Title
A Simulator for Ambulance Dispatch

Journal
The Equilibrium, 4(1)

Authors
Yuan, Hans
Lam, Timothy

Publication Date
2018

DOI
10.5070/Q24141226

A Simulator for Ambulance Dispatch

Hans Yuan
Thurgood Marshall College, Class of 2018

Timothy Lam
Earl Warren College, Class of 2018

Advisor: Professor Mauricio de Oliveira, MAE, UC San Diego

Affiliations: Instituto Tecnologico de Tijuana, Cruz Roja de Tijuana

Clinton Global Initiative University Conference in Fall 2018

ABSTRACT: Ambulance dispatch is a system where an operator directs a set of ambulances to locations where people require emergency medical assistance. A few seconds can still be decisive in whether the ambulance and the medical technicians arrive on time to save the patient. To further study how ambulance travel time affects success rates, we have written software to simulate ambulance dispatch using past data. Running this program on varying parameters such as number of ambulances and bases show minor to stark differences in the effectiveness of a group of ambulances to respond to emergency medical cases within a city. Using this tool, it becomes possible to highlight the effectiveness of increasing the number of ambulances in Tijuana in order to better execute ambulance dispatch.

1. Introduction

The Red Cross of Tijuana/Cruz Roja de Tijuana is a health services organization based in Tijuana, Mexico that manages approximately 11 ambulances for Tijuana. They respond to more than 90% of the emergency calls in the region. With a population of two million, this is close to two hundred thousand people per ambulance (World). Consequently, the performance of each ambulance is vital to the success in responding to calls for urgent assistance. Currently, a dispatcher manages ambulances from a set location using radio. On a typical call, after the ambulance is dispatched, it travels from one of eight bases throughout the city to the emergency, attends to the case, and may transport the patient to a hospital for further treatment. The ambulance then returns to its original base location.

This emergency system structure is typical in the developing world and takes little advantage of today's communication and computing technology. In recent years, the Tijuana Red Cross has been looking for ways to use mathematics and computer science to improve ambulance dispatch. Before intervening, ambulances responded to emergencies on an average of 23 minutes. With advances in software, we want to see how we can improve Cruz Roja's daily ambulance operations through improved digital user interfaces, data analysis, and live recommendations from simulations. In an earlier step of our project, Dibene et. al. (2017) used combinatorics to find the best set of starting geographic locations where ambulances can, on average, respond to emergency cases within ten minutes for 95% of the region. The previous improvement to ambulance response times was an update to a static system.

We wish to add a step to findings from Dibene et. al. and improve ambulance performance by using computers to help determine optimal choices and paths for ambulance dispatch. This new system would be dynamic - ambulances will be directed based on the live locations of all ambulances. That information is provided by the mobile and cloud networks, where each ambulance has a smartphone actively transmitting its precise location through a secured channel to a central server. With that information, it becomes possible to tie the simulator into this location system to predict optimal ambulance dispatching in real time.

We present a simulator for ambulance dispatch. It serves two main purposes:

1. To simulate historical dispatch cases with varying initial resources and dispatchment decision making
2. To take live ambulance location data and recommend to dispatchers the optimal ambulance to dispatch

The first feature allows using historical data to analyze the performance of ambulances in various hypothetical scenarios. For example, we can vary the number of bases in the simulation. Using the combinatorial procedure from Dibene et. al., we can compute the optimal locations for base placement. For varying numbers of bases, we will produce different sets of starting bases. Furthermore, we can also vary the number of ambulances and the ambulance selection algorithm. The program can simulate a set of historical cases with different values for these parameters and show heightened or worsened performance in responding to those emergency cases.

The second feature runs actively throughout the day. As the dispatcher assigns a new emergency case to an ambulance, the changes in the ambulance's state reflects in the data. The project as developed by Global Ties already allows the dispatcher to actively track the ambulances. It will become possible to plug-and-play the simulation into the deployed project, allowing the dispatcher to run different scenarios depending on which ambulance is deployed. These predictions occur within seconds, but can save an ambulance seconds or minutes towards arriving at the area of emergency.

In the future, it will become possible to use live ambulance location data that is transmitted over the internet to generate live recommendations from the ambulance dispatch simulator. This will help dispatchers decide how to respond to new calls for medical assistance. Through these tools, we anticipate that ambulance dispatching in under-resourced areas will see improvements to the amount of time it takes for an ambulance to respond to emergencies. Although our pilot program continues in Tijuana, the software and features are open-source and available at our GitHub repository. We expect that this software will help revolutionize the way ambulances are dispatched throughout the world at large.

