested in steady state conditions some attention must be given to the initial state of the system. This may be handled by allowing the simulation to start in an easily identified state and run for a period long enough to achieve steady state conditions. Other approaches include "warm-up periods" or "warm-up sections" to allow the system to become loaded with normal traffic before the desired production runs are made or sampled.

Simulation is of course a sampling technique (it is sometimes even referred to as "simulated sampling"). Although, simulation on a digital computer is fast, it is impractical to attempt to utilize the whole range of each variable since then several thousand simulation runs may be required. A statistical design should be adopted to handle some finite number of "treatment combinations". Replication is extremely desirable once the experimental design is devised. Statistical studies show that the accuracy of simulation increases as the square root of the number of samples. It is better to have the results of 4-fifteen minute runs than a single one-hour run. There are a number of techniques available to reduce the sample size and
attain the same accuracy.

Finally, it is essential that the results of simulation be compared with known real work responses to the same input in order to satisfy the analyst that modeling has been satisfactory. Validation is the process whereby the simulation model is evaluated to determine whether it satisfactorily duplicates real traffic behavior. Since, as mentioned before, it is not the intent in simulation to reproduce all minute details of a real system, it is necessary to establish in the beginning those characteristics of real traffic the model should duplicate to be useful. In other words, what criteria are to be used in the validation.

Theoretically, the model must duplicate the characteristics which the highway engineer uses as a design criteria, or the characteristics that the traffic engineer uses as an operational criterion. As is usually the case however, universally useful design and operational criteria cannot be precisely defined and each design application requires the selection of suitable criteria based on engineering judgement. One might adopt a "microscopic" philosophy in which attempts would be made to duplicate, in the computer, the specific details of field samples of moderate time length. Otherwise, a "macroscopic" approach might be utilized in which computer runs seek to reproduce gross statistical properties of field samples accumulated over long periods of time. In as much as traffic is a stochastic process, valid arguments can be raised in support of either approach.
FREEWAY MERGING SIMULATION

Steps in Simulation

In performing the simulation of a system, a normal sequence of events has evolved. It should be emphasized that the steps are neither sacred nor chronological. These steps indicate phases to be covered in an approximate order:

(1) Definition of the problem specifically, in familiar terms and symbols, with placement of the necessary limitations.

(2) Formulation of the model, including the statement of assumptions, choice of criteria for optimization, and the selection of the operational procedure or rules of the road.

(3) Construction of the block diagram establishing the functional relationship between the components of the system to be simulated.

(4) Determination of inputs for the simulation program.

(5) Preparation of the computer simulation program.

(6) Conducting experimental runs of the simulated system including experimental design to determine the number of runs, the parameter values to be used, and to establish confidence limits.

(7) Evaluation and testing of the simulated system.

The most important step in a computer simulation of a traffic system is the formulation of the model. The computer is important in that it makes solution of the model practical, and the programming merely represents the means of communication between the investigator and the computer. However, it must always be borne in mind that neither the simulation model nor the computer program represent ends in themselves, but are merely means to solving a complex problem regarding the operation of an existing traffic system or the design of a future one.

Simplifying assumptions are important in the formulation of the model for example, although the inclusion of the color of vehicles would add realism to a simulation model, the effect of a vehicle's color on traffic operation has not been documented to the extent that such information could contribute anything to the solution of a practical problem. Thus, in simulating a high type intersection in which vehicle-to-vehicle and vehicle-to-pedestrian conflicts have been eliminated, it might suffice to simulate a single approach. It is not uncommon for the novice to add unnecessary and extraneous components to his simulation.
A second important aspect of the model formulation is the establishment of the basic rules by which the design or operational improvement to be simulated can be measured. This is best accomplished by formulating the simulation model in such a way that the figures of merit are expressed as functions of the variables of the system being studied. Several such measures of effectiveness worthy of consideration are:

1. travel time and speed—^their averages, variances and distributions,
2. percent of vehicles forced to travel at some arbitrary fraction of their desired speeds,
3. acceleration noise in the system,
4. number of lane changes per vehicle-second,
5. average platoon length, and
6. the level of service as described by the energy-momentum concept.

Thus several criteria may be important or it may be desirable to use different criteria at different times. Herein lies one of the advantages of simulation: in analysis we may use only those criteria which are mathematically tractable (for example one may use least squares but not maximum absolute deviation; one may use the mean but not the entire distribution, etc.). In simulation, one may select any criterion and measure it continually if necessary.

