International Journal of Science, Technology and Society
2022; 10(1): 1-7
http://www.sciencepublishinggroup.com/j/ijsts
doi: 10.11648/j.ijsts.20221001.11
ISSN: 2330-7412 (Print); ISSN: 2330-7420 (Online)

Evacuating Orange County, California, (Part 2) — The Approximate Versus the Exact

James Riechel

Center for Information Systems & Technology (CISAT), Claremont Graduate University (CGU), Claremont, USA

Email address:
james.riechel@cgu.edu

To cite this article:
James Riechel. Evacuating Orange County, California, (Part 2) — The Approximate Versus the Exact. *International Journal of Science, Technology and Society.* Vol. 10, No. 1, 2022, pp. 1-7. doi: 10.11648/j.ijsts.20221001.11

Received: October 18, 2021; **Accepted**: January 10, 2022; **Published**: January 15, 2022

Abstract: This paper completes the science on the first paper also on the Orange County, California, evacuation problem. Here, the execution time of the exact solution is correctly reported—it is not 53+ days as originally reported, but is still over three (3) hours, 900+ times slower than the approximate solution. Comparing the Load Balancing Metric of both the approximate and exact solutions, it is clear that both produce similar results, supporting the use of the approximate solution as it takes mere seconds to complete. The Orange County, California, dataset contains 1.1 to 1.2 million addresses, both residential and business. On a map, a random 100 routes in Orange County are shown, connecting addresses (incidents) to the closest of four (4) waypoints (facilities) with respect to drive time without consideration of traffic conditions. In the Appendix, a Python toolkit for ArcGIS Pro is given that computes the approximate solution. This did not appear in the first paper.

Keywords: Vehicle Evacuation Planning By Waypoints, Nuclear Threats, Approximate Versus Exact, Load Balancing, Drive-Time Network, Traffic Conditions

1. Introduction

This section, sections 2 and 3, section 4.1, and section 6 follow Riechel (2021) closely [8].

First, you need to know what population (s) to evacuate. In this work, it is assumed that the entire (business and residential) County of Orange, California, needs to be evacuated. Yet this may not be the case. In [12], it is determined that "[b]ased on the performed analysis, we suggest avoiding evacuation if the projected first-year dose is below 500 mSv" [12]. One (1) millisievert (mSv) is the average amount of background radiation an individual will absorb in one (1) year. They base their analysis on data from the Fukushima Daiichi nuclear disaster in Japan.

"After the accidents of nuclear power plants at Chernobyl and at Fukushima, huge amounts of radioactive iodine were released into the atmosphere" [4]. Of the two million children who lived close to Chernobyl, 7,000 cases of thyroid cancer were diagnosed in 2005. Further study is required to determine if these numbers are significant, as thyroid cancer is a very common disease. Yoshimura et al. (2020) report that there were significant differences between the types and amounts of radiation exposure at Chernobyl and Fukushima [13].

In Thompson et al. (2017), it is determined that people having an evacuation plan in hand are more likely to follow evacuation instructions: "Risk perception was a consistent positive predictor of evacuation, as were several demographic indicators, prior evacuation behavior, and having an evacuation plan" [11].

In Alabdouli (2017), the case of an Orange County, California, tsunami is considered [1]. If the case of an OC tsunami were to be considered here instead of a San Onofre nuclear power plant crisis, the evacuation instructions would change to: head east to any higher ground!

2. Materials and Methods

Each line of the input file has a longitude, latitude pair (in degrees) of an address. First, convert these from degrees to radians. Then compute the *x, y, z* coordinates of each address:

$$x = \text{longitude} * r * \text{Cos}(\phi_0)$$

$$y = \text{latitude} * r$$

$$z = \text{elevation}$$

where r is the radius of Earth, and ϕ_0 is a centrally located latitude in the dataset. This forms an equirectangular projection [2].

Say (x_i, y_i, z_i) is the coordinates of an address, and (x_j, y_j, z_j) is the coordinates of a waypoint. The approximate driving distance between them is:

$$\text{distance}(i, j) = \text{Abs}(x_i - x_j) + \text{Abs}(y_i - y_j) + \text{Abs}(z_i - z_j)$$

This approximate distance, the Manhattan distance [10], is both a better approximation of the actual driving distance than the Euclidean distance and an order of magnitude faster to compute than it [5-9]. See Zeager and Stitz (2016) for a description of Euclidean distance [14].