Section 2 explores related work including a previous research step in our Cruz Roja project. Section 3 gives an overview of the simulator itself and its basic procedure. Section 4 dissects the simulation into components/models. Section 5 introduces the metrics and evaluation methods for measuring impact. Section 6 shows our simulation results. Section 7 offers a discussion on the project. Section 8 concludes this entry.

2. Related Work

In earlier work related to our Cruz Roja project, Juan Carlos Dibene used combinatorics to find the optimal set of starting locations (bases) for ambulances to idle. His work involved identifying the location of all past emergency calls, clustering them, and finding the optimal locations where ambulances may reach emergencies within 10 minutes.

Previous work related to ambulance simulations include Maxwell et. al.'s (2009) Ambulance Redeployment policies which are simulated in software using dynamic programming and simulations to find whether dispatch runtimes can be improved (Ni et. al., 2012; . Aboueljinane et. al., 2012; Aringhieri, 2007).

3. Simulation Program

Simulator: The simulator is the big-picture controller which emulates the passage of time. The basic mechanism of the simulator runs as follows: as long as there are still cases to attend, load the next case into the simulator. Before starting the case, check which ambulances are available. Then, using a selection algorithm, pick a particular ambulance to dispatch to the case. There can be different criteria for selecting an ambulance. For example, the simulator can choose an ambulance arbitrarily, choose the ambulance with the smallest projected travel time to the location, choose an ambulance that maintains the best overall coverage of the city, or a combination of these. This cycle continues until there are no more cases to run. If there are no ambulances to assign for a case, then the simulator delays it until an ambulance is available.

4. Models

To design the program intuitively, we used object-oriented programming in Python. The active elements of real life attendance to urgent medical cases include the ambulance capability, the physical locations of the ambulance, the case urgency, the hospitals, the ambulance bases, and the date and time at which events occur.

Cases: Each case includes information on when the call was placed, the location of the emergency, and the priority of the emergency (urgent, mildly urgent, not urgent). During the lifetime of the simulation, we can compute the time an ambulance was dispatched for the call, which ambulance was selected, and when the call finished. The simulation accumulates this information to allow for the summarization and analysis of performances on the cases.

Location Sets: Some elements of the ambulance dispatch require only knowledge of the GPS coordinates. The starting point of ambulances and the location of the case are examples. Often, we want to find a particular point within a set of locations. We use K-Dimensional Trees (kd-trees) to efficiently find the closest point within the set.

Ambulances: Each ambulance includes information on its license plate, its location, and whether it is deployed.

5. Metrics

To decide the effectiveness of a series of dispatches, it is necessary to define the kind of metrics we use in order to observe and measure.

Regional Coverage: Generally, the coverage of the city is a metric quantifying the percentage of the city reachable by an ambulance within some time requirement. For this project, we defined the coverage as the number of demand points covered by an ambulance within 10 minutes. There are 100 demand points. Dibene et. al. clustered these points based on the tens of thousands of cases Cruz Roja recorded.

Since Cruz Roja uses the US EMS Act (Ball & Lin, 1993) as a standard for their dispatch standard, it is useful to know the current coverage of the region. Some ambulances may already be attending to a case. These ambulances are busy and cannot be considered when determining the overall coverage of the city. If an ambulance is dispatched and no other free ambulance is nearby, then the previously covered area may become exposed.

The reverse function may also be useful. We compute the required minimum travel time radius for each ambulance to maintain a percentage coverage of a city. For example, if we were to maintain a 60% coverage as opposed to a 40% coverage, the travel time radius for each ambulance may increase. A graphical example can be seen in *Figure 1*.

Ambulance Selection: The simulation is affected by the ambulance selection policy. A few outcomes are desired, such as the need for the ambulance to reach the destination in minimal time. Sending the fastest ambulance at any given time would accomplish this. However, when the region is restrained by the number of ambulances present at once, choosing the fastest ambulance for each case may not be the optimal option in the big picture. Not all emergency cases are equally as urgent. Some cases may allow for additional time to pass without causing harm, while other cases absolutely need the closest ambulance possible. There are several metrics for choosing an ambulance. The impact of the chosen metric can be seen in *Figures 2-4*.