Inherent in the formulation of the model are the determination of the significant input and output variables. Inputs may be divided into four categories: geometrics, traffic characteristics, driver policy and vehicle performance.

Important geometric variables are curvature, grade, number of lanes, angle of convergence of ramps, sight distances, length of acceleration lane, etc. Such system considerations as interchange type, ramp configuration, frontage roads and upstream or downstream bottlenecks may be important, as well as the presence of signs, traffic signals and other control devices.

The three fundamental characteristics of traffic movement—speed, volume, and density—define the operational requirements of the traffic stream. The speed of a maneuvering vehicle has a significant effect on a particular maneuver whether it involves crossing, merging, weaving or lane changing. On the one hand, the size of gap required for the maneuver varies directly with the relative speed between the maneuvering vehicle and the stream across which, into which, or through which he desires to maneuver.

The transverse distribution and longitudinal distribution of vehicles on the simulated system may be described either in terms of volume or density. The longitudinal placement of vehicles in the traffic stream affects the driver's choice of speed. The reciprocal of this longitudinal placement yields volume if the measurement is made
in time, the density if distance is the measurement parameter. Equally important to the ability of an individual vehicle to maneuver through the simulated system is the transverse or lane use distribution of vehicles where there is more than one traffic lane in a given direction.

The principal characteristics of traffic concern the abilities, requirements and performance of the driver and vehicle, who together form the discrete unit of the simulation model. The objectives of the driver must be incorporated into the model. It may be to minimize his delay or maximize his safety. To complicate things a driver's policy may not be consistent with the capabilities of his vehicle. For example, a vehicle which gets trapped behind a slow-moving vehicle is forced to reduce its speed. When a passing opportunity presents itself, the accelerating potential of the vehicle becomes yet another significant input to the model.

The power of simulation as a tool for the study of traffic flow lies in the ability to include the effect of the random nature of traffic. We have seen that the number of variables, some associated with the characteristics of the roadway, some associated with the characteristics of the driver, and some associated with the characteristics of the vehicle, are very large for most traffic systems. These variables are expressed as frequency distribution and input into the simulation model using the techniques of the previous sections.

Basic Description of Operation

The simulation program developed to study the freeway merging process can be represented physically by the configuration diagrammed in Figure 8. The study was made as general as possible, with the ability to make changes in the input data to change the physical representation of the freeway, allowing the study to simulate a number of different freeway designs. Some of the variable design factors are:

(1) number of lanes, from one to six
(2) length of simulated freeway segment
(3) number of on-ramps and off-ramps, from none to two
(4) location and lengths of on-ramps and off-ramps
(5) length of each acceleration lane
(6) overall length of each on-ramp
(7) location of the start of a grade relative to the beginning of the freeway segment
(8) the grade itself
Figure 8. Physical Representation of the Freeway Segment
The model performs all logical operations to accomplish safe travel of a vehicle through the freeway segment. The general logical organization is shown in Figure 9.

The study is started by initializing all the parameters in the system including the vehicle chain, which will be discussed later. Vehicles are initially placed in a random array on the freeway segment ensuring that each vehicle on the freeway segment is a safe distance from neighboring vehicles. Then the vehicles in the system are processed, starting with the vehicle which is most distant from the zero reference point regardless of its lane. Vehicles are treated successively in order of decreasing distance, until all vehicles in the system have been updated.

Each vehicle is assigned a number of characteristics: current speed, desired speed, and distance from the zero reference point. There are the characteristics necessary to determine what the vehicle is doing and what it desires to do in the future.

This simulation study employed a periodic scan technique, as discussed earlier, for the processing of vehicles in the system. This technique updates information on each vehicle in the system during each time interval, repeating for each time increment, until the study is completed. The vehicle data are updated by determining which characteristics are to be altered during the time interval and then performing the required alterations. The vehicle information consists of all the characteristics of each vehicle in the system after processing for a time interval.

