Rapid Simulation of Urban Traffic using FPGAs

Gordon Russell Paul Shaw John McInnes
Department of Computer Science

Neil Ferguson
Department of Civil Engineering

University Of Strathclyde
26 Richmond Street
Glasgow, G1 1XH
Scotland, UK

May 1994
AIS-CIA-2-94*
Keywords: Traffic Simulation, SPACE Machine, FPGA

Abstract

Microscopic simulation of complex traffic models can take hours or even days on simulators build to run on a single processor machine. These simulators use continuous models for traffic movement. In this report, we introduce a discrete simulation system that models individual vehicles. Although we have abandoned the notion of continuous movement of vehicles, we achieve highly accurate results gained from our simulations of a road-merge bottleneck and a signalized junction.

Our eventual target simulation platform is field-programmable gate arrays (FPGAs) which allow us to construct arbitrary random logic circuits. Discrete modelling is essential for efficient implementation on such devices. This paper describes preliminary work towards simulating road networks at high speed directly in hardware.

1 Introduction

With the increase in road usage, traffic engineers are finding that modern traffic flow simulators are simply not up to the job. Engineers require more complex networks to be simulated, while

*This work was funded by SERC grant number GR/J09239.

wanting the run-times for the simulations to be reduced significantly from current levels. This had lead the charge for parallel implementations, with the simulation programs being distributed across multi-million dollar machines. This may be acceptable for the largest of bank-balances, but perhaps there is a much cheaper way.

In an earlier report, [3], we proposed a scheme for simulating traffic using Field Programmable Gate Arrays (FPGAs). This uses reconfigurable hardware to model road networks as logic devices. For simplicity, discrete modelling of car movements are used, as opposed to more continuous schemes. Each roadway is constructed from a number of road cells, with each cell either containing or not containing a car. The initial results of this was to indicate that such an approach was feasible, although more work would be required to allow calibration of the network components. This report documents the calibration of roadways, and demonstrates the effectiveness of discrete traffic modelling at the microscopic level.

Before starting to investigate roadway construction, first consider the macroscopic behaviour of traffic on a road. Traffic performance can be described by three parameters; flow q, density k, and speed u. Flow can be considered as the number of cars passing a certain point in the road. Density in a discrete model is probably best measured by the number of cars present within an length of roadway. Speed can not be based on cars passing a single point, since in the discrete model either a car passes or its does not. Later we will show how speed is implemented, and this (like density measurement) is based on an average speed over a number of road cells. This has directed us to break lengths of road into short stretches of road cells. We call these stretches *segments*. Measurements of speed, flow, or density can only be made over an entire segment.

Our objectives for the roadway model include:

- Each road segment should be able to mimic an average speed specified by the designer. The speed should be implemented as accurately as possible, but there is no requirement to allow for a linear range of speeds. Limiting the selection of speeds to 20, 30, 40, 50, 60, and 70 mph is acceptable, but having other possible speeds would be an advantage.

- As the flow of cars on a road segment increases, the average speed of cars should decrease slightly in line with observations. As an estimate, a minimum drop of 5% in average speed should be measured when the flow approaches saturation.

- The capacity for each road segment should be definable by the designer. The maximum flow predicted for any single lane is defined as 2500 pcu/hour. No minimum flow is specified, but no capacity below 900 pcu/hour is expected.

Once the model for the road segments has been described, it will be demonstrated in two examples; a roadway bottleneck, where approach roads carry more traffic than can be handled on subsequent road segments, and a full signalized junction, where the road segment model is simulated with junction components designed earlier.

2 Speed Control

The model used to simulate roadways is based on discrete movement of cars between road cells. Each cell can hold only a single car. Cars can move from a full cell to an empty cell in a single simulation tick. Cars cannot move into full cells, and instead wait until the destination cell empties. If every cell in a road was full, then we define the distance from the start of one car to the next as being five metres.

Consider a car cell. At maximum throughput, a cell which is empty on tick one will become full on tick two. The cell will then empty itself on tick three, and then becomes full again on tick four. This indicates that each roadcell has a throughput of one pcu every two simulation ticks. This requires that each tick be equivalent to 0.8 seconds if the maximum flow is to be 2250 pcu/hour. With each road cell covering an length of 5 metres, the simulated speed of this model must be 14 mph.

