Ambulance Simulation – Final Project

Johannes M. Halkenhaeusser

Minerva Schools at KGI

CS166 Prof. Drummond

Fall 2019

Part 1 – Simulation

The ambulance simulation's goal is to identify the best strategy for distributing ambulances at the beginning of the simulation. That is, which allocation rule minimizes the average execution time of an accident given a number of ambulances.

To achieve this goal, the simulation creates random emergencies based on the population size of the neighborhoods. The neighborhoods are connected by a network of edges. The simulation then has to find the closest free ambulance and record the total time it took for the emergency to be resolved. Therefore, each emergency needs a start time and location, each ambulance a home depot. We further need a map of the entire city and the edges between the neighborhoods.

After running the simulation for a certain time interval, we will have a list of execution times which we can then use to estimate which allocation strategy has led to a lower average execution time depending on the number of ambulances used.

Map of Berlin

To create a network map of Berlin we can designate each district/neighborhood ("district" and "neighborhood" are used interchangeably) a node in our network and the connections between the neighborhoods as the edges. The nodes have a population-size attribute (as a proxy for the likelihood of an emergency call being made) and size (to be used when getting the random value for how long an ambulance takes within the neighborhood).

Berlin can be divided into following neighborhoods and populations as shown in Fig. 1 and Tab. 1.

Figure 1. A map of Berlin and its different neighborhoods (TUBS, 2010).

Table 1. The neighborhoods and their attributes population and suburbs. The total population of Berlin is 2,883,854 (Amt für Statistik Berlin-Brandenburg, 2008).

Neighborhood	Population	No. suburbs
Mitte	230469	6
Friedrichshain-Kreuzberg	200487	2
Pankow	327985	13
Charlottenburg-Wilmersdorf	251304	7
Spandau	193641	9
Steglitz-Zehlendorf	256184	7
Tempelhof-Schöneberg	274225	6
Neukölln	234564	6
Treptow-Köpenick	227142	15
Marzahn-Hellersdorf	237231	5
Lichtenberg	231894	10
Reinickendorf	218728	11

Taking the edges of each neighborhood as the neighborhoods it borders to; we create a list of edges to be added to the network. We, thus, add undirected edges across each district border which each have a weight that is normally distributed around ten with standard deviation two. The values ten and two are arbitrary choices but are chosen to penalize having to go into a different neighborhood. This assums that there is no connection between neighborhoods that do not border each other. This disregards the possibility of an out-of-city highway that can be used to avoid having to go through the city center.

Simulation Rules

The simulation follows this basic outline:

1. Allocate ambulances
2. For every minute in the length specified:
    a. Generate an emergency

b. Solve as many emergencies as possible that are in the queue

c. For every solved emergency, record the execution time

Working backwards to optimize our initial allocation, we begin by thinking about the rules that govern the execution time of an emergency.

How long does an emergency take?

Any trip will take as much time as it takes to get from the current depot node to its edge (that is a random time controlled by the traffic aspect of each node in the network), plus the shortest path of the edges, plus the random traffic of the target node, and the wait time in the queue (Eq. 1).

$$T_E = traffic_{emergency} + shortestPath + traffic_{depot} + time\ in\ queue \qquad \textit{Eq. 1}$$

Each edge weight is initialized around a mean of 10 with standard deviation of 2 and the traffic of each node is defined by the number of different suburbs shown in Figure 1. The more suburbs the larger the size of the neighborhood and hence any trip within that neighborhood will take longer. This is a modeling choice that could potentially be improved by getting real traffic data of how long it takes to travel within a neighborhood. However, not the real values but the comparison of strategies is of primary concern.

Note that the two traffic terms include the way from the current location to the emergency and back. If there is an emergency in a neighborhood the traffic includes all movement within that neighborhood and on the edges, there and back. An ambulance has to move along any applicable path (or term within Eq 1.) twice, letting us simplify our simulation here.

The traffic in a neighborhood is calculated using the absolute, rounded value drawn from a normal distribution centered around the number of suburbs with a standard deviation of 2 minutes. There are two manipulating choices made here:

- Rounding

  The time step the simulation is operating in is minutes. Therefore, we will be using only minutes to denote the length of a task.

- Absolute value

  Notice, how there are neighborhoods such as Friedrichshain-Kreuzberg with just two suburbs. This means that there is a roughly 15% chance of a value being below 0. A value of negative time is not real and hence we take the absolute. It further seems like a plausible modeling assumption, considering that such a neighborhood is so dense that it's mean is slightly pushed up by how dense it is.

How long is an emergency in the queue?

The ambulances in the simulation will take the emergency on the top of our emergency queue and solve it next. Once an emergency call comes in it is added to the back of the queue and as soon as any ambulance is ready it will tend to the emergency. The closest ambulance will be chosen. Therefore, we want to minimize the length of the queue when an emergency comes in.

How often does and where do emergencies occur?

In 2013 there were roughly 380,000 calls to emergency services (B. Z., 2014). Therefore, we can make a rough estimation of the number of emergencies that occur every 10 minutes:

380000/365 is roughly 1000 daily calls

That then means roughly 40 calls an hour.

Hence roughly seven calls every ten minutes or 0.7 calls a minute.

Therefore, if we want to let our simulation run for minutes, we can use a probability of 0.7 emergencies per minute when creating emergencies. After having the decision made that an emergency is going to occur, we sample from a vector. This vector can be imagined by a big basket containing a number (in our simulation 1000) of balls each with a neighborhood written on it. The number of balls with each neighborhood is determined by the underlying probability that a person lives in that neighborhood, i.e., the relative population of the neighborhood. Randomly choosing a ball from this basket, models drawing from the distribution of relative population sizes.

Strategy 1: Population

The first initialization strategy is focused on the number of people who live in a neighborhood. Neighborhoods with more people are more likely have emergencies. Therefore, an ambulance will be stationed at the neighborhood of emergency with the same probability as the emergency is occurring there if we assign it with the same probability as an emergency occurring. Practically, this means that we can use the same "basket" to draw from as we use for the emergency likelihood.

Therefore, the strategy tries to reduce the time taken for an emergency to be completed from Eq. 1 to:

$$T_E = traffic_{depot} + time\ in\ queue \qquad \textit{Eq. 2}$$

Where $traffic_{depot}$ is the traffic at the home district.

The strategy tries to increase the probability of having the best case in each scenario.

Strategy 2: Degree

The first strategy accounted for the probability of an incident happening somewhere and tried to maximize the probability that there is an ambulance in the neighborhood that the emergency occurs in.

