Stochastic Modeling and Simulation of Microscopic Processes

Andreas Andersson

Bachelor's thesis
2013:K15

Faculty of Science
Centre for Mathematical Sciences
Numerical Analysis

ABSTRACT. In this thesis we study analytical and numerical methods which describe interactions and evolution of complex physical systems modeled by microscopic processes. We focus on two types of systems which are differentiated based on how modeling is taking place at the microscopic level. In that respect we study systems which can be described either by stochastic differential equations or by elementary microscopic stochastic processes.

For the case of stochastic differential equations, we illustrate how to produce their solutions both analytically and numerically. For the case of elementary microscopic stochastic processes it is unrealistic to expect that an analytic solution would always exist since often such systems can be transient and out of equilibrium. We therefore produce numerical solutions for those systems which are based on Monte Carlo simulations.

We constitute the respective analytic solutions and numerical methods from two vastly different research areas such as finance and traffic flow in order to better illustrate the wide applicability of such methods.

Contents

1. Introduction

In this thesis we study two different modeling methodologies applied in systems whose macroscopic behavior and characteristics can be obtained and are strongly related to their microscopic dynamics. In particular we study how stochastic processes can be used to describe particle interactions and evolution. As an extension this can allow us to describe complex physical systems.

Stochastic differential equations (SDEs) employ stochastic processes in order to describe microscopic particle interactions and subsequent trends of their macroscopic equivalent characteristics. Practically this is done by employing differential equations together with random variables in order to describe unknown and generally chaotic system behavior. Knowing and many times assuming some minimal information [**1**] about these random variables, such as their expected value or variance we are then in position to assert their future behavior by solving the respective SDE for that system.

Alternatively complex system dynamics [**2, 3, 4, 5**] can also be described via microscopic stochastic processes which are built, from the ground up, by directly enforcing a set of rules from which particles interact with their nearest neighbors. Under this methodology we do not assume any adhoc differential equation to describe our physical system but rather apply specific microscopic interaction rules which produces (in the asymptotic limit) the form of the differential equation responsible for the macroscopic behavior of that system.

In this thesis we will study analytical and numerical methods for both approaches and apply them to problems in finance as well as traffic flow in order to present their capabilities.

2. Stochastic Processes And Microscopic Dynamics

In this section we provide all the theory behind the stochastic processes responsible for modeling particle interactions and evolution for a given physical system. Examples of such systems range from economics to chemical reactions or traffic networks. In this section we provide the necessary mathematical background which will help us resolve system dynamics and behavior. Later in the applications section we demonstrate how these methods can be applied to such systems.

2.1 Stochastic Differential Equations

We begin by studying stochastic differential equations of the form

$$dX_t = a(t, X_t)dt + b(t, X_t)dW_t \tag{1}$$

and outline the mathematical tools such as Itô calculus in order to do so.

To understand the above differential equation we first need to explain terms such as X_t and W_t that appear in equation (1) above . To do this, we first need to define what a stochastic process is and what a sample path is.

DEFINITION. (Stochastic process and sample path)
A collection of random variables $\{X(t) \,|\, t \geq 0\}$ is called a stochastic process. For each point $\omega \in \Omega$ in the sample space, the mapping $t \to X(t, \omega)$ is the corresponding sample path.

REMARK. The stochastic process $\{X(t) \,|\, t \geq 0\}$ is often abbreviated as X_t.

The sample path $t \to X(t, \omega)$ will in general look different if we run multiple experiments while keeping all other parameters the same. For instance, if we model a stock price over a time period, the price of the stock will not have the same values for each individual time point between the simulations. The following figure illustrates how two different sample paths can look for the same stochastic process $X(t)$ for the sample paths ω_1 and ω_2.

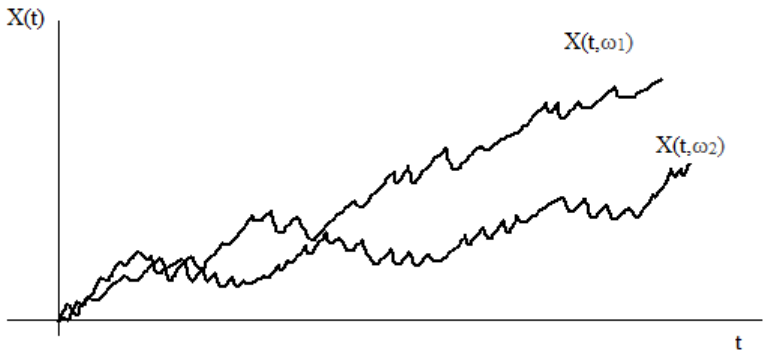

FIGURE 1. Sample paths ω_1 and ω_2 for the stochastic process X_t.

We can now define the Brownian motion $W(\cdot)$:

DEFINITION. (Brownian motion)
We say that a real valued stochastic process $W(\cdot)$ is called a Brownian motion (or a Wiener process) if it satisfies the following properties:
i) $W(0) = 0$ a.s.
ii) $W(t)$ is $N(0, t)$
iii) $W(t) - W(s)$ is $N(0, t - s)$ for all $t \geq s \geq 0$,
iv) The stochastic process $W(t)$ has independent increments i.e for $0 < t_1 < t_2 < \cdots < t_n$, the random variables $W(t_1), W(t_2) - W(t_1), \ldots, W(t_n) - W(t_{n-1})$ are independent.

We are now able to explain the details the equation (1) above with the help of the following theorem.

THEOREM. *(Itô formula)*
Let X_t be an Itô process given in differential form

$$dX_t = a(t, X_t)dt + b(t, X_t)dW_t$$

Let $g(t, x)$ be a twice differentiable function defined on $[0, \infty) \times \mathbb{R}$, i.e $g(t, x) \in C^2([0, \infty) \times \mathbb{R})$. Then

$$Y_t = g(t, X_t)$$

is again an Itô process and

$$dY_t = \frac{\partial g}{\partial t}(t, X_t)dt + \frac{\partial g}{\partial x}(t, X_t)dX_t + \frac{1}{2}\frac{\partial^2 g}{\partial x^2}(t, X_t) \cdot (dX_t)^2 \tag{2}$$

where $(dX_t)^2 = (dX_t) \cdot (dX_t)$ *is computed by use of the following rules:* $dt \cdot dt = dt \cdot dW_t = dW_t \cdot dt = 0$ *and* $dW_t \cdot dW_t = dt$.

PROOF. See [**1**]. □

REMARK. The term $dW(t)$ is interpreted as $dW(t) = W(t+dt) - W(t)$, i.e. a small step along the sample path for $W(t, \omega)$.

The reason we chose the Itô interpretation over of the Stratonovich interpretation of stochastic differential equations is because the Itô interpretation is better suited when modeling problems based on historical events. See [**1**] for a more throughout discussion regarding this.

We now illustrate how the above formula can be used to solve some elementary stochastic differential equations.

EXAMPLE 1. Consider the SDE initial value problem given by

$$\begin{cases} dX_t = tX_t dt + e^{\frac{t^2}{2}} dW_t \\ X_0 = 1 \end{cases} \tag{3}$$

To show that $X_t = (1 + W_t)e^{\frac{t^2}{2}}$ solves (3) we let $x = W_t$ so that $X_t = g(t, x) = (1 + x)e^{\frac{t^2}{2}}$. By Itô's formula (2) we have:

$$\begin{aligned} dX_t = (1 + x)e^{t^2/2} \cdot \tfrac{2t}{2} dt + e^{t^2/2} dx + \tfrac{1}{2} \cdot 0 \cdot dx dx &= (1 + W_t)e^{t^2/2} t dt + e^{t^2/2} dW_t \\ &= X_t t dt + e^{t^2/2} dW_t \end{aligned}$$

It remains to check that $X_0 = 1$. Indeed:

$$X_0 = (1 + W_0)e^0 = (1 + 0)e^0 = 1$$

So $X_t = (1 + W_t)e^{t^2/2}$ is indeed a solution to the SDE initial value problem given by (3).

Another example which involves a slightly more complicated scenario is the following SDE.

