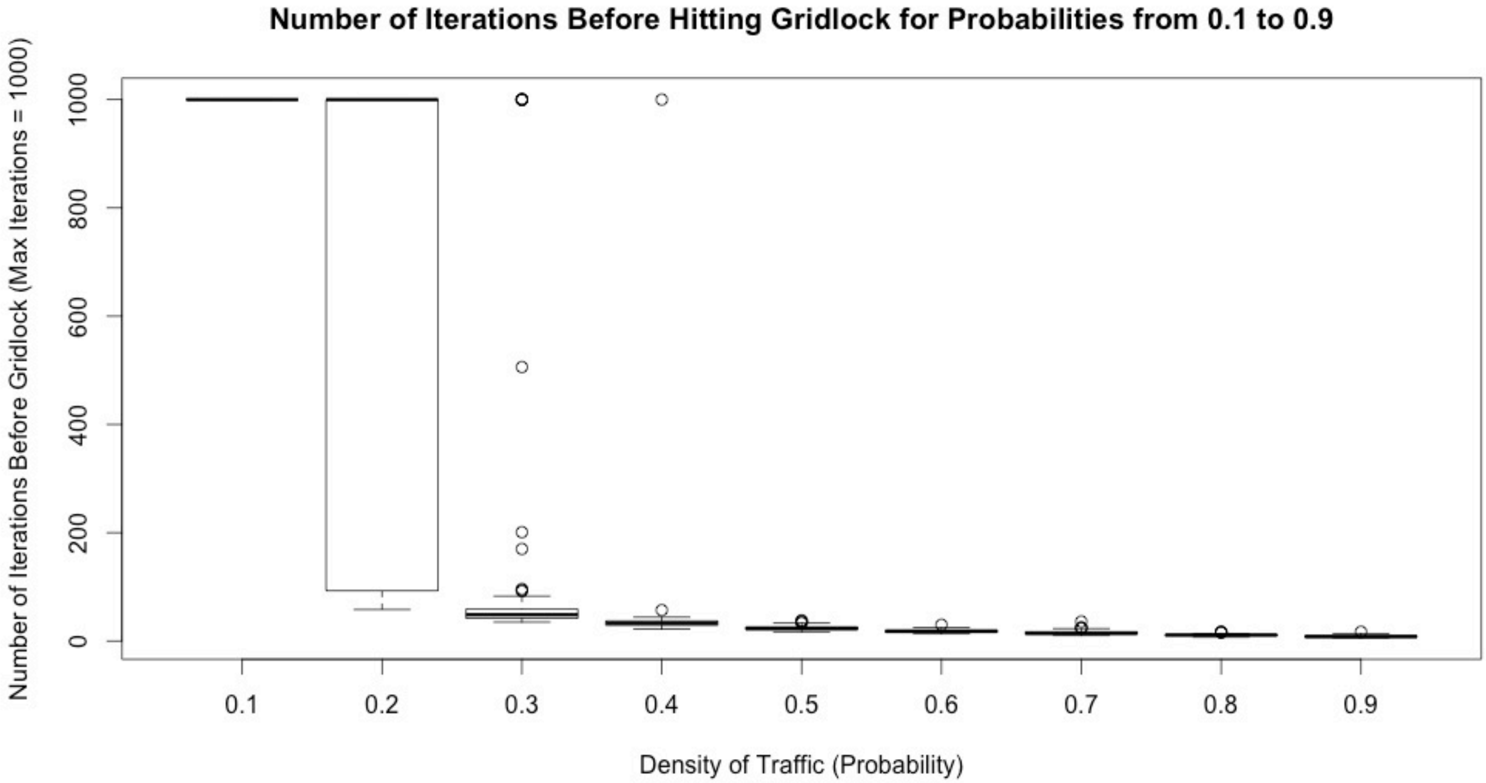
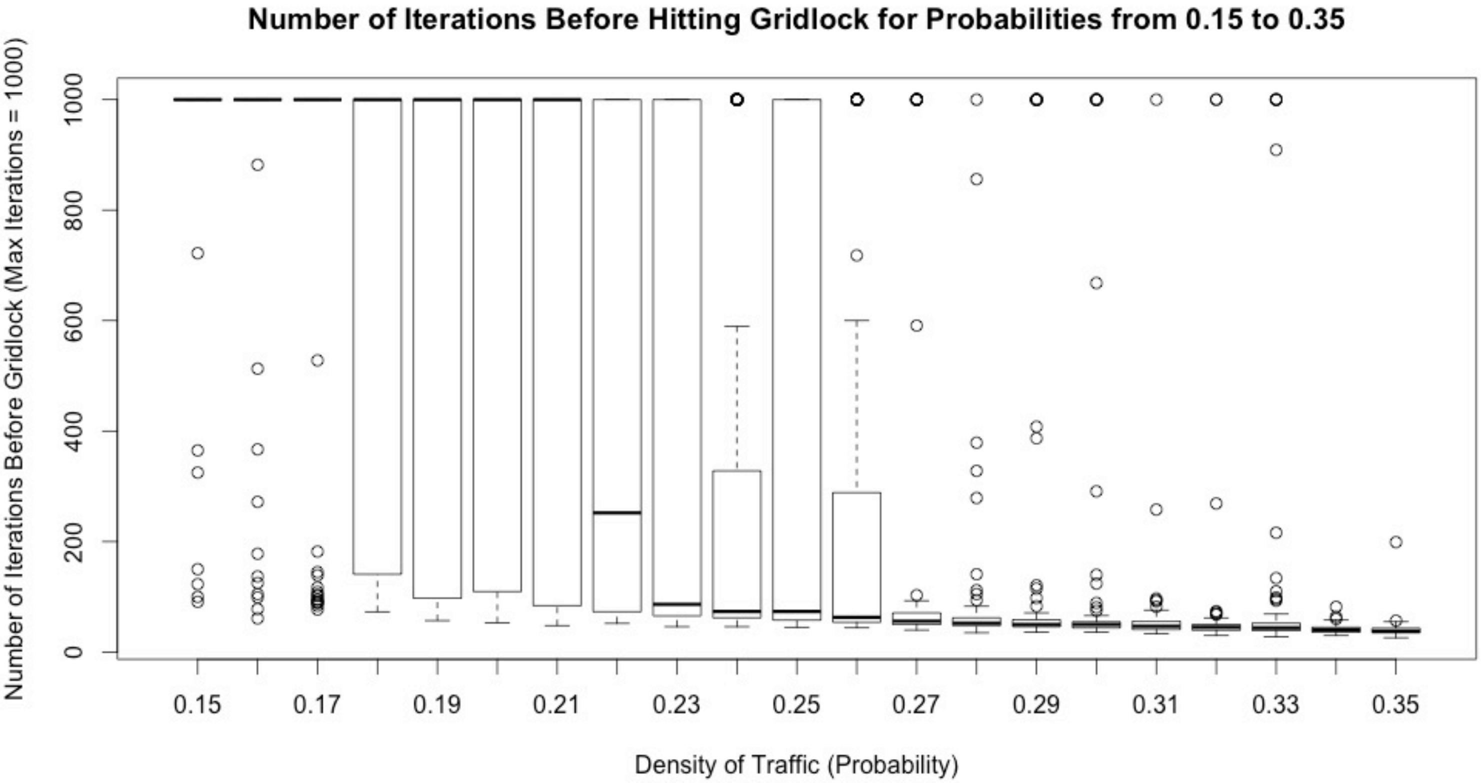
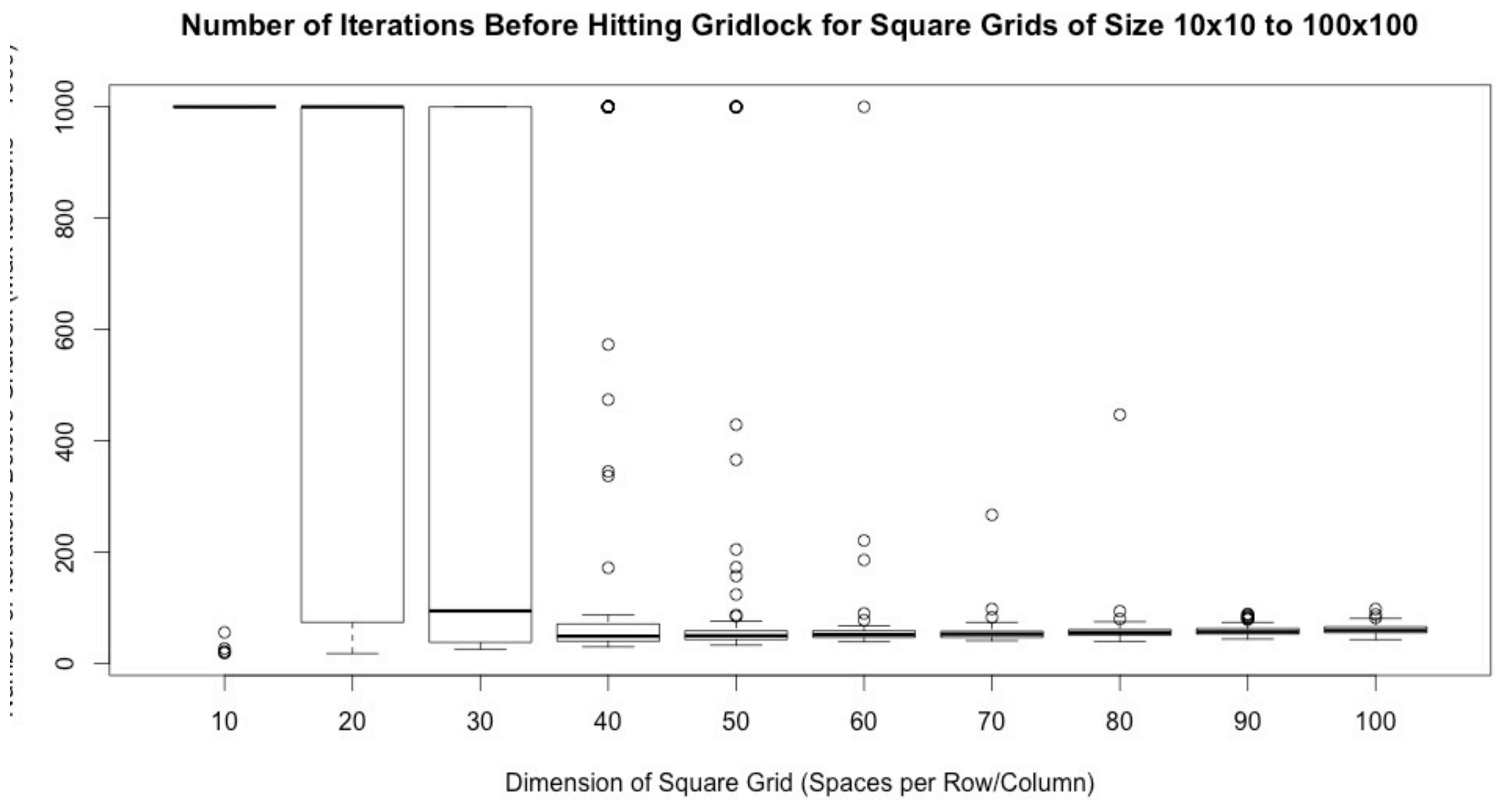


For this boxplot, I ran my bml.sim function 100 times for each value of p between 0.1 and 0.9. The maximum number of iterations I used for my bml.sim function (the number of steps) was 1000, because I felt the results would be closer to their true values than if I had used only 100 iterations. If after running my bml.sim function, it managed to run 1000 times, that means that the system did not hit gridlock and was free flowing. I decided to use a higher maximum number of iterations for more accurate results, despite the fact that the boxplot of the number of iterations before hitting gridlock vs. the density of the traffic is not particularly well formatted.

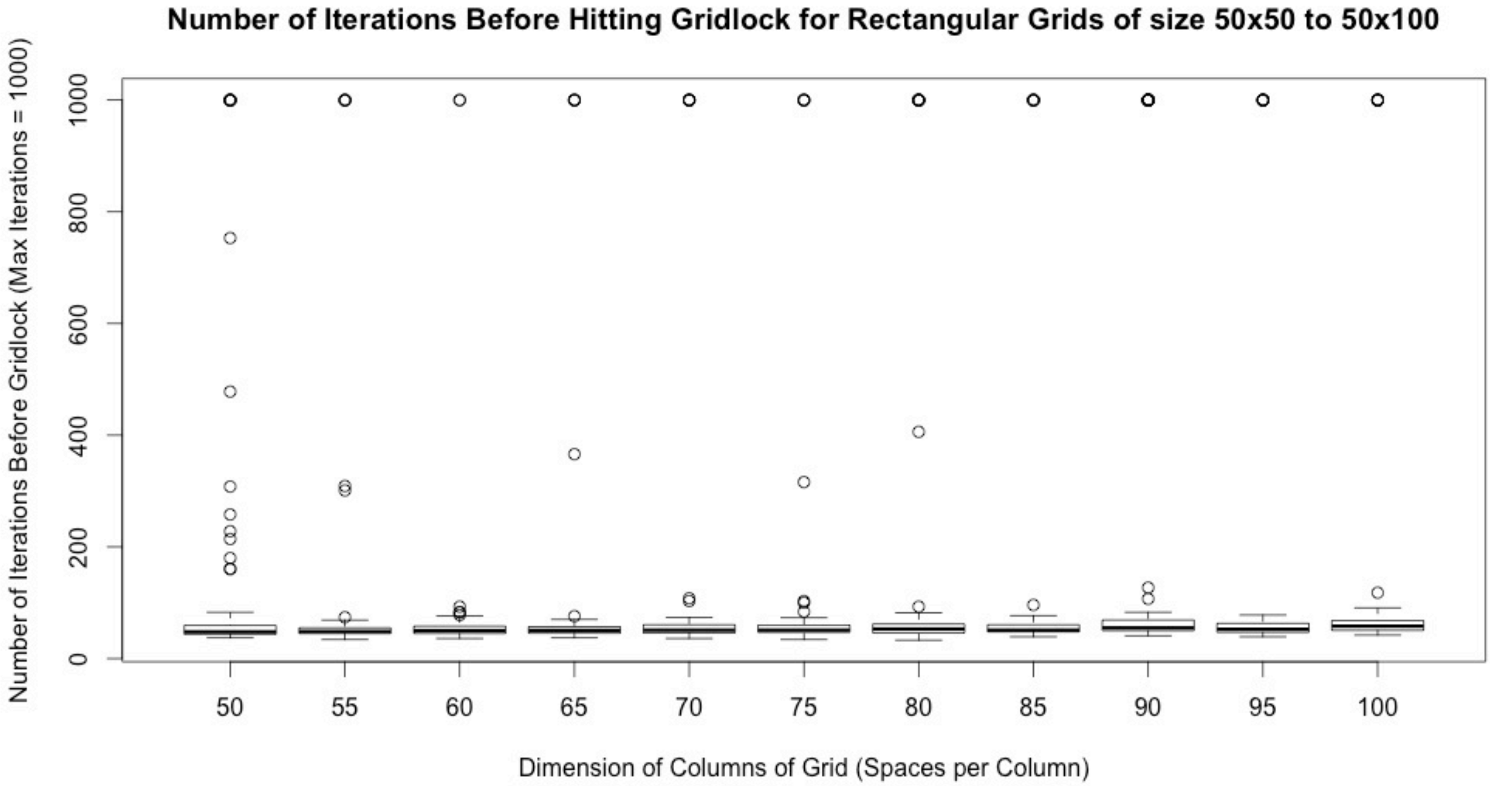
