Mathl. Comput. Modelling Vol. 16, No. 5, pp. 107–120, 1992
Printed in Great Britain.
0895-7177/92 \$5.00 + 0.00

EFFICIENT TRAFFIC FLOW SIMULATION COMPUTATIONS

A. Chronopoulos
Department of Computer Sciences
P. Michalopoulos and J. Donohoe
Department of Civil Engineering
University of Minnesota-Twin Cities
200 Union Street S.E., Minneapolis, MN 55455, U.S.A.

(Received October 1991)

Abstract—Explicit numerical methods for solving macroscopic traffic flow continuum models have been studied and efficiently implemented in traffic simulation codes in the past. We studied and implemented implicit numerical methods for solving the flow conservation traffic model. We then wrote an experimental code in C simulating a freeway (un)congested pipeline and freeway entry/exit traffic flow. Tests with real data collected from the I-35 W freeway in Minneapolis were conducted on a workstation computer. The implicit methods gave the same (and in some cases better) accuracy as the Lax method. The implicit methods were (more than twice) faster than the Lax method.

1. INTRODUCTION

Macroscopic continuum traffic models flow based on traffic density, flow and velocity have been proposed and analyzed in the past. Examples include Lighthill and Whitham's flow conservation and Payne's momentum conservation models [1,2]. These models involve partial differential equations (PDEs) defined on appropriate domains with suitable boundary conditions which describe various traffic phenomena and road geometries.

The improvement of computational efficiency in the continuum traffic models has been the focal point in the development of traffic simulation programs. It is understood that the computer execution time to solve traffic flow problems depends not only on the size of the freeway and the complexity of roadway geometries, but also on the model equations and numerical schemes used in their discretization.

Explicit numerical methods (for example Lax, Upwind) have been used by Michalopoulos and Lin and Leo and Pretty to compute the solution of traffic flow continuum models [3,4]. In these explicit schemes the space and time mesh sizes are restricted both by accuracy and numerical stability requirements. In order to reduce the computer execution time and maintain good accuracy, the total number of computations must be reduced. This can be achieved by using larger values of time and space mesh sizes. Implicit numerical methods provide the same accuracy as explicit methods and allow changes in the mesh sizes while maintaining numerical stability [5].

In this work, we use implicit numerical methods (Backward Euler, Trapezoid) to solve more efficiently the flow conservation model. We wrote an experimental code in C simulating a freeway (un)congested pipeline and freeway entry/exit traffic flow. Tests with real data collected from the I-35 W freeway in Minneapolis were conducted. These data have been collected by the Department of Civil Engineering at the University of Minnesota and the Minnesota Department

This work was supported by the Univ. of Minnesota Transporation Center and the Army High Performance Computing Research Center. We would like to acknowledge Dr. Eil Kwon (postdoctoral researcher in Civil Engineering at the University of Minnesota) and Ping Yi (graduate student in Civil Engineering at the University of Minnesota) for their comments and useful discussions on this subject.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

of Transportation. Using these data, we tested (for accuracy and efficiency) the implicit methods against the Lax method on a Sun Sparc1 workstation computer. The implicit methods yielded the same (or better accuracy) as the Lax method and they were (more than twice) faster than the Lax method.

The outline of this article is as follows. In Section 2, we review the flow conservation continuum traffic model. In Section 3, we review the Euler implicit, Trapezoidal and Lax methods. In Section 4, we describe the various theoretical and empirical curves relating the traffic flow and density. In Section 5, we describe the congested/uncongested and entry/exit freeway models. In Section 6, we present the numerical results. Section 7 contains concluding remarks.

2. A CONTINUUM MODEL OF TRAFFIC FLOW

The following conservation equation has been proposed by Lighthill and Whitham [1] as a continuum traffic model:

$$\frac{\partial k}{\partial t} + \frac{\partial q}{\partial x} = g(x,t), \tag{1}$$

where $k(x,t)$ and $q(x,t)$ are the traffic density and flow, respectively, at the space-time point (x,t). The traffic flow, density and speed are related by the equation:

$$q = k\,u, \tag{2}$$

where the equilibrium speed $u(x,t) = u(k)$ must be provided by a theoretical or empirical u-k model. For the Greenshields u-k model

$$u(k) = u_f\left(1 - \frac{k}{k_0}\right), \tag{3}$$

where u_f is the free flow speed and k_0 the jam density [6]. The generation term $g(x,t)$ represents the number of cars entering or leaving the traffic flow in a freeway with entries/exits.

3. NUMERICAL METHODS

We consider one explicit method (Lax) and two implicit methods (Euler implicit and Trapezoidal) which are used in computational fluid dynamics [5]. For each traffic model the road section (the space dimension) is discretized using uniform mesh for all numerical methods; only the time stepsizes differ between methods. We use the following notation:

$\Delta t =$ time stepsize.
$\Delta x =$ space stepsize.
$k_j^n =$ density (vehicles/mile/lane) at space node $j\,\Delta x$ and at time $n\,\Delta t$.
$q_j^n =$ flow (vehicles/hour/lane) at space node $j\,\Delta x$ and at time $n\,\Delta t$.

3.1. Lax Method

The Lax method is an explicit method. The new density value k_j^{n+1} is computed directly from the density and flow at the preceding time step n:

$$k_j^{n+1} = \frac{k_{j+1}^n + k_{j-1}^n}{2} - \frac{\Delta t}{\Delta x}\,\frac{q_{j+1}^n - q_{j-1}^n}{2} + \frac{\Delta t}{2}\left(g_{j+1}^n - g_{j-1}^n\right). \tag{4}$$

The method is of first order accuracy with respect to Δt, i.e., the error is $O(\Delta t)$. To maintain numerical stability, time and space stepsizes must satisfy the CFL condition $\frac{\Delta x}{\Delta t} > u_f$, where u_f is the free flow speed. For example, in the KRONOS traffic simulation code (using Lax) $\Delta x = 100$ feet and $\Delta t = 1$ sec are recommended.

