Modelling and Simulation: The Biham-Middleton-Levine Model

Kareem Al-Saudi - S3877043

Abstract—Traffic jams are a common problem that can be found in many a country's infrastructure on a global scale. Within the current literature, a plethora of models exist that aim to simulate their formation both in an attempt to help us better understand and reinforce our understanding of their generation as well as to potentially uncover previously unconsidered reasons associated with their emergence. This paper seeks to take a closer look at one such model, namely the Biham-Middleton-Levine traffic model, in order to get a baseline of understanding as to how and why traffic jams occur.

Index Terms—Traffic Jams, Biham-Middleton-Levine

1 Introduction

As our reliance on automobiles for travelling continues to grow so too does the increase in the congestion that occurs on our freeways. This congestion, oftentimes referred to as a *traffic jam*, is a widespread phenomenon that occurs for any one of a number of reasons including, but not limited to, an increase in traffic demand that exceeds the capacity of the freeway, a reduction in freeway operation speed as a result of geometric constraints with respect to the layout of the roads of said freeway itself or the occurrence of events that are not entirely predictable such as a traffic accident or adverse weather conditions. Regardless of the cause, the resulting economic losses as a result of delays in traffic are enormous and are cause for concern for major governing bodies [1]. Given the increased attention that problems with traffic have accrued; various approaches have been applied throughout the years in order to describe the collective properties of traffic flow [2] with the intent to optimize and improve on what has become an urgently deteriorating situation [3]. Of these approaches is a self-organizing cellular automaton traffic flow model called the Biham-Middleton-Levine (BML) traffic model. Throughout the scope of this project, and subsequently this report, we will be exploring how the BML traffic model simulates traffic flow and traffic jams by means of implementing the model from scratch and performing experiments over a variable set of hyperparameters. The following sections of the paper will be organized as follows: Section 2 serves to provide a brief overview of the history of the BML traffic model while Section 3 serves to outline the details of our experiments. Finally, Sections 4 and 5 serve to present the results of our experiments alongside a short discussion and concluding thoughts.

2 The BML Traffic Model

The Biham-Middleton-Levine (BML) traffic model was first formulated in 1992 by Ofer Biham, A. Alan Middleton and Dov Levine [4]. It consists of a variable number of cars, represented as points on a lattice that are initialized with random starting positions, where each car on said lattice belongs to either one of two classes. The first class, generally represented by cars that take on a red color, can only move towards the right on the lattice while the second class, generally represented by cars that take on a blue color, can only move downwards on the lattice. During each turn of the simulation all the cars advance by one step if and *only if* they are not currently being blocked by another car i. e., there exists a car in the space directly in front of them.

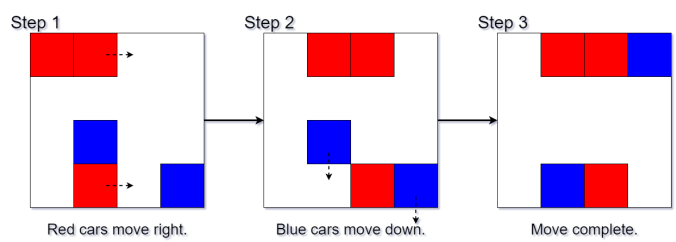


Fig. 1: Steps of the BML traffic model.

The lattice space that the cars are placed on varies in shape with the most common shapes simulated being either a square lattice, as seen in Figure 2, or a rectangular lattice. Regardless of shape, the topology of the lattice is equivalent to a torus; that is in the sense that cars that move off the right edge would reappear on the left edge and cars that move off the bottom edge would reappear on the top edge in the case of a lattice with 4 edges.

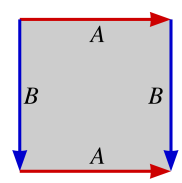


Fig. 2: The fundamental torus of the polygon on which the cars of both class A (red) and class B (blue) move on. Image source: [5], licensed under public domain.