However, when there are not enough ambulances, we could decrease our travel time if we initialized ambulances in central areas that is in areas with higher overall connectivity/degree. This strategy optimizes the ambulances for the best worst case. That is, when the best case does not occur, we try to reduce the shortest path length versus trying to increase the likelihood of the worst-case scenario not occurring at all.

Therefore, the second strategy will (similar to the first strategy) implement a sampling vector that holds each neighborhood with a likelihood dependent on its degree.

Parameters

The simulation will take the length of the simulation in minutes, the map the ambulances are traversing over, the number of ambulances, and the strategy for initialization as input. By keeping the location, length, and number of ambulances constant, we can compare the output from either strategy.

Output metrics

The most important metric is the mean time it takes for an ambulance to be attended to. Further, I will check how loaded the queue is after the simulation is finished running for a certain length of time.

As discussed earlier, the minimum execution time occurs when the emergency can be tended to immediately and within the neighborhood. Therefore, we can run an estimate a minimum bound for the mean execution time by ignoring ambulance allocation and assuming that every emergency can be attended to immediately in each neighborhood.

Theoretically, we expect it to be the weighted average of the number of suburbs in each neighborhood, where the weights are the population ratios. These ratios give the probability of a neighborhood having an emergency. The average time we take within that neighborhood is determined by the number of suburbs as they are the mean of the normal distribution. Therefore, if we take the expected values of the normal distribution and use it for Eq. 3.

$$T_{min} = \sum_{d=1}^{d=no.\ districts} \left(S_d \frac{Pop_d}{Pop_{total}}\right) \qquad \textit{Eq. 3}$$

In words: the sum of the districts number of suburbs (S_d) multiplied with its probability of having an emergency ($\frac{Pop_d}{Pop_{total}}$). Hence, our theoretical estimate would be: 8.2229 minutes.

However, our simulation does contain sampling from a normal distribution with a standard deviation of 2 and using an absolute and rounding as explained above. Since mathematically accounting for rounded and absolute values is complicated (absolute will slightly increase our mean with probability of having this value), we can sample the minimum times from the know probability distributions the same way we sample emergencies in our simulation (Fig. 2). As expected, the sampling approach yields a relatively similar value (8.24 minutes; SD = 3.98 minutes) compared to the theoretical value. The one from the basic simulation is slightly higher due to the absolute value but not high enough to be statistically significantly different. The 95% confidence interval is between 2 and 15 minutes exactly which is wide considering that we are running 1,000,000. However, that is because the sample is a representation of the underlying distribution of the neighborhood sizes plus/minus the standard deviation gotten when using the normal distribution to sample from. Therefore, the standard deviation will actually not decrease further, but approach the standard deviation of the target distribution which is 3.499.

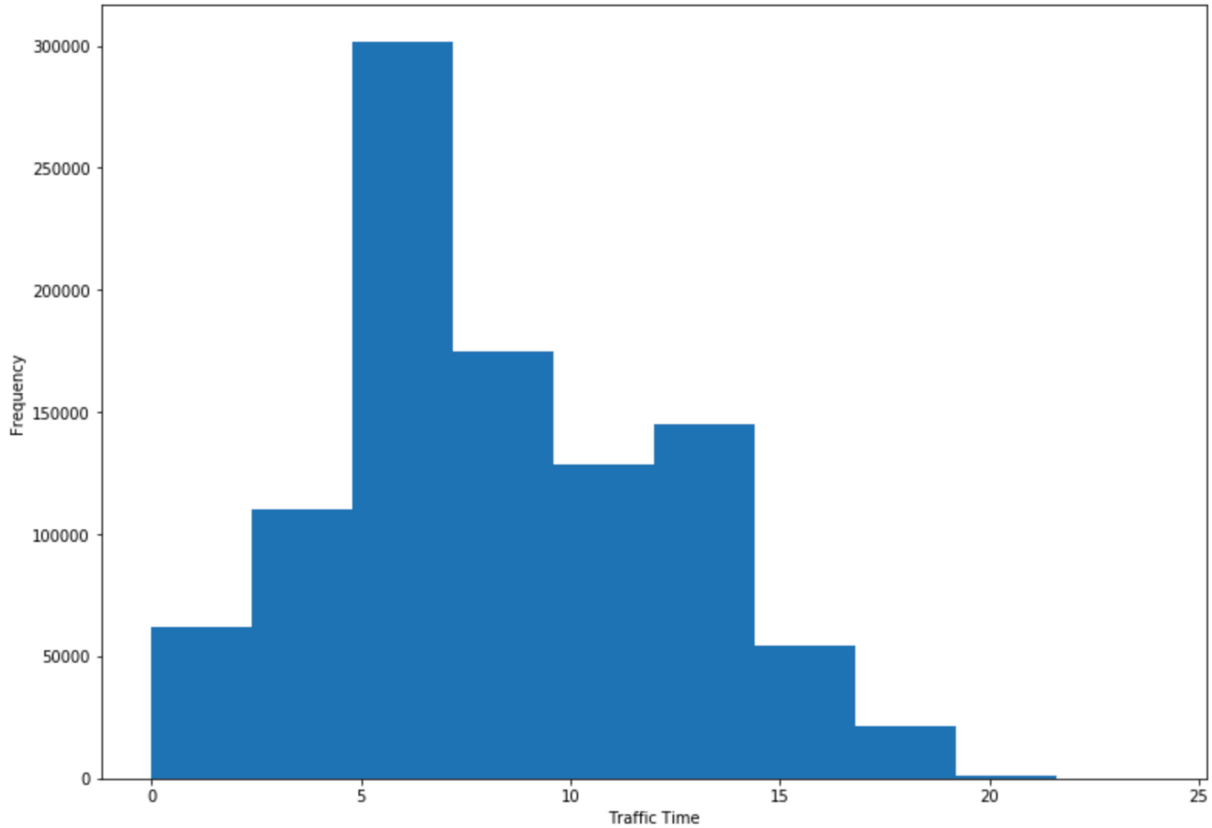

Figure 2. A histogram of the traffic times. The mean minimum execution time is 8.24 minutes with standard deviation 3.98 minutes.

Part 2 – Results

For the main analysis, we chose to a time period of twenty hours to make differences most pronounced and sample roughly 800 (≈1140*0.7) emergencies. However, the longer the time period, the higher the likelihood of the ambulances all being busy when an emergency enters the queue.

Figure 3 shows how with an increasing number of ambulances, the average execution time decreases linearly for both strategies until about 24 ambulances. This seems to be the threshold until which there are rarely ambulances in the neighborhood of the emergency. After this threshold, the two curves, approach the estimated minimum execution time with the strategy that optimizes on the population making higher marginal improvements per ambulance (Fig. 4).