EXAMPLE 2. Consider the SDE initial value problem given by

$$\begin{cases} dX_t = 3(W_t^2 - t) dW_t \\ X_0 = 0 \end{cases} \tag{4}$$

To show that $X_t = W_t^3 - 3tW_t$ is a solution to (4) we let $x = W_t$ so that $X_t = g(t, x) = x^3 - 3tx$. By Itô's formula (2) we have:

$$dX_t = -3xdt + (3x^2 - 3t)dx + \tfrac{1}{2}6xdxdx = -3W_t dt + (3W_t^2 - 3t)dW_t + 3W_t dW_t dW_t$$
$$= -3W_t dt + (3W_t^2 - 3t)dW_t + 3W_t dt = (3W_t^2 - 3t)dW_t$$

It remains to check that $X_0 = 0$. Indeed:

$$X_0 = W_0^3 - 3 \cdot 0 \cdot W_0 = 0 - 0 = 0$$

So $X_t = W_t^3 - 3tW_t$ is indeed a solution to the SDE initial value problem given by (4).

In the following example we show how to solve the Langevin equation that have many important applications in financial modeling and statistical physics

EXAMPLE 3. Consider the SDE given by

$$dX_t = \mu X_t dt + \sigma dW_t \tag{5}$$

The idea is to multiply (5) with an integrating factor $e^{-\mu t}$ and comparing it to $d(e^{-\mu t}X_t)$. By Itô's formula (2) we have

$$d(e^{-\mu t}X_t) = -\mu e^{-\mu t}X_t dt + e^{-\mu t}dX_t \tag{6}$$

In (6) we recognize that dX_t is given by (5). Inserting (5) into (6) gives

$$d(e^{-\mu t}X_t) = -\mu e^{-\mu t}X_t + e^{-\mu t}(\mu X_t dt + \sigma dW_t)$$
$$= -\mu e^{-\mu t}X_t + e^{-\mu t}\mu X_t dt + e^{-\mu t}\sigma dW_t$$

Which gives us

$$d(e^{-\mu t}X_t) = e^{-\mu t}\sigma dW_t$$

Integrating both sides gives

$$e^{-\mu t}X_t = X_0 + \sigma \int_0^t e^{-\mu s}dW_s$$

Multiplying by $e^{\mu t}$ gives

$$X_t = e^{\mu t}X_0 + \sigma \int_0^t e^{-\mu(s-t)}dW_s$$

We see that no closed form solution exists. Later we will show how the Langevin equation can be solved numerically.

2.2 Lattice Based Stochastic Modeling Of Particle Interactions

This section describes a method to simulate interacting particles on a closed periodic domain as proposed by T. Alperovich and A. Sopasakis in [**6**].

The domain is divided into $n \cdot m = N$ cells, where n denotes the number of vertical cells and m the number of horizontal cells. This lattice configuration is denoted by $\mathcal{L} = \mathcal{L}_n \times \mathcal{L}_m$, where $\mathcal{L}_n = \{1, 2, \dots n\}$ and $\mathcal{L}_m = \{1, 2 \dots m\}$. We can thus define each location in the lattice as $x \in \mathcal{L}$. Each position is uniquely determined by its horizontal and vertical cell number, i.e every x is can be written as $x = (x_k, x_l)$ where $0 \leq k \leq m$ denotes the particle's horizontal cell number and $0 \leq l \leq n$ denotes the particle's vertical cell number in the lattice.

To identify if a position in the lattice is occupied by a particle we introduce the following order parameter $\sigma(x)$ for each $x \in \mathcal{L}$ as:

$$\sigma(x) = \begin{cases} 1, \text{ if a particle occupies position x} \\ 0, \text{ if the position x is empty} \end{cases} \tag{7}$$

Next we introduce the interaction potential J. This potential determines the strength of the interaction between a particle and the particles in its surrounding area. In this work, J denotes an asymmetric short range inter-particle interaction potential defined by

$$J(y, x) = V\left(\frac{||\vec{y} - \vec{x}||_d}{L}\right), \, x, y \in \mathcal{L} \tag{8}$$

In (8) L denotes the look ahead radius, i.e. the cells a particle can look ahead in the domain. Here $\vec{x}$ and $\vec{y}$ denote the location in space for each particle. The look ahead radius lets us decide if we want to allow one sided interactions or all around interactions. The notation $||\vec{y} - \vec{x}||_d$ denotes the absolute value of the distance between two particles in the lattice and is defined by

$$||\vec{y} - \vec{x}||_d = \begin{cases} y_1 - x_1, \text{ if } y_1 \geq x_1 \\ x_1 - y_1, \text{ if } x_1 < y_1 \end{cases} \tag{9}$$

It is important to note that the distance between the particles given by (9) is only determined by the distance between them in the first coordinate.

When simulating traffic we only allow one sided interactions for the vehicles, i.e. vehicles are only allowed to look in front of them. When simulating other types of dynamics we can for instance allow all around interactions between the particles. We

also choose a uniform potential for the particle interactions. We thus define the potential $V : \mathbb{R} \to \mathbb{R}$ as

$$V(r) = \begin{cases} J_0, \, if \, 0 \leq r < 1 \\ 0, \, otherwise \end{cases} \tag{10}$$

where J_0 is a non-zero parameter which will be calibrated later depending on the application. Since particles naturally tend to move towards places in the lattice with less concentration of particles, we want the interaction between the particles to be repulsive. For traffic simulations the sign of J_0 in (10) is chosen to be positive since a negative sign would imply attraction between particles. To compute the total contributions from all particles inside the look ahead radius we let $U_e(x)$ denote the short range interactions between particles and is given by

$$U_e(x) = \sum_{z \neq x} J(x, z)\sigma(z) \tag{11}$$

We also introduce another interaction called the anisotropy interactions denoted by $U_\alpha(x, y)$. This interaction affects the particles tendency to move in a preferred direction. We define it as

$$U_\alpha(x, y) = \sum_{z_k = y_k}^{y_k + L} w\sigma(z) \tag{12}$$

where y denotes the location of the cell that the vehicle in position x will move to and $\sigma(z)$ is defined as in (7). The constant w is chosen to reflect the strength of the interaction. For traffic simulations we would generally want to encourage vehicles to move forward and discourage them from changing lanes. We thus define three different values for the parameter w as

$$w = \begin{cases} w_l, \, if \, y = x_l + 1 \\ w_r, \, if \, y = x_l - 1 \\ w_f, \, if \, y = x_k + 1 \end{cases} \tag{13}$$

To reflect the above stated preferences, we choose $w_f \gg w_r, w_l$ in (13) where $w_f, w_r, w_l > 0$.

The rate at which the process will advance a particle from a position $x \in \mathcal{L}$ to $y \in \mathcal{L}$ is defined as

$$c(x,y,\sigma) = \begin{cases} c_0 \exp\left[-U(x)\right] \; if \; \sigma(x) = 1, \; \sigma(y) = 0 \\ 0, \text{ otherwise} \end{cases} \tag{14}$$

where $c_0 = 1/\tau_0$. In (14), $U(x) = U_e(x) + U_\alpha(x,y)$ where $U_e(x)$ is given by (11) and $U_\alpha(x,y)$ is given by (12). The parameter τ_0 is called the characteristic time for the process and is calibrated in conjunction with the parameter J_0 so that they reflect realistic conditions. For a one dimensional lattice we have $y = y_k = x_{k+1}$. For a two dimensional lattice, y is given by

$$\begin{gathered} y_l = x_l + 1, \text{ for moves to the left,} \\ y_r = x_l - 1, \text{ for moves to the right,} \\ y_k = x_k + 1, \text{ for moves forward} \end{gathered}$$

The stochastic process $\{\sigma_t\}_{t\geq 0}$ that is constructed above is a continuous, jump Markov process on $L^\infty(\Sigma; R)$ with the following generator [**4, 7**]

$$(Mf)(\sigma) := \sum_{x\in\mathcal{L}} c(x,y,\sigma)[f(\sigma^x(y)) - f(\sigma(y))], \; for\, all\, 0 < y \in \mathcal{L}$$

for any bounded test function $f \in L^\infty(\Sigma; R)$ with $c(x,y,\sigma)$ defined in (14). Here $\sigma^x(y)$ denotes the configuration after a change in the value of the cell at x such that, $\sigma^x(y) = 1 - \sigma(x)$ if $y = x$ and $\sigma^x(y) = \sigma(y)$ otherwise. Therefore the observables f (test functions) evolve with the rule [**7**]

$$\frac{d}{dt}\mathbf{E}f(\sigma_t) = \mathbf{E}(Mf)(\sigma)$$

which is equivalent to Dynkin's formula [**1**]. Thus, the differential equation above gives the connection between the microscopic process and the corresponding macroscopic process that is given by a stochastic differential equation.