- *Kareem Al-Saudi is a second year MSc. of Computing Science student at the Rijksuniversiteit Groningen. E-mail: K.Al-Saudi@student.rug.nl*
- *The GitHub repository associated with this project can be found here.*

The most interesting aspects of the BML traffic model are its two highly distinguishable phases: the jammed phased and the free-flowing phase and, as the names suggest, pertain to when the cars on the lattice organize themselves to either achieve a smooth flow of traffic or reach a globally jammed state in which not a single car can make a move. In spite of the BML traffic model's simplicity, it has been subject to rigorous mathematical analysis with Omer Angel, Alexander Holroyd and James Martin [6] noting that with densities $\rho \sim 1$ or $lim_{\rho \to 1}$ the system will reach a globally jammed phase infinitely often and Tim Austin and Itai Benjamini [7] noting that the model will always reach the free-flowing phase if the total number of cars (n) is less than half the length of the edge of a square lattice $(\frac{N}{2})$. In this paper we will be exploring the relationship of ρ, or otherwise the number of the cars on the lattice, with the state of the traffic flow and attempt to recreate the findings of both Omer Angel, Alexander Holroyd and James Martin and Tim Austin and Itai Benjamini.

3 Methodology

To ascertain the assumptions made in Section 2 of this paper we will be implementing the BML traffic model from scratch and attempt to recreate the simulations under a similar set of hyperparameters.

3.1 Implementation

Implementing the BML traffic model is rather straightforward – we initialize an $X \times Y$ grid that serves to act as the lattice that the cars will move on. Following that, we populate the grid by placing $\rho \times X \times Y$ (with $0 < \rho < 1$) cars at random positions throughout the grid making sure that no two cars are occupying the same cell. Once the grid has been populated the cars take alternating turns moving depending on which class they belong to. In our configuration, blue cars move downwards at odd time periods $(t = 1, 3, 5, ...,)$ while red cars move rightwards at even time periods $(t = 2, 4, 6, ...,)$. When a car gets to the edge of the grid it "wraps" around, e. g., when a blue car gets to the bottom edge, we reset its position to the top of the grid and, similarly, when a red car gets to the right edge of the grid, we reset its position to the leftmost position on the grid. Finally, before moving, the cars have to consider the following 3 options:

1) The considered location is empty $\to$ we can move.
2) The considered location is occupied by a car of the opposite color $\to$ we can't move.
3) The considered location is occupied by a car of the same color $\to$ we check to see if the "blocking" car can move and act accordingly as per steps 1 & 2.

To wrap things up, we direct the reader to our GitHub repository located here that contains the relevant source code as well as further documentation on our (Python) implementation of the BML traffic model. The relevant hyperparameters for our experiments are listed in Table 1. We note that we will be utilizing the original (random) version of the BML traffic model when conducting the experiments outlined in Sections 3.2, 3.3 and 3.4 and point the reader towards the definition of velocity in regards to the BML traffic model; oftentimes referred to as either the *"mobility"* or the *"speed"* of the system, it denotes the number of cars that *can* make a move expressed as a fraction over the total number of cars – this is outlined in Equation 1. When assessing the validity of our experiments, the velocity of the system will serve to act as the primary metric.

Hyperparameter	Explanation
X	The size of the x-axis of the grid.
Y	The size of the y-axis of the grid.
ρ	Density of the cars on the grid $\rho = \frac{n}{X \times Y}$
t	Number of iterations.

TABLE 1: Hyperparameters of our implementation of the BML traffic model.

$$\text{Velocity}(V) = \frac{\text{Number of cars that could move}}{n} \tag{1}$$

3.2 Experiment 1

The first experiment we will be performing is to ascertain that, assuming a square lattice $(X = Y)$, if the total number of cars is less than or equal to half the length of one of the edges of lattice $\left(n \leq \frac{X}{2}\right)$ then we will *always* achieve a free-flowing state of traffic flow. To do this we will run the simulation under the set of hyperparameters outlined in Table 2 and document the results predominantly noting the mean velocity of the system for each individual run as well as the velocity at the final iteration, or rather, the number of cars that could move at the final iteration of each simulation.

Hyperparameter	Value
X	64
Y	64
ρ	$0.005 \to 0.025$ (in increments of 0.001)
t	2500

TABLE 2: Hyperparameters for experiment 1.