The results for both strategies are quite similar. That is not because our strategies are not working properly but because the strategies are both are relatively similar for the Berlin neighborhoods. When examining the distribution of population and degrees, it becomes apparent that they are relatively uniform. With such a small number of ambulances being sampled from the underlying distribution, the distribution of ambulances does not approach the actual distributions. Therefore, both strategies are doing almost the same if run individually. However, we can make the differences more pronounced by running more trials, which simulates more ambulances sampled overall and the ambulance allocations resemble the underlying distributions.

To assess the significance of the difference, we can do a simple significance test of the mean execution time. We can do this for the example of 30 ambulances. Figure 7 shows the distribution of execution times after 100 trials of 24 simulation hours with thirty ambulances of the both strategies. This is a snapshot of the variability of the results in Figures 3 and 4. The 95% confidence interval for the population strategy is between 9.64 and 13.63 minutes and for the degree strategy between 9.96 and 14.34 minutes. Therefore, the results are not significantly different at the 95% confidence level.

To make our results more robust, we can increase the number of trials or the length of simulation. In both cases the number of emergencies and the number of recorded execution times increases and will more closely represent the difference in the assimilation of the strategies to the underlying distributions they are fitted to. However, we will run into the same limitation as before, which is that the underlying distribution of emergency times, is also dependent on the distribution of emergencies (like in the simulated optimal). Hence, the utility of increasing trials is limited. The standard deviation here is lower because we have not quite reached the underlying distributions yet. However, through the general observation of the trends, practical significance of the one strategy being superior can be inferred.

For Berlin, we see that if we have passed the threshold of how optimizing on the best case (that is after 24 ambulances), it would be most suitable to employ ambulances using the population-based strategy. If we do not have ambulances enough ambulances to so that we have to cross neighborhood borders often and amount a cue, it would be most suitable to allocate ambulances using the degree-based strategy. Neither case means designating ambulances to the neighborhood with the highest population or the highest degree but based on the underlying distributions of these.

To extend our intuition, let us simulate two different maps, each tailored to either strategy. One with the population being distributed with a power-law distribution but the graph is completely connected and the other where the population is perfectly uniform, but the edges are distributed in a power-law distribution.

The first map should be better tailored to the population strategy as a strong power-law distribution means that the probabilities of an emergency are less evenly distributed and, therefore, a correctly placed ambulance has higher utility. Hence, we use a complete graph to simulate Berlin but change the population. The number suburbs for each neighborhood is held at a constant 10 for all nodes and we will therefore assume an optimal execution time of around 10. As expected, the population-driven strategy outperforms the degree-oriented strategy clearer than in the Berlin proxied map, as following the underlying distributions is rewarded more (Fig. 8).

Now, we are able to improve the significance of the difference in execution times so that after only four simulation-hours at 30 ambulances, the population strategy is also statistically significantly better than the degree-based strategy at the 95% interval (Fig. 9).

Moving on to the second map, we use a Barabasi-Albert (BA) graph. The power-law distribution of edges is generated by the preferential attachment of the BA graph. Here, the degree-oriented strategy will outperform the population strategy, as being able to allocate at the central nodes is strongly rewarded with lower travel time. Like with our Berlin map, this advantage will disappear as soon as we have enough ambulances to cater to each neighborhood. Figure 10 validates this expectation as now, the degree-oriented strategy outperforms the population strategy more strongly (though not statistically significant (see appendix)) before reaching the critical point at which allocation to distinct neighborhoods becomes more important.

Hence, we can improve our suggestion from before and add that the relative distribution and connectivity of the city also matters. That is, if we have great connectivity but an uneven population distribution, using the population strategy might actually outperform the degree-strategy across the board. Regardless of the strategy, however, trying to decrease the traffic within the neighborhood would be beneficial when trying to reduce the emergency execution time as this time drives the equations.

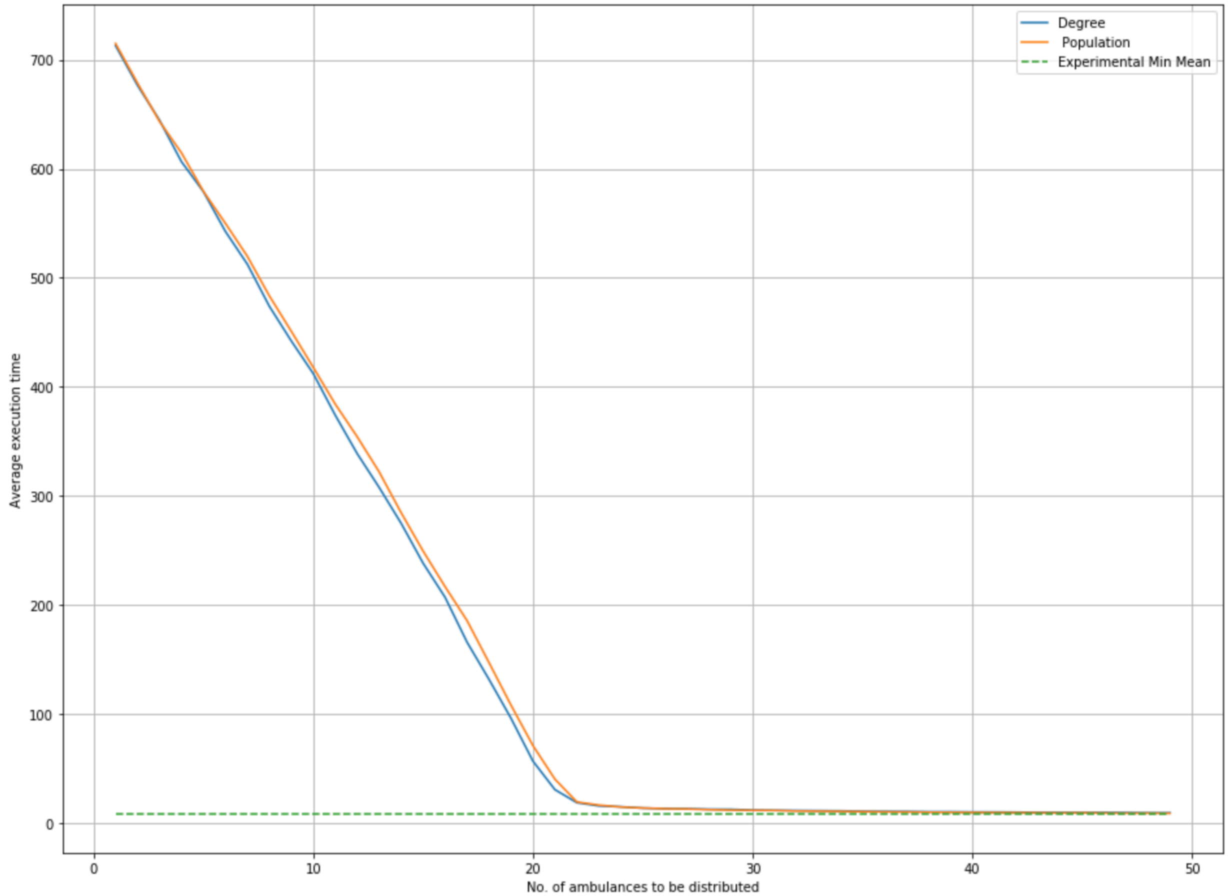

Figure 3. The average execution time after ten hours runtime for an increasing number of input ambulances using 100 trials.

