Applying Linear Programming to Optimize Fire Station Placement

Sharad Parulekar, Ashish Bhave, Matthew Cady,

Samuel Kenyon, Akarsh Tripathi

NASA Aeronautics Research Institute

Abstract

The rising trend in wildfire occurrence and severity has put a strain on wildfire management organizations by spreading out limited resources to meet increasing demand. There has been extensive prior research and data collection to determine the areas of highest risk and to predict regional wildfire damages based on historical trends. Our group aims to utilize this data to best determine fire station placement, optimizing where resources are allocated to reduce the time and investment needed to effectively mitigate wildfires. Using Data Basin, MATLAB image processing, and Python linear programming we are able to calculate optimal fire station locations by utilizing a single-objective facility location problem algorithm incentivized for cost reduction. Our findings can be applied to evaluate existing or planned fire station placement in order to effectively allocate resources.

Introduction

Over the past several decades wildfires have become increasingly frequent and have caused hundreds of millions of dollars worth of damage, on top of the mounting threat to human lives. The California wildfire community has risen to the occasion and met the challenge so far, but as climate continues to change and the wildfire season continues to lengthen the already limited resources available will be challenged. Despite effective mitigation, wildfires are inflicting more and more damage over the past decade. From 2009 to 2018 there were 1.4 times greater fires and 1.6 times larger acres burned than the per decade-averages from 1979 to 2009 (Buechi 2021). The average annual loss due to wildfires during the 2009-2018 decade was almost $1 billion, more than the $0.62 billion of the prior three decades combined (Buechi 2021). The life loss has increased comparably to the acres burned and monetary loss, with 4 total civilian deaths prior to 2000 increasing to a total of 84 civilian deaths from 2000-2018 (Buechi 2021).

Rather than developing novel wildfire fighting technologies, our team chose to focus on optimizing existing infrastructure, "Given wildfires will only get worse in the hotter, drier future, solely being reactive to them isn't a winning strategy" (Hasseltine 2022). The group decided to create an algorithm capable of determining optimal fire station placement locations based on wildfire risk factors. While reviewing relevant literature, the team found several cases of facility location problem (FLP) algorithms for similar use cases. A 2011 paper from Turkey applied their FLP algorithms for optimal placement of hospitals and fire stations in Kadikoy (Şen 2011). Inspired by their methods, the team opted to apply a custom limited-scope FLP algorithm to regions of California to determine where best to place fire stations in order to combat wildfires.

Methodology

Data Collection

Before the facility location algorithm could generate a solution, it first had to be supplied with data to operate on. We reviewed various existing data sources to determine which would best be able to both provide the information we needed and do so in an easily accessible format. Conservation Biology Institute's Data Basin was selected as it provided convenient access to both the EPA's EnviroAtlas (percent forest per 12-digit HUC) and Cal Fire's FRAP (fire hazard severity zones) data overlaid on the California map.

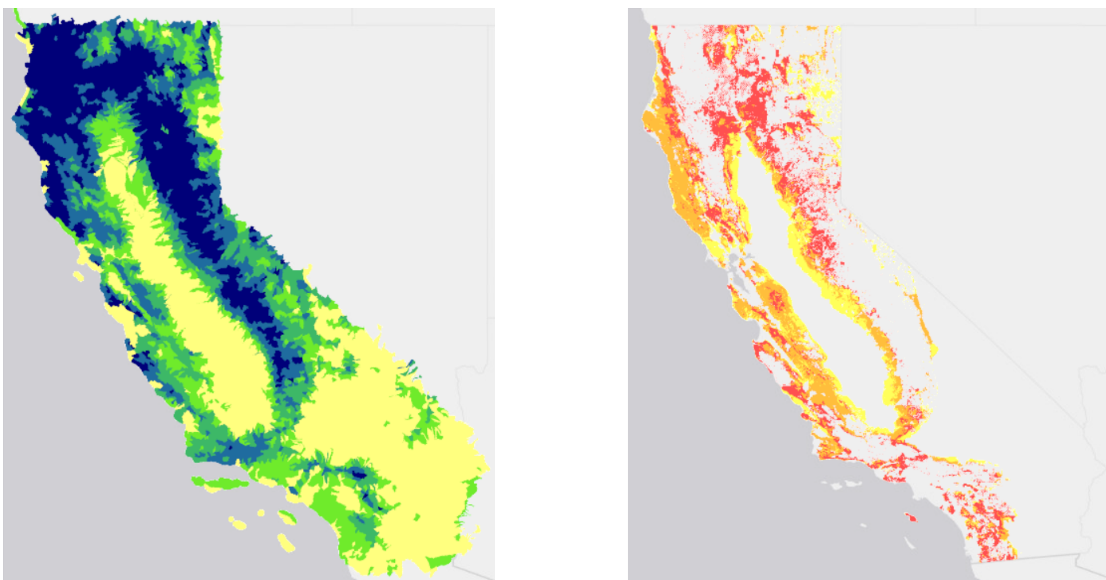
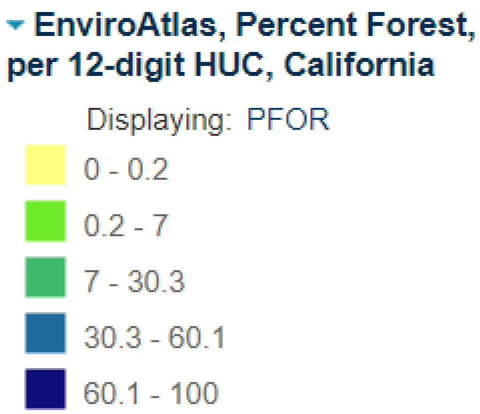