3.2. Euler Implicit

The Euler implicit method applied to the nonlinear PDE (1) generates a nonlinear recursion involving all space nodes at each time step. To solve numerically this recursion, Beam and Warming have suggested using one Newton linearization steps [5]. Each Newton step constructs a tridiagonal linear system with unknowns $\Delta k_j = k_j^{n+1} - k_j^n$.

$$-\frac{\Delta t}{2\Delta x}\left(\frac{dq}{dk}\right)_{j-1}^{n}\Delta k_{j-1} + \Delta k_j + \frac{\Delta t}{2\Delta x}\left(\frac{dq}{dk}\right)_{j+1}^{n}\Delta k_{j+1} = -\frac{\Delta t}{2\Delta x}\left(q_{j+1}^n - q_{j-1}^n\right) + \frac{\Delta t}{2}\left(g_{j+1}^n - g_{j-1}^n\right).$$

This tridiagonal linear system is solved by a variant of the Gaussian elimination called the Thomas algorithm. The solution is then advanced to the next time step simultaneously at all space nodes by computing $k_j^{n+1} = k_j^n + \Delta k_j$. This method is of first order accuracy with respect to Δt and it is unconditionally stable.

Artificial smoothing is often added to reduce oscillatory behavior in the numerical solution. This is achieved by adding a fourth order damping term d_j to each term k_j

$$d_j = -\frac{\omega}{8}\left(k_{j-2} - 4\,k_{j-1} + 6\,k_j - 4\,k_{j+1} + k_{j+2}\right).$$

We have tested several damping coefficients from $\omega = 0$ (no damping) to $\omega = 1$. The choice $\omega = 1$ gave the best results.

3.3. Implicit Trapezoidal Method

The Trapezoidal method is identical to the Euler implicit method except for the constants used in the tridiagonal linear system equations.

$$-\frac{\Delta t}{4\Delta x}\left(\frac{dq}{dk}\right)_{j-1}^{n}\Delta k_{j-1} + \Delta k_j + \frac{\Delta t}{4\,\Delta x}\left(\frac{dq}{dk}\right)_{j+1}^{n}\Delta k_{j+1} = -\frac{\Delta t}{2\Delta x}\left(q_{j+1}^n - q_{j-1}^n\right) + \frac{\Delta t}{2}\left(g_{j+1}^n - g_{j-1}^n\right).$$

It is of second order accuracy with respect to Δt and unconditionally stable. However, for discontinuous problems Euler Implicit may yield more accurate results. As with the Euler method, explicit damping is added at each time step. We note that the Trapezoidal and the Euler implicit methods require also the computation of the Jacobian $\frac{dq}{dk}$. It is clear that these methods involve more computations per time step than Lax. However, they allow much larger stepsizes which may make them overall faster than Lax.

4. FLOW RATE-FLOW DENSITY (q-k) MODEL CURVES

A u-k model curve (see (2) and (3) above) is an indispensable part of the flow conservation model. This relation can be used to express the flow rate as a function of the flow density, i.e., $q = q(k)$. This function is a nonlinear function which must satisfy some general requirements. The equations that define the q-k curve are used in the programs to convert from density to flow and from flow to density.

These general requirements on the q-k curve can be derived from the following observations on traffic flow modeling [7].

- For uncongested flow an increase in density corresponds to an increase in flow, up to a critical density k_c, where the flow becomes congested.
- Maximum flow occurs at the critical density: $q_{max} = q(k_c)$.
- For congested flow an increase in density corresponds to a decrease in flow, up to the jam density k_{jam}, where flow stops.

A q-k model curve must also be adapted to characteristics of the freeway section which it represents. Theoretical q-k model curves can not be adapted to the special roadway characteristics and so such a model function must be constructed from empirical data. Greenshields q-k curve is derived from Equations (2) and (3) and appropriate choices for the free flow speed u_f and jam density $k_{jam} = k_0$. In our applications, we chose u_f =(60 miles/hour) and k_0 =(180 vehicles/mile). The Greenshields curve has the basic features described above but can not be tuned to local characteristics of a freeway section. However, we used Greenshield's q-k curve in the initial development of the programs and as a baseline for comparisons.

4.1. Experimental q-k Model Curves

Field data for constructing the q-k model curve (see Table 1) were collected in I-35 W in Minneapolis. With these discrete data a piecewise linear q-k curve was derived [8]. Such a curve must have parameter ranges reflecting the road characteristics of the freeway section it represents [9]. With our discrete data the experimental q-k curve must have following parameter ranges:

- The critical density k_c should be about 70 to 75 vehicles/mile/lane.
- The maximum flow $q_{\max}$ should be less than 2500 to 2700 vehicles/hour/lane.
- The slope of the curve at $k = 0$, which represents the free-flow speed u_f, should be approximately 65 to 75 miles/hour.

We have used several curve fitting methods to construct continuous q-k curves from the set of (k, q) discrete data points available. Our objective was to find a general method that produces a curve which is based on the discrete data, has the basic features of a q-k curve, has the parameter ranges (described above), and also works well in the numerical methods for solving (1). We used three different methods **piecewise linear**, **cubic spline**, and **least squares** to approximate q-k curves from field data.

The simplest method consists of connecting the q-k data points with straight line segments, yielding a **piecewise linear** q-k curve. This is a continuous curve that passes through all data points but the slope of the curve (which is used in the implicit methods) is discontinuous at the line segments intersections.

In an effort to find a curve that interpolated all of the q-k data points and that also had a continuous first derivative, we constructed a **cubic spline**. The cubic spline is a collection of third-degree polynomials, one polynomial for each interval between q-k data points. We tested both clamped (slope at endpoints is specified) and natural (slope at endpoints is unspecified) splines and found that for our field data set the splines were nearly identical. All cubic spline programs used the natural cubic spline.