2.3 Lattice Free Stochastic Modeling Of Particle Interactions

When simulating real physical systems it is often unrealistic to assume that the particles move with integer jumps to a surrounding location in the domain. Instead it would be more realistic to allow particles to move to any surrounding location in the domain. This motivates us to simulate the particles in a lattice free domain as suggested by A. Sopasakis in [**8**] and [**9**]. Instead of dividing the lattice into equally sized cells, we now divide the domain in a union D of disjoint sets. The way we do this is to partition the domain into disjoint sets O and E where O denotes space on the domain that is occupied by particles and E the is the remaining empty space. If we have k particles and l empty sets in the domain then D can be written as

$$D = O \bigcup E = V_1 \cup V_2 \cup \cdots \cup V_k \cup E_{k+1} \cup \cdots \cup E_{k+l}$$

The analogy to the order parameter $\sigma(x)$ given in the previous section for the lattice free dynamics is given by

$$\sigma(i) = \begin{cases} 1, \text{ if at } V_i \\ 0, \text{ if at } E_i \end{cases}$$

where $1 \leq i \leq k + l < M$ if there are k particles in the domain and l empty sets. We see that $\sigma(i) = 1$ if there exist a particle at $i \in \{1, \ldots, k\}$ and $\sigma(i) = 0$ if there's no particle at $i \in \{k + 1, \ldots, k + l\}$.

We define the new interaction potential J for the lattice free dynamics between particles at V_i and V_j by

$$J(i-j) = \frac{1}{L} W\left(\frac{i-j}{L}\right), \text{ if i,j} \in I_O \text{ and } W(r) = \begin{cases} J_*, \text{ for } 0 \leq r \leq 1 \\ 0, \text{ otherwise} \end{cases} \tag{15}$$

In (15) L denotes the range of interaction, I_O denotes the index set for the set O of occupied positions in the domain and J_* is a parameter that must be calibrated.

The corresponding spin-exchange rate for the lattice free dynamics is given by

$$c(i, j, \sigma) = \begin{cases} d_0 w(j) \exp[-U(i, \sigma)], \text{ if } \sigma(i) = 1 \text{ and } \sigma(j) = 0 \\ 0, \text{ otherwise} \end{cases} \tag{16}$$

where

$$U(i, \sigma) = U_e + U_\alpha = \sum_{j=1}^{k+l} J(i-j)\sigma(j) + \sum_{j=i}^{k+L} w\sigma(j)$$

where U_α is recognized as the anisotropy interaction defined in the same way as in (12).

Note also the introduction of the factor $w(j)$ above defined as

$$w(j) = \begin{cases} |E_i| - |V|, \text{ if } |E_i| > |V| \\ 0, \text{ otherwise} \end{cases}$$

This factor dictates how far a particle can move depending on the space available ahead.

3. Numerical Applications

We now apply the theory presented in the previous section to produce respective numerical solutions and simulations of given physical systems. We will specifically produce numerical solutions for a) well-known problems in financial mathematics which are modeled via stochastic differential equations and b) evolution of vehicular traffic streams.

3.1 Numerical Approximation Of Stochastic Differential Equations

One of the most widely used method for producing the solutions of stochastic differential equations is the Euler-Maruyama method. The method is simple and easy to program.

The Euler-Maruyama method is given by the following pseudo code which approximates the solution of $dX_t = a(X_t)dt + b(X_t)dW_t$ in the interval $t \in [0, T]$ with initial value $X_0 = x_0$:

1. Partition the interval $[0, T]$ into N equal sub intervals (i.e $\Delta t = \frac{T}{N}$).
2. Define recursively $Y_{n+1} = Y_n + a(Y_n)\Delta t + b(Y_n)\Delta W_n$

where $\Delta W_n = W_{n+1} - W_n$ and $Y_0 = X_0 = x_0$.

As one can see, the Euler-Maruyama method is very similar to the Euler method for ordinary differential equations.

REMARK. The order of the error for the Euler-Maruyama method is $\mathcal{O}\left(\sqrt{\Delta t}\right)$ [**10**], where the notation $\mathcal{O}\left(\sqrt{\Delta t}\right)$ denotes the fact that the error for the approximation scales as the square root of the step size Δt.

The Milstein method is a generalization of the Euler-Maruyama. The Milstein method approximates the derivative of the function $b(X_t)$ in order to reduce the error to just Δt. It is similar to the Euler-Maruyama except in the computation of Y_{n+1} above. Instead we compute Y_{n+1} as

$$Y_{n+1} = Y_n + a(Y_n)\Delta t + b(Y_n)\Delta W_n + \frac{1}{2}b(Y_n)b'(Y_n)\left((\Delta W_t)^2 - \Delta t\right)$$

REMARK. The order of the error for the Milstein method is $\mathcal{O}(\Delta t)$ which is better than the Euler-Maruyama method [**10**].

It is possible to apply the above methods to problems that are not autonomous if we assume a small time interval and that the coefficients are continuous. This will however incur extra error for the approximation.

EXAMPLE 4. Consider the SDE initial value problem given by

$$\begin{cases} dX_t = X_t dt + X_t dW_t \\ X_0 = 1 \end{cases} \tag{17}$$

The initial value problem (17) has the analytical solution $X_t = e^{\frac{1}{2}t+W_t}$ [**10**]. Applying the Euler-Maruyama method with step size of $\Delta t = 0.01$ on the interval $0 \leq t \leq 5$ gives the following approximate and analytical solution as can be seen in Figure 2. The absolute error of the approximated solution is presented in Figure 3.

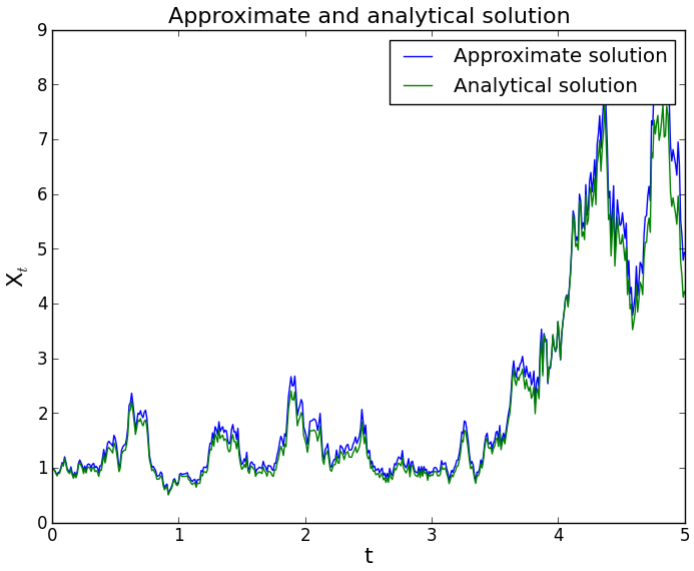

FIGURE 2. Approximate and analytical solution to the initial value problem (17) using the Euler-Maruyama method.

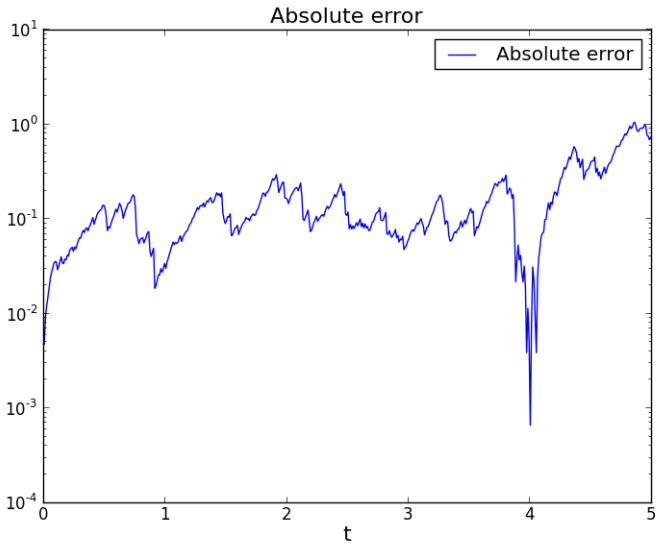

FIGURE 3. Absolute error for the approximated solution of the initial value problem (17).