Figure 1: Data Basin EnviroAtlas Forest Density and Fire Hazard Severity Maps

We chose to pull data from the two selected Data Basin maps by screen clipping the desired portion of the map and saving it as a PNG. The program Greenshot was used in order to reliably clip the same portion of the screen repeatedly as well as measure the pixel dimensions of the clipped region. Two separate clips were taken of the same region, one of the EnviroAtlas forest density map and one of the Cal Fire FRAP fire hazard severity map.



0 - 0.2 → Land
0.2 - 7 → Light
7 - 30.3 → Moderate
30.3 - 60.1 → Heavy
60.1 - 100 → Very Heavy

Figure 2: Forest Density and Fire Hazard Severity Map Legends

Both PNGs were then imported into MATLAB where they were split into RGB matrices. For the forest density map, each pixel's RGB value was checked with the forest density legend to classify the pixel accordingly. The same process was repeated for the fire hazard severity map. Pixels whose RGB value did not match any of those from the corresponding legend were classified as null. Null values most frequently occurred due to image compression interpolating boundaries between colors (lower zoom) or when the image included roads (higher zoom). The cells that were classified as unforested "land" (A-type cells) were considered potential fire station locations, and all other forested cells (B-type cells) were considered potential sites of fire occurrence. Once all the pixels from the forest density and fire hazard severity images were classified, they were output into respective matrices with numeric values associated with each legend item. These two CSV files were exported for use in the Python facility location algorithm.

Integer Linear Programming

$$\min \sum_{i=1}^{N}\sum_{j=1}^{M} d_j t_{ij} y_{ij} + \sum_{i=1}^{N} f_i x_i$$

Figure 3: FLP Objective Function and Constraints (Cantlebary 2020)

The facility location integer linear programming model operates by minimizing an objective function, in our case the total cost of transportation and station construction (*Fig. 3*). The objective function consists of two summation terms. The two index summation term iterates over each facility *i* and customer *j* to multiply d_j (the total demand of customer *j*), y_{ij} (the fraction of d_j that facility *i* satisfies) and t_{ij} (transportation cost between facility *i* and customer *j*). This is added to a summation over each facility *i* that multiplies the corresponding facility's fixed cost, f_i, and the binary variable x_i which denotes whether or not a station exists.

$$\begin{aligned} s.t.\ & \sum_{i=1}^{N} y_{ij} = 1 \ \ \forall j \in \{1, \ldots, M\} \\ & \sum_{j=1}^{M} d_j y_{ij} \leq k_i x_i \ \ \forall i \in \{1, \ldots, N\} \\ & y_{ij} \geq 0 \ \ \forall i \in \{1, \ldots, N\}, \forall j \in \{1, \ldots, M\} \\ & x_i \in \{0, 1\} \ \ \forall i \in \{1, \ldots, N\} \end{aligned}$$

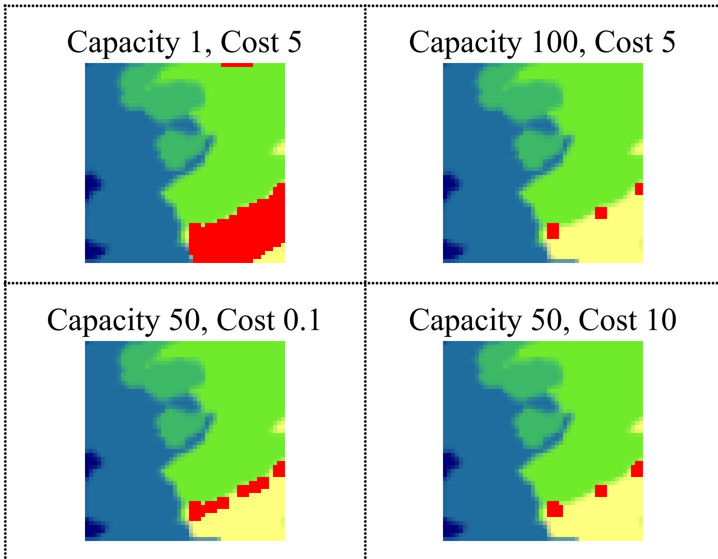
Figure 4: FLP Objective Function Constraints (Cantlebary 2020)