The normal logic allows a vehicle to be processed along the freeway segment under certain restrictions. If the vehicle being processed can maintain its desired speed, it is simply allowed to proceed down the freeway. If it encounters a slower vehicle in its lane, an attempt is made to weave. The weaving logic will allow a vehicle to attempt to weave left then right, whenever possible. If the weaving attempt is unsuccessful, the vehicle will decelerate. The normal logic also allows a freeway vehicle to exit on an exit ramp from the shoulder lane. If a vehicle is not in the shoulder lane when it approaches its desired exit ramp, it attempts to weave right. If it is unable to weave right, it will decelerate and try to weave during the next time period. If it passes the desired exit ramp before it enters the shoulder lane, it will keep trying to weave right and will exit on any of the following exit ramps. The normal logic also initiates the merging logic to allow a ramp vehicle to enter the shoulder lane from the acceleration lane of an entrance ramp. If a ramp vehicle reaches the end of the acceleration
Weaving Logic

Initialize Parameters and Vehicle chain

Initialize Freeway Segment

Find Vehicle to be Processed

Weaving Logic

Proced Normally?

Exit on Ramp?

Attempt to Merge?

All Freeway Vehicles done?

Update Ramp Vehicles

Study Finished?

Normal Logic

Figure 9. Logic Flow Diagram of the System
lane without merging, it must stop before entering the freeway segment. After the normal logic has processed all freeway vehicles in the system, the ramp operation logic updates the ramp vehicles.

**Computer Considerations**

Several high-level simulation languages such as GPSS and SIMSCRIPT have been developed. These do not however, allow as much freedom in varying parameters as do certain other languages. On the other extreme, a large machine language program takes a great deal of coding effort and a long time to develop. Therefore, a language such as FORTRAN IV which gives good flexibility in varying parameters without an overwhelming amount of coding can be easily employed in simulation work. FORTRAN IV also can be used on most large-scale computers and for these reasons was chosen for this simulation program. A brief discussion of how FORTRAN IV was used to good advantage in this study follows.

On the digital computer, the FORTRAN IV language allows internal storage cells to be addressed singly or as arrays. Each part of an array is given a particular subscript to distinguish it from any other part of that array. Each one of the parts of such an array is individually addressable by using the proper subscript. In this simulation study an array is used for each characteristic assigned to the vehicles in the study. As an example, the array "DIST" contains the distance of a vehicle from the zero reference point. Each vehicle in the system is assigned its own subscript number, allowing no other vehicle in the system to have the same number. Any characteristic associated with a vehicle in the array of that characteristic is addressable in that array with the use of the subscript number of that vehicle. As an example, the current speed of a vehicle which is assigned the subscript "L", where L is any positive integer between 1 and 500, can be determined by addressing the current speed array "CV" as follows: CV(L).

If the simulation is allowed to operate over a long period of time, a considerable quantity of data is collected. In order to conserve storage in the computer, only the characteristics of the vehicles presently in the system are stored in memory. The characteristics of a vehicle which has passed the end of the study section are no longer stored. This is accomplished by reassigning the subscript number of the vehicle using a chaining technique.
Chaining Technique

The chaining technique is a method of logically organizing the relative position of a vehicle to all other vehicles in the system. There is assigned to each vehicle a value in each of two characteristic arrays which contain the subscript numbers of the vehicles immediately ahead of and behind that vehicle and in line with its movement. The arrays of these characteristics are named "LAST" and "NEXT" respectively. This gives the processing program access to the characteristics of the vehicles between which a vehicle being processes is situated. This allows determination of those characteristics of the vehicles which must be changed.

Since each vehicle has characteristics in arrays "LAST" and "NEXT", the processing program has an overall picture of the placement of vehicles in the system. These two arrays can be thought of as being a chain with each link being one subscript of a vehicle. For example a chain array might appear as shown in Figure 10.

<table>
<thead>
<tr>
<th>Subscript Number</th>
<th>'LAST'</th>
<th>'NEXT'</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>8</td>
<td>23</td>
</tr>
<tr>
<td>23</td>
<td>10</td>
<td>49</td>
</tr>
<tr>
<td>49</td>
<td>23</td>
<td>75</td>
</tr>
<tr>
<td>75</td>
<td>49</td>
<td>76</td>
</tr>
</tbody>
</table>

Figure 10 Array Organization of a Chain

A chain from the arrays "NEXT" and "LAST" containing the subscripts of vehicles shows the organization of one lane of traffic. From Figure 10 it can be seen that the lane represented by this chain would physically look like Figure 11.
The system is initialized with vehicles on the freeway segment and then the possibility of vehicles entering the freeway is examined each time interval. At the zero reference point a location called the generation pool is constructed for each lane of the system, always containing one vehicle already assigned its characteristics. When the logic determines that a vehicle will enter the freeway, a new vehicle and its characteristics are generated. The vehicle in the generation pool of that lane enters the freeway, and the newly generated vehicle is placed into the generation pool and remains there until another vehicle is generated. This method of always having one vehicle in each generation pool allows the chain associated with each lane to have an end link at the zero reference point, facilitating processing.