Even in urban simulations, 14 mph along all roads is not going to produce satisfactory results. At this point, we define the current road cell as being a *normal* cell, and introduce a new *fast* road cell. This cell acts as a normal cell if either the destination cell or itself holds a car, otherwise it operates in a transparent mode. In this mode, the fast cell can be considered as being invisible, with the previous cell appearing to be connected directly to the cell immediately after the fast cell.

The fast cell has the effect of teleporting cars over its 5 metres of simulated road whenever the subsequent car cell is empty. If the next cell is full, then the fast cell acts as a normal one. When normal and fast cells are interspersed on a road in a 1:1 ratio (*i.e.* one fast cell, one normal cell, one fast cell, *etc.*), then the simulated speed of cars is doubled; cars travel between normal road cells in a single tick (under free-flow conditions), appearing to travel 10 metres (a normal cell and a fast cell). This corresponds to an average speed of 28 mph. By varying the interspersion of fast and normal cells, the speed of a road can be controlled with reasonable accuracy.

To calculate the number of fast and normal cells required for a specified speed v (measured in mph) over a length d (in car cells) of road segment, use:

$$\text{normal} = 14d/v \tag{1}$$

$$\text{fast} = d - \text{normal} \tag{2}$$

Since the number of normal and fast cells are integers, then as the length of each road segment increases, the number of possible speeds reproducible increases. At low speeds (less than 40 mph), relatively short segments ($<$ 15 cells, which would represent 75 metres) can produce reasonable accuracies of requested speeds. At higher speeds, the number of cells required for reasonable accuracy increases, although road segments of 40 car cells (*i.e.* 200 metres) can produce just about any desired road speeds. This is not a major problem, since high speeds will most likely mean longer roads between neighbouring junctions, which in turn allows for longer road segments. Where the distance between junctions is small, speeds tend to be less than 40 mph anyway. High speeds in short road segments are always possible, but the accuracy of the resultant average speed may be outwith desirable limits.

3 Controlling the Speed/Flow Relationship

One of the initial aims of the roadway model was to implement the slight decrease in vehicle speed as the flow increases towards the saturation limit. If we consider a good model, such as the three-regime (proposed by [1]), plotted as speed against flow, then the non-blocking area of the model is mimicked by a line with a slightly downward slope. It is suggested this this effect could be realized in a road segment by converting a fast cell to a normal cell once the flow at that point in the segment approaches saturation. This is implemented using a *pseudo* cell.

The pseudo cell monitors the condition of the cell immediately after itself, coupled with a state bit held internally. In normal (state bit clear) operation, the pseudo cell acts as a fast cell. If the cell propagates a car or becomes full itself, then the state bit becomes set. Whenever the state bit is set, the pseudo cell operates as a normal cell. The state bit can only be cleared when both the next cell and itself have been empty for an entire tick. In this way, the pseudo cell becomes a normal cell when the flow of cars at that point in the segment have reached half of the 2250 pcu/hour maximum capacity.

To understand the effect of using pseudo cells, consider a road way constructed using a single lane. The road is constructed using a number of road segments, each of 20 road cells calibrated for 30 mph (just over 48 kph). Due to rounding errors, the average speed on the road segment is 50 kph. Each segment has either none, one, two, or four pseudo cells in place of a corresponding number of fast cells. A few hours worth of traffic is played into the roadway. This simulation is also performed using a traffic light on the roadway, causing a range of blocking delays on the road. The resultant speed/flow measurements is shown in figure 1. For reference, the three-regime model is also plotted onto the graph. All speeds below about 15 kph were caused by the simulation containing a traffic light (to demonstrate the effects of queueing).