Finally, several **least squares** approximations were tried. In this method the data points (k_i, q_i) are used to construct a rectangular matrix with row i composed of powers of k_i and a right-hand-side vector containing the q_i. Then the matrix is reduced using the singular value decomposition method (SVD) available in the LINPACK package or the Matlab package [10]. The reduced matrix is then used to find the coefficients of the curve that minimizes the total squared error between the data points and the curve. This method will produce curves of any degree up to the number of data points. Quadratic, cubic and quartic least-squares polynomial curves were found using the Matlab's (SVD).

The quartic curve

$$q = -1.7156 \times 10^{-5} k^4 + 7.1802 \times 10^{-3} k^3 - 1.2514 k^2 + 94.8463 k - 69.1588$$

appeared to be the lowest-degree least-squares approximation to the discrete data that satisfies the q-k curve criteria. This quartic polynomial must be evaluated at each node for each time step so it is important to use a polynomial of the least degree.

The choice of q-k curve was found to have a large effect on the stepsize selection of the implicit methods. Implicit methods using the smooth q-k curves generated by the least squares and Greenshield's methods were able to use larger time steps than the programs using cubic spline or piecewise linear curves.

Table 1. Empirical q-k data.

k = density (veh/mile/lane) q = flow (veh/hour/lane)			
k	q	k	q
0	0	66	2376
10	650	76	2432
20	1260	98	2352
30	1860	124	2232
32	1952	150	1500
35	2100	175	525
36	2124	186	0

Table 2. Uncongested flow field data.

Initial condition: 271.67 veh/5 min/2 lanes uncongested flow at all nodes q_u = upstream flow, q_d = downstream flow, q_c = flow at check station							
Time	Flow (veh/5 min/2 lanes)			Time	Flow (veh/5 min/2 lanes)		
(min)	q_u	q_d	q_c	(min)	q_u	q_d	q_c
5	272	277	270	65	275	267	277
10	285	284	288	70	283	289	285
15	287	283	279	75	253	261	255
20	301	293	296	80	262	259	260
25	285	286	284	85	245	240	237
30	318	310	315	90	259	257	260
35	321	319	318	95	294	287	291
40	344	356	348	100	232	241	236
45	322	322	330	105	227	234	229
50	300	303	295	110	259	247	250
55	329	319	328	115	272	273	279
60	313	327	317	120	249	240	243

5. FREEWAY TRAFFIC MODEL CASES

Three model cases of freeway traffic flow were used to test the numerical methods described above. These model cases describe congested/uncongested pipeline and entry/exit freeway traffic flows. Each model case consists of a section of a single-lane freeway. The data sets used with each model case give the number of vehicles counted crossing each boundary (all lanes) during each 5-minute interval. In all model cases, these numbers, vehicles/5 minutes, are then multiplied by 12 and divided by the number of lanes to yield an average single-lane flow rate in vehicles/hour/lane. The field data are collected as follows: A detector is placed at a *check station* which counts the volume of cars passing that road point every 5 minutes. Check stations are set at the upstream and downstream boundaries and at one more locations in the freeway stretch in between. These measurements provide a flow-time function. This function has the form of a step function. The flow at the boundaries is used to set up the boundary conditions of the PDE (1). The flow at intermediate points is used to compute the deviation of the computed model solution from the field data.

In our modeling we distributed linearly the flow within the 5-minute intervals. The resulting boundary flow-time function is piecewise linear. In the programs boundary flow rates are converted to density (vehicles/mile/lane) using a q-k relation. This gives boundary density-time functions which are piecewise linear. This is implemented as follows. At each time step the boundary densities must be assigned to the boundary nodes (upstream and downstream), and then the boundary flow is determined from the density. In these programs the boundary density at each time step is found by linear interpolation between the known density values at $t = 5$ min, 10 min, etc. Then boundary flow is found at the point on the q-k curve corresponding to the boundary density.

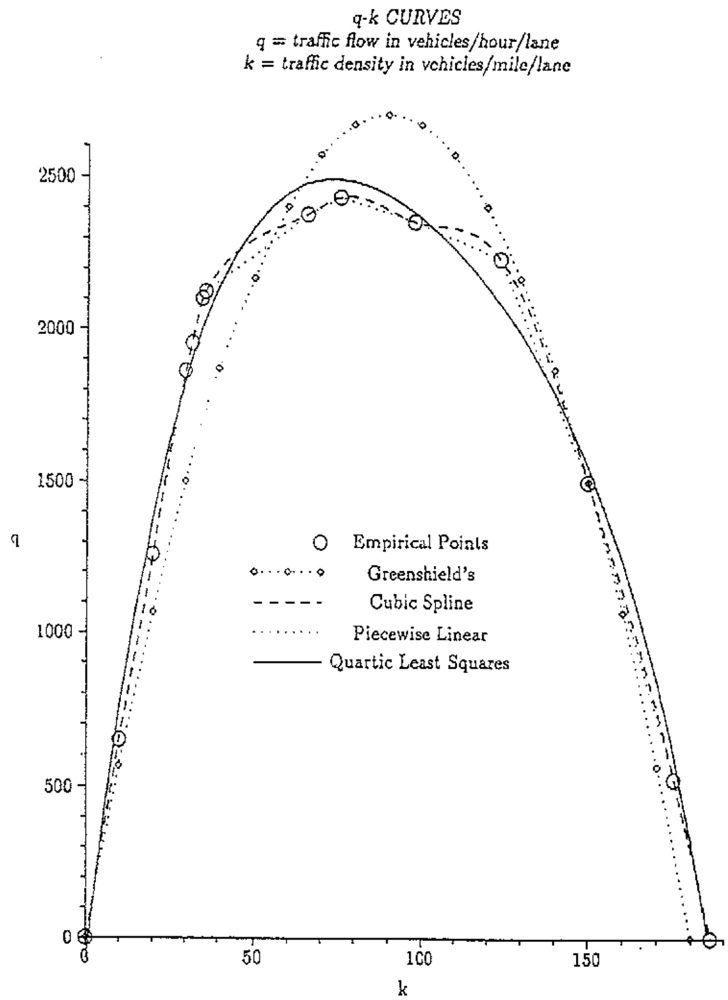

Figure 1.

specified so we have assumed the initial flow to be the average of the flow values at the check stations at 6:05am, and constant along the road section. For this report, only the measurements made at the first check station were used in comparisons with the numerical program output.