This experiment relates to the deterministic question, posed by Tim Austin and Itai Benjamini. They state that, "Given N (where N refers to the length of an edge of a square lattice $N \Leftrightarrow X = Y$), for which n does any initial configuration of the cars inevitably attain speed one *(free-flowing state of traffic)*, and for which is there some initial configuration of the cars which gets stuck *(reaches a jammed state)*." The first proposition made in their paper, and the notion that we will be tackling as part of experiment 1, states: "If $n \leq \frac{1}{2}(N)$ the system must attain speed *(velocity)* one."

3.3 Experiment 2

The second experiment that we will be performing is done to ascertain that as ρ approaches 1 the model will reach a globally jammed phase infinitely often. The relevant hyperparameters for this experiment can be found in Table 3.
The only notable differences between experiments 1 and 2 are the range in values of ρ that we will be exploring.

Hyperparameter	Value
X	64
Y	64
ρ	$0.005 \rightarrow 1$ (in increments of 0.025)
t	2500

TABLE 3: Hyperparameters for experiment 2.

3.4 Experiment 3

The final experiment is done to explore the intermediate phase, frequently occurring close to the transition density, which combines features from both the jammed as well as the free-flowing phases. The two principal intermediate phases are **disordered** and **periodic**. We'll specifically be looking at whether we can achieve periodic orbits on rectangular lattices with coprime dimensions as per the findings of D'Souza [8]. Unlike experiments 1 and 2, we will be utilizing a rectangular lattice of coprime dimensions (144×89) and we will be exploring values of ρ that lie in between values that lead to free flowing states of traffic and globally jammed states in order to find ρ values that lead to periodic intermediate phases.

4 Results and Discussion

As mentioned earlier, this section serves to present a graphical illustration of the results relevant to the experiments outlined in Section 3 alongside a brief discussion.

4.1 Experiment 1

The first of our experiments explored the relationship between the number of cars on a square lattice. Notably, we examined whether the BML traffic model would achieve a free-flowing state of traffic given that the total number of cars (n) did not exceed the length of one of the sides of the aforementioned square lattice $\left(n \leq \frac{X}{2}\right)$. To ascertain this, we ran simulations using our implementation of the BML traffic model under the set of hyperparameters listed in Section 3.2 and averaged the results over a total of 10 runs. Figure 3 denotes the (averaged) mean velocity for each simulation under varying values of ρ. We note that for values of n where $n \sim \frac{N}{2}$, the system tends to consistently reach a velocity of 1 and a free-flowing state of traffic which is in line with the theory outlined by Tim Austin and Itai Benjamini in their paper. In accordance with their paper, we observe that, for values of $n < \frac{1}{2}(N)$, regardless how the cars are distributed in our $N \times N$ lattice then there will always exist an empty arc (over the diagonals of the square lattice in question) of length at least 2, and so the system will self-organize to a velocity of 1 in a finite amount of iterations. In the interest of preserving time and preventing this paper from being overtly length, we direct the reader to the original paper, located here, to better understand how Tim Austin and Itai Benjamini came about to prove this proposition.

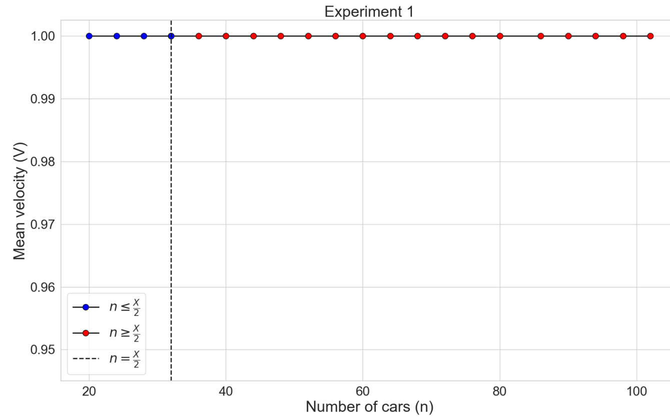


Fig. 3: Mean velocity of the simulations run under the set of hyperparameters pertaining to experiment 1. A mean velocity of 1.0 denotes that the system experienced a smooth flow of traffic straight from the get-go and (in the worst case) experienced a number of collisions around the start.

