For office use only
T1 ________________
T2 ________________
T3 ________________
T4 ________________

For office use only
F1 ________________
F2 ________________
F3 ________________
F4 ________________

2013

16th Annual High School Mathematical Contest in Modeling (HiMCM) Summary Sheet

(Please attach a copy of this page to your Solution Paper.)

Team Control Number: 4552
Problem Chosen: A

Please type a summary of your results on this page. Please remember not to include the name of your school, advisor, or team members on this page.

The continental United States is divided into several states, counties, and districts all with their own populations, geography, and needs. We were provided with one such county and asked to find the most efficient allocation of ambulances in order to maximize the number of people who could be reached in under a certain response time. Using only the county's demographics and the average travel times between the county's zones, we devised a model to determine in which zones ambulances should be placed. This model is both scalable and efficient, and can be applied to every county, regardless of size or population distribution.

We started by determining the number of people who could be reached by an ambulance if it were placed in a certain zone. Despite the minimal information provided, we managed to extrapolate the reach of an ambulance as a function of the county's population distribution. This provides a realistic model of the situation by accounting for the size of each zone and the fact that an ambulance may only reach a portion of the people in a zone.

Using this data, we developed a greedy heuristic that weighed population against travel times to determine a hierarchical order for the placement of ambulances. Each zone was ranked based on how much additional coverage an ambulance placed in it would provide, and using this information, we allocated the ambulances into said zones. We then applied our algorithm and the population distribution data for a specific county to test the efficacy of the model. With only three ambulances, we managed to cover 99.4% of the county's population of 270,000 within the given response time—an optimal solution. Using our model, we can decrease the ambulance response time for people across the United States.

Emergency Service Location Optimization in a Multi-Zone County based on Population Distribution

Team 4552

November 15, 2013

Contents

1 Problem Restatement

Given a county made up of six separate zones and average travel times between zones, create a model to optimize the placement of n ambulances in order to maximize the number of civilians reached in an eight minute period. Consider the scenarios where $n = 3$, $n = 2$, and $n = 1$, respectively, and for each scenario, keep track of how many people are not being reached with each possible solution.

Then, consider a scenario in which a large scale disaster affects a single location (e.g. the September 11th attacks), and discuss how an Emergency Service Coordinator would cover the situation. Examine how a real-world city or county would prepare for such a disaster. Finally, write a two page memo detailing the model and its analysis for the Emergency Service Coordinator.

2 Assumptions and Justifications

1. **Zones as Regions**
   Each zone is not a point, but rather a region with a finite area. This means that the population of each zone is not concentrated on one point, but rather is spread out across its area.

2. **Evenly Distributed Populations within Zones**
   We assume, for lack of better data, that population is evenly distributed in each zone.

3. **Zone to Zone Travel**
   The travel times given are considered to be optimal routes from the center of one zone to the center of the another, regardless of any other zones that may lie in between. The time to travel from one zone to another is not necessarily equal to the reverse.

4. **Travel Inside Each Zone**
   The travel times given for travel from any zone to itself is the average of all the travel times of the different routes within that zone.

5. Directional Travel Inside Each Zone

The ratio of the time it takes to travel inside zone A toward a second zone B to the time it takes to travel inside zone A away from zone B, is equal to the ratio of travel time from A to B compared to the travel time from B to A. For example, traffic congestion from A to B slows down all movement in that direction, even inside A.

6. Ambulance Placement

Ambulances can only be placed in the very center of a zone. Because we do not know the spacial position of the zones relative to each other, we cannot consider the ambulances being closer to one zone or another.