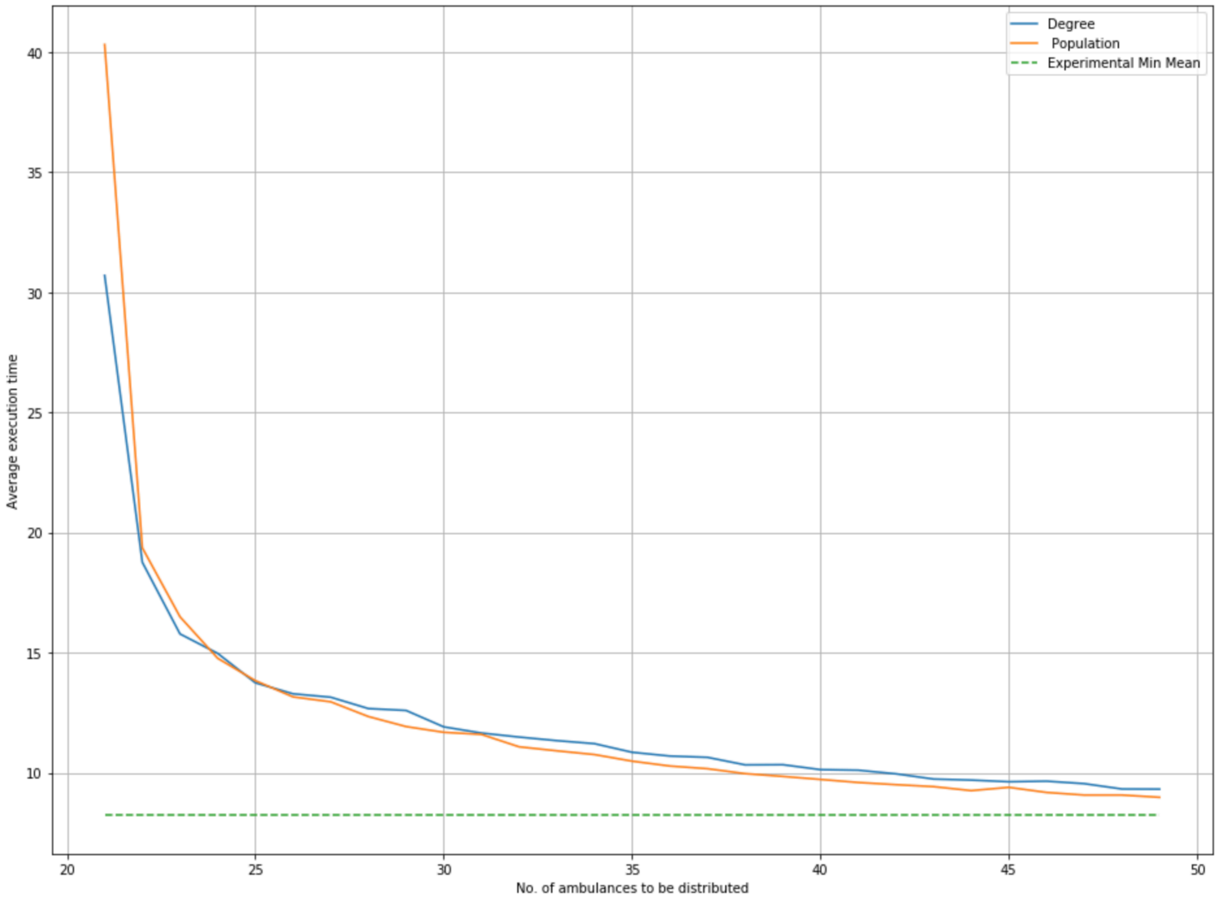

Figure 4. After around a threshold of around 24 ambulances being allocated, the allocation strategy using ambulances outperforms the degree allocation as they approach the optimal minimum average execution time.

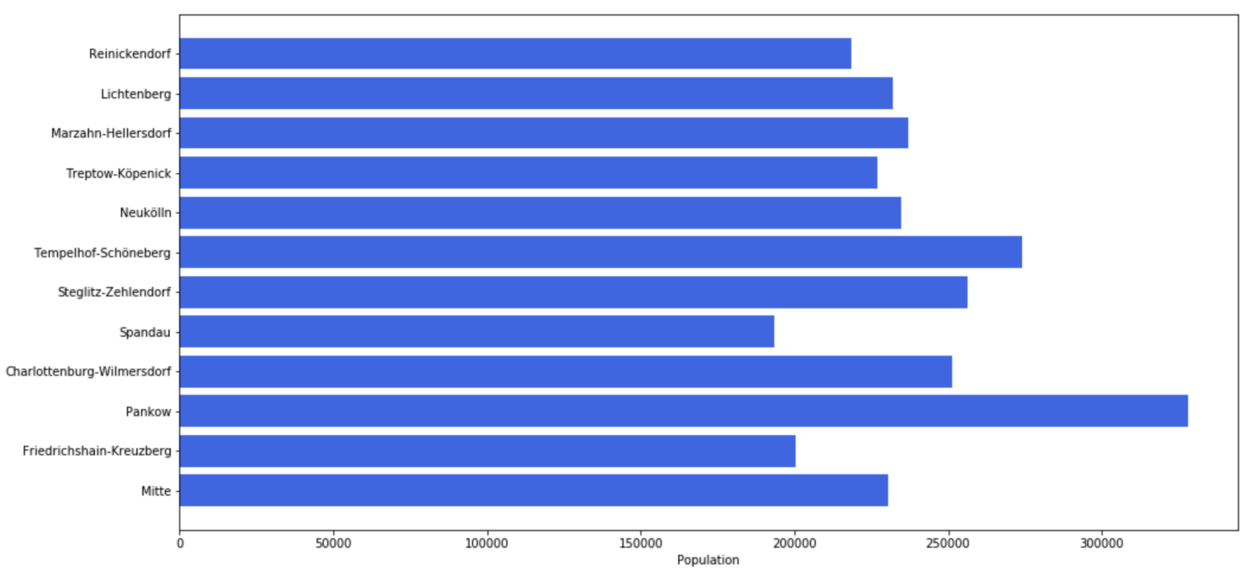

Figure 5. Bar graph of the different neighborhoods and their populations. It is relatively uniform.