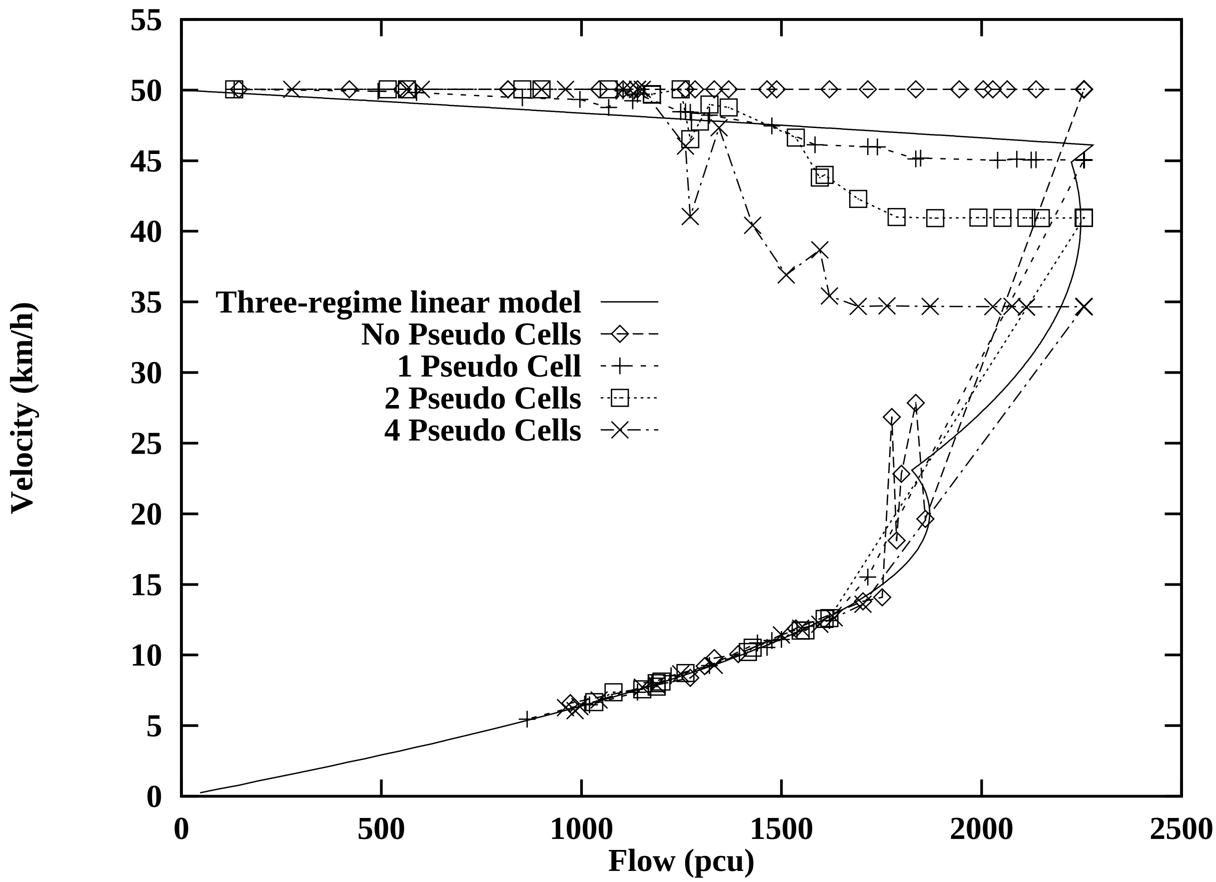

Figure 1: The Effect of Pseudo Cells on the Speed/Flow Relationship (30 mph)

With no pseudo cells, the non-blocking part of the graph (the top half) remains completely flat until the saturation flow of 2250 pcu/hour is reached. With a single pseudo cell, a close match to the three-regime model is achieved. With more than a single pseudo cell, the resultant graph deviates strongly from the theoretical model. This would suggest that a single pseudo cell per 20 road cells is adequate for implementing the required decrease in velocity suggested by the three-regime model.

4 Capacity (Saturation Flow) Control

The capacity of each road segment is currently 2250 pcu/hour. The specification of the road model requires that the saturation flow be variable, with a maximum allowable flow of 2250 pcu/hour. What is not required is some way to control the maximum flow on a segment.

Consider the distribution of cars at maximum flow. Only every second car cell will be full, with the other half of the cells empty. If the desired flow was half of 2500 pcu/hour, then the spacing (or headway) between cars would have to be increased from 2 to 4. This indicates that the headway d required for a particular flow q would be:

$$d = 4500/q \tag{3}$$

At this point a new cell is introduced, the *propagating normal cell*. This cell is a duplicate of the normal cell, except that when empty, it propagates back the state of the next cell. Thus a propagating normal cell and a normal cell would propagate back a full signal when either or both of the cells were full. This propagation does not interfere with the normal cell handshaking, as it is turned off when handshaking for delivery of a car commences.

If a stretch of road (with cells named A,B,C,. . .) begins with a lead delay cell, which stops traffic when the next cell signals back that it is full, then if cell A is a normal cell the lights have no effect. If cell A is a propagating cell but cell B is a normal cell, then the lights would ensure that cell A was empty whenever cell B was full. This equates to an inter-car spacing of 2. If cell A and B were propagating cells, and cell C was a normal cell, then the inter-car spacing over cells A to C would be three, and so on.