7. Partial Coverage

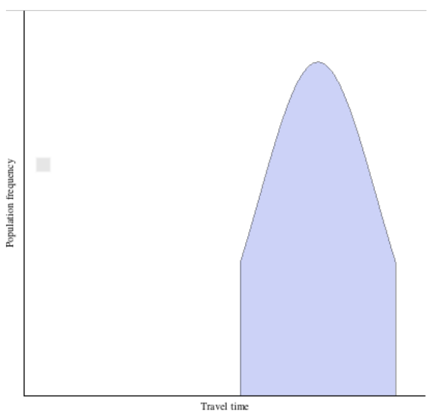
Partial ambulance coverage of a zone will be treated as incomplete even coverage over the entire zone. Because ambulances of ideal placement are unlikely to overlap in partial coverage, we assume that multiple ambulances will cover different areas of the zones, thus evening out coverage. (If partial coverages from multiple ambulances sum to above the total population of a zone, we assume that zone is completely covered by the ambulances.)

3 The Model

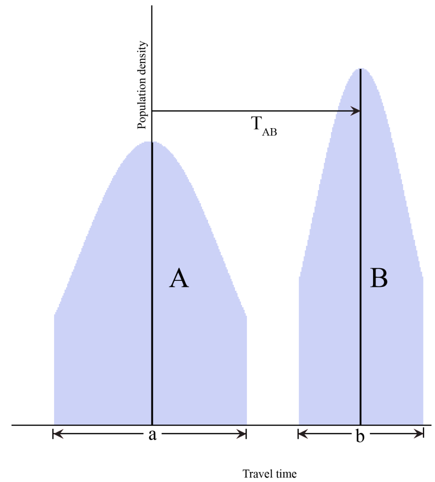
3.1 Model Approach

Given that a county has n zones, k ambulances need to be placed to maximize the number of people who can be reached in eight minutes or fewer. This system can be represented as a graph of n nodes and n^2 vertices. By analyzing all possible configurations, we can identify the ideal nodes to place the ambulances. This algorithm can be made more efficient—albeit less optimal—by employing a greedy heuristic that only focuses on the nodes that provide the most added coverage.

Each zone is a region, and therefore the population is not concentrated on one point. As a result, ambulances will often be able to cover only a fraction of a zone's total population. In order to find these fractions, we developed a model of population density for each zone by extrapolating from the provided average travel times between zones.



(a) An example of a zone's population density as a function of travel time from another zone.



(b) The width of A and B are shown. T_{AB} is the travel time between the centers (i.e. the provided average).

Figure 1: Examples of population density as a function of travel time

Travel time from a zone A to another zone B is not necessarily the same as the travel time in the opposite direction. This inconsistency could be due to a variety of factors, including differences in traffic, roads, and conditions. As a result, these two cases for each pair of zones must be handled separately.

3.2 Zone-to-Zone Population Dynamics

The population that an ambulance can cover in a second zone, B, from a first zone, A, will be referred to as zone-to-zone coverage. Zone-to-zone coverage also applies if A and B are the same zone, and represents the coverage of an ambulance in the zone it is stationed in.

A lack of spatial understanding of the county entails the use of a different type of population density. Rather than expressing population as a function of area or distance, we chose to express population density in terms of travel time from a reference point. This is analogous to plotting the number of people a vehicle is likely to meet at each time as it drives along a one-dimensional path. See Figure 1a.

Because we have more than four different zones, it is not possible for every zone to be adjacent with all the other zones. Therefore, the ambulance may travel through another zone C in an optimal route between zones A and B. However, we are only calculating the zone-to-zone population coverage of B from A, we can consider any time spent in zone C as a gap (i.e. containing a population density of 0). The zone-to-zone coverage of C from A is already accounted for in a separate calculation of A to C.