This next example illustrates how the Milstein method can be applied to solve a different SDE initial value problem.

EXAMPLE 5. Consider the SDE initial value problem given by

$$\begin{cases} dX_t = W_t dt + \sqrt[3]{9X_t^2}\, dW_t \\ X_0 = 0 \end{cases} \tag{18}$$

The initial value problem (18) has the solution $X_t = \frac{1}{3}W_t^3$ **[10]**. Applying the Milstein method with step size of $\Delta t = 0.01$ on the interval $0 \leq t \leq 5$ gives approximated and analytical solution presented in Figure 4 and absolute error presented in Figure 5.

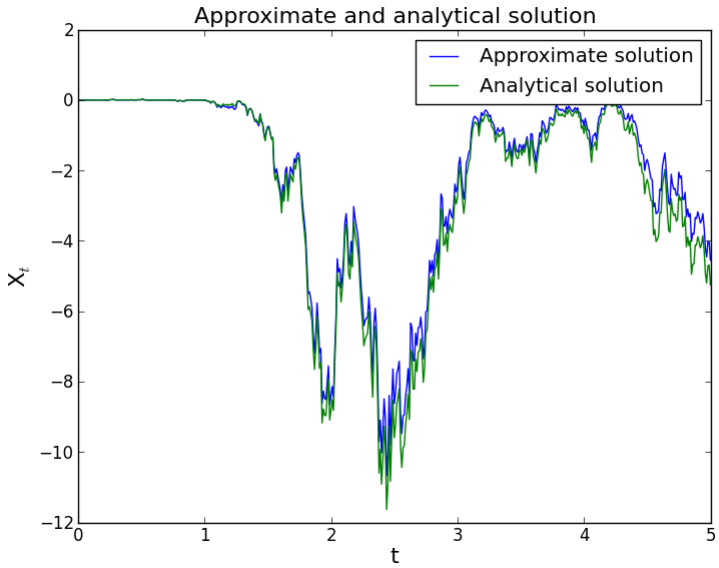

FIGURE 4. Approximate and analytical solution to the initial value problem (18) using the Milstein method.

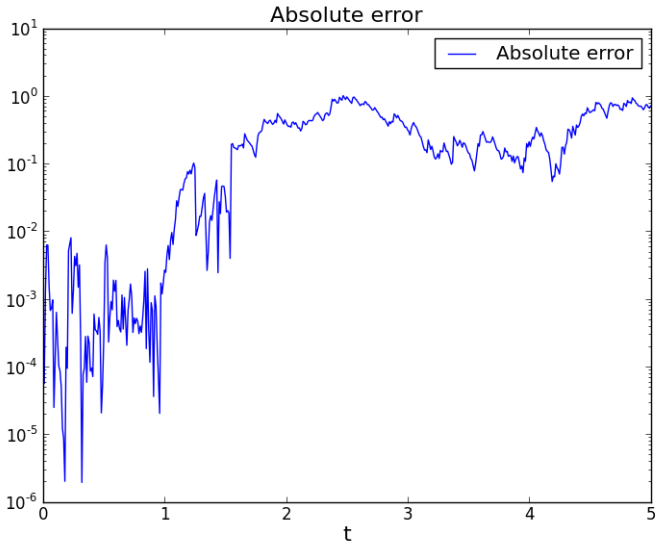

FIGURE 5. Absolute error for the approximated solution to the initial value problem (18).

As we can see from the solution to the Langevin equation in Example 3 we could not obtain closed form analytical solution. However, with the methods we have presented here we are able to produce an approximation to the equation. We use the Euler-Maruyama method to solve the problem since the term σ in front of dW_t is constant, we would not gain additional accuracy by using the Milstein method. The approximated solution is presented in Figure 6.

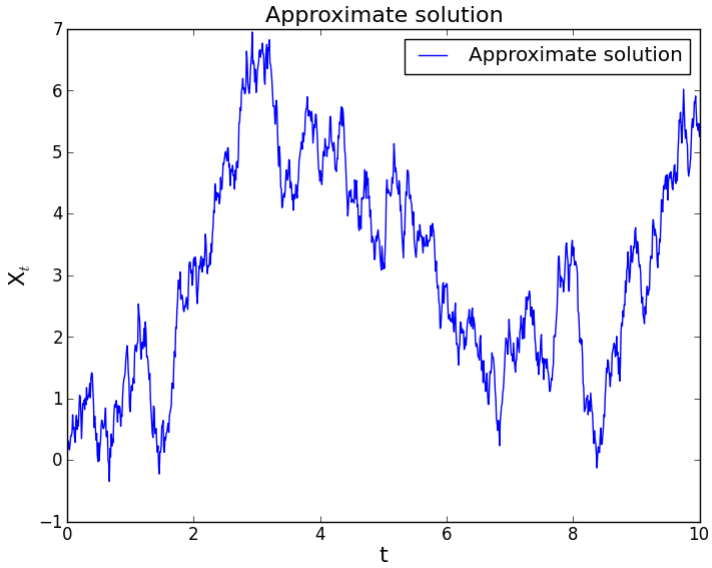

FIGURE 6. Approximate solution to $dX_t = \mu X_t dt + \sigma dW_t$ with $\mu = 0.1$, $\sigma = 2.0$.

3.2 Lattice Based Traffic Flow Simulation On A Closed Periodic Highway

In this section we apply the lattice-based stochastic processes illustrated above toward modeling vehicular behavior on a highway. We will therefore be able to study vehicular interactions and evolutions as well as produce traffic stream evolution and dynamics through our numerical simulations.

For our simulations we initialize the vehicles in a closed periodic lattice domain representing a periodic highway (or "ring-road"). This means that the vehicles can switch lanes upward and downward until they reach the end of the highway. However, when the vehicles have traveled forward to the end of the highway their next step forward will put them in the beginning of the highway. We initialize a given number of vehicles on the highway and keep the number of vehicles constant throughout the simulation. We do not allow vehicles to exit or enter the highway.

To simulate the traffic we implement the Kinetic Monte Carlo (KMC) algorithm [**11**] by first computing the transition rates for all the vehicles on the highway by (14). The rate for movements to occupied cells is always 0 as seen by (14). After the rates have been computed, we calculate the total exchange rate for all vehicles and pick a random number between 0 and the total rate. This random number decides which vehicle should move to a target cell. The pseudo-code for the spin-exchange Monte Carlo Pseudo-code is given bellow [**6**]

(1) Calculate the rate $c(l)$ for all the l nodes in the lattice.
(2) Calculate the total rates by $R = \sum_l c(l)$
(3) Generate a random number ρ.
(4) Index all the rates in an array of size $|\mathcal{L}|$.
(5) Find the node at lattice position k where $0 < k < |\mathcal{L}|$ such that

$$\sum_{j=0}^{k} c(j) \geq \rho R > \sum_{j=0}^{k-1} c(j)$$

(6) Update the time, $t = t + \Delta t$ where $\Delta t = \frac{1}{R}$.
(7) Repeat from step 1 until dynamics of interest have been captured.

The above pseudo-code was implemented in C# to produce the simulations.

To ensure that our model gives accurate simulations of the traffic we need to calibrate the parameters τ_0 and J_0. This is done so that the forward and backwards velocity for the traffic is close to 67 mph and -12 mph respectively which corresponds to realistic traffic conditions[**6**]. In practice, this is done by running the simulations for different values for the two parameters. If one of the velocities is too fast or too slow, the corresponding parameter value is adjusted. In some cases, both parameters needs to be adjusted simultaneously since they each affect each others to a certain degree, especially when it comes to fine tuning the velocities.

We present both a time-space plot of the traffic evolution (Figures 7-9) as well as the important fundamental diagrams (see Figures 10-12) for traffic engineering research. We simulate a multi-lane environment (up to 3 lanes total).