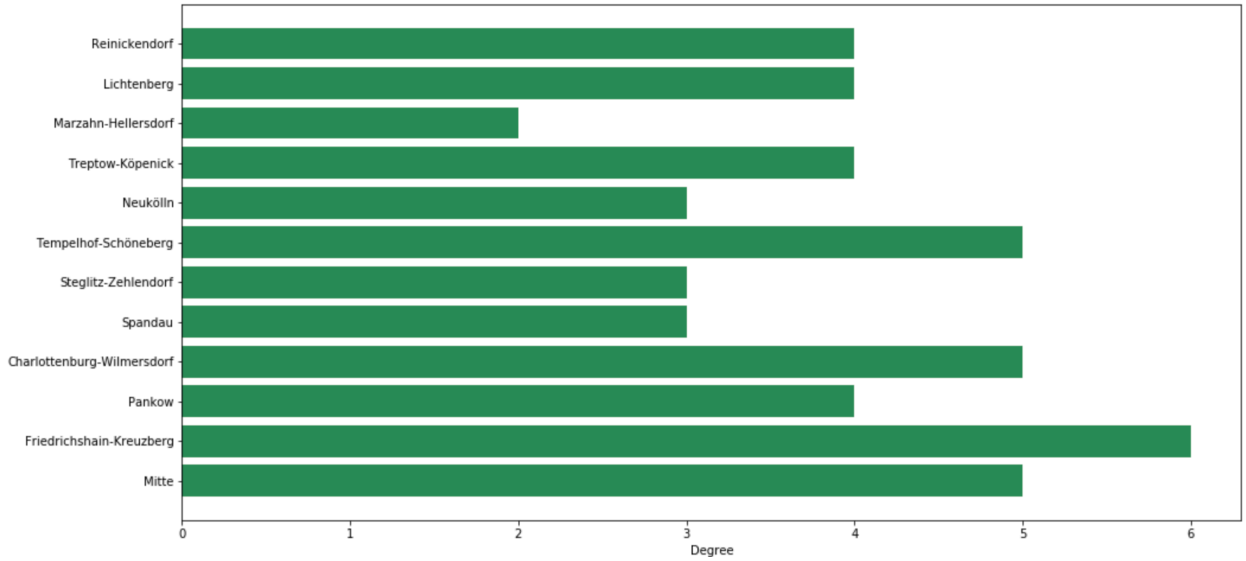

Figure 6. Bar graph of the different neighborhoods and their degrees. They are again relatively uniform.

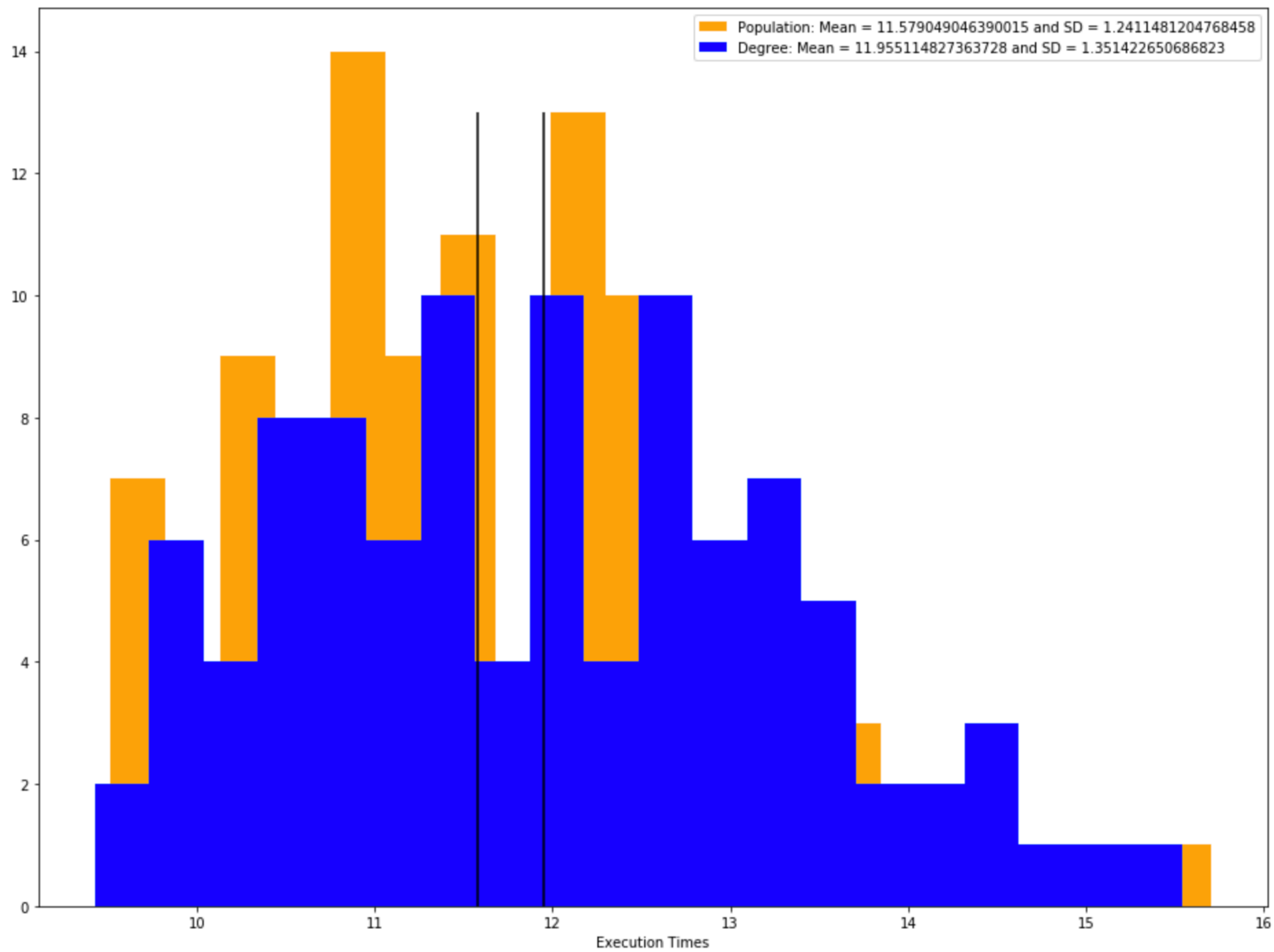

Figure 7. *Distributions of the execution times, comparing the different strategies. The left black line is the mean of the population execution times and the right black line the mean of the degree execution times.*