In this simplified model case the flow is assumed to be uncongested at all times. Merging flow from the entry ramp is added to the flow at the first node downstream of the ramp. If merging flow plus mainstream flow exceeds the maximum flow $q_{\max}$, the flow value at the entry node is set to $q_{\max}$ and any excess merging flow is not used in the calculations. Exiting flow is subtracted from the first node downstream of the exit ramp.

6. RESULTS

Table 1 contains the data for the empirical q-k curve. These data were used in constructing the piecewise linear, cubic spline and least squares approximations shown in Figure 1. Tables 2–4 contain the field data for the uncongested/congested and entry/exit freeway traffic flow tests.

For the tests, the time stepsize selection was made as follows. For the Lax method, we set $\Delta t = 1$ sec. This is required in order to maintain numerical stability. For the implicit methods we increased the time stepsize subject to the restriction that the maximum error does not exceed that of the Lax method. For the uncongested and entry/exit flow cases a single time stepsize was selected. For the congested flow case two different time stepsizes were used. One *small* stepsize was used in the 5-minute time intervals of change from congested to uncongested (or vice versa) and another *large* stepsize was in other time intervals.

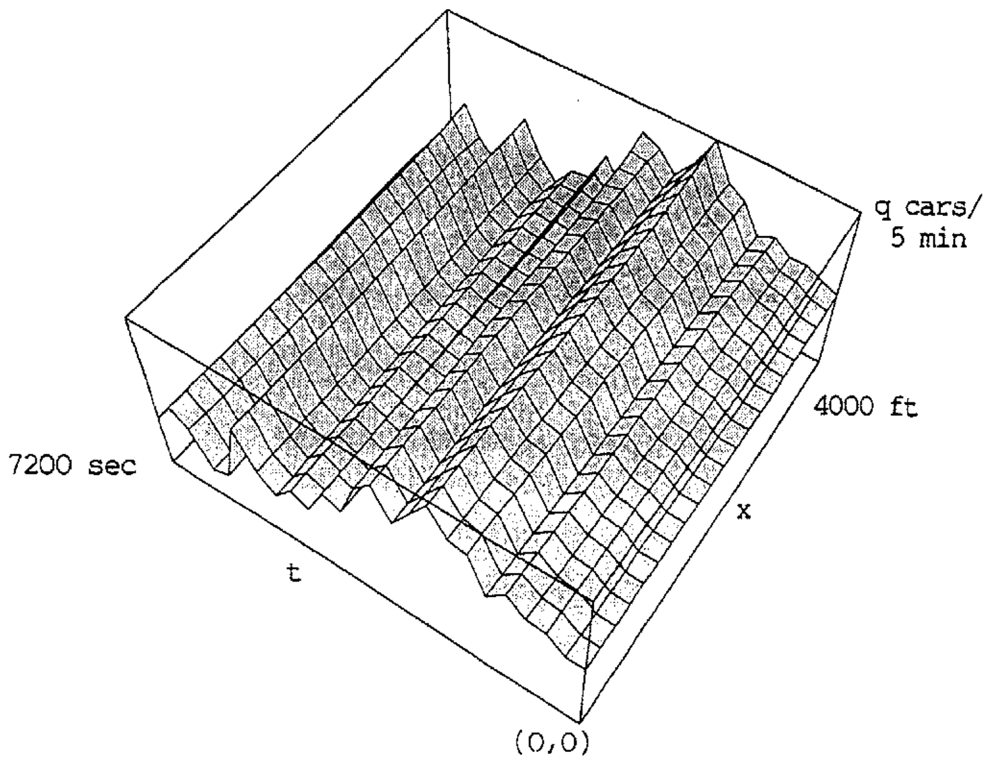

Figure 2. Lax Method. Uncongested Pipeline Flow (veh/5 min/2 lanes). Quartic Least Squares $q-k$ Curve, $dx = 200$ ft, $dt = 1$ s.

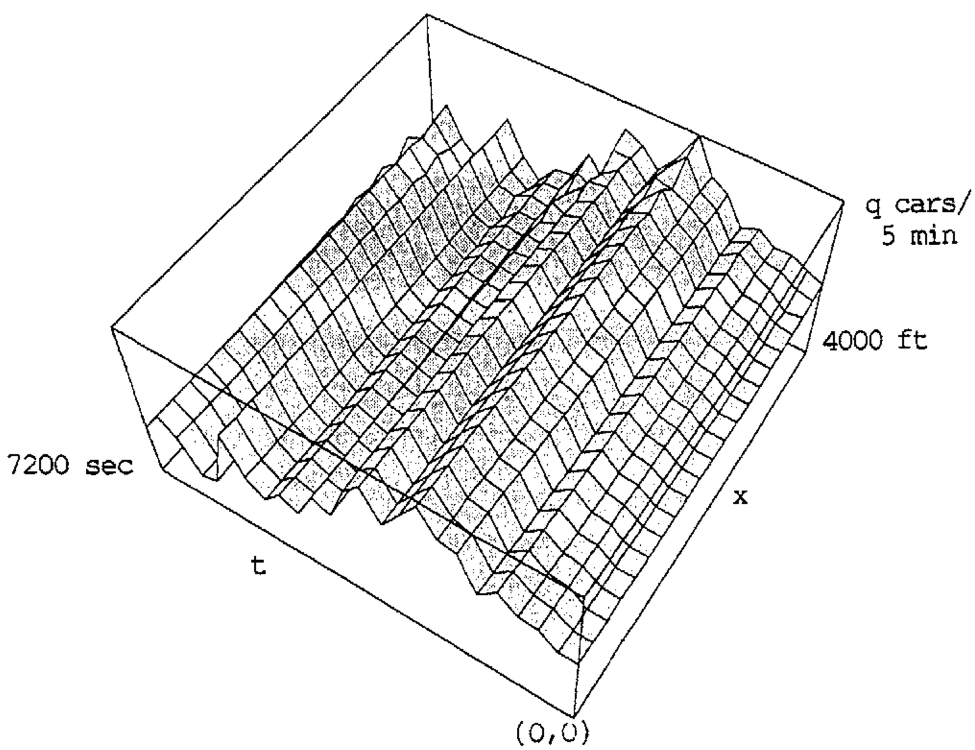

Figure 3. Implicit Euler Method. Uncongested Pipeline Flow (veh/5 min/2 lanes). Quartic Least Squares q-k Curve, $dx = 200$ feet, $dt = 15$ s, omega $= 1.0$.