4.2 Experiment 2

The results of experiment 2 are meant to ascertain that, as ρ approaches 1, the BML traffic model will reach a globally jammed phase, in which no car can make a move, infinitely often. To ascertain this, we ran we ran simulations using our implementation of the BML traffic model under the set of hyperparameters listed in Section 3.3, and once again averaged the results over a total of 10 runs per value of ρ. Our findings are outlined in Figures 5, 6, 7, and 8. The basis of the arguments formulated by Omer Angel, Alexander Holroyd and James Martin revolve around the existence of blocking paths in configurations where $\rho > \rho_c$. To best understand this, take an initialization in which $\rho = 1$: each and every car is immediately blocked by a car positioned directly in front of it and in this way no cars can make a move due to an infinite chain of blocking cars that define a set of *"blocking paths"*. While this logic does not immediately extend to configurations in which $\rho < 1$, as such a chain will always be broken by an empty space, if we consider the previously defined network of blocking paths for a configuration at $\rho = 1$, then it is likely that these blocking paths still exist in systems configured at values of $\rho < 1$ (considering the fact that taking $\rho < 1$ is akin to removing a proportion of cars from a $\rho = 1$ configuration.) See Figure 4 for an illustration.

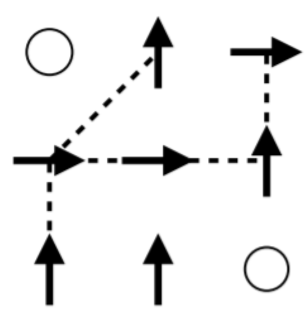


Fig. 4: An illustration of blocking paths as per the findings of Omer Angel, Alexander Holroyd and James Martin. Image source: [6] licensed under CC BY 3.0.

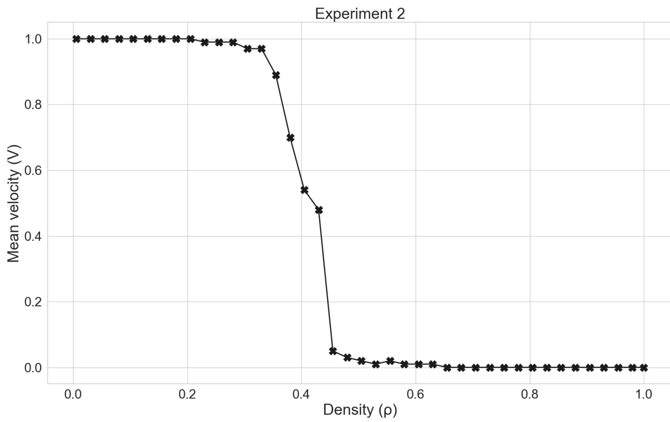


Fig. 5: Mean velocity per simulation averaged over a total of 10 runs per value of ρ.

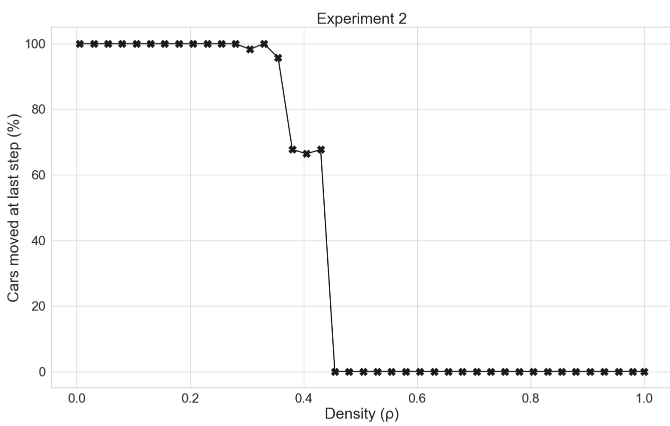


Fig. 6: Percentage of cars that moved at the last time step recorded for each of our simulations averaged over a total of 10 runs per value of ρ.