In model terms, inter-car spacing can be enforced by beginning with the lead delay cell, followed by $\lfloor d - 2 \rfloor$ propagating normal cells. Naturally, we can only use the integer part of d, as there can be only an integer number of cells in the model.

The fractional part of d can be catered for using a random delay element. The delay required is small enough not to interfere with the average speed of vehicles moving along the roadway. This would not be true if the delay was used for both the integer and fractional part of d. The random delay element is placed after the propagating normal cells, and uses a delay d calculated from:

$$d = \frac{1}{\frac{1}{d - \lfloor d \rfloor} + 1} \tag{4}$$

As an example of the saturation flow control system, consider a roadway made from a number of road segments, calibrated to 30 mph. Cars are delivered onto the roadway in a simulation of peak traffic flow, with a maximum flow of 2250 pcu/hour. The roadway is simulated three times, using saturation flows of 1400, 2000, and 2250 pcu/hour. The resultant speed/flow relationship can be found in figure 2.

As the saturation flow is decreased, the effects of the pseudo cell are removed. For simplicity in constructing the road segments, the pseudo cell is used independently of the requested saturation flow.

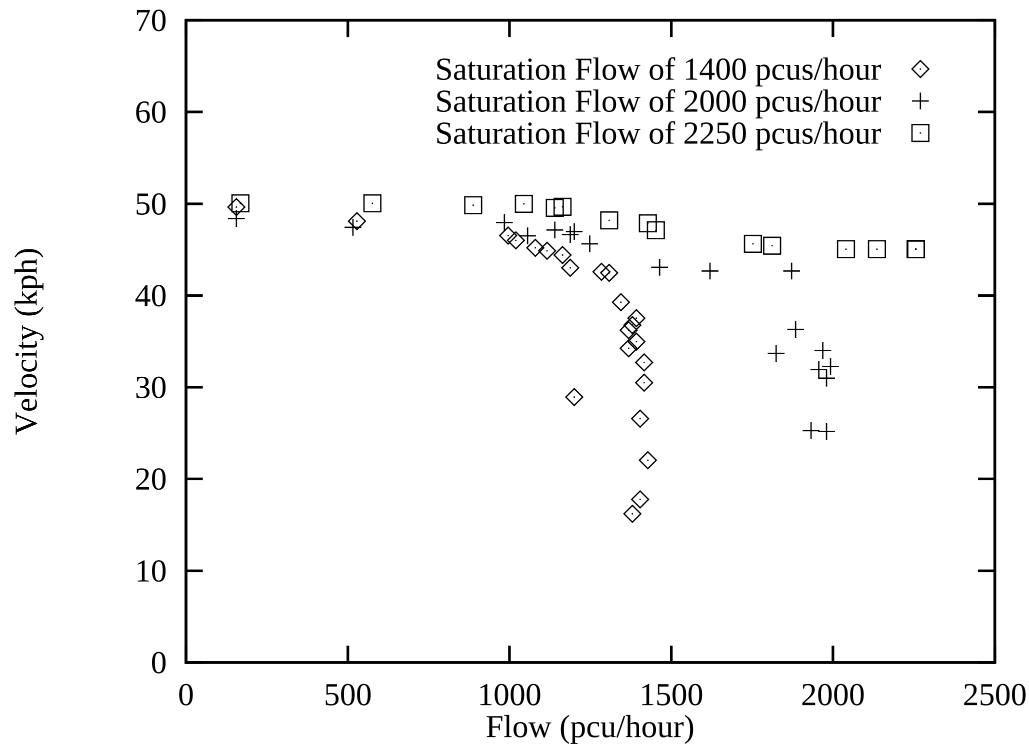

Figure 2: Effects of Saturation Flow Control (30 mph)

5 Layout of a Road Segment

There are two layouts used in the construction of a road segments, and these are shown in figure 3. Whenever possible, road segments are 20 cells long. The lead delay cell is only included if propagating normal cells are required. The pseudo cell is always used, independent of the requested saturation flow.

If the length of road requested means that less than 20 road cells must be used, then the requested length is used instead. This may result in less accurate average speeds.