The Manhattan distance is extremely fast to compute:

Table 1. *Execution speed of Manhattan distance, aka "approximate driving distance".*

	Running time of 100 million calls (milliseconds)	Operations per second
Manhattan distance, aka "approximate driving distance"	296	337,837,838

A pilot study was performed to determine how accurate Manhattan distances are to actual driving distances. In this nonrepresentative study of 200 green taxi cabs rides in New York City on January 1, 2016, starting at 12 AM EST, the following distribution was produced:

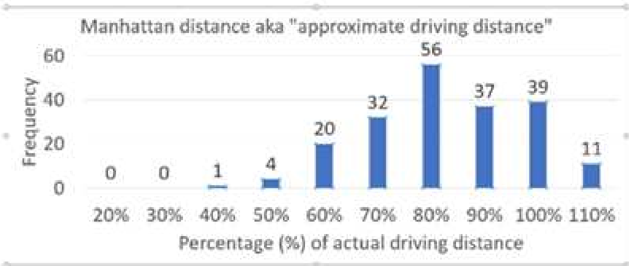

Figure 1. *Accuracy of Manhattan distance in NYC pilot study.*

This distribution is confirmed as normal. On average, the Manhattan distance is about 80% of the actual driving distance, and the Manhattan sometimes overestimates actual driving distance.

There are many reasons why Manhattan distance might underestimate actual driving distance, including:

Traffic controls (U-turns, one-way streets)

Manmade obstacles (bridges)

Natural obstacles (lakes, hills, mountains)

Etc.

Also, the equirectangular projection tends to underestimate distance.

The approximate driving distance algorithm (Riechel, n.d.-a, n.d.-b, 2020, 2021, 2019) makes this evacuation planning possible. Using actual driving distances would take too long and be too expensive. The waypoint suggestions are just that: suggestions. In some cases, the approximate driving distance algorithm might not choose the closest waypoint.

3. Case Study

In the following map, the San Onofre nuclear power plant is labeled "R" for "reactor." The four waypoints out of Orange County, California, are labeled "W1," "W2," "W3," and "W4." The four waypoints exit Orange County on the 405, 5, 57, and 91 freeways, respectively. Once a waypoint is reached, evacuees can travel in any direction, except back toward Orange County.

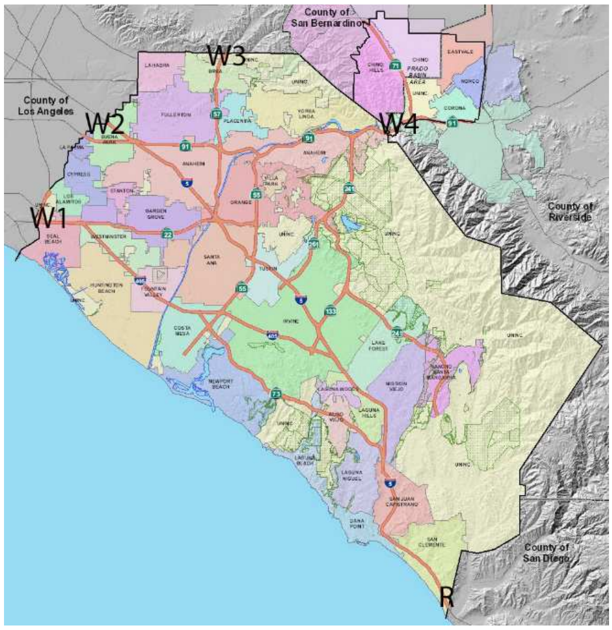

Figure 2. *The reactor and the four waypoints: map courtesy of Orange County, California (https://www.ocgis.com/ocpw/landrecords/).*

The following table gives more detailed information on the waypoints:

Table 2. *The four waypoints.*

Waypoint	Description	Longitude (degrees)	Latitude (degrees)
W1	405 on LA/Orange County border	-118.0931346	33.7859273
W2	5 on the LA/Orange County border	-118.0114088	33.8748110
W3	57 on the LA/Orange County border	-117.8683537	33.946003
W4	91 on Riverside/Orange County border	-117.6717187	33.8695321

Each address is sent a color map of Orange County (much like Figure 2 above), showing the nuclear reactor R and all four waypoints: W1, W2, W3, and W4.