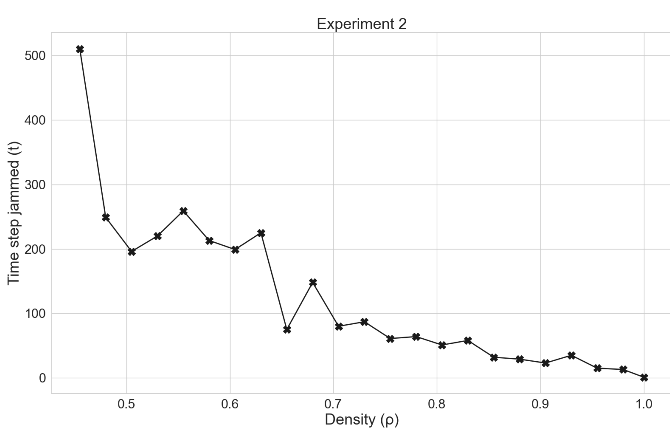


Fig. 7: Time step that the system jammed (if applicable) for each of our simulations averaged over a total of 10 runs per value of ρ.

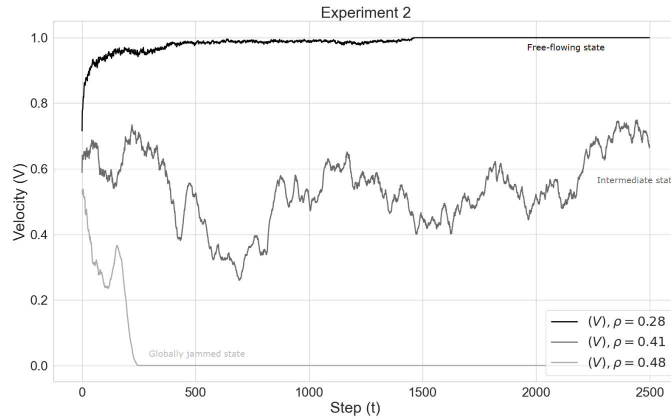


Fig. 8: A deeper look at the velocity per iteration of 3 different systems at 3 different densities that exhibit the different states of traffic flow associated with the BML traffic model.

4.3 Experiment 3

The final experiment pertains to the emergence of periodic (with respect to time) intermediate states in the BML traffic model that capture the essence of both the jammed state as well as the free-flowing state. In particular, the findings of D'Souza, show that for a lattice of coprime dimensions and for values of $\rho \sim \rho_c$ the system will self-organize into periodic arrangements that consists of both jams as well as smoothly flowing traffic. To attempt to recreate this we initialized a 144×89 rectangular lattice with a density of 38% and ran the simulation for a total of $7,500$ iterations. We noted the velocity per time step of the simulation and found that at around $t \sim 4,000$ the system self-organized to the previously defined intermediate state. The findings of our results can be seen in Figures 9, 10, and 11.

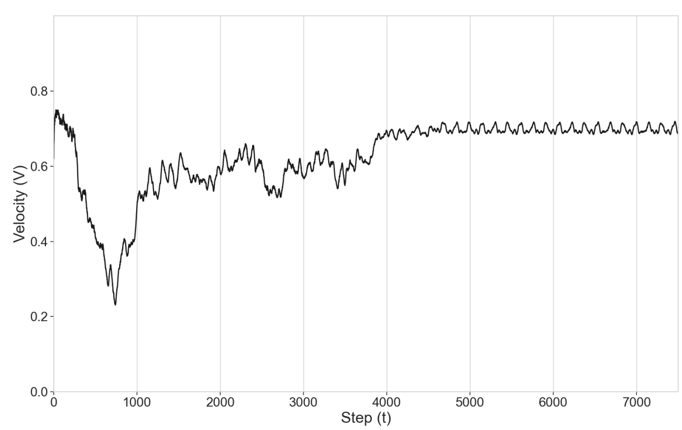


Fig. 9: Velocity per time step of a simulation run on a rectangular lattice with coprime dimensions over a total of $7,500$ time steps.