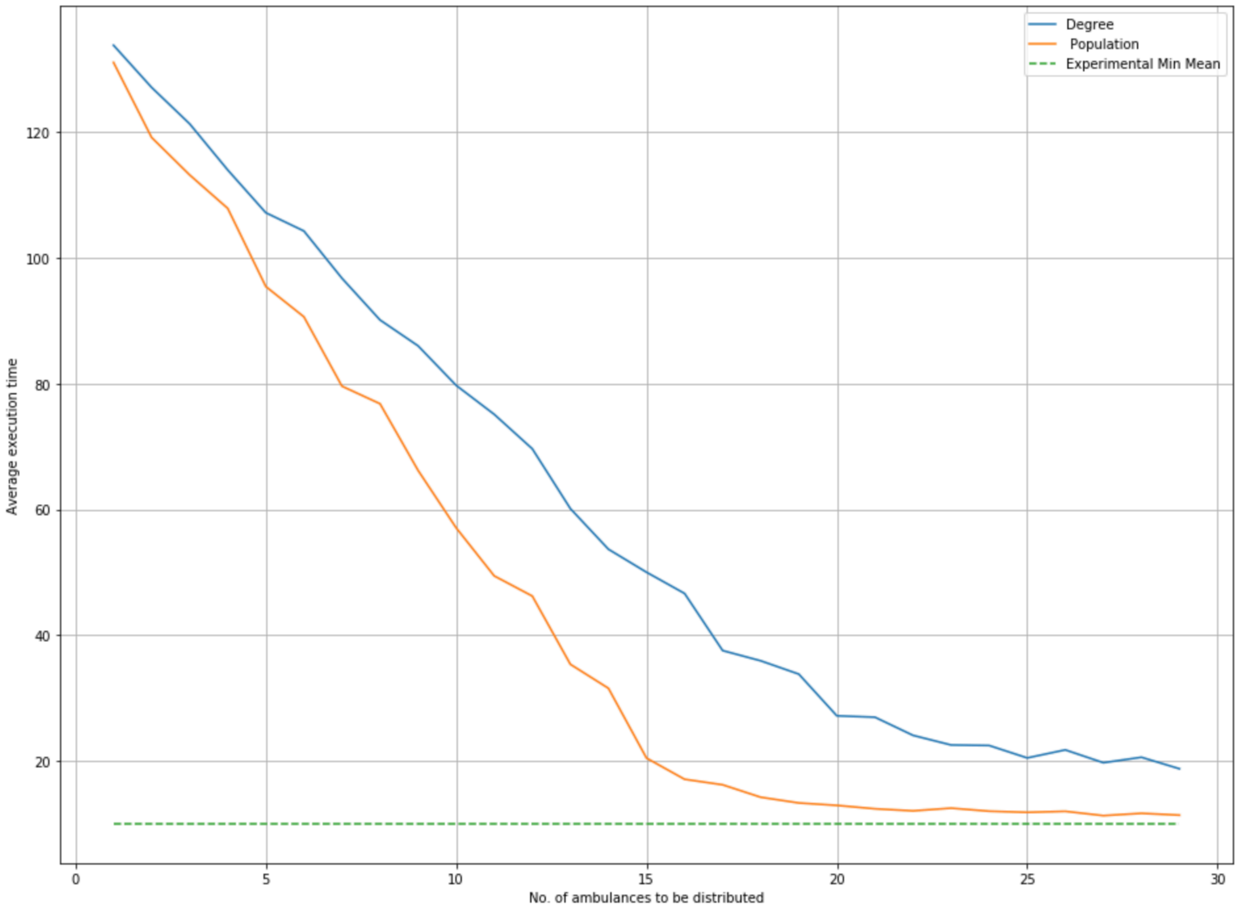

Figure 8. *Using a complete graph and power law distribution for the population. The Population oriented strategy beats the degree-oriented strategy clearly within four simulation-hours. The size of the network is the same.*