The parameter values were chosen according to the following table

Number of lanes	1	2	3
τ_0	0.23	0.223	0.22
J_0	0.12	0.10	0.10
Forward velocity (mph)	67.2	67.1	67.5
Backward velocity (mph)	-12.4	-12.4	-12.3

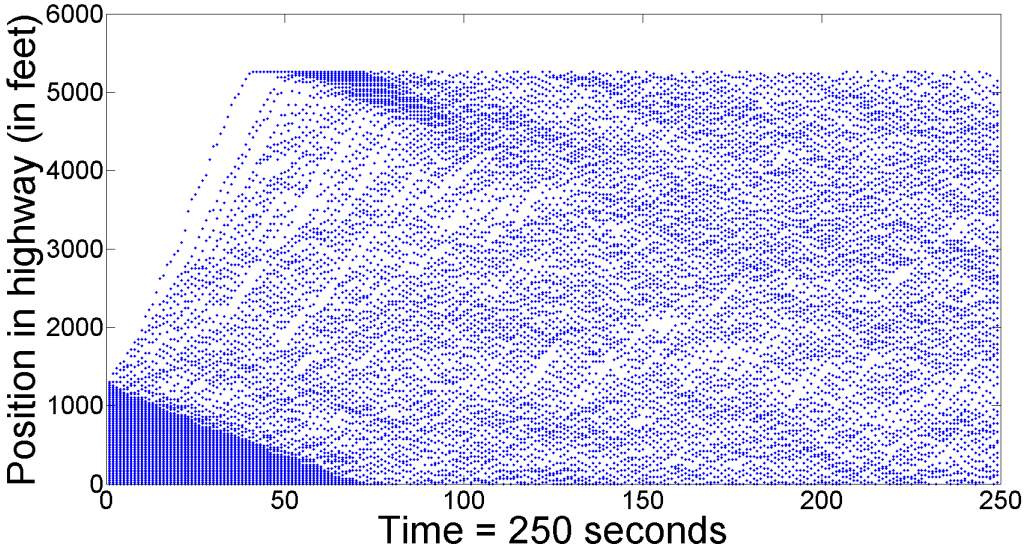

FIGURE 7. Time history plot for a 1 lane highway of 1 mile.

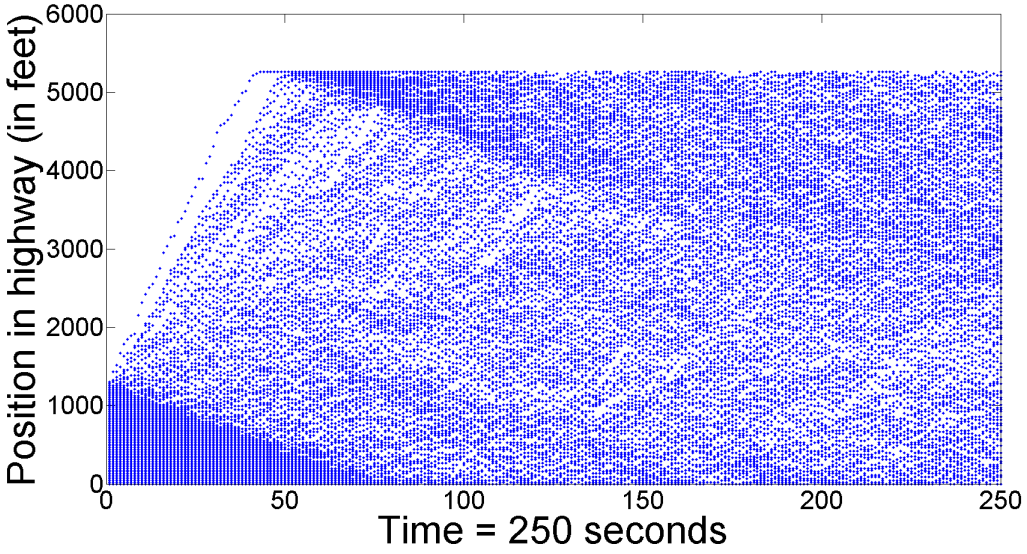

FIGURE 8. Time history plot for a 2 lane highway of 1 mile.

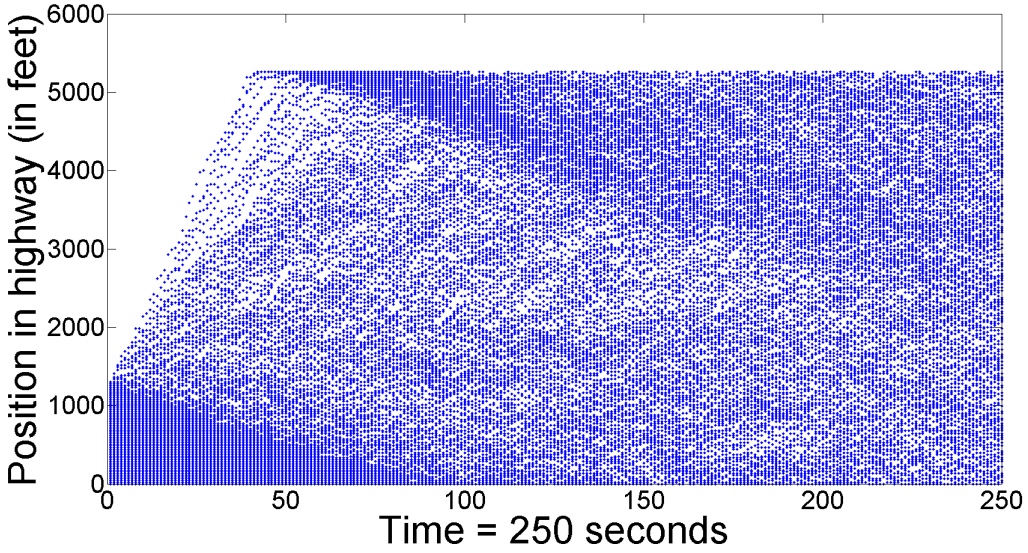

FIGURE 9. Time history plot for a 3 lane highway of 1 mile.

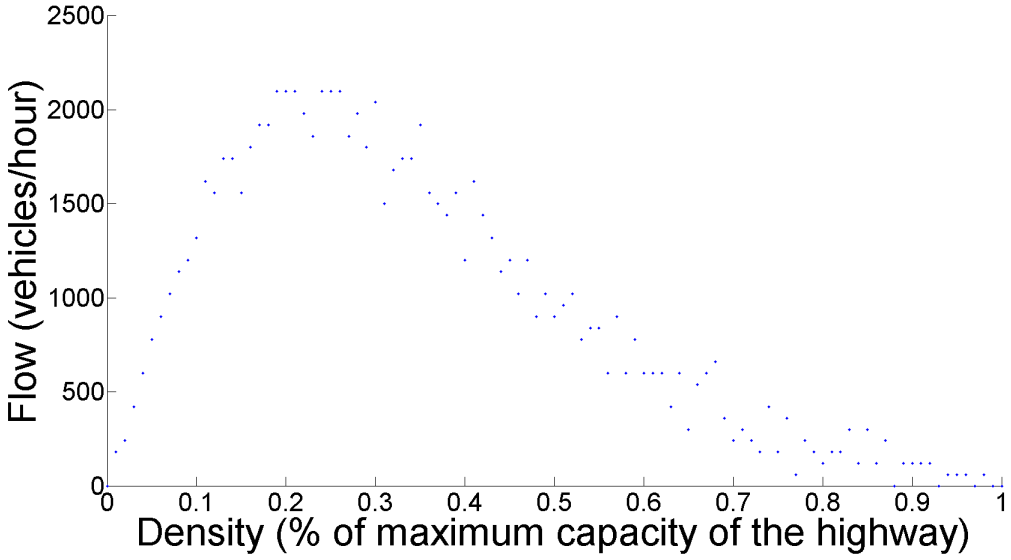

FIGURE 10. The fundamental diagram for a 1 lane highway of 1 mile.