For each address in Orange County (residential or business), the approximate driving distance algorithm is used to find the nearest waypoint. Along with the map, a suggested or recommended waypoint is given.

Residents do not have to follow the recommended waypoint. Instead, they can choose any of W1, W2, W3, or W4 in the event of a crisis at R.

4. Results

4.1. The Approximate Solution

The following map shows the results of evacuating County of Orange, California, to the four waypoints:

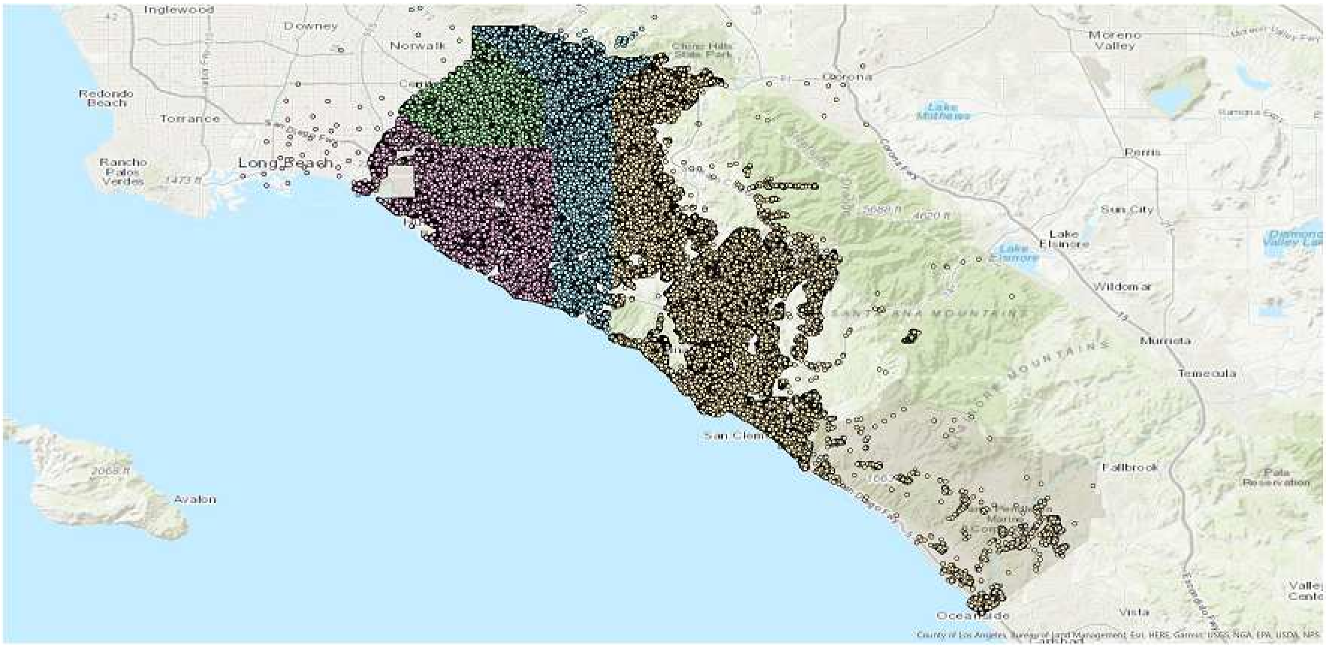

Figure 3. *The approximate results.*

The addresses are color-coded to specify suggested evacuation waypoints:

Table 3. *Explanation of color codes in Figure 3.*

Color	Evacuation waypoint
Purple	Evacuate by 405 freeway
Green	Evacuate by 5 freeway
Blue	Evacuate by 57 freeway
Brown	Evacuate by 91 freeway

The following table presents the number of addresses for each color code, and the total number of addresses:

Table 4. *The number of addresses routed to each waypoint, and the total number of addresses.*

Waypoint	Count (addresses)
W1	306,249
W2	129,219
W3	266,898
W4	445,429
TOTAL	1,147,795

4.2. The Exact Solution

The following map (Figure 4) shows 100 out of the 1,147,795 incidents from the master datafile. Incidents appear as yellow squares, and the route to the nearest of four facilities (waypoints) are shown. The selection of waypoint is based on drive time without consideration of traffic conditions.

