Travel Time

Author: Nathan Bunker  
Date: August 30, 2016

# Introduction

The “travelling salesman” problem has long been an example of problems that are difficult for computers to solve. A full explanation of the problem is given here on Wikipedia: <https://en.wikipedia.org/wiki/Travelling_salesman_problem>

The American Immunization Registry Association (AIRA) works with state and local health departments across the US. The question was asked, if AIRA wanted to send a single staff person to visit every site, how long would this take? And what would be the most efficient route. This was a theoretical question posed to the student group, AIRA is not expecting to actually visit every site.

# Setup

The total of 58 travel sites were identified and travel times were calculated between all of these sites. Assumption was made that travel times were equivalent both directions (this reduced the amount of preparatory data that had to be collected.)

The staff member was expected to visit each site for 5 hours, between 9am and 5pm Monday through Friday. Travel time commenced at the end of a meeting, with the staff person traveling to the next site. Once travel was completed the staff member had 12 hours of down time and then would make another visit. Staff member was not expected to make visits on Saturday or Sunday. The simulation we explored two possibilities:

* Continuous Travel: Staff member visits all 58 sites before returning home.
* Disconnected Travel: Staff member returns home each weekend

In both cases the starting point of the staff member was not known. The simulation assumed that at least 8 hours were needed to travel to first location and that 7 hours were need to travel back on Friday. This setup mean that the first day of the travel week (the first Monday for continuous and every Monday for disconnected) was always dedicated to travel. It would also mean in disconnected travel weeks that the simulation was likely to prefer long travel legs to occur at the end or beginning of weeks. (Since the home travel assumptions were the same no matter how far the site might be from the potential home site.)

Calculating a complete solution was estimated to require review of 58! / 2 = 1.17 x 10^78 possibilities. Too many for a brute force solution. So a genetic algorithm was chosen.

# Architecture

|  |  |
| --- | --- |
| **Java Class** | **Responsibilities** |
| domain/DataStore | Holds all data used during calculations. |
| domain/TravelAgent | Represents a specific scenario or “creature” that is a potential solution to the problem. |
| domain/Trip | The Travel Agent organizes a list of Trips, each trip lasting a week. |
| domain/TripStop | The Trip organizes a list of Trip Stops, each trip stop representing a place visited |
| domain/TravelTime | Represents the amount of time to take one segment of the trip |
| logic/TripBuilderInterface | Represents a class that can built a Travel Agent. |
| logic/TripBuilderFactory | Allows for creation of different types of trip builder factories |
| logic/TripBuilderContinuous | Builds a Travel Agent that details a continuous trip |
| logic/TripBuilderDisconnected | Builds a Travel Agent that details a disconnected trip |

The core of the genetic algorithm logic is contained in the TripBuilders. Here is the interface declaration:

**public** **interface** TripBuilderInterface {

**public** List<Trip> makeTrip(DataStore dataStore, List<Destination> destinationList);

**public** List<Trip> makeTrip(DataStore dataStore);

**public** List<Trip> makeNewTripList(List<Trip> tripList1, List<Trip> tripList2, DataStore dataStore, String generation) ;

**public** List<Trip> clone(TravelAgent travelAgent) ;

}

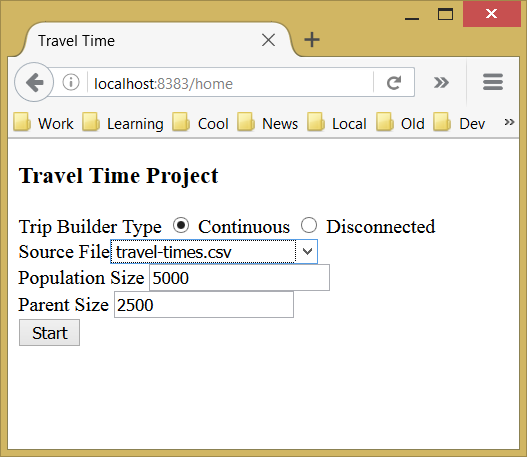
|  |  |
| --- | --- |
| makeTrip(dataStore, destinationList) | Creates a new trip based on the destinations passed in. This supports creating a trip where the route is already determined. |
| makeTrip(dataStore) | Creates a new random trip. |
| makeNewTripList(tripList1, tripList2, dataStore, generation) | Creates a new trip based on two parent trips. The new trip will share traits of both parent trips. |
| Clone(travelAgent) | Creates a new trip that is a copy of a single parent trip. Copy is subject to the potential of a random mutation that changes one item of the itinerary. |