However, it does clearly show that with a density of 0.1, the system was always free flowing. With a density of 0.2, the system starts to show some gridlocks but is mostly still free flowing. With a density of 0.3, the gridlocked systems become more common than the free flowing systems, since the average number of iterations is well below 1000, despite the few outlying systems at this probability that do reach 1000 iterations. For densities 0.4 and higher, the average number of iterations reached only continues to decrease, as gridlocks happen sooner and sooner, and at a density of 0.5 and higher, no systems are free flowing.



I decided to focus on the probabilities between 0.15 and 0.35, since that was where the jamming seemed to begin to occur for my simulation. Similarly for the boxplot above, I also ran my bml.sim function 100 times for each probability between 0.15 and 0.35, and also for each bml.sim function, set the maximum number of iterations to 1000. For densities of 0.15 to 0.17, the systems are mostly free flowing, with very few gridlocked outliers. For densities of 0.18 to 0.21, the systems are still mostly free flowing, but the first quartile has come to incorporate the larger number of gridlocked systems. Starting with a density of 0.22, the average number of iterations becomes much lower than 1000, indicating a high number of gridlocked systems. However, up until a density of about 0.25, the number of free flowing systems is still common. Based on my simulations, the traffic became less free flowing and more gridlocks occurred at a density of about 0.21.



For this boxplot, I decided to see if systems were affected by the initial size of the grid. I tested square matrices of size 10x10 up to 100x100 for my bml.sim function. In addition, I set the density for each system to 0.3, the max number of iterations to 1000, and ran the bml.sim function 100 times for each differently sized matrix. As you can see, with a 10x10 matrix, the systems were very predominantly free flowing. With a 20x20 matrix, the systems were still mostly free flowing, though there were quite a few gridlocked systems. For a 30x30 matrix, the systems start becoming more gridlocked, as the average number of iterations is much lower than 1000. For matrices larger than 30x30, the systems are mainly gridlocked.



For this plot, I decided to see if systems were affected by rectangular grids rather than square grids. I set the maximum number of iteration to 1000, set the density to 0.3, and ran the bml.sim function 100 times for each matrix of size 50x50, 50x60, … , 50x100. As you can see from the boxplot, rectangular matrices as compared to square matrices seem to have very effect on whether systems are free flowing or if they gridlock.

According to my results, whether or not systems are free flowing depends mainly upon the size of the matrices and the density of the traffic.