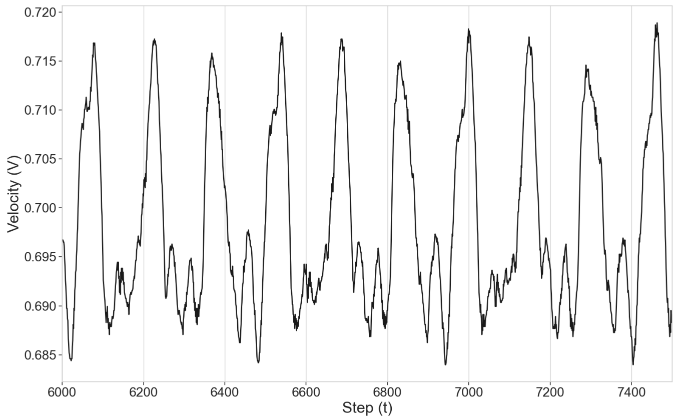


Fig. 10: A closer look at the velocity of the system at iterations $6,000 \to 7,500$.

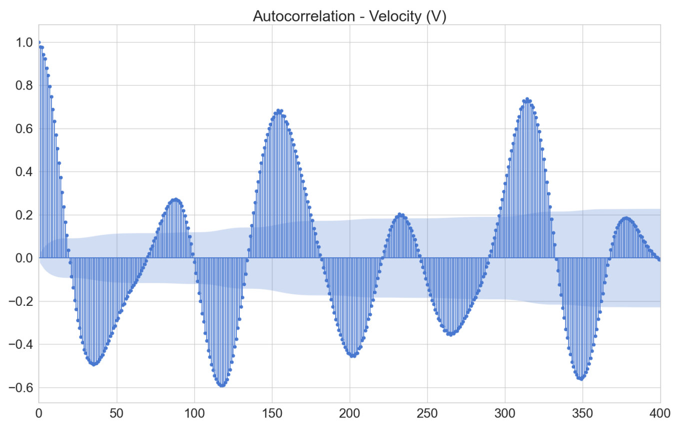
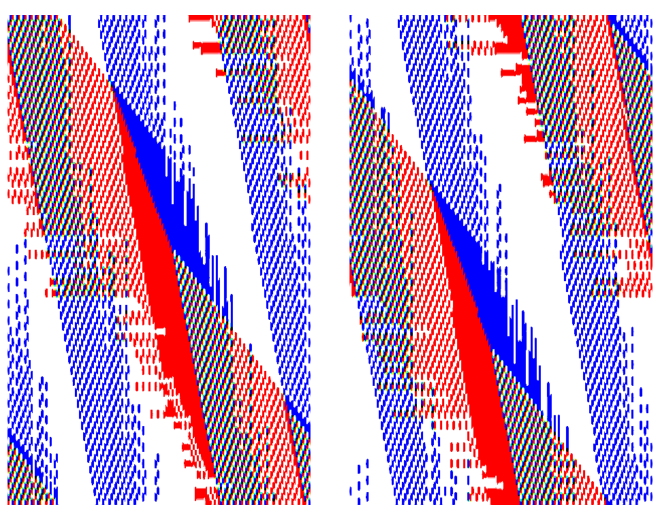


Fig. 11: Plotting the autocorrelation of the velocity of the system in an attempt to illustrate the periodic nature of the intermediate state that the system self-organized to.



(a) Time step: 6250.

(b) Time step: 6415.

Fig. 12: A snapshot of our system at 2 separate iterations (165 iterations apart) illustrating the intermediate phase characterized by arrangements of traffic jams as well as smoothly flowing traffic.

5 Conclusion

To conclude, in this paper we recreated major findings with respect to the BML traffic model by means of a Python implementation of the model and simulation runs over a variable set of hyperparameters that pertained to the aforementioned findings. In doing this, we explored the relationship of the traffic density and state of traffic flow and, with respect to the BML traffic model, we were able to better understand how simple changes with respect to initial configurations of the system can have drastic effects on whether or not the model would ultimately self-organize to either extreme with regards to traffic flow, be that a free-flowing state of traffic or a globally jammed state. Given the computational limitations and time constraints at the time of writing this project, the number of runs per experiment were limited alongside the overall size of the lattice(s) we explored which, given the nature of the project, did not come with any major ramifications although it would have been interesting to explore a wider array of initial configurations and even finer increments with respect to ρ. Exploring variations of the model, i. e., , on a k-dimensional lattice ($k \geq 3$), or the *Klein bottle* variation or either of the non-deterministic, randomized variation or the open-boundary (chain) variation would also have been interesting and can be considered for future work. Other cellular automaton models, such as the Nagel-Schreckenberg model that bring in the concepts of car velocity and acceleration, would also be interesting to look for future work.