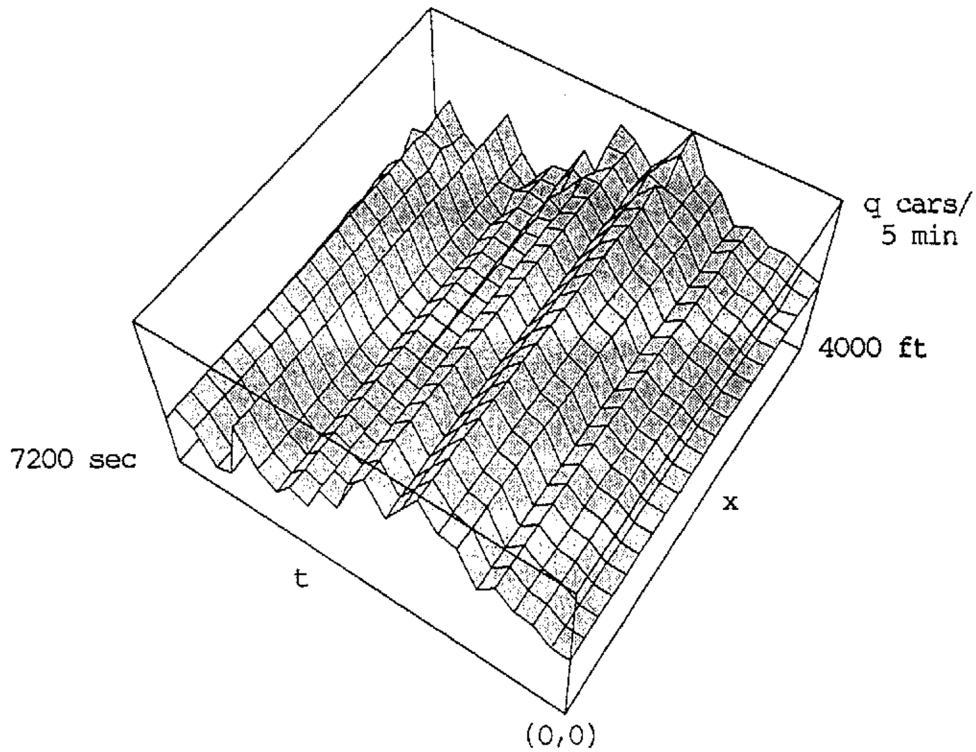

Figure 4. Trapezoid Method. Uncongested Pipeline Flow (veh/5 min/2 lanes). Quartic Least Squares q-k Curve, $dx = 200$ feet, $dt = 15$ s, omega $= 1.0$.

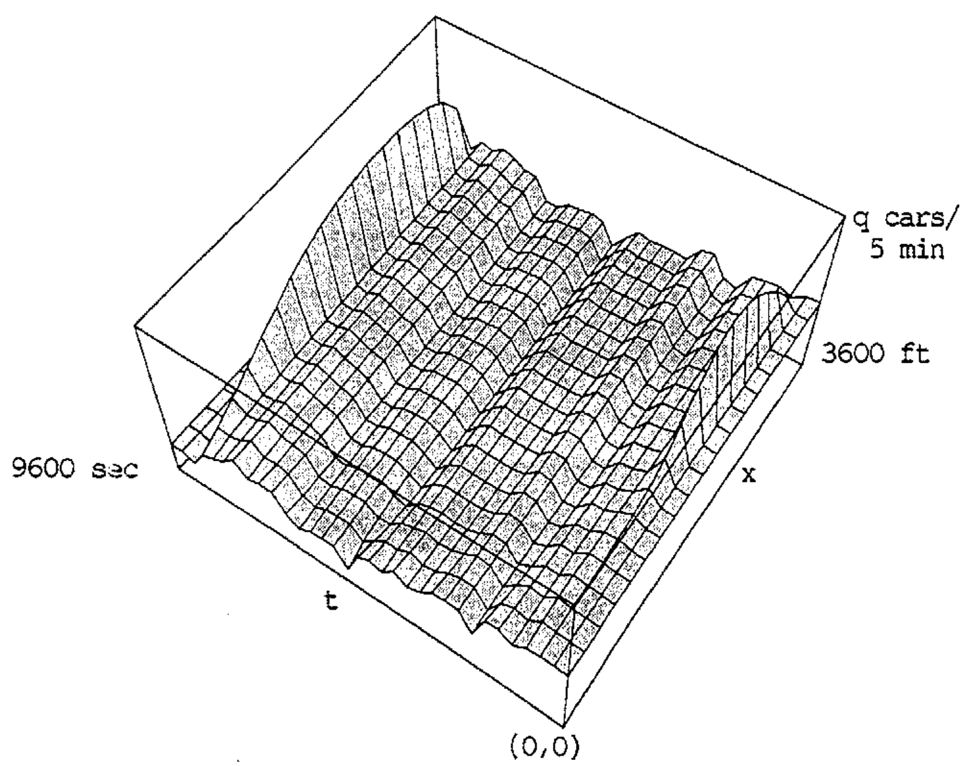

Figure 5. Lax Method. Uncongested Pipeline Flow (veh/5 min/4 lanes). Quartic Least Squares q-k Curve, $dx = 200$ feet, $dt = 1$ s.