The vehicle most distant from the zero reference point in each lane has a value in the "LAST" array as does every other vehicle in the system. Since this vehicle does not have a vehicle between it and the end of the freeway segment, a fictitious vehicle is placed outside the freeway segment to represent the vehicle ahead of the most distant vehicles. Only one fictitious vehicle is generated for all lanes and is assigned a complete set of characteristics and a unique subscript number. This allows all chains to begin with the same unique vehicle logically terminating the chains. As a vehicle passes through the freeway segment its characteristics are stored, but once it passes off the end of the freeway segment its characteristics are no longer saved and its subscript number is removed from the chain of that lane. The vehicle directly behind the vehicle just processed off in the same lane considers the fictitious vehicle as the vehicle now ahead of it.

As a vehicle is introduced into a generation pool, it is assigned a subscript number. To prevent it from accidentally acquiring the subscript of some vehicle already in the system, the new vehicle obtains its subscript number from a free links pool. Initially one chain is formed which contains all the subscript numbers available for the system linked together using the "NEXT" and "LAST" arrays. When the system is initially set up, the subscript numbers of the vehicle not placed on the freeway segment form the free links pool. This pool subsequently contains every available subscript number not being used in the system. As a vehicle passes off the end of the freeway segment
its subscript number is placed in the free links pool, and when a vehicle enters a generation pool it obtains its subscript number from the free links pool.

The purpose of the chaining logic is to permit an organized handling of vehicle characteristics while using a method which is easily adapted for digital computation. In the digital computer this process reduces the number of internal storage locations required. If this process were not used and specific locations were required for the characteristics of each vehicle, either on or off of the system, the number of vehicles passing through the system would be more restricted. Using the chaining technique, the restriction imposed on the system is that not more than 500 vehicles be on the freeway segment at one time.

Logic Organization

The program is organized into independent logical divisions with one monitor division to direct the control among the other divisions. By using separate divisions, experimentation can be carried out in one division without changing any other logic. The monitor division incorporates the tasks of initializing all parameters including the chains, initializing the freeway segment for the first time period, and handling for the normal or through flow of traffic. This section determines which of the vehicles on the freeway segment is the most distant non-processed vehicle and perform a series of tests on it. The tests are in the form of the following questions:

1. Is this vehicle going to travel past the end of the freeway segment during the next time period?
2. Does this vehicle exit on a ramp?
3. Does a vehicle merge in front of this vehicle from the acceleration lane of an entrance ramp?
4. If this vehicle is not in the shoulder lane and desires to exit soon, can it weave into the shoulder lane?
5. Is this vehicle traveling as fast as it desires?

Once the monitor division has operated on all freeway vehicles and has updated them, the ramp vehicles are updated.

Normal Flow

The normal flow section determines the behavior of vehicles in the same lane and is divided up into two segments; the distance section and the desired speed section. A flow diagram of the distance
The distance logic stems from the fact that certain points on the freeway segment are of particular importance in the processing of vehicles. These are the end of the freeway segment, the beginning of the exit ramps, the exit decision stations, and the area next to the acceleration lanes. The number of these points and their distances from the zero reference point can be varied to give the scheme flexibility. This section determines whether a vehicle has passed one of the important points during a given time period by calculating a tentative distance \( D_{t,i} \).

\[
D_{t,i} = D_{o,i} + (S_{c,i})T
\]

(27)

\( D_{t,i} \) represents the position of vehicle \( i \) if no speed changes occur. If this tentative distance is greater than the distance to one of the important points, and the original distance \( D_{o,i} \) of vehicle \( i \) is less, vehicle \( i \) has passed this point for the first time. Depending on which important point was passed, different algorithms are employed. If a vehicle has passed other significant points, other processing is initiated.