The following calculations are used to find the coverage function for any pair of zones. Separate calculations must be done for every pair, as we as for both directions for each pair. We will be using the following variables, equations, and constants:

$\mathbf{Z_{AB}(t)}$
: The zone-to-zone coverage function of B from A; to be solved

$\mathbf{B(t)}$
: Probability density function of zone B with respect to travel time (from the center of A)

$\mathbf{a}, \mathbf{b}$
: Width of A and B respectively

$\mathbf{v}$
: Required ambulance response time; given (8 minutes)

$\mathbf{T_{AB}}, \mathbf{T_{BA}}$
: Travel time from A to B and travel time from B to A; given

$\mathbf{P_B}$
: Total population of B; given

$\mathbf{T_B}$
: Average travel time inside of B; given

$\mathbf{T_{B+}}, \mathbf{T_{B-}}$
: Average travel time inside of B in the direction of A to B
Average travel time inside of B in the direction of B to A

$\mathbf{F_B(s)}$
: Frequency distribution of travel times from one random point in zone B to another point in B

The function for zone-to-zone coverage in respect to travel time can be written as the product of the population of B with the percent of the population that is covered from A in less than v minutes. The percent is the area under the probability curve, $B(t)$. As shown in Figure 1b, the bounds begin at $T_{AB} - \frac{b}{2}$ and end at v. If v is less than $T_{AB} - \frac{b}{2}$, the coverage of B is 0.

$$Z_{AB}(t) = P_B \int_{T_{AB}-\frac{b}{2}}^{v} B(t)dt$$

$B(t)$ is a population density function with a zone-specific constant b, which is the width of zone B. Every zone has a different b and, as a result, a different $B(t)$. In order to find this specific $B(t)$, we must obtain b. Until then, we cannot directly evaluate $B(t)$.

Using the average travel time within B that is provided to us, we can calculate the value for b. However, since we are provided with a general average travel time within each zone—which does not account for the direction of travel—we must extrapolate both the average travel time within B towards A and within B away from A.

The average travel time in the direction of A to B inside of B, T_{B+}, and the average travel time in the direction of B to A inside of B, T_{B-}, have a total average travel time of T_B. We can also write the ratio of T_{B+} and T_{B-}, which is equivalent to the ratio of T_{AB} and T_{BA}.

$$T_B = \frac{T_{B+} + T_{B-}}{2}$$

$$\frac{T_{B+}}{T_{B-}} = \frac{T_{AB}}{T_{BA}}$$

This allows us to solve for T_{B+}.

$$T_{B+} = \frac{2T_B}{1 + \frac{T_{AB}}{T_{BA}}}$$

Now that we have the average travel time within B away from A, we can set that equal to the definition of the average travel time within B away from A, as shown below, and solve for the b originally sought. The definition is the area under the distribution of all the possible travel times from one random point in B to another point in B.

$$T_{B+} = \frac{2T_B}{1 + \frac{T_{AB}}{T_{BA}}} = \int_0^b sF_B(s)ds$$

To find this distribution of travel times, we take the cross correlation of $B(t)$ on itself, with s being travel time plotted against frequency.

$$F_B(s) = \int_{T_{AB}-\frac{b}{2}}^{T_{AB}+\frac{b}{2}} B(t)B(t-s)dt$$

Combining the previous two equations, we can finally solve for b in terms of known values. b is a zone-specific constant embedded within the general equation $B(t)$. Depending on the type of equation that $B(t)$ is, b can be solved for in different ways.

$$\frac{2T_B}{1 + \frac{T_{AB}}{T_{BA}}} = \int_0^b \int_{T_{AB}-\frac{b}{2}}^{T_{AB}+\frac{b}{2}} sB(t)B(t-s)dtds$$

With b solved and $B(t)$ fully defined, we can finally evaluate the original integral for zone-to-zone coverage.

$$Z_{AB}(t) = P_B \int_{T_{AB}-\frac{b}{2}}^{v} B(t)dt$$

Although our model is robust and supports any symmetrical function for $B(t)$, we assume for lack of better data that population is evenly distributed in each zone (i.e. $B(t)$ is a constant). This simplifies the process of finding the optimal ambulance locations.

3.3 Ambulance Distribution Optimization

Once the zone-to-zone coverages have been calculated, we must still decide where to place the ambulances to maximize total coverage. Total coverage is defined as the sum of the coverages of the individual zones (i.e. the number of people that can be reached by at least one ambulance within eight minutes).