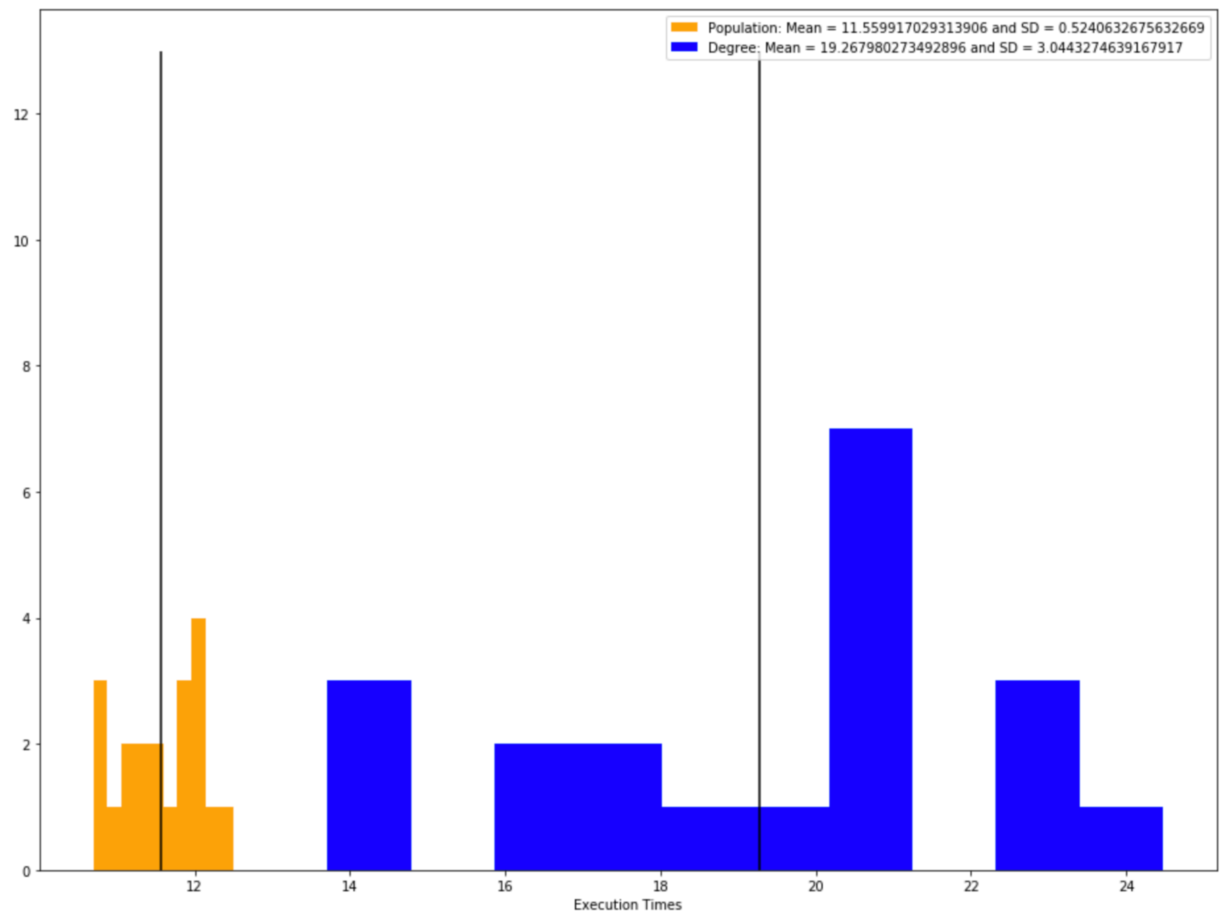

Figure 9. Distributions of the execution times, comparing the different strategies with the power law distribution for population. The left black line is the mean of the population execution times and the right black line the mean of the degree execution times. The population-strategy 95% confidence interval ranges between 10.69 and 12.25 minutes. The degree-strategy 95% confidence interval ranges between 14.37 and 23.29 minutes. Hence, the execution times are statistically significantly different.

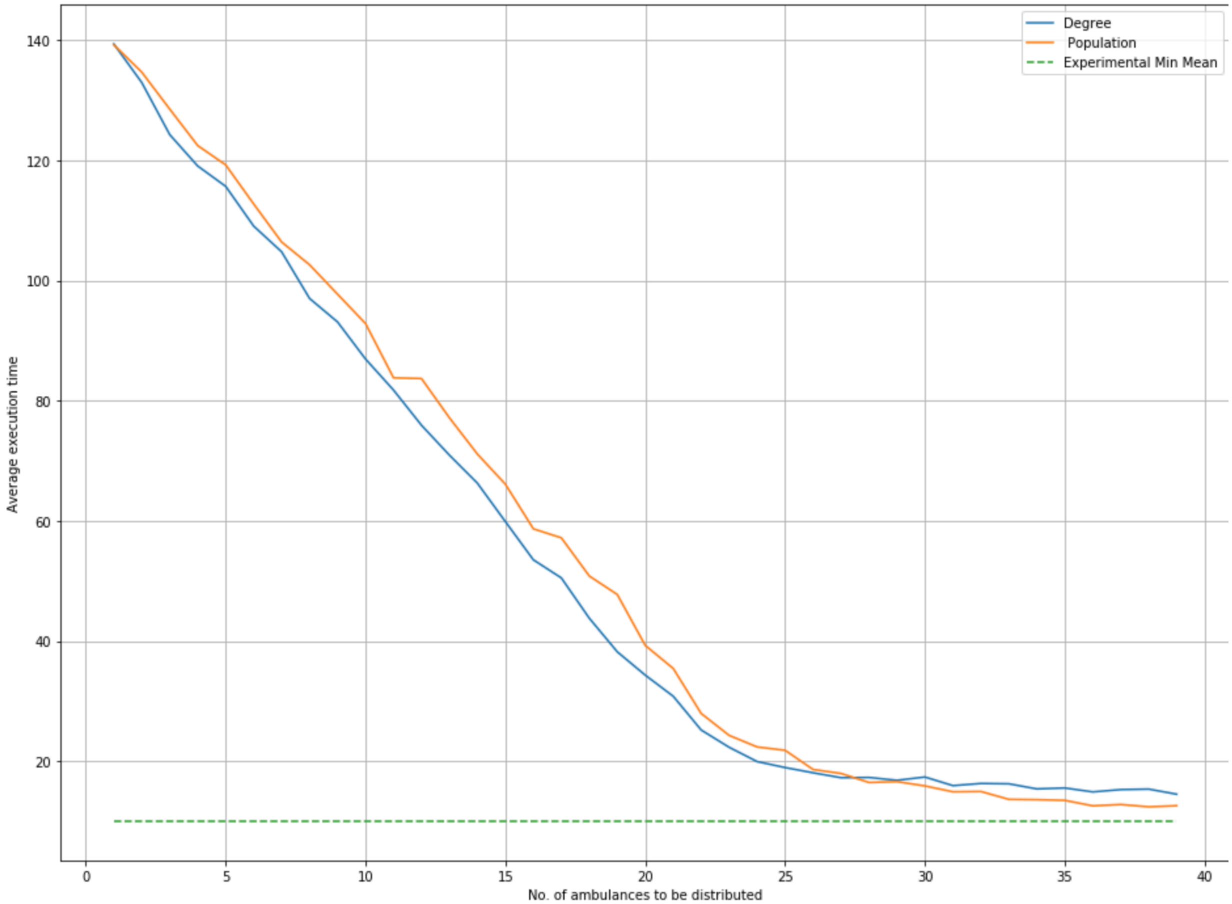

Figure 10. Using the Barabasi-Albert Graph, the degree-oriented simulation outperforms the population-based strategy in the first 4 hours and for around 26 ambulances. Afterwards the population strategy starts outperforming again.

Limitations and possible extensions

The Berlin map is a simplistic representation of traffic, connectivity, and population of the city. The traffic between neighborhoods is estimated using a simple proxy of time that penalizes for going outside the neighborhood. It does not actually represent how long it takes to get from one neighborhood to the other. To improve accuracy, real data for the edge weights should be considered. To make it even more realistic, these edge weights could change with rush hours or emergencies having an attribute "car crash" lets the traffic component of their neighborhood change if there is car crash. This feeds into another modeling assumption that is a limitation to the real-life applicability of the nodes. The traffic component in each neighborhood is purely proxied by the number of suburbs. However, in real life, Mitte may be a high-traffic neighborhood as it is quite dense and busy. Hence, even though here it has a relatively low score of 4, in real life it should be a different criterion, such as the number of vehicles passing through 30 randomly selected intersections, that estimates the traffic at a node.

Further, to make a more suitable simulation for Berlin, using a more fine-grained network that includes every single suburb would be more suitable. Using a directed graph would also allow to more closely simulate rush hours, with certain edges being busier during the morning when commuters block the streets towards the city center and in the opposite direction in the evening.