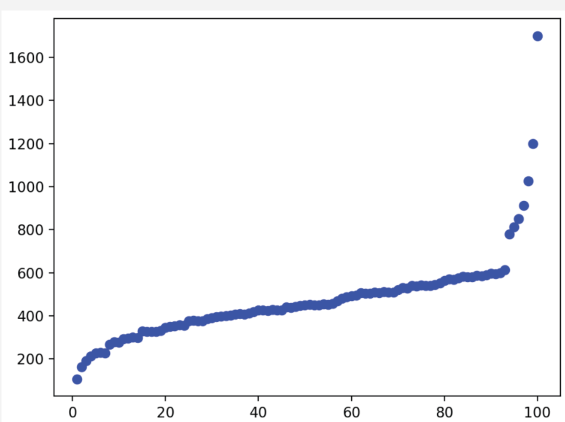

Figure 1. Given the desired coverage of Tijuana after an ambulance is dispatched (x axis in percentage), returns the number of seconds required to achieve that coverage. As the desired coverage increases from 90%, the amount of time needed for each ambulance increases drastically.

Point	Meaning
Blue	Demands covered
Red	Demands exposed
Yellow	Bases
Green	Location of medical emergency
Black	Chosen ambulance

Legend for Figures 2, 3, and 4.

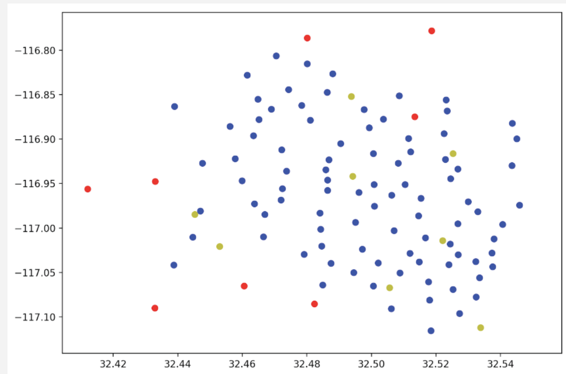

Figure 2. The coverage of Tijuana at a given time. X-Axis: Longitude. Y-Axis: Latitude.

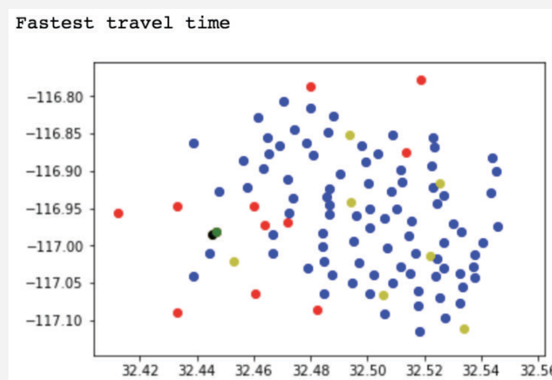

Figure 3.Urgent medical attention needed at the green point, the fastest ambulance is at the black point. Certain previously covered demands points are now exposed.

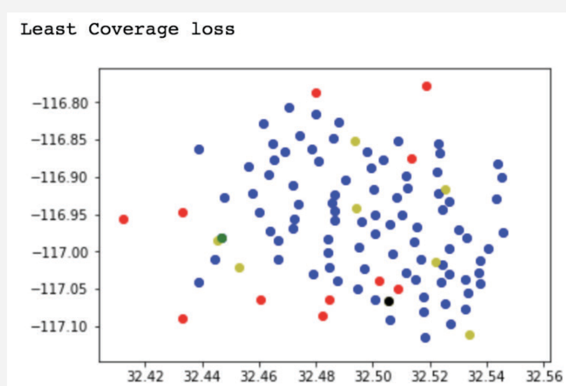

Figure 4. Urgent medical attention needed at the green point, the ambulance that disrupts the coverage the least is at the black point.

When a call comes in, if the case is urgent, then we may send the ambulance that has the lowest projected travel time to maximize the chance of arriving on time. If the call coming in is completely non-urgent, then we could send the ambulance that least disrupts the overall coverage of the region in anticipation for urgent calls. There are issues with the above two choices. The case may not always require the fastest ambulance, but most cases still need ambulances to arrive at a reasonable time. Thus, there is a tension between choosing a fast enough ambulance while anticipating a case where the emergency is dire.

Using the simulation, dispatchers will be able to quickly rank ambulances based on their location and availability on demand. The simulator returns two lists: ambulances ranked by fastest travel time to the destination, and ambulances ranked by lowest disruption on the regional coverage. With this information, dispatchers are presented with various options instantly to select an ambulance based on whether faster travel times or better coverage is desired for that particular case. Examples can be seen in *Figures 5 and 6*.

Quantifying the Desired Outcome: While ranking the fastest travel times and best regional coverage are each desired, it is useful to find a numerical relationship between the mentioned metrics. Since the travel times should be minimized while the coverage is maximized, we can relate them by dividing the travel time by the coverage. In *Figures 5 and 6*, this is the "composite" value.