The terms are subject to several constraints (*Fig. 4*). The first constraint states that each fire's demand has to be met by some combination of stations' resources. The second states that the capacity of each station has to meet or exceed the total demand of all forest fires on that station. The third states that the fraction of the total demand d_j of fire j satisfied by station i must be greater than or equal to zero. The fourth and final constraint defines x_i as a binary variable with value 0 representing no station, and value 1 representing a station. Note indices i and j are integer values, constraining the system to integer numbers of fire stations and fires.

The forest density and fire hazard severity matrices generated from MATLAB are imported into the Python code for further processing before being fed into a CVXPY solver and generating the calculated optimal fire station placements. The first step upon being imported is to translate the forest density CSV file into candidate fire station and potential fire occurrence locations. If it is a type-A cell (unforested land) its coordinates are added to a station location matrix, and the number of fire stations variable is incremented. If it is a type-B cell (forested land) its coordinates are added to a fire location matrix, and the number of fires variable is incremented. In the case of a type-B cell, the coordinates' forest density CSV value is added with the fire severity hazard CSV value to create a combined risk factor variable. Using the station location and fire location matrices, a transportation cost matrix is calculated using the relative pixel distance between every single candidate station and every single possible fire. The solver is then passed the necessary parameters and outputs a binary matrix with optimal fire station placements. This binary matrix is exported as a CSV into MATLAB to visualize the results.

Results

In order to better understand the outputs of the algorithm, the group decided to conduct a sensitivity analysis. There are several parameters that could be varied in order to effect the output, including station capacity, station cost, transportation cost, and how demand is determined. We chose to vary station capacity and station cost, and recorded their effects on the number of stations placed as well as the objective function (solution “score”, lower is better).



Note: Red cells represent fire stations. Darker cells represent denser forest. For density legend see *Fig. 2*.

Figure 5: Max / Min Results of Sensitivity Analysis

The sensitivity analysis revealed the relationship of inputs to outputs. Station capacity and station costs were independent variables whose sensitivity were individually tested. The number of stations placed by the algorithm and the objective function were dependent variables measured over a range of independent variable values. When capacity’s sensitivity was measured, cost was held constant at 5 and capacity was varied from 1 to 100. When cost’s sensitivity was measured, capacity was held constant at 50 and cost was varied from 0.1 to 10. Both capacity and cost had two orders of magnitude between their lowest and highest tested value. When not being measured, the other independent variable was set to the middle of the range. The range of values chosen gave us a good look into the edge behavior of the algorithm. For example, with values of capacity 1 and cost 5, the algorithm outputted 229 stations out of a possible 260 (*Fig. 5*). This aligns with expected behavior, as due to the ILP constraints the algorithm must place enough stations to meet demand. With each station having low capacity, many stations are needed.