Vehicle \( i \) has passed the end of the freeway segment if its \( D_{t,i} \) is greater than the distance to the end of the freeway segment. A sequence is then initiated to remove \( i \) from the chain of that lane and place it into the free links pool. The exit decision stations are points before reaching the exit ramps which give a vehicle desiring to exit time to weave into the shoulder lane. After a vehicle has passed a station, it will attempt to weave right into the shoulder lane, and continue to attempt to weave until it is successful, even after it has passed the desired ramp. When a vehicle has been determined to be in the shoulder lane, the desires of the vehicle to exit are examined. If this vehicle desires to exit, its \( D_{t,i} \) and \( D_{o,i} \) are compared with the distance to each exit ramp on the freeway segment to see if this vehicle is now next to one of the exit ramps. If vehicle \( i \) is now next to an exit ramp and this is the desired ramp, it will be processed into this ramp.

The distance logic is based on the assumption that there is no change in the speed of a vehicle during a time interval. This is, of course, not necessarily true. After the position of a vehicle of the freeway segment has been compared with all the important points and this vehicle does not leave the freeway, a closer look is taken to see if
Figure 12. Distance Logic Flow Diagram

Part I
Desired Speed Logic
its speed must be changed.

The desired speed logic is broken down into three parts based on the relationship between the vehicle being processed and the vehicle ahead of it in the same lane. The following distance and the acceptable following distance are calculated for a pair of vehicles, such as, i and j in Figure 13 in the Equations (28) and (29) respectively.

\[ D_{f,ij} = D_{o,j} - D_{o,i} - V_{s,i} \]  
(28)

\[ D_{a,ij} = V_{s,i} + S_{c,i} + 0.5(S_{c,i} - S_{c,j})^2 + D_{d,ij}(S_{c,i}) K \]  
(29)

where \( K = 0 \) if \( S_{c,i} < S_{c,j} \)

and \( K = 1 \) if \( S_{c,i} > S_{c,j} \)

![Figure 13 Position of Vehicles i and j on the Freeway](image)

The need for different values of \( K \) can be seen by the fact that if \( i \) is slower than \( j \), \( i \) can be closer to \( j \) and not be in any danger of running into it. A flow diagram of the desired speed logic is shown in Figure 14 and Figure 15.

First the following distance between vehicle \( i \) and vehicle \( j \) is compared with the calculated value of the acceptable following distance. The first part of the desired speed logic deals with the situation where \( D_{f,ij} \) is less than \( D_{a,ij} \). If vehicle \( i \) is moving faster than vehicle \( j \), \( i \) will not have enough time to weave but will decelerate by either
Figure 14. Desired Speed Logic Flow Diagram - Part I
Figure 15. Desired Speed Logic Flow Diagram - Part II
B_{x,i} or B_{b,i} depending on how close i and j are. If i is slower than j, the distance between them is increasing and the relationship between the two is becoming safer as time progresses and no speed changes occur.

The second part considers the case where the following distance is equal to the acceptable following distance and can be broken down into two divisions. The first division deals with the speed of vehicle i being greater than the speed of vehicle j. If vehicle i can slow down in one time increment to the speed of vehicle j, i and j will be a minimum safe distance apart and have the same speed. If i cannot decelerate fast enough, i will decelerate by the amount B_{b,i}. The second division of this part is for the logic where S_{c,i} is less than S_{c,j}. Vehicle i is accelerated by A_{x,i} insuring that if its speed is slower than it desires, an attempted weave is made.

The last part of the logic is concerned with D_{a,ij} being less than D_{f,ij} which is the most desirable condition, since vehicle i is a safe distance away from vehicle j. A look is now taken at the present situation to see if i will come too close to j during the present time period if its speed remains the same. A new tentative following distance is calculated from

\[ D_{u,ij} = D_{o,j} - D_{o,i} - (S_{c,i})T - V_{s,i} \]  

If the same D_{a,ij} is greater than D_{u,ij}, a hazardous condition will exist if the speed of i is not changed; therefore vehicle i is decelerated by the amount B_{b,i}. If D_{a,ij} is then less than the following distance, the distance between i and j is still a safe one, but there is a matter to be considered of processing the vehicle to eliminate erratic movement. When speed is decreased only in increments of B_{x,i} and B_{b,i}, vehicle i appears to move down the freeway in a "jerky" manner. This cannot be avoided in all cases, but it can in non-hazardous situations. The idea of closing the distance between vehicles i and j gradually, is used to give a smoothing deceleration to vehicle i as it closes in on vehicle j and is expressed by

\[ B_{s,i} = \frac{\left[ S_{c,j} - S_{c,i} \right]^2}{\left[ D_{u,ij} - D_{a,ij} \right]^2} \]
This gives the deceleration that i must take for it to be moving at the same velocity as j when the following distance between them equals the acceptable following distance. If this smoothing deceleration is greater than $B_{n, i}$, vehicle i will try to weave. If vehicle i cannot weave, its velocity is decreased by $B_{s, i}$. If on the other hand $B_{s, i}$ is less than $B_{n, i}$ the possibility of accelerating vehicle i is explored and a new following distance is calculated, assuming $S_{c, i}$ and $S_{c, j}$ constant during the following time period.