6. Results

Simulation Effects: Selecting the fastest ambulance and selecting the ambulance that least disrupts coverage produce different results. Sending the fastest ambulance tends to leave many demand points uncovered. On the other hand, sending the ambulance that minimizes coverage disruptions incurs a higher response time to the case. Demand points on the outskirts of Tijuana were noticeably uncovered by the ambulances stationed in bases placed in denser areas of the city. In *Figure 1*, we notice that ambulances must cover a larger distance, as measured in seconds, to maintain an increasing minimum coverage. In particular, travel times dramatically increase when we wish to maintain a minimum coverage of around 90% in Tijuana.

Runtime Performance: The simulation was tested using 8 months of data containing 23,000 unique cases collected by Cruz Roja. In general, the lower the number of ambulances and bases, the faster the algorithm runs. The following times were gathered from running three simulations simultaneously on a 2016 Macbook Pro, 15 inch. For three ambulances and three bases, the simulation took 9 minutes and 37 seconds. Running the simulation for 7 ambulances and 7 bases took 20 minutes and 4 seconds. In general, the runtime increased as the number of ambulances and bases rose.

Figure 1 models a tension between the ambulance coverage travel time radius and the minimum required coverage in Tijuana. *Figure 2* displays the Tijuana coverage at a given time. *Figures 3 and 4* display the location of a single case during the simulation and mark the ambulance chosen for dispatchment. *Figures 5 and 6* show the chosen ambulances' rankings via different ambulance selection methods.

```
Sort by travel time.
(4, 99,  88.,  4.,  1.125)
(6, 789,  88.,  4.,  8.96590909)
(1, 1064,  76.,  16.,  14.)
(3, 1144,  89.,  3.,  12.85393258)
(0, 1279,  75.,  17.,  17.05333333)
(7, 1298,  89.,  3.,  14.58426966)
(5, 1625,  88.,  4.,  18.46590909)
(2, 1691,  76.,  16.,  22.25)
Shortest Travel:  (4, 99,  88.,  4.,  1.125)

Sort by maximal coverage.
(3, 1144,  89.,  3.,  12.85393258)
(7, 1298,  89.,  3.,  14.58426966)
(4, 99,  88.,  4.,  1.125)
(5, 1625,  88.,  4.,  18.46590909)
(6, 789,  88.,  4.,  8.96590909)
(1, 1064,  76.,  16.,  14.)
(2, 1691,  76.,  16.,  22.25)
(0, 1279,  75.,  17.,  17.05333333)
Least disruption:  (3, 1144,  89.,  3.,  12.85393258)
```

Figure 5. Sample rankings of ambulances. Each row's values are (ambulance ID, travel time in seconds, resulting percentage covered in Tijuana, the number of demands newly exposed, and the composite value). Ambulance #4 can reach the destination in 99 seconds while sacrificing 1% more coverage than the ambulance which maximizes the resulting coverage.

```
Sort by travel time.
(3, 484,  89.,  3.,  5.43820225)
(7, 695,  89.,  3.,  7.80898876)
(0, 757,  75.,  17.,  10.09333333)
(6, 805,  88.,  4.,  9.14772727)
(1, 1271,  76.,  16.,  16.72368421)
(4, 1313,  88.,  4.,  14.92045455)
(5, 1600,  88.,  4.,  18.18181818)
(2, 2081,  76.,  16.,  27.38157895)
Shortest Travel:  (3, 484,  89.,  3.,  5.43820225)

Sort by maximal coverage.
(3, 484,  89.,  3.,  5.43820225)
(7, 695,  89.,  3.,  7.80898876)
(4, 1313,  88.,  4.,  14.92045455)
(5, 1600,  88.,  4.,  18.18181818)
(6, 805,  88.,  4.,  9.14772727)
(1, 1271,  76.,  16.,  16.72368421)
(2, 2081,  76.,  16.,  27.38157895)
(0, 757,  75.,  17.,  10.09333333)
Least disruption:  (3, 484,  89.,  3.,  5.43820225)
```

Figure 6. Sample rankings of ambulances. Each row's values are (ambulance ID, travel time in seconds, resulting percentage covered in Tijuana, the number of demands newly exposed, and the composite value). Ambulance #3 is the optimal ambulance for both metrics: it is the fastest ambulance and least disrupts the coverage.

7. Discussion

Limitations: The simulation software piece is composed of pieces of logic to model a real world scenario. Thus, the current simulation form is static and deterministic: running the simulation on the same cases with the same parameters will result in the same performance.