6 Model of a Roadway Bottleneck

As an example of the ability to model traffic over roadways, consider a roadway carrying traffic to the central business district during the morning peak period. Segments of interest on the roadway is marked A, B, C, and D. Segment A marks the beginning of a two-lane roadway, with each lane calibrated to 1500 pcu/hour saturation flow. Segment B is the last segment in this two-lane roadway. Segment C is a single lane, with a saturation flow of 2000 pcu/hour. Cars from the two lanes in B are merged and passed into C. After C, cars travel along across

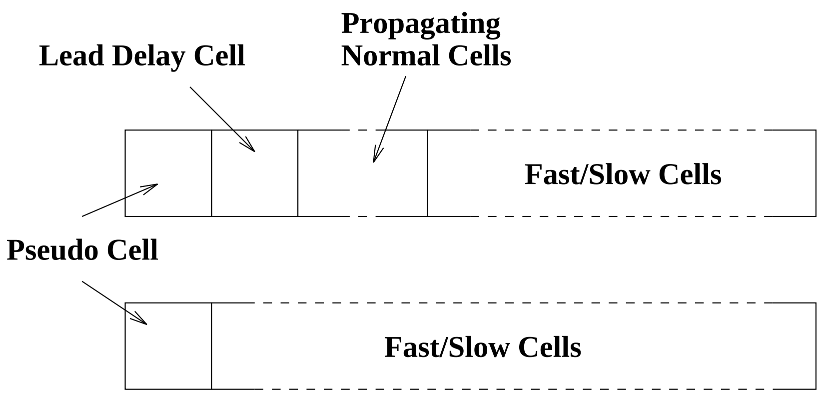

Figure 3: Layout of Road Segments

an second 2000 pcu/hour single lane, before being split into two lanes (at segment D), each with a flow of 1500 pcu/hour.

The simulation is split into two timezones. In the first zone, between 6:00 and 7:30 a.m., the traffic demand increases from a very small flow up to a flow equivalent to 1000 pcu/hour/lane. This should bring the flow up to saturation on segment C (the bottleneck). The results of this are drawn on figure 4 as diamonds. While segments A, B, and D are operating at only two-thirds of capacity with relatively high speeds and low densities, segment C has reached its capacity with significantly lower speeds and higher densities. In the graphs, speed in measured in kph, flows in pcu/hour, and density in cars/km.

Now assume that the demand from 7:30 to 8:00 a.m. increases to a flow equivalent to 1250 pcu/hour/lane. The results of this part of the simulation is plotted using pluses in figure 4. This second timezone lasts for only 30 minutes, and should result in a queue of 125 cars per lane. Segment A is 200 car cells away from the bottleneck, and so no queueing ever reaches that point. Segment B, which is effected by the queueing, becomes congested with high densities and low flows. Segment C, has a capacity of only $2/3$ the saturation flow of the approach roads, operates at its saturation flow with fairly optimum density and flow. Finally, segment D, being supplied only with 2000 pcu/hour (the capacity from segment D), operates with fairly high speeds and low densities.

In both timezones, the results gathered are pretty much those expected, and there is a close correlation between these results and those described under a similar experiment in [2, pages 228-290]. This would suggest that, even with the simple traffic behaviour model used here, reasonably accurate traffic simulations can be performed.