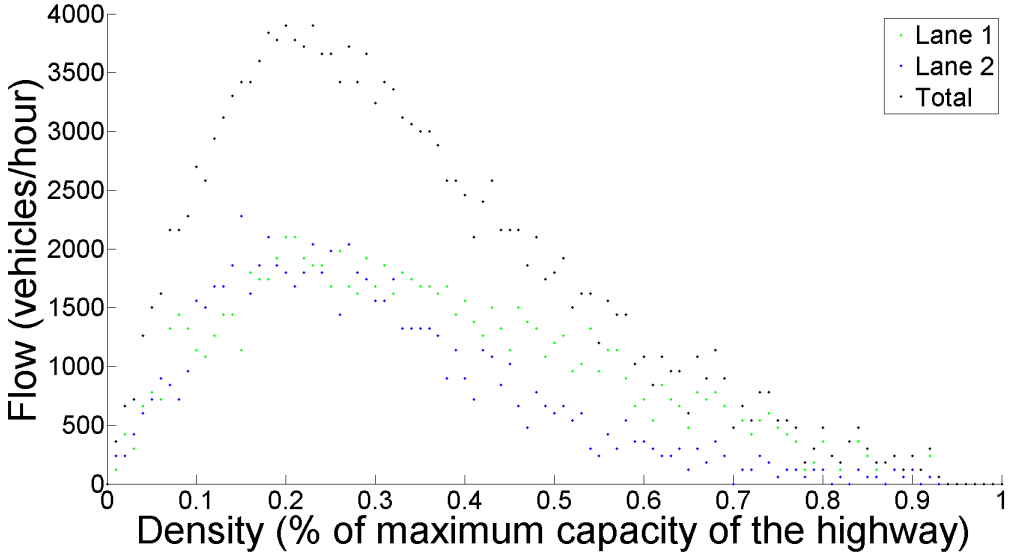

FIGURE 11. The fundamental diagram for a 2 lane highway of 1 mile. Showing the flow in the two lanes respectively as well as the total flow.

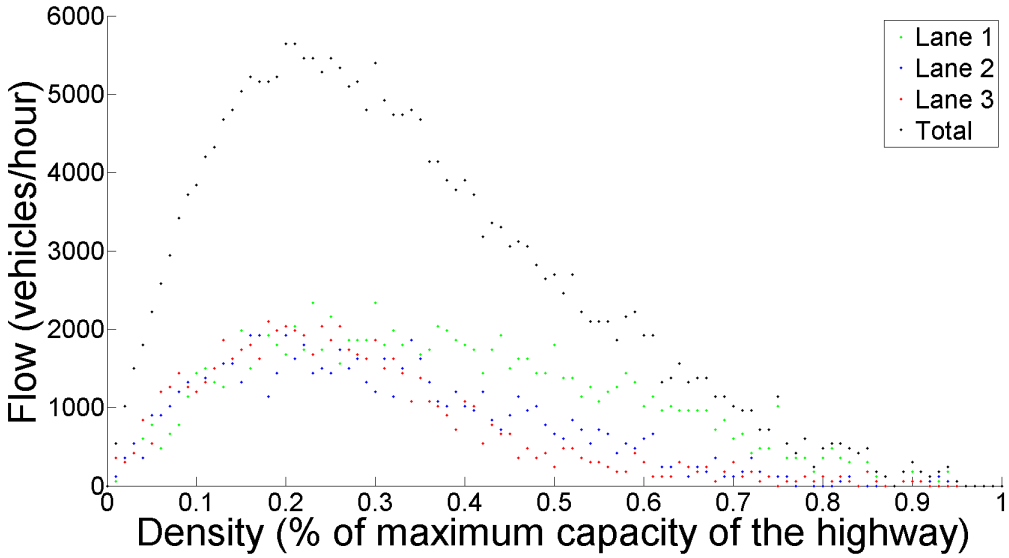

FIGURE 12. The fundamental diagram for a 3 lane highway of 1 mile. Showing the flow in the three lanes respectively as well as the total flow.

3.3 Lattice Free Traffic Flow Simulation On A Closed Periodic Highway

We now also apply the theory for lattice-free stochastic dynamics in order to compare solutions and possible differences between lattice-based versus lattice-free models.

The Monte Carlo algorithm used to produce the simulation is very similar to the one for the lattice based dynamics. The main difference now is that the random number between 0 and the total rate also specifies how far ahead the vehicle should move (up to a maximum of 22 feet forward corresponding to the length of one vehicle). In the lattice based dynamics, each vehicle only took a vertical step when changing lanes. Now, we allow vehicles to move forward in the same time step to create more realistic dynamics. The calibration of the parameters d_0 and J_* is done in the same way as for the lattice based simulations. The Kinetic Monte Carlo algorithm [**11**] was implemented according to the following pseudo code [**9**]

(1) Calculate and index all transition rates c given by (16).
(2) Calculate the total rate $R = \sum_i c(i)$.
(3) Using a random number ρ find the set $j = m$ for which

$$\sum_{j=0}^{m} c(j) \geq \rho R > \sum_{j=0}^{m-1} c(j)$$

(4) Perform the indicated move from an occupied set V to an empty set E signifying motion within the roadway. The random number ρR also decides how far forward the vehicle moves on the highway.
(5) Update the time $t = t + \Delta t$ where $\Delta t = \frac{1}{R}$
(6) Repeat from the beginning until dynamics of interest have been captured.

The above pseudo code was implemented in C# to produce the simulations.

The parameter values were chosen according to the following table

Number of lanes	1	2	3
d_0	9.0	4.5	3.85
J_*	1.7	0.35	0.2
Forward velocity (mph)	67.9	67.1	67.1
Backward velocity (mph)	-12.2	-12.3	-12.3

Figures 13-15 presents the time history of the traffic flow and Figures 16-18 presents the fundamental diagrams for the lattice free simulations.

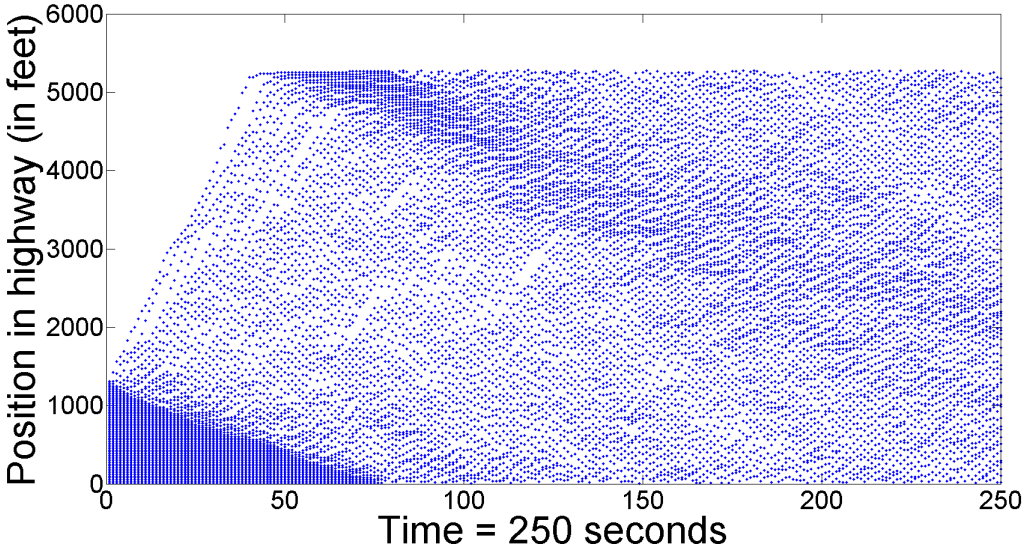

FIGURE 13. Time history plot for a 1 lane highway of 1 mile.