$$D_{u, ij} = D_{o, j} + (S_{c, j})T - D_{o, i} - 2(S_{c, i})T - V_{i}$$  \hspace{1cm} (32)$$

If the acceptable following distance is greater than this new following distance, i will decelerate by the amount $B_{s, i}$ or will attempt to weave to another lane. If the $D_{a, ij}$ is still less than $D_{u, ij}$ then i is accelerated by the amount $A_{s, i}$ where

$$A_{s, i} = A_{x, i} \left( S_{d, i}/S_{c, i} \right) - A_{x, i}$$  \hspace{1cm} (33)$$

Lane Changing

When it is determined that a vehicle desires to weave into another lane, the possibility of weaving left is explored and then right. If a vehicle is on either the shoulder lane or the innermost lane, only one weave is attempted. If all weave attempts are unsuccessful, the vehicle remains in its present lane. If a weave is successful, the weaving vehicle is removed from its old lane to the new lane by rearranging the appropriate chains. After a lane has been chosen into which an attempted weave will be made, the current speed of the weaving vehicle is increased by the amount $A_{x, i}$. This new speed is called the weaving speed and is designated $S_{w, i}$. The vehicle will weave if it will not come hazardously close to the vehicles ahead or behind it in the new lane. If the gap is inadequate the weave attempt is unsuccessful. The weaving logic is diagrammed in Figure 16.

The first condition for a successful weave has three parts to it. First the acceptable following distance is calculated as well as a following distance as expressed in Equations 34 and 35 respectively for vehicles i and j in Figure 17.
Figure 16. Lane-to-Lane Weaving Logic Flow Diagram
\[
D_{a,ij} = V_{s,i} + \left[ S_{w,i} + D_{d,ij} (S_{w,i}) \right] K
\]

where \( K = 0 \) if \( S_{w,i} < S_{c,j} \)

and \( K = 1 \) if \( S_{w,i} > S_{c,j} \)

\[
D_{u,ij} = D_{o,j} - D_{o,i} - V_{s,i} - (S_{w,i})T
\]

The first part deals with \( D_{a,ij} \) being greater than \( D_{u,ij} \). There is more than a safe distance between the two vehicles and only the third part of condition one is examined. If \( D_{u,ij} \) is the greater of the two, both the remaining parts are examined. The second part of condition one tries to alter the speed of vehicle \( i \) such that a safe distance will then exist between the two vehicles. If the speed can be altered properly the third part is examined otherwise the logic is terminated. The third part of the condition requires that the weaving vehicle will be weaving into a faster lane. This is done by calculating a smoothing deceleration in Equation 36 to test whether vehicle \( i \) is closing in on vehicle \( j \) too quickly. If it is, this indicates that the new lane is slower and no weave is made.

\[
B_{s,i} = \frac{\left[ S_{c,j} - S_{w,i} \right]^2}{2 \left[ D_{o,j} - D_{o,i} - (S_{w,i})T \right]}
\]

The second condition is concerned with the relative speeds of vehicles \( i \) and \( k \) as shown in Figure 17 and is broken into two distinct parts. The first part deals with the case where vehicle \( i \) is traveling faster than vehicle \( k \). If there is at least a distance of \( \frac{1}{2} V_{s,i} \) between them, the second condition is fully met and the weave is successful.
If the distance is smaller the weave is unsuccessful. The other part of the condition is the case of $S_{c,i}$ is less than $S_{c,k}$ and it is dependent on lag time. Lag time is defined as the time required by vehicle $K$ to come within the minimum safe distance of $i$ as given by

$$L_{c,i} = \frac{[D_{o,i} - D_{o,k} - V_{s,i} - D_{a,ki}]}{[S_{w,i} - S_{c,k}]}$$

(37)

This is compared with the minimum acceptable lag time, $L_{a,i}$, and if it is less, the weave is successful; otherwise it is not.