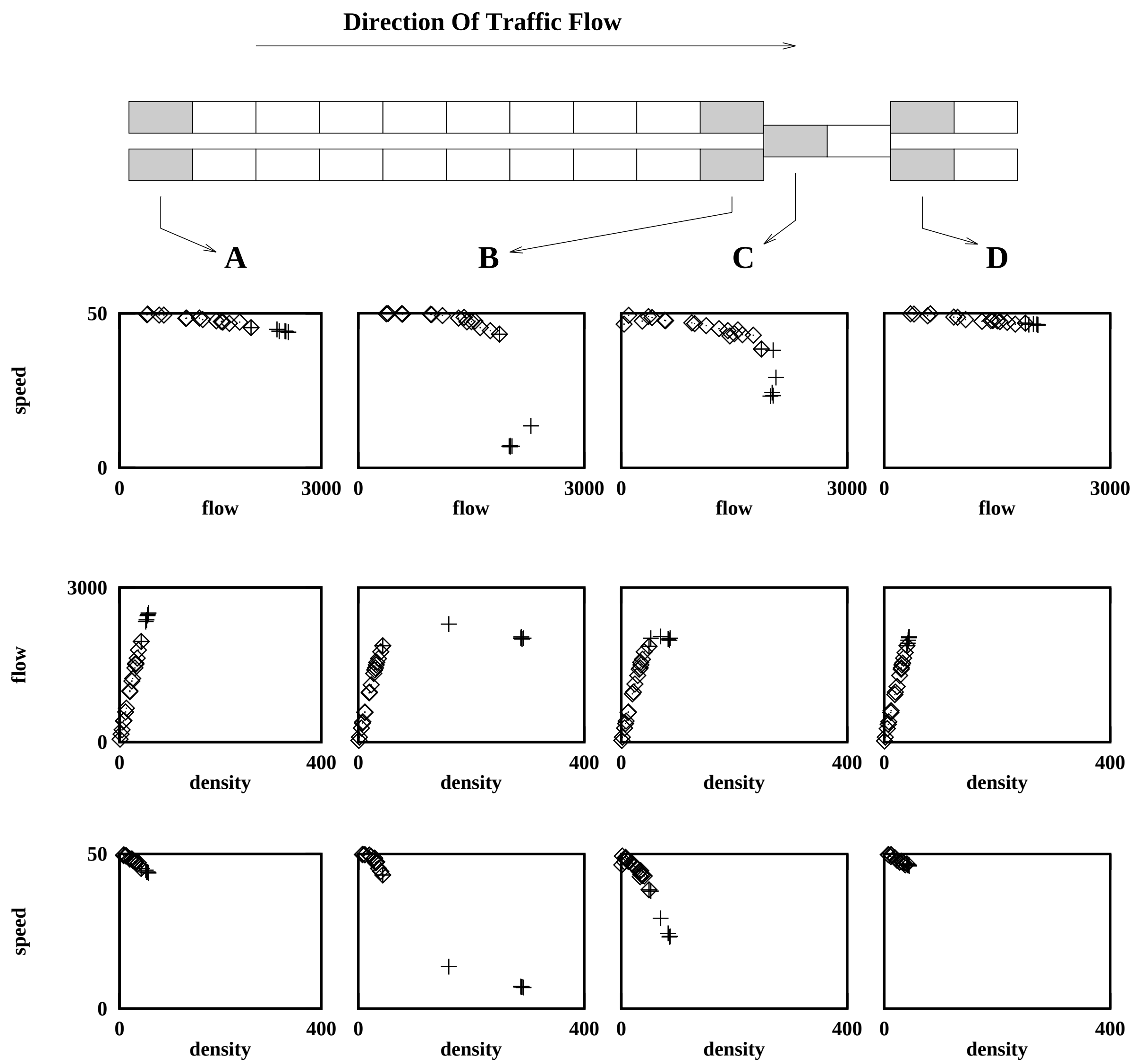

Figure 4: Results of the Bottleneck Simulation

7 Model of a Signalized Junction

A four-way, two-phase, signalized junction was simulated using the roadway models described. Statistics were gathered on the average delay per vehicle that the traffic lights incur. That is, what is the average time a car takes to traverse the traffic lights, less the time taken when the lights are green and there is no traffic queued back?

Each of the junction's four input roads has a saturation flow of 1800 pcu/hour and speed of 50 kph. The junction was fed with a consistent and equal load q on all four links, and simulated for a long enough period of time for the random fluctuations in the average delay per vehicle to reduce to acceptable levels. We have constructed this particular model such that there is sufficient space within the junction for cars to bypass others waiting to turn right. Green times on each phase are equal, and there is a total lost time of 10 seconds per cycle.

The experiment was run for four different input loadings such that $q \in \{400, 600, 700, 750\}$. In each case a range of light cycles, between 10 and 160 seconds, were simulated, and the average delay per vehicle recorded. Each light cycle simulation was executed seven times to help remove random variations, although these are still visible in the results obtained. These results are shown in figure 5. These results suggest optimum cycle times of approximately 36, 59, 90, and 110 seconds for each of the four loadings. This compares well with [4][Table 36.1] which gives optimum theoretical times of 36, 59, 91, and 125 seconds, obtained using a more traditional simulation model.

The measurement of delay has to take into account a number of factors. A delay of one tick is assumbed both when a car has not moved in the last timestep, and the same delay is also assumbed when a fastcell accepts a car without automatically propagating it. Note these delays are not cumulative within a single cell in a single clock tick.