4.3. Comparison of Approximate and Exact

Two simulations were performed to compare the approximate and exact solutions to the Orange County evacuation problem. Of interest were two metrics of performance: the time to complete each simulation (most important), and the accuracy of the approximate solution in comparison to the exact (of secondary importance).

Table 5. *Execution times for the approximate and exact solutions.*

Execution Times

Algorithm	Time to complete
Approximate	13.2 seconds
Exact	3 hours, 18 minutes, and 41 seconds

Obviously, the approximate solution is incredibly faster: it is over 900 times faster than the exact solution.

By "exact" the driving time on a network of Orange County without consideration of traffic conditions is meant.

Figure 4. *Incidents (yellow sqaures), facilities/waypoints (colored circles), and the routes that connect them.*

Table 6. *Comparison of the approximate and exact for the entire dataset.*

Accuracy

Waypoint	Using the approximate driving distance algorithm (no network)	Using drive time without traffic (using a network)	Comparison (approximate / exact)
W1	26.682%	30.757%	86.751%
W2	11.258%	22.011%	51.147%
W3	23.253%	8.975%	259.086%
W4	38.807%	38.257%	101.438%
Load Balancing Metric	19.630511	21.785909	

A comparison metric of 85% to 100% would have been ideal. This was obtained for W1 (86.751%), and almost obtained for W4 (101.438%).

W2 (51.147%) and W3 (259.086%) were well outside the ideal, but there is an explanation: The usual route to get to the 57 freeway in OC is off the I-5 freeway in the City of Santa Ana. But if already on the I-5, it is a much shorter trip to stay on the I-5 to the Los Angeles and Orange County border, rather than switch to the 57, where the county line is farther.

A Load Balancing Metric was computed for the approximate (19.630511) and exact (21.785909) solutions. A metric closer to zero (0) indicates better load balancing (each of the waypoints are used roughly the same). A load balancing metric of zero (0) would indicate perfect load balancing. The approximate and exact metrics are nearly identical for both solutions, so one is not clearly better than the other.

5. Limitations and Discussion

The presented software for the approximate solution routed 1.1 to 1.2 million addresses in OC to the closest of four (4) waypoints in about 13.2 seconds, because the approximate driving distance algorithm was used on a state-of-the-art laptop.

The exact solution using the Network Analysis tool in ArcGIS Pro took over 3 hours.

The results for both were about the same.

Li et al. (2016) introduce the “six Vs” of Geospatial Big Data: volume, variety, velocity, veracity, visualization, and visibility [3]. “In a world filled with Big Data, where the volume of data points to compute distances between, and the velocity at which these distances are expected to be computed, are both extremely high, a fast algorithm for computing approximate distances may be the only choice. This introduces the issue of veracity: How reliably accurate these approximate distances are” [6].

6. Conclusion and Future Work

Every evacuation scenario differs and requires custom software to solve.

The approximate driving distance algorithm might prove key to other evacuation problems, not just that of OC.

The approximate driving distance algorithm aids in the development and testing of evacuation software as it can be done quickly in real time.

For future work, the author would like to try solving more evacuation problems in the United States.

Acknowledgements

Mike Vuong, GIS Supervisor, County of Orange, California, Fire Authority.

Full Clinical Professor Brian N. Hilton and Professor Warren Roberts, Center for Information Systems &

Technology (CISAT), Claremont Graduate University (CGU).