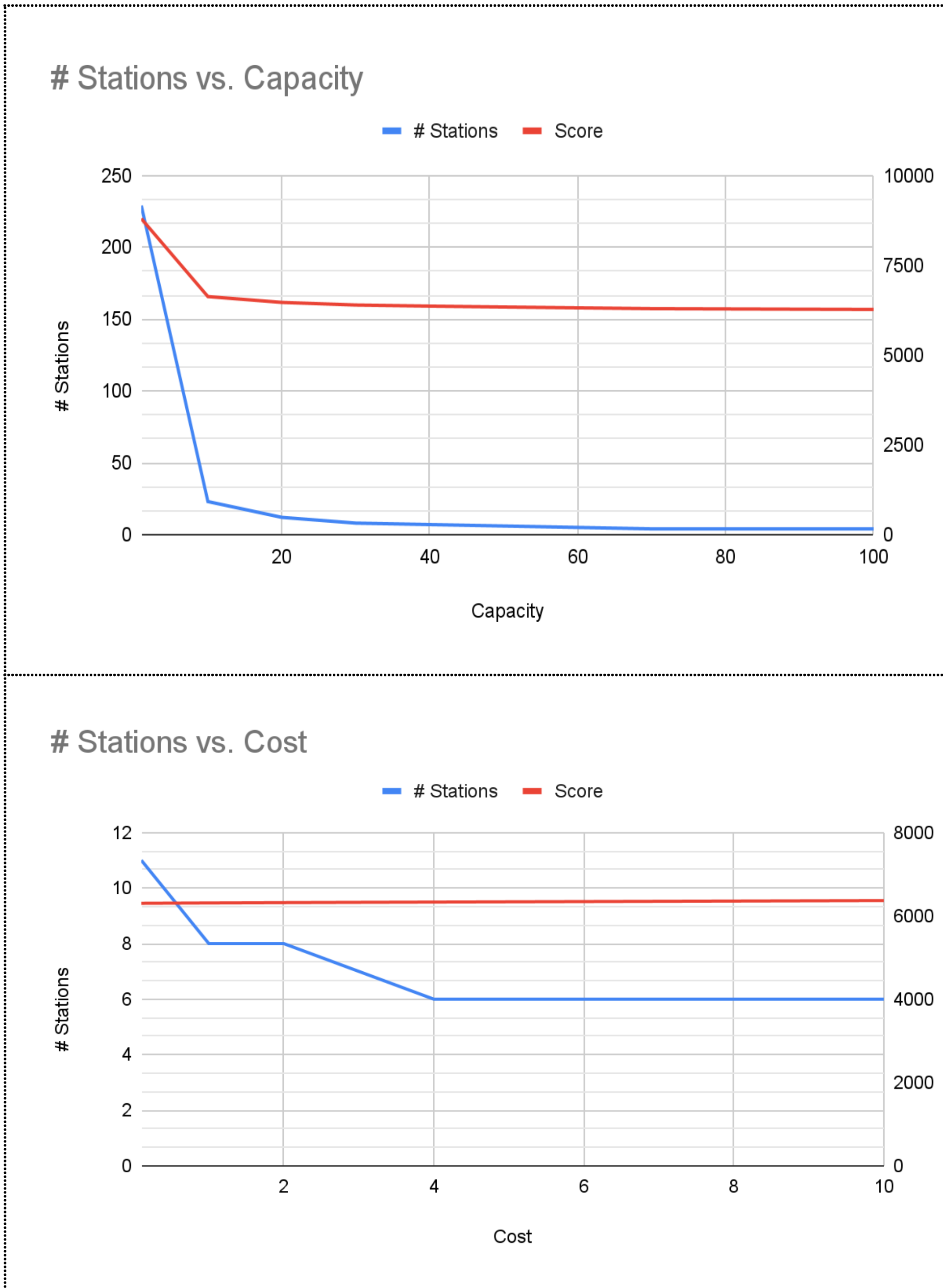


Figure 6: Sensitivity Analysis Results

As we increase the per-station capacity, we see a decrease in the number of stations placed as well as the objective score. Unsurprisingly, increasing station capacity results in less stations being needed. However the sensitivity analysis also reveals diminishing returns on increasing station capacity, as from a capacity of 70 to 100 the number of stations placed remains the same at 4 (*Fig. 6*). As the costs associated with building a fire station are increased, we can observe a decrease in the number of fire stations placed as well as an increase in the objective score. Since the objective score indirectly measures costs, its increase is to be expected. The algorithm responds to the rising objective score by placing a fewer number of stations, however it must still meet the necessary demand. Because the algorithm is able to reduce the number of placed stations even with the capacity remaining constant, we can reaffirm that the FLP algorithm minimizes the objective score rather than the number of stations. In other words, because the algorithm is able to place fewer stations and still meet demand, configurations with more stations have capacity exceeding the demand by a non-trivial amount.

Conclusion

While our project was largely a proof-of-concept, our linear programming model could readily be improved to become more accurate and thorough. The model could be adapted to support multiple types of fire stations. As NARI Interns, we focused our attention and research on fire stations that utilized aerial vehicles to combat wildfires, however the same algorithm could be used to place urban fire stations or other emergency services. The algorithm currently solves a single-objective FLP, with transportation cost being the minimized objective. If the model were to be expanded to solve multi-objective FLPs, it could account for proximity to water sources, roadways, or even prioritize certain fire sites based on the projected burn rates and/or damages.

A major improvement to the current algorithm would be to improve the estimates for station capacity and station cost. In the proof-of-concept, the values used are relative with no correlation to real values. The values are relative to fire demand, which is arbitrarily calculated from the forest density and fire hazard severity matrices. Basing capacity and cost off of practical real-world values would give the outputted results more credibility and applicability.