If a weaving attempt is successful the arrangement of the chains is changed. If vehicle $i$ weaves from a position between vehicles $k$ and $j$ to one between vehicles $m$ and $n$, the chains are changed as indicated in Figure 18.

Ramp Discipline

The behavior of vehicles on an entrance ramp is treated in two parts; the movement of vehicles on the ramp and the merging of vehicles onto the freeway segment from the ramp.

The movement of vehicles on a ramp is handled similarly to the vehicles on the freeway segment, but the ramp vehicles are not allowed to weave while on the ramp. The ramp vehicle closest to the end of the ramp is processed as a special case because this vehicle must stop at the end if it is unable to merge. This vehicle is gradually decelerated as it comes within a small distance of the end of the ramp. Once this vehicle reaches the end its speed is set to zero. The remaining ramp vehicles are processed similarly to the freeway vehicles by calculating an acceptable following distance, given by

$$D_{a,ij} = V_{s,i}/4 + \left[\frac{S_{c,i}/2 + V_{s,i}/4}{K}\right]$$

(38)

where $K = 0$ if $S_{c,i} < A_{b,i}$

and $K = 1$ if $S_{c,i} > A_{b,i}$
### Arrangement of the Chains

<table>
<thead>
<tr>
<th>Old Lane</th>
<th>New Lane</th>
<th>Old Lane</th>
<th>New Lane</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle</td>
<td>NEXT</td>
<td>LAST</td>
<td>Vehicle</td>
</tr>
<tr>
<td>h</td>
<td>g</td>
<td>i</td>
<td>m</td>
</tr>
<tr>
<td>i</td>
<td>h</td>
<td>j</td>
<td>n</td>
</tr>
</tbody>
</table>

### Position of Vehicles on the Freeway Segment

<table>
<thead>
<tr>
<th>Old Lane</th>
<th>New Lane</th>
<th>Old Lane</th>
<th>New Lane</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle</td>
<td>NEXT</td>
<td>LAST</td>
<td>Vehicle</td>
</tr>
<tr>
<td>g</td>
<td>h</td>
<td>i</td>
<td>j</td>
</tr>
<tr>
<td>l</td>
<td>m</td>
<td>n</td>
<td>o</td>
</tr>
</tbody>
</table>

Figure 18. Arrangement of a Chain on a Weave
This acceptable following distance is shorter than the ones for the through vehicles, since it is known that ramp vehicles crowd closer together when attempting to merge. The procedure determines whether a vehicle will accelerate or decelerate by comparing \( D_{a,ij} \) with its following distance during the last time period, the present time period, and the next time period. Again smooth processing is the key criterion and when a vehicle approaches another vehicle it will do so gradually.

Gap Acceptance Subroutine

The merging subroutine processes vehicles from the acceleration lane onto the shoulder lane of the freeway segment. A successful weave will occur if the following conditions are met: a gap is present on the shoulder lane into which a ramp vehicle can merge and merging vehicles will not come hazardously close to the vehicles ahead of and behind it on the freeway segment.

The merging logic is diagrammed in Figure 19. After a ramp vehicle is chosen to attempt to merge into a given gap, the merging logic examines the relationship between the merging vehicle \( i \) and the vehicle \( l \) ahead of it on the freeway segment. Normally a ramp vehicle will accelerate when merging onto a freeway; therefore the merging speed \( S_{m,i} \) is defined to be the sum of \( S_{c,i} \) and \( A_{x,i} \). A following distance and an acceptable following distance are calculated for vehicles \( i \) and \( l \) as expressed in Equations 39 and 40 respectively.

\[
D_{f,il} = D_{o,l} - D_{o,i} + S_{m,i}(T) - V_{s,l}
\]  

\[
D_{a,il} = V_{s,i} + S_{m,i} + D_{d,il} (S_{m,i}) - \\
\left[ 0.5(V_{s,i}) + S_{n,i} + (D_{d,il})S_{m,i} \right] K
\]

where \( K = 0 \) if \( S_{m,i} > S_{c,l} \)

and \( K = 1 \) if \( S_{m,i} < S_{c,l} \)

One part of the second condition is the case where \( D_{a,il} \) is greater
Figure 19. Merging Logic Flow Diagram
than $D_{f_{i1}}$. This part satisfies the entire second condition and the third condition is examined. The other part of the second condition is where $D_{a_{i1}}$ is the smaller than $D_{f_{i1}}$. If vehicle 1 is faster than vehicle i, a merge is abandoned, but if vehicle i is faster, an attempt is made to decelerate vehicle i to make the distance between the two vehicles a safe one. If vehicle i can decelerate enough this condition is met and the next condition is examined.