Since ambulance placements are assumed to be in the center of zones, putting two ambulances in the same zone does not increase coverage. Therefore, ambulances should be placed in different zones to maximize coverage.

Under the given conditions, we can easily find the optimal ambulance placement using a brute force method: trying all $\binom{6}{3}$ choices of ambulance placements, and picking the one with the highest total coverage. However, this solution is not scalable to larger cities, as the brute force method has time complexity $O(n, k) = nk\binom{k}{n}$ where k is the number of zones to cover, and n is the number of ambulances to place, which becomes intractable as n and k increase (since it has to check $\binom{k}{n}$ placements, and calculating the total coverage of each placement requires nk computations, summing the k zone-zone coverages for each of the n ambulances). Quadrupling the size of the city increases the computations required by a factor of 2 million (assuming the average zone size stays the same, and the number of ambulances increases proportionally).

Therefore, a more scalable solution is needed. We discovered that a greedy heuristic also gives a good solution, except with a much lower time complexity of $O(n, k) = nk^2$. The greedy heuristic works by assigning ambulances to zones one at a time. For each assignment, we iterate through each zone without an ambulance already assigned, and see how many additional people it would cover; then we assign an ambulance to the zone which would most increase the number of covered people.

4 Model Analysis

4.1 The n Ambulance Problem

With the n ambulance problem, we are trying to figure out where to place each of the n ambulances so that the reach of each ambulance covers the most amount of people. For our purposes, this reach is solely determined by how far the ambulance can travel in an eight minute span.

With the given county, made up of six zones, we consider the location optimization under three scenarios; with three ambulances, with two ambulances, and finally, with one ambulance. Our goal is to see a) if we can cover everybody in the county in an eight minute span, and b) if not, how many people are we failing to cover.

The number of people the ambulances are able to cover is a function of our population density function, while the actual location optimization is done by our greedy optimizer.

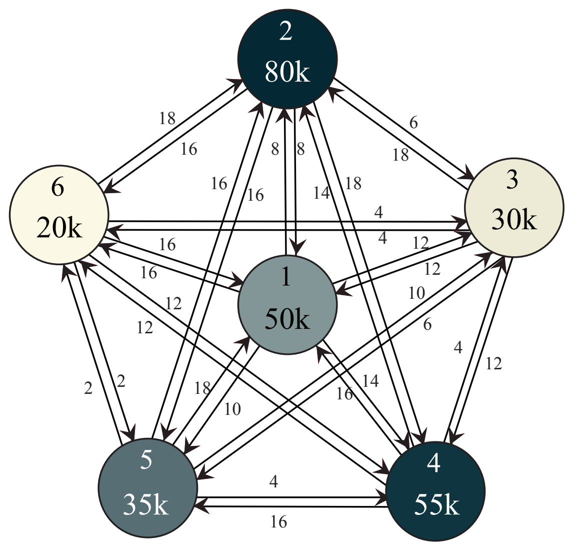


Figure 2: Graph of zones, populations, and travel times

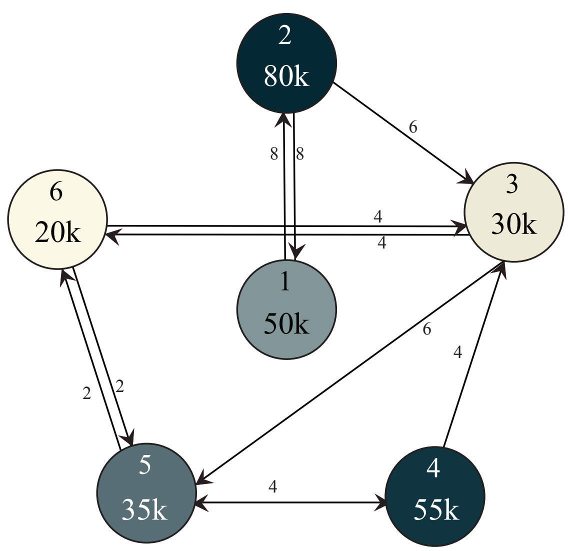


Figure 3: Graph in fig. 2, with edges longer than 8 minutes removed

4.1.1 $n = 3$

Figure 2 above shows every possible route between zones, with their respective weight, or in this case, the time necessary to travel from one zone to another. For example, traveling from Zone 1 to Zone 6 would take 16 minutes, and vice versa.