Reality is much more complex and random. For example, traffic patterns may shift for unknown reasons, delaying an on-route ambulance more than usual. A possible solution for this is to consider the way the simulator calculates ambulance travel time. Since the components of the simulation are modularized into different classes, developers need to simply improve or replace the component to account for these situations.

Thus far, we have worked with only Cruz Roja in the development of this simulator. Considering ambulance dispatch at scale, for example in different countries, there may be different types of information useful for the simulation. The solution is the same as above. The simulation is broken into various steps and modules. In order for this simulation to work with different inputs of data, parts of the simulation would be modified or replaced. Modifications can be easily made because of the adoption of object-oriented programming and the Python programming language.

Future Work: In future quarters, the Global Ties Cruz Roja undergraduate team will connect the dispatcher user interface and the core server runtime with this simulator. Currently, this simulation can only use past data to simulate the ambulances. Upon plugging in the simulator, it will become possible to use the already-existing live ambulance trackers to run the simulation. While the past data may be useful to inform travel times in the live simulation, it may also be useful to integrate e.g. Google Maps API calls into the simulation, giving a second opinion for travel times.

While the simulation itself is useful for comparing the performance of ambulances given different scenarios of ambulance counts and the number of starting locations, it will also be useful to have the simulation portray its real world counterpart accurately. Machine learning may help to improve the accuracy of the simulation itself.

8. Conclusion

The software program described here is a proof-of-concept that shows usage of historical data and varying parameters such as ambulance count and starting locations to show increases or decreases in ambulance dispatch performance. We discussed the possibility of further generalizing this simulation to plug into a live-dispatching system to produce realtime information. We will further implement these ideas and test them in Tijuana. Our efforts offer a promising look at optimizing ambulance dispatch to potentially revolutionize emergency services at scale.

Acknowledgements

Professor Mauricio de Oliveira mentored us on doing research and developing high quality software. He spent years advising each layer of the Cruz Roja Project. Without his ideas, patience, and expertise, we would not have been able to implement this project.

Professor Scott Klemmer offers candid feedback. His lessons on human-centered design and the design process were pivotal to our development.

References

Aboueljinane, L., Jemai, Z., & Sahin, E. (2012). "Reducing ambulance response time using simulation: The case of Val-de-Marne department Emergency Medical service," Proceedings of the 2012 Winter Simulation Conference (WSC), Berlin, 2012, pp. 1-12. doi: 10.1109/WSC.2012.6465018

Aringhieri, Roberto. (2007). Ambulance location through optimization and simulation: the case of Milano urban area.

Ball, M. & Lin, F. (1993). "A reliability model applied to emergency service vehicle location," Operations Research, vol. 41, pp. 18-36.

Dibene, J.C., Maldonado, Y., Vera, C., Oliveira d.M., Trujillo, L. & Schtze, O. (2017). Optimizing the location of ambulances in Tijuana, Mexico. Comput. Biol. Med. 80, C (January 2017), 107-115. DOI: https://doi.org/10.1016/j.compbiomed.2016.11.016

Maxwell, M. S., Henderson S. G., & Topaloglu, H. (2009). "Ambulance redeployment: An approximate dynamic programming approach," Proceedings of the 2009 Winter Simulation Conference (WSC), Austin, TX, 2009, pp. 1850-1860. doi: 10.1109/WSC.2009.5429196

Ni, E. C., Hunter, S. R., Henderson, S. G., & Topaloglu, H. (2012). "Exploring bounds on ambulance deployment policy performance," Proceedings of the 2012 Winter Simulation Conference (WSC), Berlin, 2012, pp. 1-12. doi: 10.1109/WSC.2012.6465198

World Population View http://worldpopulationreview.com/world-cities/tijuana-population/

Hans Yuan

Computer Science

Hans graduated from UC San Diego with a Computer Science undergraduate degree. Passionate about teaching, he worked as a TA and has done Computer Science education research. His other interests include engineering leadership and management, politics, and science. Involved with the Cruz Roja Project for the third year, he will be attending the Clinton Global Initiative University conference to share the project and learn about humanitarian efforts done by other students. He'll use the new insight to continue contributing to the project.

Timothy Lam

Computer Engineering

Timothy Lam graduated from the University of California, San Diego in March 2018 and is currently pursuing a Masters in Computer Science there. By becoming involved with the Global TIES organization and the Qualcomm Institute at UC San Diego he developed an interest in the intersection between health, humanitarian work, and technology. With a variety of experience in mobile programming, web technologies, and machine learning, he is taking steps towards understanding how data can be collected, stored, and manipulated into solving health issues that plague the world today.