References

[1] D. Yin and T. Z. Qiu, "Traffic Jam Modeling and Simulation," in *2012 15th International IEEE Conference on Intelligent Transportation Systems*, 2012, pp. 1423–1428. DOI: 10.1109/ITSC.2012.6338916.

[2] T. Nagatani, "Effect of Car Acceleration on Traffic Flow in 1D Stochastic CA Model," *Physica A: Statistical Mechanics and its Applications*, vol. 223, no. 1-2, pp. 137–148, 1996. DOI: 10.1016/0378-4371(95)00292-8.

[3] D. Helbing, "Improved Fluid-Dynamic Model for Vehicular Traffic," *Physical Review E*, vol. 51, no. 4, pp. 3164–3169, 1995. DOI: 10.1103/PHYSREVE.51.3164.

[4] Ofer Biham, A. Alan Middleton and Dov Levine, "Self-organization and a Dynamical Transition in Traffic-flow Models," *Phys. Rev. A*, vol. 46, pp. 6124–6127, 10 Nov. 1992. DOI: 10.1103/PHYSREVA.46.R6124.

[5] (Apr. 2021). "TorusAsSquare," [Online]. Available: https://commons.wikimedia.org/wiki/File:TorusAsSquare.svg.

[6] Omer Angel, Alexander Holroyd and James Martin, "The Jammed Phase of the Biham-Middleton-Levine Traffic Model," *Electronic Communications in Probability*, vol. 10, pp. 167–178, 2005. DOI: 10.1214/ECP.V10-1148. [Online]. Available: https://doi.org/10.1214/ECP.v10-1148.

[7] Tim Austin and Itai Benjamini, "For What Number of Cars Must Self Organization Occur in the Biham-Middleton-Levine Traffic Model From Any Possible Starting Configuration?," Aug. 2006. [Online]. Available: https://arxiv.org/abs/math/0607759.

[8] R. M. D'Souza, "Coexisting Phases and Lattice Dependence of a Cellular Automaton Model for Traffic Flow," *Physical Review E*, vol. 71, no. 6, 2005. DOI: 10.1103/PHYSREVE.71.066112.

Appendix

List of Figures

1 Steps of the BML traffic model. 1
2 The fundamental torus of the polygon on which the cars of both class A (red) and class B (blue) move on. Image source: [5], licensed under public domain. 1
3 Mean velocity of the simulations run under the set of hyperparameters pertaining to experiment 1. A mean velocity of 1.0 denotes that the system experienced a smooth flow of traffic straight from the get-go and (in the worst case) experienced a number of collisions around the start. 3
4 An illustration of blocking paths as per the findings of Omer Angel, Alexander Holroyd and James Martin. Image source: [6] licensed under CC BY 3.0. 3
5 Mean velocity per simulation averaged over a total of 10 runs per value of ρ. 4
6 Percentage of cars that moved at the last time step recorded for each of our simulations averaged over a total of 10 runs per value of ρ. 4
7 Time step that the system jammed (if applicable) for each of our simulations averaged over a total of 10 runs per value of ρ. 4
8 A deeper look at the velocity per iteration of 3 different systems at 3 different densities that exhibit the different states of traffic flow associated with the BML traffic model. 4
9 Velocity per time step of a simulation run on a rectangular lattice with coprime dimensions over a total of $7,500$ time steps. 4
10 A closer look at the velocity of the system at iterations $6,000 \rightarrow 7,500$. 5
11 Plotting the autocorrelation of the velocity of the system in an attempt to illustrate the periodic nature of the intermediate state that the system self-organized to. 5
12 A snapshot of our system at 2 separate iterations (165 iterations apart) illustrating the intermediate phase characterized by arrangements of traffic jams as well as smoothly flowing traffic. 5

List of Tables

1 Hyperparameters of our implementation of the BML traffic model. 2
2 Hyperparameters for experiment 1. 2
3 Hyperparameters for experiment 2. 3

Glossary

BML Biham-Middleton-Levine. 1–5, 7