Appendix

A1. Python toolkit for ArcGIS Pro for computing waypoints based on the approximate driving distance algorithm.

```
# -*- coding: utf-8 -*-

import arcpy


class Toolbox(object):
    def __init__(self):
        "Define the toolbox (the name of the toolbox is the name of the
        .pyt file)."
        self.label = "Evac"
        self.alias = "evac"

        # List of tool classes associated with this toolbox
        self.tools = [Evac]


class Evac(object):
    def __init__(self):
        "Define the tool (tool name is the name of the class)."
        self.label = "Tool"
        #self.label = "Evac"
        self.description = ""
        self.canRunInBackground = False

    def getParameterInfo(self):
        "Define parameter definitions"
        param0 = arcpy.Parameter(
            displayName="Number of waypoints",
            name="numway",
            datatype="GPLong",
            parameterType="Required",
            direction="Input")

        param1 = arcpy.Parameter(
            displayName="Waypoint file",
            name="waypointfile",
            datatype="DEFile",
            parameterType="Required",
            direction="Input")

        param2 = arcpy.Parameter(
            displayName="Input file",
            name="inputfile",
            datatype="DEFile",
            parameterType="Required",
            direction="Input")

        param3 = arcpy.Parameter(
            displayName="Output file",
            name="outputfile",
            datatype="DEFile",
            parameterType="Required",
            direction="Output")

        #params = None
        #return params
        params = [param0, param1, param2, param3]
        return params


    def isLicensed(self):
        "Set whether tool is licensed to execute."
        return True

    def updateParameters(self, parameters):
        "Modify the values and properties of parameters before internal
        validation is performed.  This method is called whenever a parameter
        has been changed."
        return

    def updateMessages(self, parameters):
        "Modify the messages created by internal validation for each tool
        parameter.  This method is called after internal validation."
        return

    def execute(self, parameters, messages):
        "The source code of the tool."
        print(parameters)

        import math
        #numwaystr = input("Number of waypoints = ")
        #numwayint = int(parameters[0][1])
        numwayint = int(parameters[0].valueAsText)
        print("Number of waypoints =", numwayint)
        #waypointfile = input("Waypoints file = ")
        waypointfile = parameters[1].valueAsText
        f = open(waypointfile, "r")
        waylonglat = []
        for x in range(numwayint):
            waypoint = f.readline()
            #print(waypoint)
            waypoint2 = waypoint.split(",")
            waylonglat.append([float(waypoint2[0]), float(waypoint2[1])])
        f.close()
        for x in range(numwayint):
            print(waylonglat[x])
        #inputfile = input("Input file = ")
        inputfile = parameters[2].valueAsText
        f = open(inputfile, "r")
        firstline = f.readline()  # read past longitude,latitude line
```

```
    avglat = float(0.0)
    numlines = 0
    for line in f:
        #print(line)
        longlat = line.split(",")
        #print(longlat)
        avglat = avglat + float(longlat[1])
        numlines = numlines + 1
    f.close()
    print("Number of lines =", numlines)
    avglat = avglat / float(numlines)
    print("Average latitude =", avglat)
    avglat = avglat / 180.0 * math.pi
    cosavglat = float(math.cos(avglat))
    xmult = cosavglat * 6371000.0 / 180.0 * math.pi  # Earth radius
    ymult = 6371000.0 / 180.0 * math.pi  # Earth radius
    wayxy = []
    for x in range(numwayint):
        wayxy.append([waylonglat[x][0] * xmult, waylonglat[x][1] * ymult])
    print("Waypoints in [x, y] format:")
    for x in range(numwayint):
        print(wayxy[x])
    #outputfile = input("Output file = ")
    outputfile = parameters[3].valueAsText
    fin = open(inputfile, "r")
    fout = open(outputfile, "w")
    firstline = fin.readline()  # read past longitude,latitide line
    line = "longitude,latitude,"
    for x in range(numwayint):
        line = line + "W" + str(x + 1) + ","
    line = line + "Waypoint\n"
    fout.write(line)
    for x in range(numlines):
        line = fin.readline()
        line2 = line.split(",")
        long = float(line2[0])
        lat = float(line2[1])
        long2 = long * xmult
        lat2 = lat * ymult
        dist = []
        outline = str(long) + "," + str(lat) + ","
        for y in range(numwayint):
            dist.append(abs(long2 - wayxy[y][0]) + abs(lat2 - wayxy[y][1]))
            outline = outline + str(dist[y]) + ","
        min_distance = 1000000
        min_index = -1
        for y in range(numwayint):
            if (dist[y] < min_distance):
                min_distance = dist[y]
                min_index = y
        outline = outline + str(min_index + 1) + "\n"
        fout.write(outline)
    fin.close()
    fout.close()
    return
```