The third condition is dependent on the speed of the following vehicle $k$ and the vehicle $i$. If vehicle $i$ is faster than vehicle $k$ and at least $V_{s_{i1}}$ away from it, the merge is successful. If vehicle $k$ is faster, a gap time is calculated which is the time it takes the vehicle $k$ to crowd vehicle $i$ excessively. This gap is given by

$$G_{c_{1}} = \frac{D_{o_{i1}} - D_{o_{k1}} - V_{s_{i1}} - S_{c_{k1}}(T) - D_{d_{ki}}(S_{c_{k1}}) - V_{s_{k1}}}{S_{c_{k1}} - S_{c_{i1}}}$$

If $G_{c_{1}}$ is less than the maximum crowding gap time vehicle 1 will accept, $G_{c_{1}}$, the merge is successful. If a merge is successful, the merging vehicle is updated, taken off the ramp, and placed onto the shoulder lane by manipulating the "NEXT" and "LAST" arrays as was done for the weaving logic.

There were several approaches investigated as to which ramp vehicle will be allowed to merge. The first was to allow only the ramp vehicle closest to the end of the acceleration lane to merge. This was not realistic, but it did prove the general merging logic to be feasible. This was expanded to allow each vehicle on the acceleration lane a chance to merge into the gap presented to the merging logic. The logic now will attempt to merge each vehicle on the acceleration lane into a gap in succession starting with the vehicle closest to the end of the acceleration lane. Even if one vehicle does merge into a gap, the remaining ramp vehicles behind it still have an opportunity to merge into the gap.

**Time-Space Output**

The normal output of a simulation system is a table of average values of quantities such as travel times and volumes processed. This does give an overall view of the operation of a system, but it does not give any information about individual vehicle movement while a number of small mistakes can be present in the logic without being detected.
A table of values indicating vehicle movement is cumbersome for a large study and, at best, is difficult to interpret. A more complete picture of the operation was developed as part of this research, displaying the movement of vehicles on the freeway segment in a graphical form by means of a time-space diagram.

The abscissa of the time-space diagram represents time and the ordinate represents distance from the zero reference point while one continuous line represents the movement of an individual vehicle along the freeway segment. This plot gives the position of each vehicle on the freeway segment for the entire length of the study plus a picture of relative speeds and positions. Different lanes and the ramp movement can be plotted using continuous lines and dashed lines or different colored lines to distinguish among them. A plot with a large number of lanes can become quite cluttered and will lose its effectiveness. Each curve of a plot is made up of a number of points with a line drawn between them, where each point represents the position and time of a vehicle in the system. Since the digital computer program makes a periodic scan, it decides when data are to be saved and writes these items on magnetic tape, one record per block, indicating the name of the vehicle, its position on the freeway segment, and the time. After the study is complete this tape is sorted so that the data on the tape are in a proper form to plot the time space diagrams. An example of the time-space output is shown in Figure 20 and serves to demonstrate the realism of the simulation model. Note that vehicles are clustered in queues moving down the freeway segment at the same speed. Note also that vehicles are weaving into and out of the shoulder lane. Vehicles move down the entrance ramp in an orderly fashion at a slower speed than the through lane vehicles. The ramp vehicles start to merge as soon as they reach the beginning of the acceleration lane, merging anywhere along the acceleration lane. If there is no opportunity for a vehicle to merge before it reaches the end of the ramp, it stops, waiting for an acceptable gap. All these features of Figure 20 show that the merging model and the overall model present a true representation of the movement of vehicles on a freeway.
As a Dashed Line Meets a Straight Line a Merge Occurs

Ramp Vehicle Stopped at the End of the Acceleration Lane when Merging

Freeway Queue

Vehicle Weaving out of Shoulder Lane

Vehicle Weaving into Shoulder Lane

--- Vehicles on the Ramp
---- Vehicles on the Freeway

Figure 20. Time Space Diagram
REFERENCES