Design of the TripBuilder is critical to the success of the algorithm. In order for this algorithm to work the following conditions must be true:

* The “DNA” of the TravelAgent must be something that can be randomly generated and work.
* The TripBuilder must be able to make a new TravelAgent based on two other TravelAgents that preserves something of the attributes of both parents. In this way good and bad traits may be passed from one generation to the next.
* A metric must be selected to determine best fit. (In this case it’s simply the total number of hours spent traveling, the best solution minimizes this time.)
* The search space must include many potential solutions that are roughly as good or nearly as good as the best solution.

# Operation

Because this was just a demonstration project the algorithm is manually run.



The application is started in Eclipse using an embedded Jetty server. The home page is not password protected, and the user is asked to indicate:

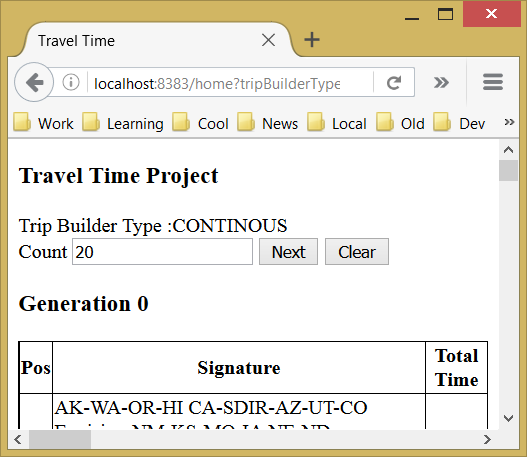
**Trip Builder Type**: Pick between the types of trips. Continuous means the staff traveling never goes home and disconnected means the staff member goes home each weekend.

**Source File**: A comma-separated-value file that contains the travel times between nodes. These are pre-loaded and included in the project.

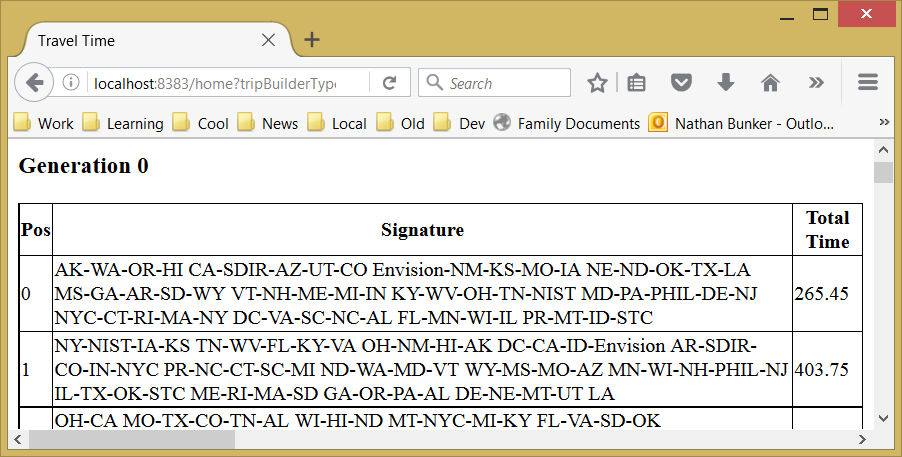
**Population Size**: The number of TravelAgents that will be created. Larger size means the process takes longer but might find a better solution.

**Parent Size**: The number of the population that can survive to the next generation.

Click start creates all the creatures and puts the user on a page to start the process.