The population that is used to represent Berlin is outdated and lacks specificity in demographics and distribution. The estimates are from 2007 and, therefore, outdated. However, testing them against more recent statistics from 201. Further, they do not specify the demographics of the city which contributes to the likelihood of emergencies. For example, Steglitz-Zehlendorf is has a higher average age than Friedrichshain-Kreuzberg (Amt für Statistik Berlin-Brandenburg, 2019). With age being a major factor to the likelihood of a medical emergency, making more precise predictions about the likelihood of emergencies in a neighborhood would be preferable.

Practically, the simulation would start getting very slow with the increments that are used. Having to iterate through every minute of the simulation becomes very costly. Another room for run-time improvement would be when finding the closest ambulance. Currently, the shortest-path length to all ambulances are found and the minimum taken. However, using a greedy strategy that goes by out in a radius of edges would suffice in finding the closest ambulance as the edge weights are not varied enough that an ambulance two edges away could actually be closer than one that is one edge away. In a large graph, calculating the shortest path length starts becoming highly computationally expensive as with a large number of ambulances for every emergency, and every ambulance, the shortest path length has to be computed. Another alternative approach would be to calculate the shortest path length between to nodes at the beginning of the simulation and store it in a matrix that can then be accessed.

However, the purpose of this simulation is to compare different strategies in a descriptive way instead of trying to achieve accurate results for the execution times.

Works Cited

Amt für Statistik Berlin-Brandenburg. (2008). *Melderechtlich registrierte Einwohner im Land Berlin am 31. Dezember 2007.* Berlin: Amt für Statistik Berlin-Brandenburg.

Amt für Statistik Berlin-Brandenburg. (2019). *Demografische Daten zu Berlin*. Retrieved from Berlin: https://www.businesslocationcenter.de/wirtschaftsstandort/berlin-im-ueberblick/demografische-daten/

B. Z. (2014, 06 17). *In neun Minuten kommt der Notarzt*. Retrieved from B. Z.: https://www.bz-berlin.de/berlin/alle-2-minuten-rueckte-die-feuerwehr-aus

TUBS. (2010, May 26). *File:Berlin Subdivisions.svg*. Retrieved from Wikimedia Commons: https://commons.wikimedia.org/wiki/File:Berlin_Subdivisions.svg

LOs and HCs

#rightproblem: Throughout the assignment strived to solve the underlying problem. the simulation is trying to tackle, which is to find the best strategy of allocation. It is not to precisely or accurately estimate the execution times or even the occurrences of emergencies. Getting these details right does contribute to solving the problem more accurately and with higher internal validity but is of secondary concern since any misspecification of, e.g., traffic times is a systematic error that shifts the results but does not affect the underlying comparison of the strategies. I successfully use this HC, as the entire assignment is tailored towards comparing and contrasting the results of the strategies.

#networks: Understanding how the underlying network characteristics affect the way emergencies are generated and execution times are affected meant that I was able to implement and compare different strategies successfully. A particularly good example of this is my comparison of how to manipulate the network map I am using to show stronger differences in results by using a completed graph with nodes having a power-law distribution, and a BA graph to show different edge distributions.

#mcmodeling: While referring to the underlying goals of the simulation, I make use of my analysis of the network and the rules of the simulation, to optimize on the underlying distribution by sampling from it. This allows me to discuss and interpret the results of the simulation and analyze how the underlying distribution of population and degree matter.

#mcanalysis: I correctly identify and analyze how the model rules of determining the execution time matters when analyzing the model results. Understanding and showing how the underlying network distributions affect the results, can be used to evaluate the two strategies in a highly sophisticated way.

Appendix

Population 90% confidence interval from [57.9968709 75.91090348] Degree 90% confidence interval from [56.23666667 66.36086827]

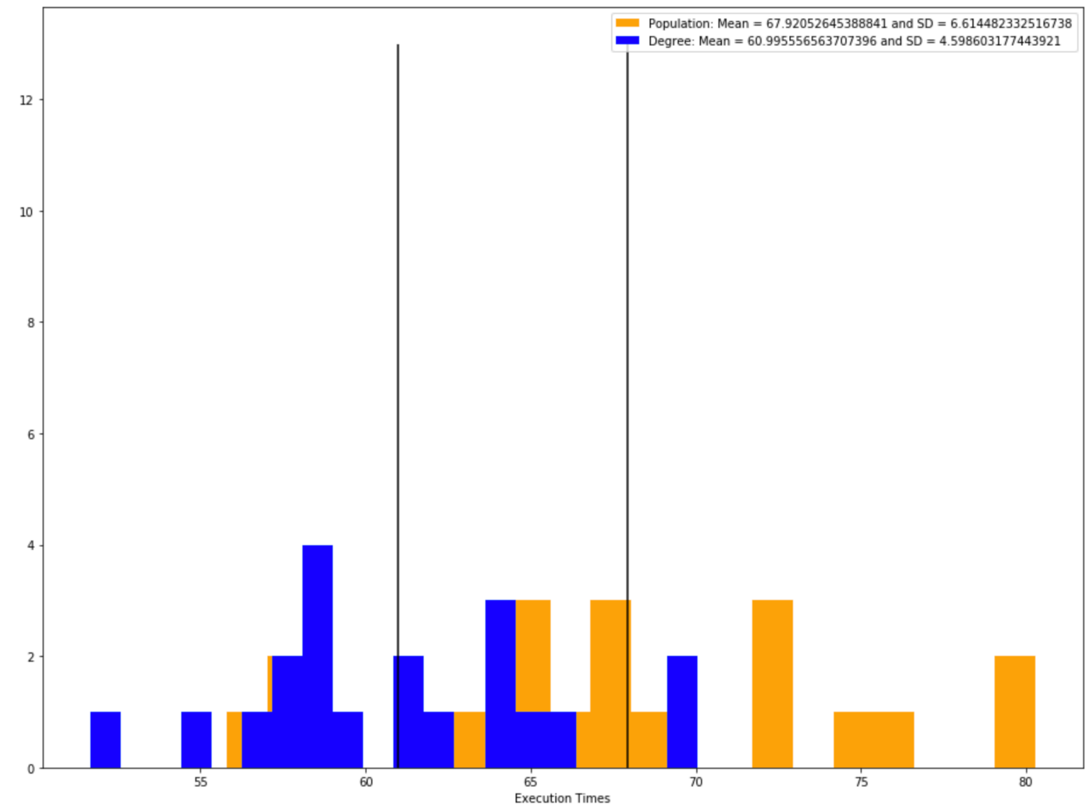

Figure 11. The nonsignificant difference in strategy outcomes for 15 ambulances with the BA graph.