However, if you look at each of the routes, you see that some paths take longer than 8 minutes to travel. In this case, it would be impossible for an ambulance to make it from one zone to another in the necessary 8 minute time interval, and therefore, it would be impossible to save anyone along that route.

In order to combat this, we look at Figure 3, above, which has significantly fewer paths. By eliminating the paths that require a travel time of more than 8 minutes, we have effectively cut our search time significantly, and allows to intuitively see some features of the graph.

First and foremost, if you look at Zone 2, you see two paths going out to Zones 1 and 3, respectively. Because of the short travel time it takes to get to both of the zones, the reach of an ambulance coming out of Zone 2 is slightly larger, and you are more likely to save more people. Furthermore, the overall population density of the three zones (1+2+3) is large, weighing Zone 2 as one of the more important nodes on the graph. Therefore, intuitively, Zone 2 makes sense as a starting point.

In the same way, if you look at Zone 5, you see two other paths going outwards to Zone 4 and Zone 6, respectively. Because neither of these two zones is represented with placing an ambulance in Zone 2, it follows that another ambulance should be placed in Zone 5, therefore covering the bulk of the county.

Finally, if you look at Zone 1, you see that while though redundant, the additional population covered by having an ambulance stationed there is the next greatest. For that reason, it makes the most sense to place the third ambulance in Zone 1.

Intuitively then, just by looking at the graphs, it makes the most sense that Zones 2, 5, and 1 are the most likely in which to place the ambulances in order to reach the most people in 8 minutes. Therefore, it shouldn't be surprising that when we ran the greedy algorithm for $n = 3$ we got that ambulances should be placed in zones 2, 5, and 1. When we matched this up with our population coverage math, we found that this arrangement has ambulances cover all of zones 1-2, 4-6 and most (28333/30000) of zone 3, accounting for a majority of the county.

4.1.2 $n = 2$

When we ran the greedy algorithm for $n = 2$ we got that ambulances should be placed in zones 2 and 5, which also intuitively follows from the graph. This arrangement has ambulances cover all of zones 2, 4-6, half (25000/50000) of zone 1, and most (28333/30000) of zone 3.

4.1.3 $n = 1$

Finally, when we ran the greedy algorithm for $n = 1$ we found that our ambulance should be placed in zone 2. This arrangement allows the single ambulance to cover zone 2 and parts of 1 and 3, which represents the highest population on the map.

For each of these scenarios, our greedy heuristic got the exact same answer as the brute-force algorithm as well as the intuitive solution from the graph. Because our greedy heuristic found the same answer as the brute-force algorithm (which looked for the ideal, optimal answer) our greedy heuristic seems to be performing efficiently and accurately.

4.2 Localized Single Zone Disaster

In a localized catastrophic event such as those that happened on September 11, 2001, there will be many casualties in a single location that require aid. Ambulances, and other emergency vehicles, such as police cars and fire trucks, will need to respond quickly in order to aid the most critically injured first, who will be sorted by triage [1]. Therefore, it is very important for ambulances to be able to reach the emergency as quickly as possible, regardless of the initial location of the ambulances and the location of the emergency.

Catastrophic events will cause more casualties than emergency personnel will be able to service, so as a result, ambulances will have to act as shuttles to hospitals for the injured. Therefore, to maximize throughput cities should arrange hospitals to minimize the average time between all areas and the nearest hospital.