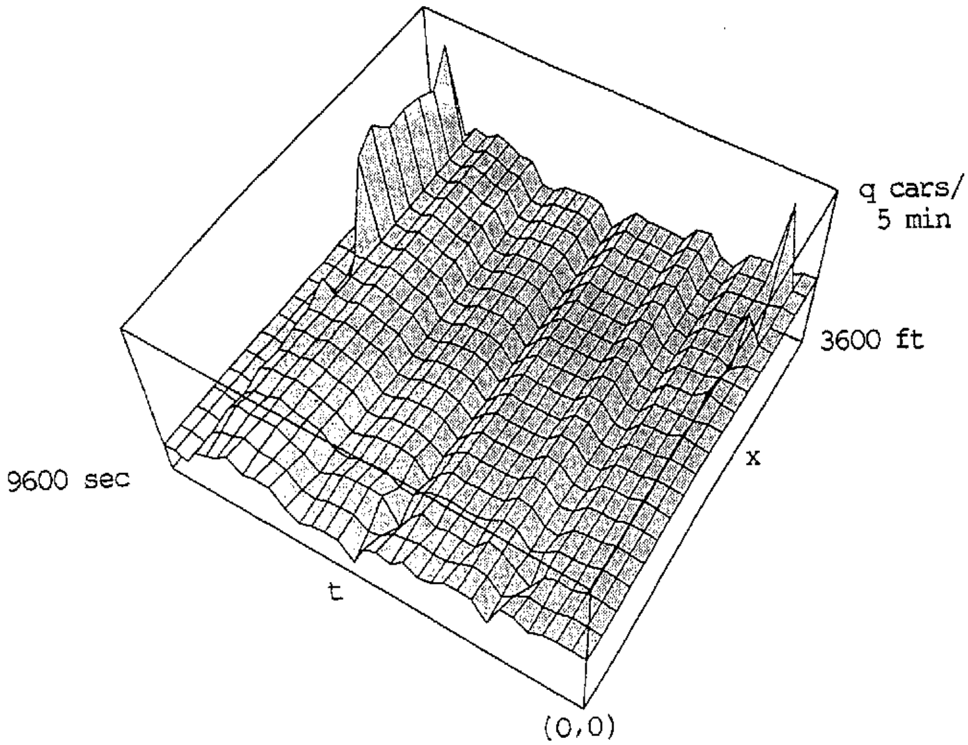

Figure 6. Implicit Euler Method. Uncongested Pipeline Flow (veh/5 min/4 lanes). Quartic Least Squares q-k Curve, $dx = 200$ feet, $dt = 15$ s in regular regions, $dt = 3$ s in congestion-change regions, omega $= 1.0$.

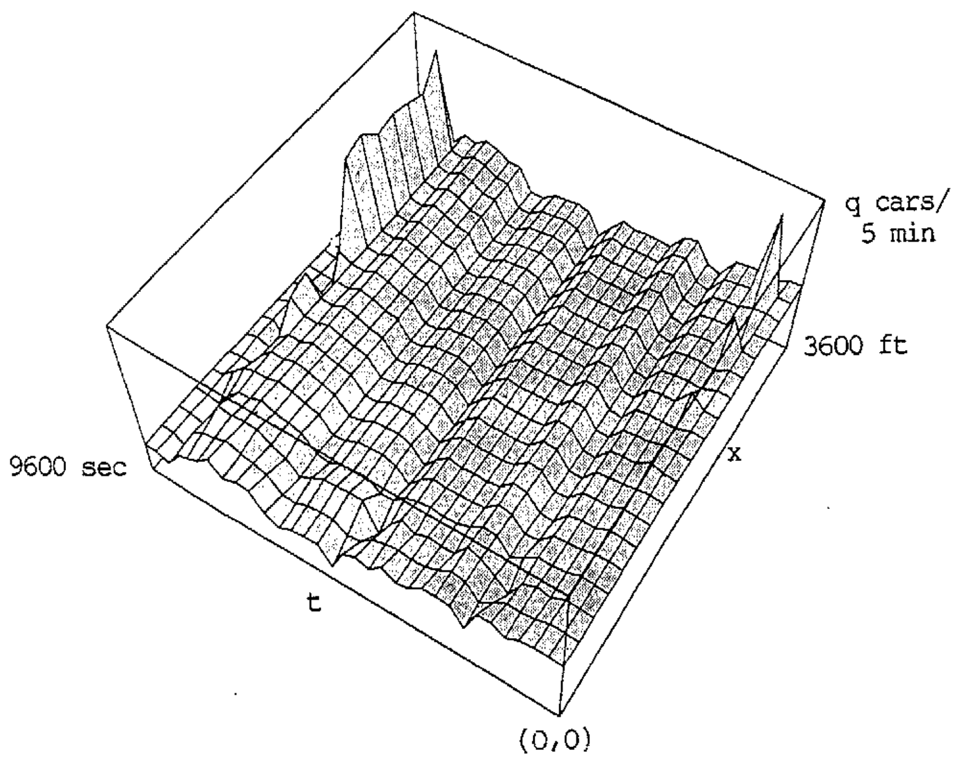

Figure 7. Trapezoid Method. Uncongested Pipeline Flow (veh/5 min/4 lanes). Quartic Least Squares q-k Curve, $dx = 200$ feet, $dt = 15$ s in regular regions, $dt = 3$ s in congestion-change regions, omega $= 1.0$.