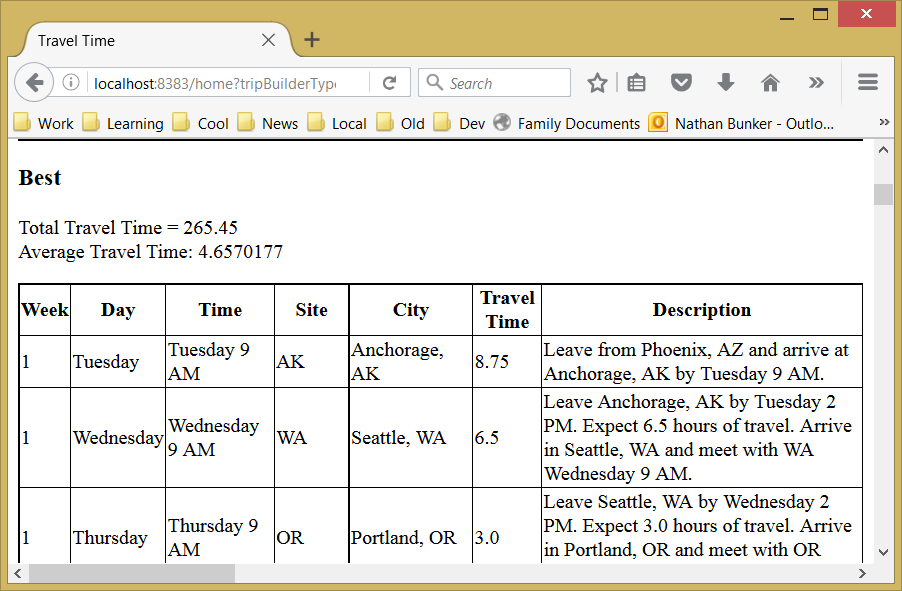


The **Count** field is where you specify the number of generations that should be generated for the next page. A higher number takes longer, a lower number allows you to more closely follow what is going on.

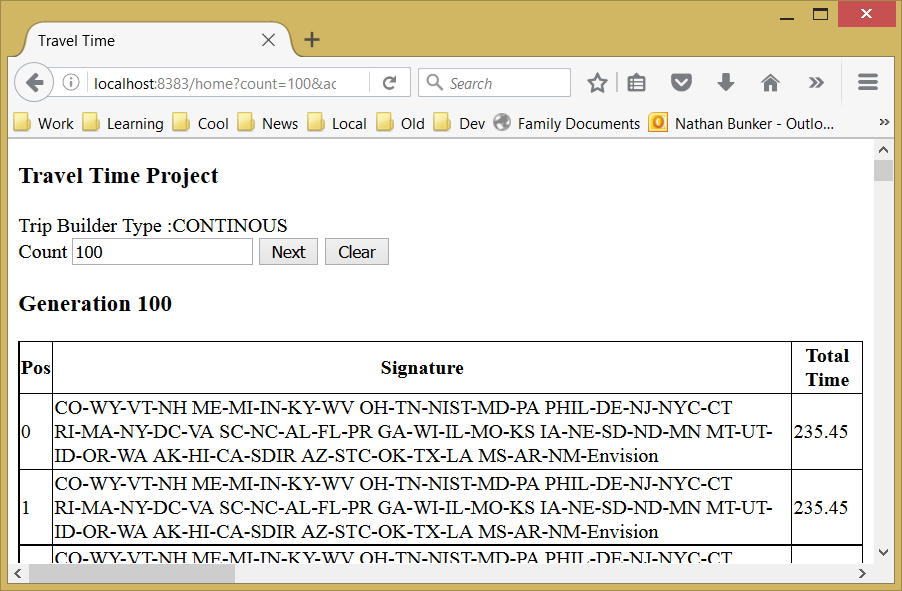
Below this is a table showing the signature of the best route. The first one shown was actually generated by the Christofides method: <https://en.wikipedia.org/wiki/Christofides_algorithm> Christofides solves the travelling salesman problem with a solution that is mathematically guaranteed to be no more than 50% longer than the best solution. This solution is included in the initial set as a starting point, the additional travel agents listed below are randomonly generated.

From this initial solution we can also know the absolute floor, the best solution cannot be less than 2/3 the Christofides solution. We think the real solution is probably higher than this line, but we know it’s not below this. So we set this as a bound in the rest of the report.

Moving down the page, more information can be shown. The best trip is shown in detail:

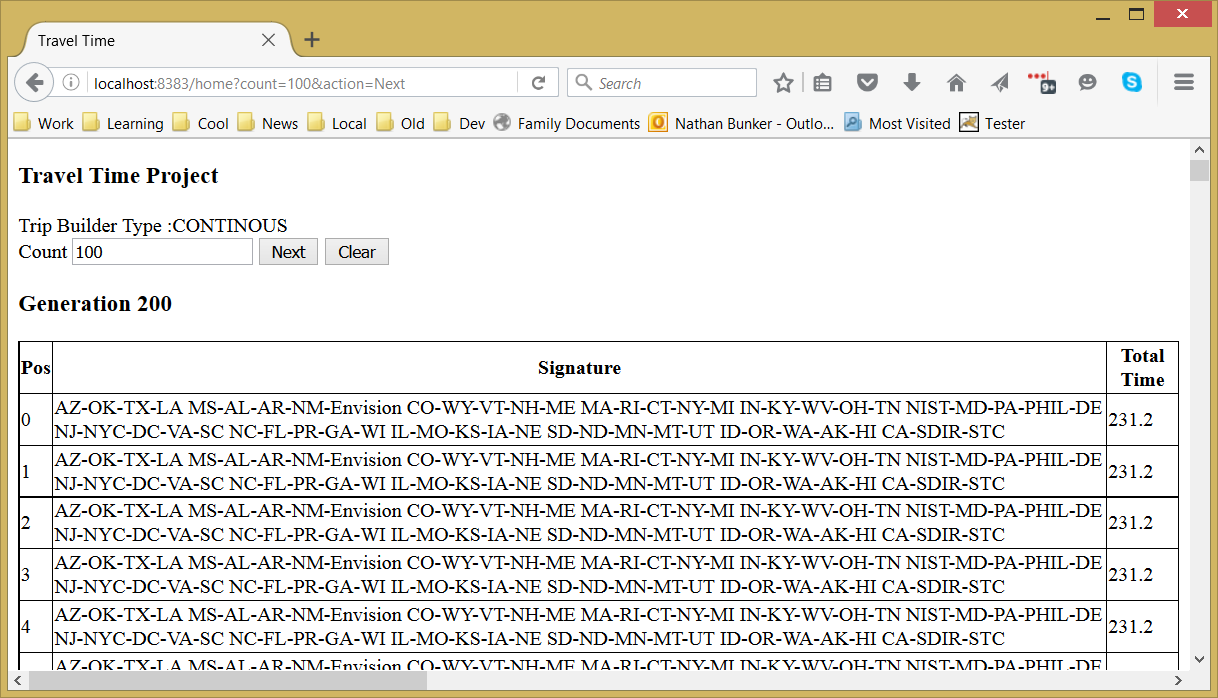


Now the user clicks and moves forward 100 generations. The following can be shown:



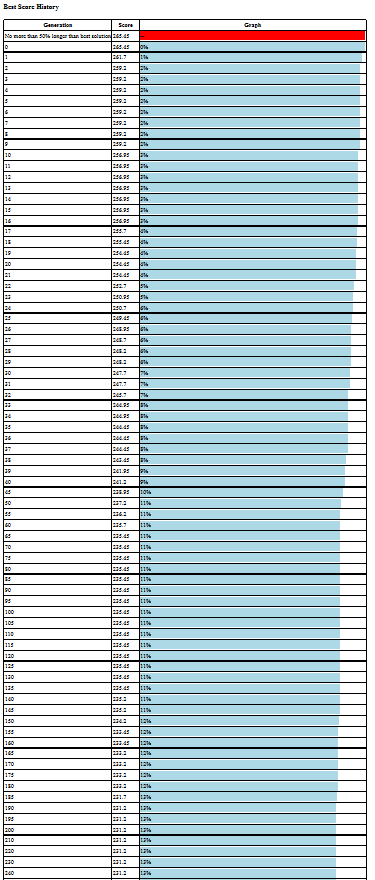
The algorithm has found a solution at 235.45 hours, compared to 265.45. The absolute minimum possible is calculated at 176. (However we have no idea how likely the best solution is to this lower bound, it’s probably that the best solution is closer to 235 than to 176.)

Running the algorithm 100 more generations comes with this solution:



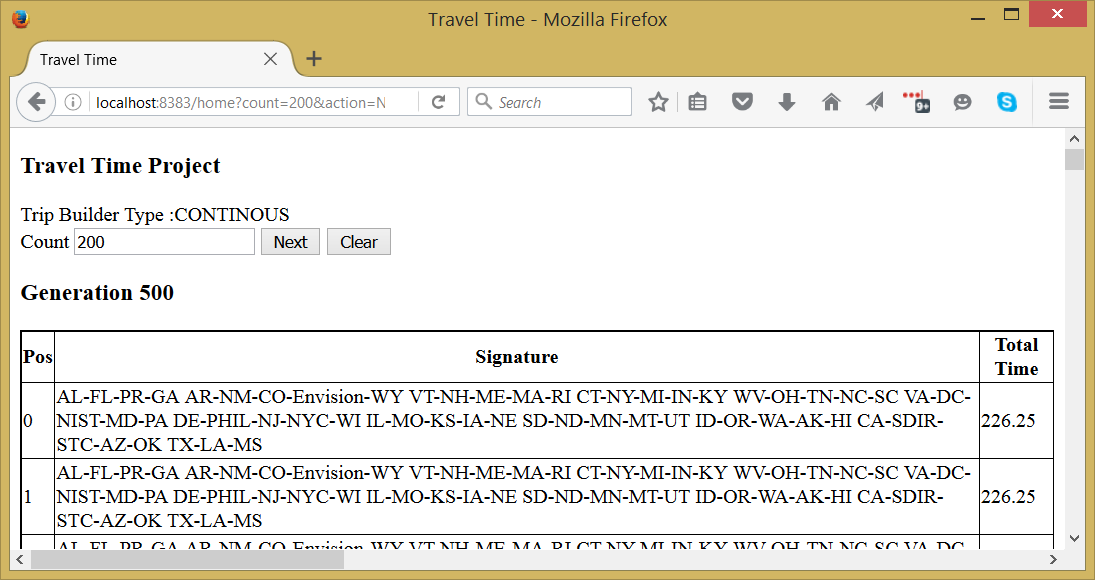
Now the solution improved by a further four hours to 231.20. And then again was run 100 more and improved further to 229.70.

The page shows the history in a rough graph:



The graph shows the optimization makes fast gains at first but as time goes by the gains begin to slow.

After running for 1,000 generations here is the best result:



Here is the detailed print out of the best solution:

### Best

Total Travel Time = 226.25  
Average Travel Time: 3.9692984

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Week** | **Day** | **Time** | **Site** | **City** | **Travel Time** | **Description** |
| 1 | Tuesday | Tuesday 9 AM | AL | Montgomery, AL | 5.0 | Leave from Jackson, MS and arrive at Montgomery, AL by Tuesday 9 AM. |
| 1 | Wednesday | Wednesday 9 AM | FL | Tallahassee, FL | 6.0 | Leave Montgomery, AL by Tuesday 2 PM. Expect 6.0 hours of travel. Arrive in Tallahassee, FL and meet with FL Wednesday 9 AM. |
| 1 | Thursday | Thursday 10 AM | PR | San Juan, PR | 8.0 | Leave Tallahassee, FL by Wednesday 2 PM. Expect 8.0 hours of travel. Arrive in San Juan, PR and meet with PR Thursday 10 AM. |
| 1 | Friday | Friday 10 AM | GA | Atlanta, GA | 6.75 | Leave San Juan, PR by Thursday 3 PM. Expect 6.75 hours of travel. Arrive in Atlanta, GA and meet with GA Friday 10 AM. |
| 2 | Monday | Monday 9 AM | AR | Little Rock, AR | 4.5 | Leave Atlanta, GA by Friday 3 PM. Expect 4.5 hours of travel. Arrive in Little Rock, AR and meet with AR Monday 9 AM. |
| 2 | Tuesday | Tuesday 10 AM | NM | Santa Fe, NM | 7.25 | Leave Little Rock, AR by Monday 2 PM. Expect 7.25 hours of travel. Arrive in Santa Fe, NM and meet with NM Tuesday 10 AM. |
| 2 | Wednesday | Wednesday 9 AM | CO | Denver, CO | 4.25 | Leave Santa Fe, NM by Tuesday 3 PM. Expect 4.25 hours of travel. Arrive in Denver, CO and meet with CO Wednesday 9 AM. |
| 2 | Thursday | Thursday 9 AM | Envision | Denver, CO | 0.0 | Leave Denver, CO by Wednesday 2 PM. Expect 0.0 hours of travel. Arrive in Denver, CO and meet with Envision Thursday 9 AM. |
| 2 | Friday | Friday 9 AM | WY | Cheyenne, WY | 2.0 | Leave Denver, CO by Thursday 2 PM. Expect 2.0 hours of travel. Arrive in Cheyenne, WY and meet with WY Friday 9 AM. |
| 3 | Monday | Monday 9 AM | VT | Montpelier, VT | 1.25 | Leave Cheyenne, WY by Friday 2 PM. Expect 1.25 hours of travel. Arrive in Montpelier, VT and meet with VT Monday 9 AM. |
| 3 | Tuesday | Tuesday 9 AM | NH | Concord, NH | 2.0 | Leave Montpelier, VT by Monday 2 PM. Expect 2.0 hours of travel. Arrive in Concord, NH and meet with NH Tuesday 9 AM. |
| 3 | Wednesday | Wednesday 9 AM | ME | Augusta, ME | 2.75 | Leave Concord, NH by Tuesday 2 PM. Expect 2.75 hours of travel. Arrive in Augusta, ME and meet with ME Wednesday 9 AM. |
| 3 | Thursday | Thursday 9 AM | MA | Boston, MA | 2.75 | Leave Augusta, ME by Wednesday 2 PM. Expect 2.75 hours of travel. Arrive in Boston, MA and meet with MA Thursday 9 AM. |
| 3 | Friday | Friday 9 AM | RI | Providence, RI | 1.25 | Leave Boston, MA by Thursday 2 PM. Expect 1.25 hours of travel. Arrive in Providence, RI and meet with RI Friday 9 AM. |
| 4 | Monday | Monday 9 AM | CT | Hartford, CT | 1.75 | Leave Providence, RI by Friday 2 PM. Expect 1.75 hours of travel. Arrive in Hartford, CT and meet with CT Monday 9 AM. |
| 4 | Tuesday | Tuesday 9 AM | NY | Albany, NY | 2.0 | Leave Hartford, CT by Monday 2 PM. Expect 2.0 hours of travel. Arrive in Albany, NY and meet with NY Tuesday 9 AM. |
| 4 | Wednesday | Wednesday 9 AM | MI | Lansing MI | 5.75 | Leave Albany, NY by Tuesday 2 PM. Expect 5.75 hours of travel. Arrive in Lansing MI and meet with MI Wednesday 9 AM. |
| 4 | Thursday | Thursday 9 AM | IN | Indianapolis, IN | 4.75 | Leave Lansing MI by Wednesday 2 PM. Expect 4.75 hours of travel. Arrive in Indianapolis, IN and meet with IN Thursday 9 AM. |
| 4 | Friday | Friday 9 AM | KY | Frankfort, KY | 3.25 | Leave Indianapolis, IN by Thursday 2 PM. Expect 3.25 hours of travel. Arrive in Frankfort, KY and meet with KY Friday 9 AM. |
| 5 | Monday | Monday 9 AM | WV | Charlston, WV | 3.0 | Leave Frankfort, KY by Friday 2 PM. Expect 3.0 hours of travel. Arrive in Charlston, WV and meet with WV Monday 9 AM. |
| 5 | Tuesday | Tuesday 9 AM | OH | Columbus, OH | 2.75 | Leave Charlston, WV by Monday 2 PM. Expect 2.75 hours of travel. Arrive in Columbus, OH and meet with OH Tuesday 9 AM. |
| 5 | Wednesday | Wednesday 9 AM | TN | Nashville, TN | 4.25 | Leave Columbus, OH by Tuesday 2 PM. Expect 4.25 hours of travel. Arrive in Nashville, TN and meet with TN Wednesday 9 AM. |
| 5 | Thursday | Thursday 9 AM | NC | Raleigh, NC | 4.25 | Leave Nashville, TN by Wednesday 2 PM. Expect 4.25 hours of travel. Arrive in Raleigh, NC and meet with NC Thursday 9 AM. |
| 5 | Friday | Friday 9 AM | SC | Columbia, SC | 3.75 | Leave Raleigh, NC by Thursday 2 PM. Expect 3.75 hours of travel. Arrive in Columbia, SC and meet with SC Friday 9 AM. |
| 6 | Monday | Monday 9 AM | VA | Richmond, VA | 5.5 | Leave Columbia, SC by Friday 2 PM. Expect 5.5 hours of travel. Arrive in Richmond, VA and meet with VA Monday 9 AM. |
| 6 | Tuesday | Tuesday 9 AM | DC | Washington, DC | 2.25 | Leave Richmond, VA by Monday 2 PM. Expect 2.25 hours of travel. Arrive in Washington, DC and meet with DC Tuesday 9 AM. |
| 6 | Wednesday | Wednesday 9 AM | NIST | Gaithersburg, MD | 1.0 | Leave Washington, DC by Tuesday 2 PM. Expect 1.0 hours of travel. Arrive in Gaithersburg, MD and meet with NIST Wednesday 9 AM. |
| 6 | Thursday | Thursday 9 AM | MD | Baltimore, MD | 1.0 | Leave Gaithersburg, MD by Wednesday 2 PM. Expect 1.0 hours of travel. Arrive in Baltimore, MD and meet with MD Thursday 9 AM. |