References

[1] Alabdouli, K. (2017). Hypothetical Tsunami Scenarios and Evacuation Intention: A Case Study of Orange County, California. *Journal of Geography & Natural Disasters*, *7* (3). https://doi.org/10.4172/2167-0587.1000212

[2] Hargitai, H., Willner, K., & Hare, T. (2019). Fundamental Frameworks in Planetary Mapping: A Review. In H. Hargitai (Ed.), *Planetary Cartography and GIS* (pp. 75–101). Springer International Publishing. https://doi.org/10.1007/978-3-319-62849-3_4

[3] Li, S., Dragicevic, S., Castro, F. A., Sester, M., Winter, S., Coltekin, A., Pettit, C., Jiang, B., Haworth, J., Stein, A., & Cheng, T. (2016). Geospatial big data handling theory and methods: A review and research challenges. *ISPRS Journal of Photogrammetry and Remote Sensing*, *115*, 119–133. https://doi.org/10.1016/j.isprsjprs.2015.10.012

[4] Ory, C., Leboulleux, S., Salvatore, D., Le Guen, B., De Vathaire, F., Chevillard, S., & Schlumberger, M. (2021). Consequences of atmospheric contamination by radioiodine: The Chernobyl and Fukushima accidents. *Endocrine*, *71* (2), 298–309. https://doi.org/10.1007/s12020-020-02498-9

[5] Riechel, J. (n.d.-a). *Extending Manhattan Euclidean and Actual Driving Distances into 3D.pptx*. Retrieved August 27, 2021, from https://drive.google.com/file/d/16rkw9Ysn8BWfT7CpfvlnlaSwUgBw4Y4v/view?usp=sharing&usp=embed_facebook

[6] Riechel, J. (n.d.-b). Extremely Fast “Solution” to the Large-Scale and Very Large-Scale Vehicle Routing Problem. *Google Docs*. Retrieved August 27, 2021, from https://drive.google.com/file/d/12BWEcFeLOQZ0_nFAV6qtMrs9fKFoC_S4/view?usp=sharing

[7] Riechel, J. (2020). *Comparing Manhattan, Euclidean, and Actual Driving Distances.pptx*. CalGIS 2020, Long Beach, CA. https://drive.google.com/file/d/1_wgjePJH6LXM6OAYHg-PJkr2o-wVr8AK/view?usp=sharing&usp=embed_facebook

[8] Riechel, J. (2021). EVACUATING ORANGE COUNTY, CALIFORNIA, IN ABOUT ELEVEN (11) SECONDS. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLIV-M-3–2021*, 143–147. https://doi.org/10.5194/isprs-archives-XLIV-M-3-2021-143-2021

[9] Riechel, J. A. (2019). A fast algorithm for computing approximate distances in the Cartesian plane. *URISA GIS-Pro 2019, New Orleans, LA*. https://urisa.library.esri.com/cgi-bin/koha/opac-detail.pl?biblionumber=183148&query_desc=kw%2Cwrdl%3A%20riechel

[10] Singh, A., Yadav, A., & Rana, A. (2013). K-means with Three different Distance Metrics. *International Journal of Computer Applications*, *67* (10), 13–17. https://doi.org/10.5120/11430-6785

[11] Thompson, R. R., Garfin, D. R., & Silver, R. C. (2017). Evacuation from Natural Disasters: A Systematic Review of the Literature. *Risk Analysis*, *37* (4), 812–839. https://doi.org/10.1111/risa.12654

[12] Yanovskiy, M., Levi, O. N., Shaki, Y. Y., & Socol, Y. (2020). Consequences of a large-scale nuclear accident and guidelines for evacuation: A cost-effectiveness analysis. *International Journal of Radiation Biology*, *96* (11), 1382–1389. https://doi.org/10.1080/09553002.2020.1779962

[13] Yoshimura, K., Saegusa, J., & Sanada, Y. (2020). Initial decrease in the ambient dose equivalent rate after the Fukushima accident and its difference from Chernobyl. *Scientific Reports*, *10* (1), 3859. https://doi.org/10.1038/s41598-020-60847-0

[14] Zeager, J., & Stitz, C. (2016). *College Algebra*. http://dspace.calstate.edu/handle/10211.3/180387