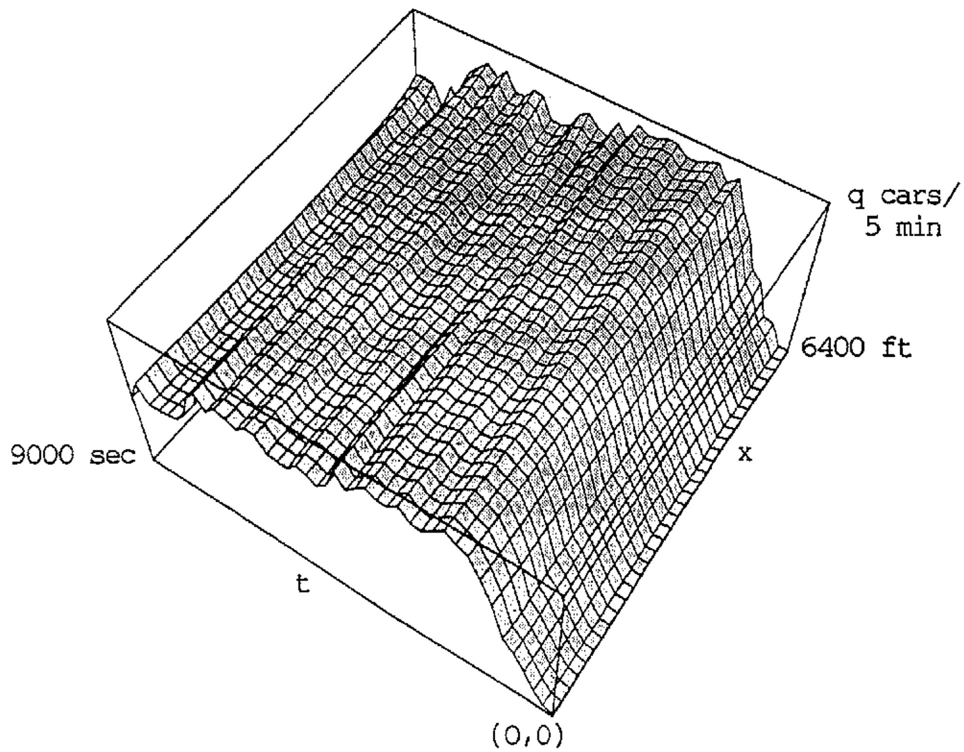

Figure 8. Lax Method. Uncongested Entry/Exit Flow (veh/5 min/3 lanes). Quartic Least Squares q-k Curve, $dx = 200$ feet, $dt = 1$ s.

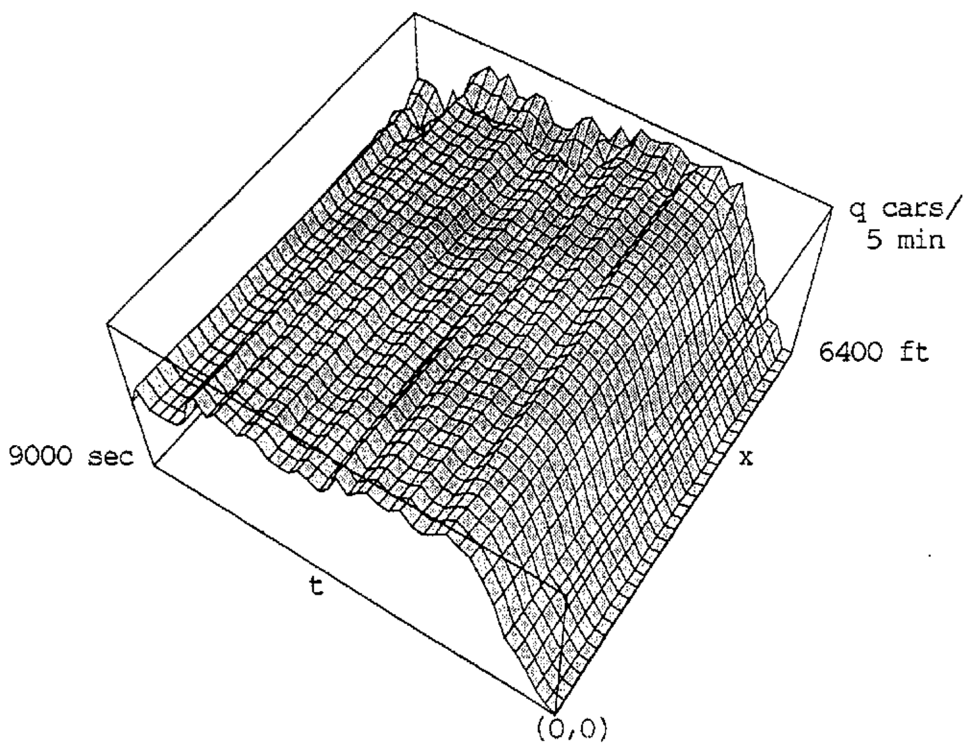

Figure 9. Implicit Euler Method. Uncongested Entry/Exit Flow (veh/5 min/3 lanes). Quartic Least Squares q-k Curve, $dx = 200$ feet, $dt = 15$ s omega = 1.0.

Table 6. Uncongested flow results.

Quartic Least Squares q-k Curve $dt = 15$ s Time = exec time on SPARCstation 1			
Method	Error (veh/5 min/2 lanes) maximum	average	Time (s)
Lax	9.61	3.93	2.6
Euler	9.84	4.01	0.6
Trapezoid	9.83	4.03	0.6

Table 7. Congested flow results.

Quartic Least Squares q-k Curve $dt = 15:3$ s Time = exec time on SPARCstation 1			
Method	Error (veh/5 min/4 lanes) maximum	average	Time (s)
Lax	273.56	24.99	3.2
Euler	77.32	17.33	1.6
Trapezoid	106.77	20.88	1.6