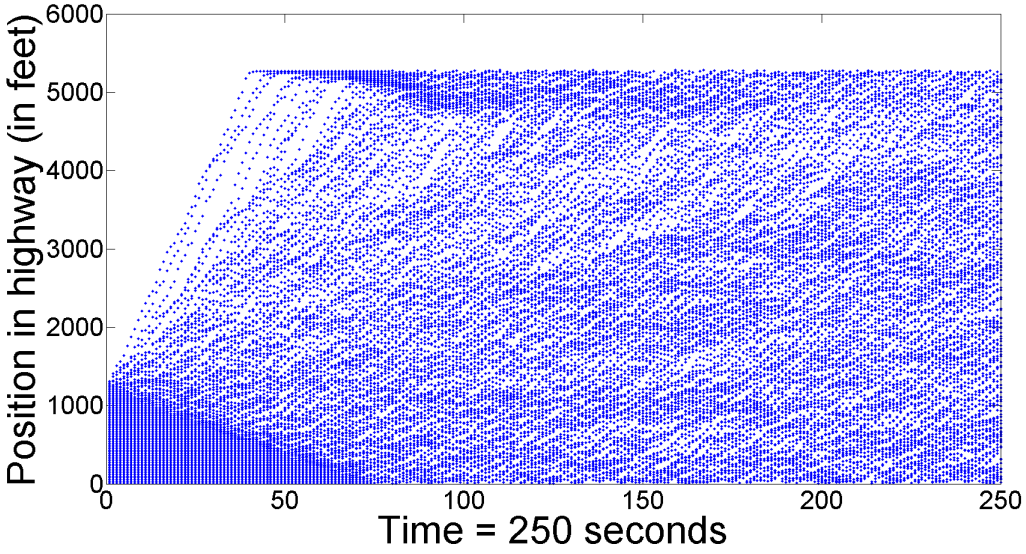

FIGURE 14. Time history plot for a 2 lane highway of 1 mile.

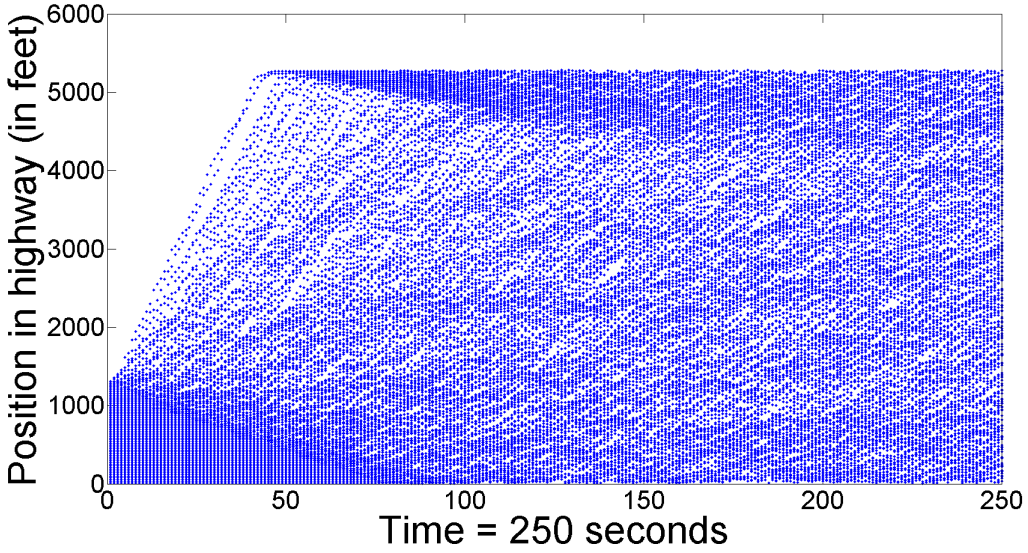

FIGURE 15. Time history plot for a 3 lane highway of 1 mile.

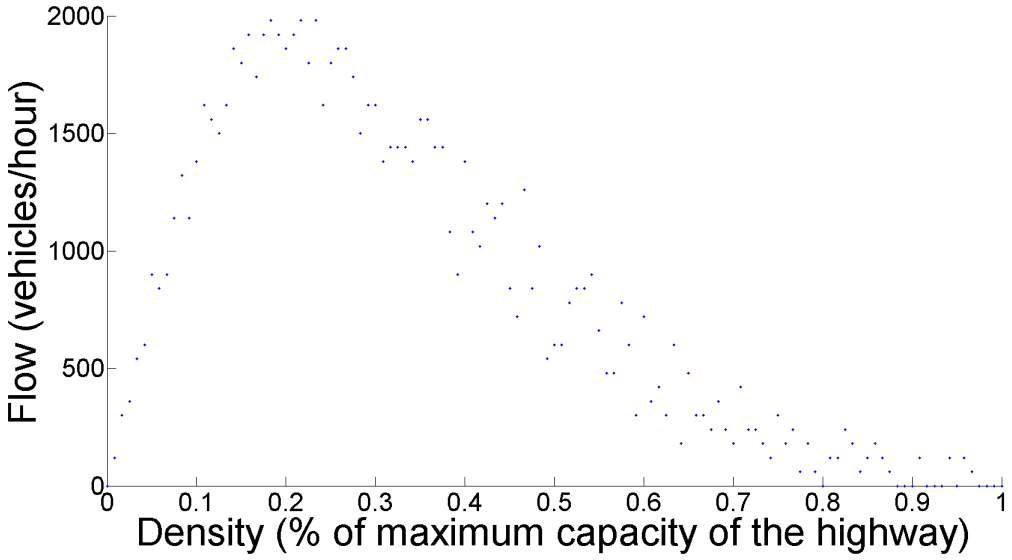

FIGURE 16. The fundamental diagram for a 1 lane highway of 1 mile.

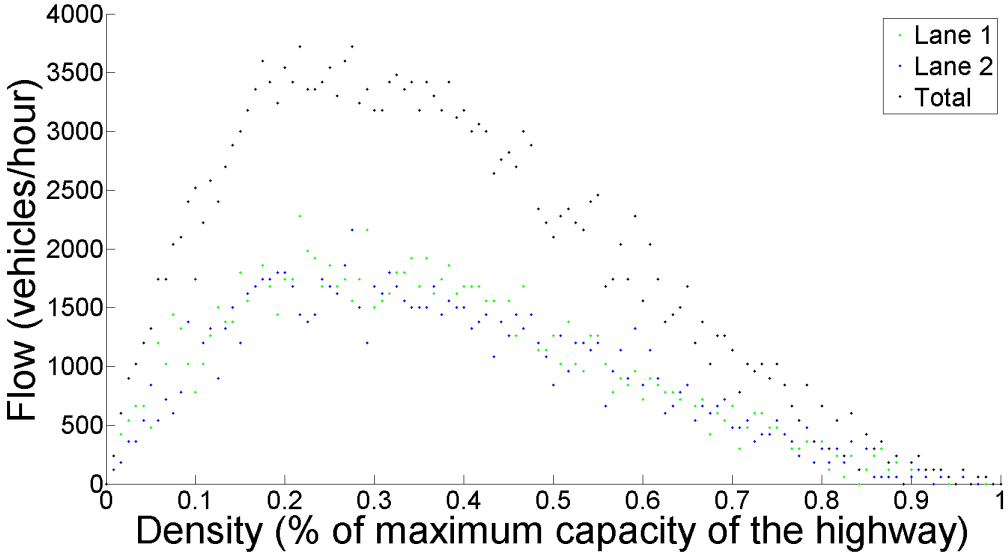

FIGURE 17. The fundamental diagram for a 2 lane highway of 1 mile. Showing the flow in the two lanes respectively as well as the total flow.

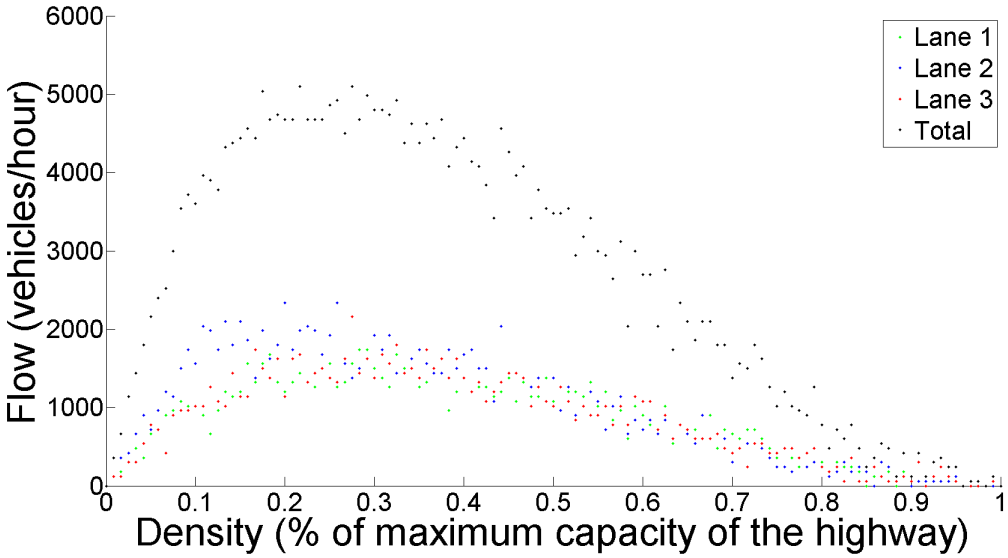

FIGURE 18. The fundamental diagram for a 3 lane highway of 1 mile. Showing the flow in the three lanes respectively as well as the total flow.