Since metropolitan areas are most likely to be affected by catastrophic events, and catastrophic events there will affect significantly more people, to maximize patient throughput cities should arrange hospitals to minimize the average time between all areas and the nearest hospital. Futhermore, due to differences in population density, statistically, an accident in a metropolitan area will demand more urgent care than a similar accident in a rural or suburban area. Therefore, when the city lays out its stations for ambulances, attempting to minimize travel times to any area, it should put significantly more weight on minimizing travel time to metropolitan areas.

5 Strengths and Weaknesses

Strengths	**Weaknesses**
1. Our model, by virtue of its polynomial-time greedy selection algorithm, is extremely scalable. We can scale up to large datasets, consisting of many, larger zones. Also, because of a non-exponential time algorithm, our model is also extremely efficient.	1. Because of the greedy nature of our algorithm, we are not guaranteed to find the optimal solution for the location of the ambulances.
2. Our model is also adaptable to various different population distributions. Population Distributions can vary with the addition of any new data. The more accurate the data, the more accurate the model becomes.	2. In order to find the optimal solution, our brute force optimizer works in exponential time, so at a large scale, finding the optimal placement of ambulances is near impossible.
3. Completely flexible model. Depending on any number of nodes, any number of ambulances, and any population distribution, we will be able to find a good answer relatively quickly.	

Table 1: Model Strengths & Weaknesses

6 Extensions

Though our model performs fairly well already, providing us with the optimal locations of our n ambulances to maximize the number of people saved, there are a number of ways we can improve the model so that we can make more accurate predictions as to the number of people we can reach in an eight minute period. Furthermore, we can also improve the overall efficiency of the solution algorithm itself, by looking at potential dynamic programming approaches to the problem.

While any algorithmic changes may not be necessary with only six zones, and a maximum of $n = 3$ ambulances, if we scale this problem up to a large city, or across multiple counties, algorithm efficiency becomes an important problem that requires attention.

6.1 Population Distribution

As it currently stands, we have very little information regarding the actual population distribution of each zone in the scope of the county, which provides us with a slightly skewed perception. While we justified the use of a standard distribution for population above, we have the ability to greatly improve our model with a better idea of how spread out a population of a zone actually is, in relation to its epicenter. This information allows us to better predict how many people an ambulance would miss (not be able to cover) in a certain zone.

The number of missed people is an important statistic that an Emergency Service Coordinator needs to have in order to fully optimize their resources. In our system now, we predict this number with a certain level of accuracy, but we are still prone to a relatively large amount of error. With a better understanding of the population and geospatial makeup of each individual zone, we would be able to better optimize our resources and potentially save more people.

6.2 Dynamic Approach

With only a total of six zones and a maximum of $n = 3$ ambulances to allocate, overall algorithm efficiency is not a pressing issue. As we have shown above, even a brute force attempt at optimizing the ambulance locations takes a negligible amount of time. However, as we look to expanding the model to fit a larger area, with more zones, and more ambulances, it becomes increasingly more important to look for ways to make our algorithm more efficient.

Even after our greedy algorithm reduces the overall solution time to an $O(n, k) = nk^2$, or polynomial time, there are still possible ways to improve it. One such method would be the use of dynamic programming methods to split the problem up into a series of smaller subproblems, and then merge the results together. As a result of such a method, we would reduce the problem to one with an $O(n, k) = nk \log(k)$, an algorithm that would easily scale to larger data sets.

However, the implementation of such a method is slightly less simple. One such way to implement a solution in this manner would be in the following fashion:

1. Given a graph of each node (zone), and the edges between them (time it takes to travel between nodes), immediately remove all infeasible edges, with edge length (time) greater than the eight minute baseline.

2. Identify a set of nodes such that there is a large number of outgoing edges, thereby identifying potential split points, to divide each county into smaller sub-counties. Make sure to account for overlap.

3. Allocate n ambulances across the sub-counties. If there are fewer ambulances than are necessary to cover the entire county, weight each sub-county by population size (save more people per ambulance).