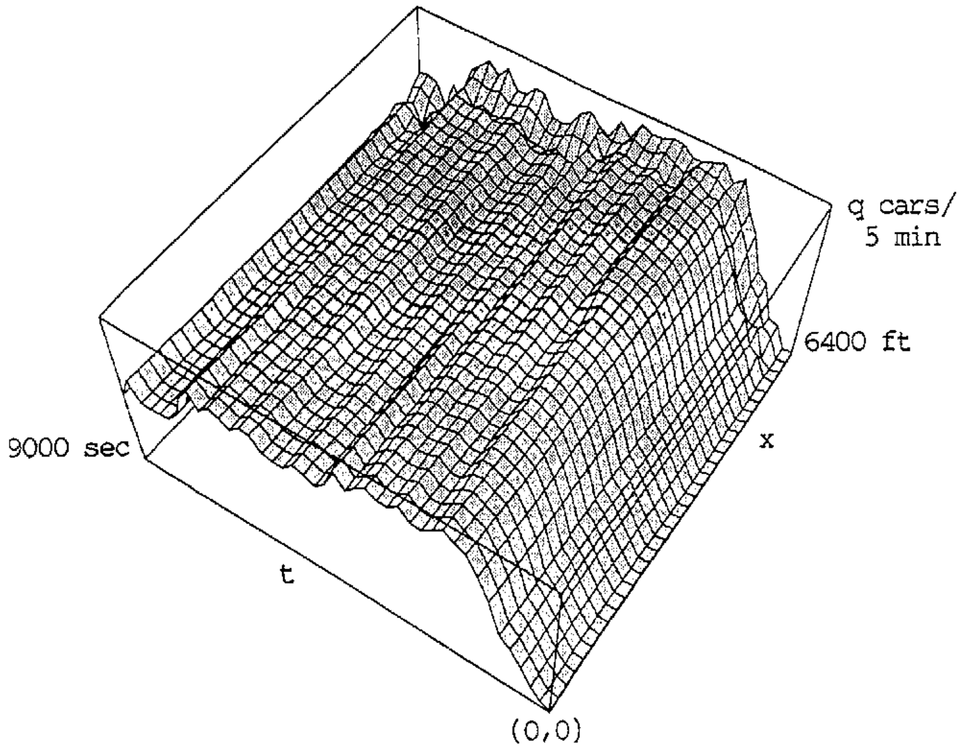

Figure 10. Trapezoid Method. Uncongested Entry/Exit Flow (veh/5 min/3 lanes). Quartic Least Squares q-k Curve, $dx = 200$ feet, $dt = 15$ s omega = 1.0.

The tests for selecting the best empirical q-k curve pointed to the quartic least squares approximation. This allows the largest stepsize combinations in the implicit methods yielding the smallest maximum error. The results for the congested case are contained in Table 5. The largest of the stepsizes was found to give the smallest maximum error in the uncongested and entry/exit flow cases. Table 6 shows the results for accuracy and execution time obtained on the Sun SPARCstation 1, using a time step size $\Delta t = 15$ sec for the implicit methods programs on the uncongested field data. Table 7 shows the results obtained on the same machine using congested field data and $\Delta t = 15$ sec during regular intervals and $\Delta t = 3$ sec during congestion-change intervals. Table 8 shows the results obtained on the same machine using the entry/exit field data and $\Delta t = 15$ sec.

The best performance in accuracy and execution time was obtained with the Euler implicit method using three (Newton) iterations per time step in congestion-change intervals. This method showed a large improvement over the Lax method in both error and time required. In the uncongested and entry exit cases (Tables 6 and 8) the maximum errors in all three methods are of the same magnitude. In the congested case (Table 7) the maximum error produced by the Euler method was about one fourth of the maximum error produced by the Lax method. The implicit methods are more than twice as fast the Lax method (Tables 6–8 last column) for the congested case and more than three times in the uncongested and entry/exit cases.

Table 8. Entry/exit flow results.

Quartic Least Squares q-k Curve $dt = 15$ s Time = exec time on SPARCstation 1			
Method	Error (veh/5 min/3 lanes)		Time
	maximum	average	(s)
Lax	42.66	11.58	7.2
Euler	42.51	11.38	1.6
Trapezoid	42.52	11.50	1.6

The 3-D figures, Figure 2–10, show the Lax, Euler and Trapezoid solution using the empirical quartic least squares approximation. These solutions plots look very close to each other in all cases except the congested flow case. The high oscillations in the congested/uncongested change intervals appear in the Lax method more than the Euler method.

7. CONCLUSIONS

We have studied the use of implicit numerical methods solve the flow conservation continuum model. We have written an experimental code in C simulating a freeway (un)congested pipeline and freeway entry/exit traffic flow. Tests with real data collected from the I-35 W freeway in Minneapolis were conducted on a workstation computer. Our tests show that the implicit methods are more efficient than the Lax method and provide the same or better accuracy. This could increase if iterative methods are used instead of the Gaussian elimination in solving the tridiagonal linear systems required by the implicit methods.

REFERENCES

1. M.H. Lighthill and G.B. Witham, On kinematic waves: II A theory of traffic flow on long crowded roads, *Proceed. R. Soc. Ser.* **A 229** (1178), 317–345 (1955).
2. H.J. Payne, Models of freeway traffic and control, *Proceed. Math. of Publ. Syst.*, Published by Simul. Council 1 (1), 51–61 (1971).
3. P.G. Michalopoulos and J. Lin, A freeway simulation program for microcomputers, In *Proceed., 1st National Conf. on Microcomputers in Urban Transp.*, pp. 330–341, ASCE, California, (1985).
4. C.J. Leo and R.L. Pretty, *Some Comments and Numerical Tests on Upwind Finite Difference Schemes for Macroscopic Traffic Modeling*, Dept. of Civil Eng., Univ. of Queensland, Australia, (1990).
5. C. Hirsch, *Numerical Computation of Internal and External Flows*, Vol. 2, John Wiley and Sons, (1988).
6. B.D. Greenshields, A study of traffic capacity, In *Proc. Highway Res. Board*, Vol. 14, pp. 448–477, (1934).
7. W. McShane and R. Roess, *Traffic Engineering*, Prentice Hall, (1990).
8. P.G. Michalopoulos, Ping Yi, D.E. Beskos and A.S. Lyrintzis, Continuum modeling of traffic dynamics, In *Proc. of the 2nd Int. Conf. on Appl. of Advanced Tech. in Transportation Eng.*, pp. 36–40, ASCE, Minneapolis, Minnesota, (1991).
9. P.G. Michalopoulos, Eil Kwon and J. Kang, Enhancements and field testing of a dynamic freeway simulation program, *Transportation Research Records* (1991).
10. J.J. Dongarra *et al.*, *LINPACK User's Guide*, SIAM, (1979).