3.4 Comparison Between Lattice Based And Lattice Free Models

As we can see from the time history in Figures 7-9 and Figures 13-15 the forward velocity is approximately 67 mph since it takes the first vehicle approximately 40 seconds to reach the end of the 5280 feet long highway, starting 1320 feet in. We can also see that the backwards velocity is approximately -12 mph since the last vehicle in the traffic jam starts to move after 75 seconds. This is true when we have 1, 2 or 3 lanes. We can also see that the first vehicle has to wait after it has traveled to the end of the highway before it can move (the end of the highway is defined by our starting reference point since it is a periodic highway). This validates the fact that there are no stacking issues for the vehicles. The first vehicle has to wait before it can move because the last vehicle in the initial jam has yet to move. We can also see that the traffic builds up at the end of the roadway, creating a secondary traffic jam, before dissolving completely after 200 seconds where we observe free flow for the traffic.

The fundamental diagram presented in Figure 19 gives the relationship between traffic density and traffic flow on a highway. Traffic density is supposed to be measured in vehicles per mile while traffic flow is measured in vehicles per mile per hour.We see that the flow should increase until the density reaches c_{crit} which is the critical density at around 25 percent. It should thereafter decrease until the density reaches the jam density c_{jam} [**12**].

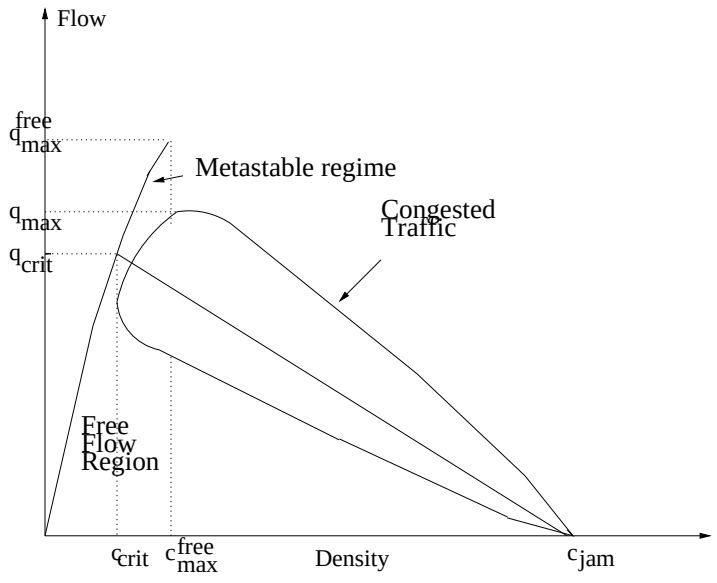

FIGURE 19. The fundamental diagram (flow-density relationship) [**12**].

In the fundamental diagrams presented in Figures 10-12 and Figures 16-18 we see that they agree with the fundamental diagram in Figure 19. This is a strong validation that our simulations produces accurate results since the overall shape of the fundamental diagrams corresponds well with widely accepted standards for traffic research [**6, 12**].

The fundamental diagram is a very important tool for validating traffic simulations. The way those diagrams were produced was by randomly placing vehicles corresponding to a density on the highway and then monitor how many of them passes a camera that is placed on the side of the highway. From this we can determine the flow of vehicles on the highway under a certain time period. The interesting question now is which of the two methods presented here gives the most realistic description of traffic flow?

As we can see when comparing the fundamental diagrams for the lattice based method (Figures 10 - 12) with the fundamental diagrams produced using the lattice free method (Figures 16-18) we see that there are differences. The peak for the critical density occurs earlier for the lattice based stochastic process and it is also more narrow compared to the lattice free stochastic process. We can also observe that the flow declines more rapidly for the lattice based method. This indicates that there is a difference between the methods.

To determine which of the two processes produces the most accurate results we use the Palasti conjecture. According to this conjecture objects of equal size are placed randomly in a closed domain, the overall coverage of the domain will be approximately 0.746 on average [**13**]. Figure 20 bellow shows the domain size vs density for both the lattice based and the lattice free method as well as the Palasti conjecture.

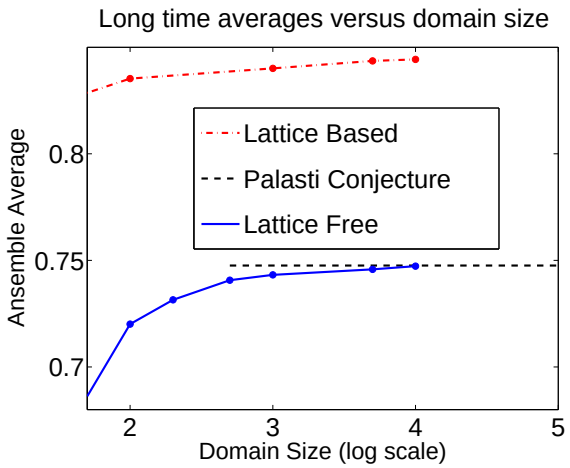

FIGURE 20. Initialization density vs domain size [**9**]. Figure reproduced here with permission from author.

As we can see from Figure 20, the lattice free approximation is in complete agreement with the Palasti conjecture. From this we can conclude that the lattice free method indeed gives a more realistic description of traffic flow.

Provided this significant finding for lattice free versus lattice based methods it would only be natural to use such a more accurate approach for our next project in modeling swarming behavior. Such a stochastic description proposed above will be tested in simulating the interactions of schools of fish. It will also be interesting to extend further our model in order to include predatory behavior from sharks for instance.

Bibliography

[1] Bernt Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer; 6th edition, Heidelberg Dodrecht London New York, 2010.

[2] K.Nagel and M. Schreckenberg. A cellular automaton model for freeway traffic. *J. Phys. I*, 2:2221–2229, 1992.

[3] A. Schadschneider. Traffic flow: a statistical physics point of view. *Physica A*, 312:153, 2002.

[4] A. Sopasakis and M. Katsoulakis. Stochastic modeling and simulation of traffic flow: asymmetric single exclusion process with arrhenius look-ahead dynamics. *SIAM Journal on Applied Mathematics*, 66:921–944, 2006.

[5] D.G.Vlachos and M.A.Katsoulakis. Derivation and validation of mesoscopic theories for diffusion of interacting molecules. *Phys. Rev. Let.*, 85 (18):3898, 2000.

[6] T. Alperovich and A. Sopasakis. Stochastic description of traffic flow. *Journal of Statistical Physics*, 133:1083–1105, 2008.

[7] T.M Ligget. *Interacting Particle Systems*. Springer, Berlin, 1985.

[8] A. Sopasakis. Lattice free stochastic dynamics. *Communications in Computational Physics*, 12:691–702, 2010.

[9] Alexandros Sopasakis. Traffic updating mechanisms for stochastic lattice-free dynamics. In *20th International Symposium on Transportation and Traffic Theory (ISTTT 2013)*. Elsevier Ltd., 2013.

[10] Timothy Sauer. *Numerical Analysis*. Pearson;, USA, 2005.

[11] A. B. Bortz, M. H. Kalos, and J. L. Lebowitz. A new algorithm for monte carlo simulation of ising spin systems. *Journal of Computational Physics*, 17:10–18, 1975.

[12] A. Sopasakis. Stochastic noise approach to traffic flow modeling. *Physica A: Statistical Mechanics and its Applications*, 342:741–754, 2003.

[13] I. Palasti. On some random space filling problems. *Publ. Math. Inst. Hungar. Acad. Sci.*, 5:353–360, 1960.

Bachelor's Theses in Mathematical Sciences 2013:K15
ISSN 1654-6229

LUNFNA-4002-2013

Numerical Analysis
Centre for Mathematical Sciences
Lund University
Box 118, SE-221 00 Lund, Sweden

http://www.maths.lth.se/