4. Solve each sub-county for the optimal ambulance location, thereby finding a solution for the set as a whole.

While we are not guaranteed to find the optimal solution to ambulance location, we will be provided with a relatively good one, in a short amount of time. There are tradeoffs to using such a model, but especially at a large scale, this methodology is realistically the one an Emergency Service Coordinator would use.

7 Non-Technical Memo

To whom it may concern,

After looking at several scenarios, and forming a strategic model, we believe that we have some suggestions for you regarding how to optimize ambulance placement across zones in your county.

To us, it seems that the problem is as follows: A county with a six zones needs to maximize the number of people that can be reached within eight minutes using its three ambulances. Given the average amount of time necessary to travel between each of the zones, we must determine the optimal placement of the ambulances to meet the three conditions.

In order to solve this problem, we first needed to develop a firm understanding of how many people a single ambulance can reach within eight minutes. To get an accurate representation of how many people an ambulance can actually reach, we took into account the fact that a zone has an area and that instead of being concentrated at one specific point, the population of a zone is spread out over its entire area. For our purposes, we assumed that the population density distribution was linear, but without some additional data, it is impossible for us to draw any significant conclusions about what it should be. However, we designed our solution to be flexible enough to implement any symmetrical population distribution, thereby ensuring that our model will work regardless of the data provided.

Because distance in this scenario is an unknown quantity (We are only given travel times in the data provided to us), we are forced to use another unit to represent the population density. Given that we want to travel from zone A to zone B, we chose to model the population density of zone B as a function of travel time from zone A. Using this function, we were able to calculate how many people we would be able to reach in B (within a timespan of eight minutes) from A. By taking the number of people covered and the given travel times between each of the zones, we were able to calculate how many people an ambulance could reach in an eight minute period from any zone A.

After computing this quantity, we still need to figure out where we should put the ambulances to cover the most people. We approach this problem in two ways: first, we use a brute-force method, enumerating all possible combinations of ambulance positions, and picking the one that covered the most people (finding us the optimal solution); second, using our brute-force algorithm as a base, we create a greedy heuristic, which finds a solution relatively quickly in comparison, especially for larger cities/more ambulances. However, while the brute-force algorithm is guaranteed to pick the optimal solution, the greedy heuristic is only guaranteed to find a good one. Yet, especially when this problem is scaled up to multiple zones or counties, the greedy heuristic proves to be superior than the brute-force algorithm, with a high level of efficacy and precision.

When applied to the county at hand, the greedy heuristic succeeded in finding a solution. Better yet, when compared to the brute-force algorithm, we found that the two methods yielded identical solutions, meaning that the greedy algorithm found the optimal solution as well. The results are as follows: For 3 ambulances across the county, we split them between zones 1, 2, and 5; this allows us to cover all of zones 1-2, 4-6 and 94% of zone 3, covering 99.4% of the county population-wise in eight minutes. If one or two ambulances are busy, the other two ambulances should be relocated to zones 5 and 2; ambulances in zones 5 and 2 will cover 90.1% of the county, and a single ambulance in zone 2 will cover 49.4% of the county. These results indicate that a majority of the county will be covered at any given time, and that all citizens can be reached in under 8 minutes.

With these suggestions in mind, I hope you take the necessary steps to implement our solution in this county, and perhaps in others as well.