The existing algorithm itself could also be improved without changing the FLP objective or further research. Rather than using Python's CVXPY library, the algorithm may benefit from using MATLAB's Optimization Toolbox or a custom solver. The group ran into numerous issues with the algorithm taking longer than expected to run, sometimes even crashing from excessive RAM usage on Google Colab. MATLAB may be able to perform the required matrix operations faster and more efficiently. Another way to reduce run-time might be to avoid "redundant stations." For example, the visualized outputs in *Fig. 5* of [capacity 100, cost 5] and [capacity 50, cost 10] look very similar despite having different numbers of stations, 4 and 6 respectively. This is due to the algorithm placing fire stations very close to each other. If the algorithm were improved to avoid placing fire stations within a certain proximity of one another, it would cut down on the "redundant" placement.

Fig. 7 shows an output over the largest region the algorithm was able to reliably process. The region shown is 75 pixels by 125 pixels. Converting the pixels to miles, the region is 35.156 miles by 58.594 miles, or 2059.9 square miles. Using the arbitrarily chosen values for station capacity of 50 and station cost of 5, the algorithm placed three fire stations. The three stations placed by the algorithm in *Fig. 7* can be compared to existing fire station locations to gauge how well resources are being allocated. Of the three stations placed, community wildfire officials may be able to select one that would be most effective according to their considerations, and construct a station there. Although our model in its current form will need refinement to be more practical for determining fire station placements, it can still be used to evaluate existing fire station locations and allow for redistribution of fire fighting assets to critical stations and regions.

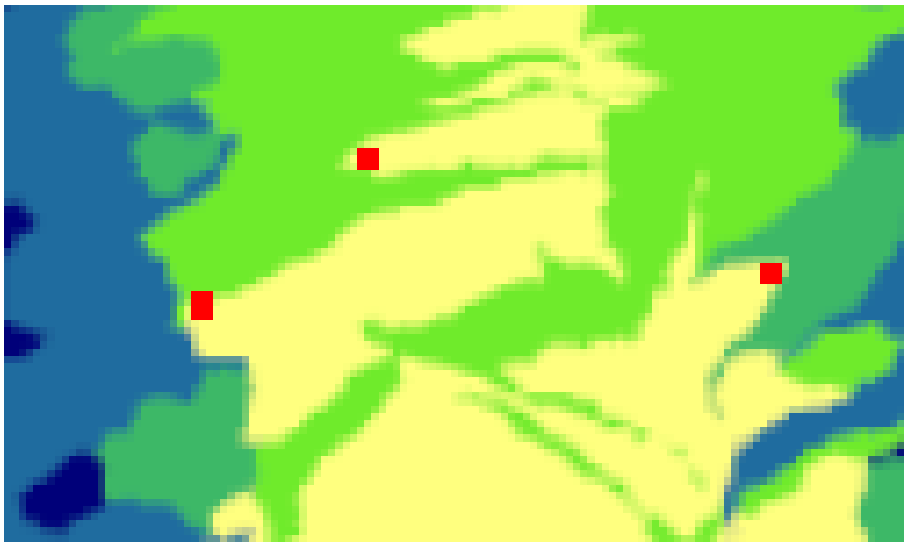


Figure 7: Final Result of FLP Algorithm

References

Buechi, H., Weber, P., Heard, S., Cameron, D., & Plantinga, A. J. (2021). Long-term trends in wildfire damages in California. *International Journal of Wildland Fire*, *30*(10), 757. https://doi.org/10.1071/wf21024

Cantlebary, L., & Li, L. (2020, December 21). *Facility location problem*. Facility location problem - Cornell University Computational Optimization Open Textbook - Optimization Wiki. Retrieved August 3, 2022, from https://optimization.cbe.cornell.edu/index.php?title=Facility_location_problem

Conservation Biology Institute. (2015). *California forest coverage and existing fire protection facilities*. Data Basin. Retrieved August 3, 2022, from https://databasin.org/maps/64bf4f194ecf4658ab8d029ef0189d35/active/#

Hasseltine, D. W. and D. (2022, July 30). *Wildfire season is here. California needs to fight back smarter*. San Francisco Chronicle. Retrieved August 11, 2022, from https://www.sfchronicle.com/opinion/openforum/article/Wildfire-season-California-fire-17337351.php

Şen, A., Önden, I., Gökgöz, T., & Şen, C. (2011). A GIS Approach To Fire Station Location Selection. Paper presented at the Gi4DM 2011 - GeoInformation for Disaster Management, Retrieved from www.researchgate.net

Acknowledgements

We would like to thank Dr. Parimal Kopardekar for his mentorship, Christine Clark for her support throughout the project, and members of the California wildfire community for speaking with us about the challenges they face.