| 6 | Friday | Friday 9 AM | PA | Harrisburg, PA | 1.5 | Leave Baltimore, MD by Thursday 2 PM. Expect 1.5 hours of travel. Arrive in Harrisburg, PA and meet with PA Friday 9 AM. |
| 7 | Monday | Monday 9 AM | DE | Dover, DE | 2.75 | Leave Harrisburg, PA by Friday 2 PM. Expect 2.75 hours of travel. Arrive in Dover, DE and meet with DE Monday 9 AM. |
| 7 | Tuesday | Tuesday 9 AM | PHIL | Philidephia, PA | 1.5 | Leave Dover, DE by Monday 2 PM. Expect 1.5 hours of travel. Arrive in Philidephia, PA and meet with PHIL Tuesday 9 AM. |
| 7 | Wednesday | Wednesday 9 AM | NJ | Trenton, NJ | 1.0 | Leave Philidephia, PA by Tuesday 2 PM. Expect 1.0 hours of travel. Arrive in Trenton, NJ and meet with NJ Wednesday 9 AM. |
| 7 | Thursday | Thursday 9 AM | NYC | New York, NY | 1.25 | Leave Trenton, NJ by Wednesday 2 PM. Expect 1.25 hours of travel. Arrive in New York, NY and meet with NYC Thursday 9 AM. |
| 7 | Friday | Friday 9 AM | WI | Madison, WI | 6.25 | Leave New York, NY by Thursday 2 PM. Expect 6.25 hours of travel. Arrive in Madison, WI and meet with WI Friday 9 AM. |
| 8 | Monday | Monday 9 AM | IL | Springfield, IL | 5.25 | Leave Madison, WI by Friday 2 PM. Expect 5.25 hours of travel. Arrive in Springfield, IL and meet with IL Monday 9 AM. |
| 8 | Tuesday | Tuesday 9 AM | MO | Jefferson City, MO | 4.5 | Leave Springfield, IL by Monday 2 PM. Expect 4.5 hours of travel. Arrive in Jefferson City, MO and meet with MO Tuesday 9 AM. |
| 8 | Wednesday | Wednesday 9 AM | KS | Kansas City, KS | 3.0 | Leave Jefferson City, MO by Tuesday 2 PM. Expect 3.0 hours of travel. Arrive in Kansas City, KS and meet with KS Wednesday 9 AM. |
| 8 | Thursday | Thursday 9 AM | IA | De Moines, IA | 3.5 | Leave Kansas City, KS by Wednesday 2 PM. Expect 3.5 hours of travel. Arrive in De Moines, IA and meet with IA Thursday 9 AM. |
| 8 | Friday | Friday 9 AM | NE | Lincoln, NE | 3.5 | Leave De Moines, IA by Thursday 2 PM. Expect 3.5 hours of travel. Arrive in Lincoln, NE and meet with NE Friday 9 AM. |
| 9 | Monday | Monday 9 AM | SD | Pierre, SD | 7.0 | Leave Lincoln, NE by Friday 2 PM. Expect 7.0 hours of travel. Arrive in Pierre, SD and meet with SD Monday 9 AM. |
| 9 | Tuesday | Tuesday 9 AM | ND | Bismark, ND | 6.5 | Leave Pierre, SD by Monday 2 PM. Expect 6.5 hours of travel. Arrive in Bismark, ND and meet with ND Tuesday 9 AM. |
| 9 | Wednesday | Wednesday 9 AM | MN | St Paul, MN | 5.0 | Leave Bismark, ND by Tuesday 2 PM. Expect 5.0 hours of travel. Arrive in St Paul, MN and meet with MN Wednesday 9 AM. |
| 9 | Thursday | Thursday 9 AM | MT | Helena, MT | 6.25 | Leave St Paul, MN by Wednesday 2 PM. Expect 6.25 hours of travel. Arrive in Helena, MT and meet with MT Thursday 9 AM. |
| 9 | Friday | Friday 9 AM | UT | Salt Lake City, UT | 4.5 | Leave Helena, MT by Thursday 2 PM. Expect 4.5 hours of travel. Arrive in Salt Lake City, UT and meet with UT Friday 9 AM. |
| 10 | Monday | Monday 9 AM | ID | Boise, ID | 4.25 | Leave Salt Lake City, UT by Friday 2 PM. Expect 4.25 hours of travel. Arrive in Boise, ID and meet with ID Monday 9 AM. |
| 10 | Tuesday | Tuesday 9 AM | OR | Portland, OR | 4.25 | Leave Boise, ID by Monday 2 PM. Expect 4.25 hours of travel. Arrive in Portland, OR and meet with OR Tuesday 9 AM. |
| 10 | Wednesday | Wednesday 9 AM | WA | Seattle, WA | 3.0 | Leave Portland, OR by Tuesday 2 PM. Expect 3.0 hours of travel. Arrive in Seattle, WA and meet with WA Wednesday 9 AM. |
| 10 | Thursday | Thursday 9 AM | AK | Anchorage, AK | 6.5 | Leave Seattle, WA by Wednesday 2 PM. Expect 6.5 hours of travel. Arrive in Anchorage, AK and meet with AK Thursday 9 AM. |
| 10 | Friday | Friday 12 AM | HI | Honolulu, HI | 9.25 | Leave Anchorage, AK by Thursday 2 PM. Expect 9.25 hours of travel. Arrive in Honolulu, HI and meet with HI Friday 12 AM. |
| 11 | Monday | Monday 9 AM | CA | Richmond, CA | 9.0 | Leave Honolulu, HI by Friday 5 PM. Expect 9.0 hours of travel. Arrive in Richmond, CA and meet with CA Monday 9 AM. |
| 11 | Tuesday | Tuesday 9 AM | SDIR | San Diego, CA | 5.0 | Leave Richmond, CA by Monday 2 PM. Expect 5.0 hours of travel. Arrive in San Diego, CA and meet with SDIR Tuesday 9 AM. |
| 11 | Wednesday | Wednesday 9 AM | STC | Phoenix, AZ | 4.25 | Leave San Diego, CA by Tuesday 2 PM. Expect 4.25 hours of travel. Arrive in Phoenix, AZ and meet with STC Wednesday 9 AM. |
| 11 | Thursday | Thursday 9 AM | AZ | Phoenix, AZ | 0.0 | Leave Phoenix, AZ by Wednesday 2 PM. Expect 0.0 hours of travel. Arrive in Phoenix, AZ and meet with AZ Thursday 9 AM. |
| 11 | Friday | Friday 9 AM | OK | Oklahoma City, OK | 5.0 | Leave Phoenix, AZ by Thursday 2 PM. Expect 5.0 hours of travel. Arrive in Oklahoma City, OK and meet with OK Friday 9 AM. |
| 12 | Monday | Monday 9 AM | TX | Austin, TX | 5.5 | Leave Oklahoma City, OK by Friday 2 PM. Expect 5.5 hours of travel. Arrive in Austin, TX and meet with TX Monday 9 AM. |
| 12 | Tuesday | Tuesday 9 AM | LA | New Orleans, LA | 4.5 | Leave Austin, TX by Monday 2 PM. Expect 4.5 hours of travel. Arrive in New Orleans, LA and meet with LA Tuesday 9 AM. |
| 12 | Wednesday | Wednesday 9 AM | MS | Jackson, MS | 3.5 | Leave New Orleans, LA by Tuesday 2 PM. Expect 3.5 hours of travel. Arrive in Jackson, MS and meet with MS Wednesday 9 AM. |