A Code

The following Python script (optimizer.py) contains the code for both our brute force location optimizer, as well as the greedy optimizer. Both functions take a parameter *nambulances*, representing the total number of ambulances we have to allocate across the county.

```
import numpy as np
import sympy as sp
from sympy import Function, integrate
import itertools
from data import traveltimes, populations

NODES = 6  # number of nodes to care about
MAX_TIME = 8  # maximum allotted time to cover a destination
zz_coverage = np.zeros((NODES, NODES))

a, x, s, t = sp.symbols('a, x, s, t')


# Probability/population distribution curves.
class quad_curve(Function):
    nargs = 2

    @classmethod
    def eval(cls, a, x):
        return -(12.0 / 11 / a) * x ** 2 / a ** 2 + (12.0 / 11 / a)


class const_curve(Function):
    nargs = 2

    @classmethod
    def eval(cls, a, x):
        return 1 / a

PROB_CURVE = const_curve  # selected model for population distribution

expected_value = integrate(integrate(PROB_CURVE(a, t) * PROB_CURVE(a, t - s),
                                     (t, -a / 2, a / 2)) * s, (s, 0, a))
# integral that should equal t_bbp


def calc_cover(start, target):
    '''
```

```
    Calculates coverage of a single ambulance for a zone.
    '''
    t_bb = traveltimes[target, target]
    t_ab = traveltimes[start, target]
    t_ba = traveltimes[target, start]
    t_bbp = 2 * t_bb / (1 + t_ab / t_ba)
    b = sp.solve(expected_value - t_bbp)[0]
    b_start = -b / 2
    b_end = min(b / 2, MAX_TIME - t_ab)
    if b_start > b_end:
        return 0
    return int(integrate(PROB_CURVE(b, t), (t, b_start, b_end)) *
               populations[target] + 0.5)


for i in xrange(NODES):
    for j in xrange(NODES):
        zz_coverage[i, j] = calc_cover(i, j)


def greedyoptimize(nambulances):
    '''
    Optimize ambulance placement using a greedy algorithm.
    Selects additional ambulances locations based on how
    many additional people it would cover.
    '''
    covered = np.zeros(NODES)
    startnodes = []
    for i in xrange(nambulances):
        maxcoverage = 0
        bestnode = 0
        for start in xrange(NODES):
            if start in startnodes:
                continue  # don't duplicate zones
            coverage = 0
            for target in xrange(NODES):
                # additional coverage will increase the coverage to,
                # at most the population of the zone
                coverage += min(zz_coverage[start, target],
                                populations[target] - covered[target])
            if coverage > maxcoverage:
                maxcoverage = coverage
                bestnode = start
        startnodes.append(bestnode)
        for target in xrange(NODES):
```

```
            covered[target] += min(zz_coverage[bestnode, target],
                                   populations[target] - covered[target])
    return ([i + 1 for i in startnodes], covered)  # change 0 to 1-indexing


def bruteforce(nambulances):
    '''
    Takes all possible combinations of nambulances from NODES,
    and arranges them so that there is maximum coverage among zones.
    '''
    maxcoverage = np.zeros(NODES)
    for startingnodes in itertools.combinations(xrange(NODES),
                                                nambulances):
        covered = np.zeros(NODES)
        for target in xrange(NODES):
            for start in startingnodes:
                covered[target] += min(zz_coverage[start, target],
                                       populations[target] - covered[target])
        if np.sum(covered) > np.sum(maxcoverage):
            optimal = startingnodes
            maxcoverage = covered
    return ([i + 1 for i in optimal], maxcoverage)  # change 0 to 1-indexing
```

The following file (data.py) contains the data we used for running our optimizer, taken from the provided problem constraints.

```
import numpy as np

'''
avg. travel times from sector to sector.
traveltimes[i, j] is travel time from sector i to sector j.
0-indexed, so watch out!
'''
traveltimes = np.array([[1, 8, 12, 14, 10, 16],
                        [8, 1, 6, 18, 16, 16],
                        [12, 18, 1.5, 12, 6, 4],
                        [16, 14, 4, 1, 16, 12],
                        [18, 16, 10, 4, 2, 2],
                        [16, 18, 4, 12, 2, 2]])

'''
Populations of each sector.
0-indexed, watch out!
'''
populations = np.array((5e4, 8e4, 3e4, 5.5e4, 3.5e4, 2e4))
```

References

[1] Jeffrey B. Goldberg. Operations Research Models for the Deployment of Emergency Services Vehicles. *EMS Management Journal*, March 2004.