8 Conclusions

Microscopic simulation of complex traffic models can take hours or even days on simulators build to run on a single processor machine. These simulators use continuous models for traffic movement. Our simulation system still models cars (as opposed to just flow), but the model itself supports only discrete movement. Although intuitively this would suggest that the results gained from such a model would be highly inaccurate, the results gained from both the simple bottleneck network and the four-way signalized junction suggests that our modelling technique has a high degree of accuracy.

Discrete modelling is essential for efficient implementation onto the target execution platform; the FPGAs. Implementing continuous models of traffic would be expensive in logic

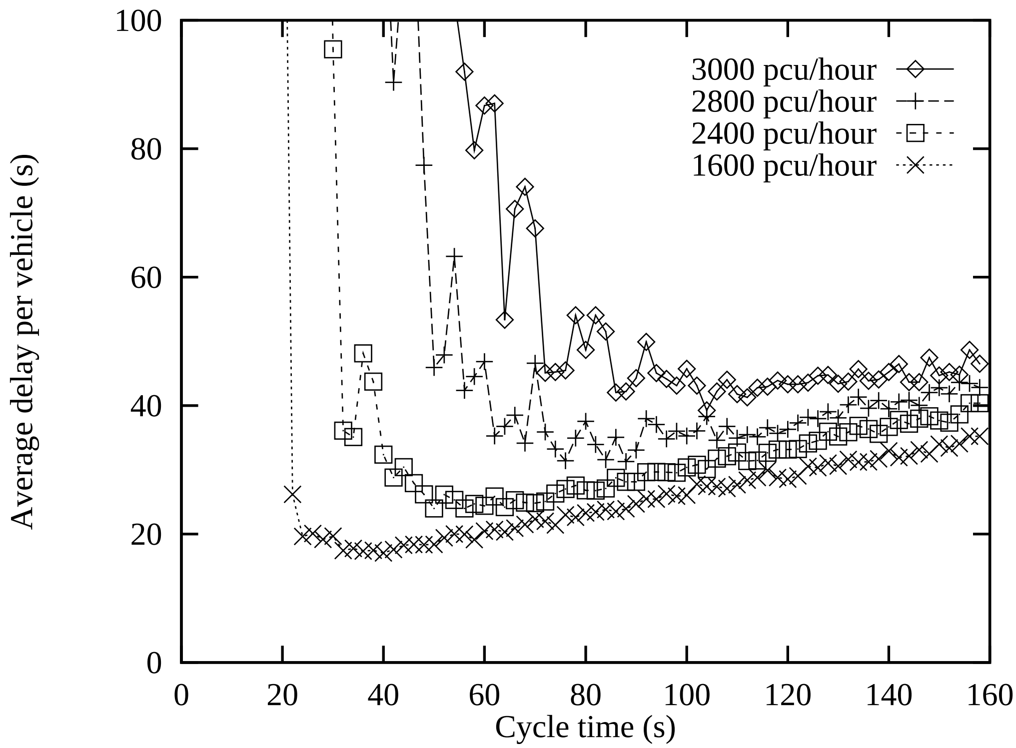

Figure 5: Delays per Vehicle for a Signalized Junction

terms, making successful implementation of our model in reconfigurable hardware practically impossible. The discrete modelling approach may have other uses, since the amount of effort required in simulating continuous traffic motion in software is much greater than that required for discrete modelling. This would suggest that a software-based traffic simulator could be rewritten to use discrete modelling, gaining increased execution speed without loss of accuracy.

The next stage of the work involves moving the modelling mechanism from the current software simulation to the FPGA implementation. To aid in this, the components have been described in Circal, allowing us to verify the implementation of the components in logic against the Circal specification. This should reduce our debug time considerably. Once the conversion to hardware is complete, more complex traffic problems can be examined.

References

[1] J. S. Drake, J. L. Schofer, and A. D. May. A statistical analysis of speed density hypotheses. In *Third International Symposium on the Theory of Traffic Flow Proceedings*, New York, 1967. Elsevier.

[2] Adolf D. May. *Traffic Flow Fundamentals*. Prentice Hall, 1990.

[3] Gordon Russell, Alex Cowie, John McInnes, Martin Bate, and George Milne. Simulating vehicular traffic flows using the circal system. Technical Report aiscia-1-94, University of Strathclyde, June 1994.

[4] R. J. Salter. *Highway Traffic Analysis and Design*. MacMillan Education, second edition, 1990.