The final solution was arrived at in the 400th generation and no improvement was found after that. The entire process took less than 5 minutes to complete.

The advantage to this process is that it can also be applied to solutions that do not match the original problem. While the above solution might seem useful at first, it has to be noted that no staff person is going to take 12 week business trip with no break to go home. It would be more practical to generate a series of week long trips that still minimizes travel times. This can be easily calculated using the same process. Here is the solution that we come up with that:

### Best

Total Travel Time = 389.5  
Average Travel Time: 6.8333335

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Week** | **Day** | **Time** | **Site** | **City** | **Travel Time** | **Description** |
| 1 | Tuesday | Tuesday 9 AM | PHIL | Philidephia, PA | 8.0 |  |
| 1 | Wednesday | Wednesday 9 AM | DE | Dover, DE | 1.5 |  |
| 1 | Thursday | Thursday 9 AM | DC | Washington, DC | 2.0 |  |
| 1 | Friday | Friday 9 AM | NIST | Gaithersburg, MD | 1.0 |  |
| 2 | Tuesday | Tuesday 9 AM | NYC | New York, NY | 8.0 |  |
| 2 | Wednesday | Wednesday 9 AM | NJ | Trenton, NJ | 1.25 |  |
| 2 | Thursday | Thursday 9 AM | PA | Harrisburg, PA | 2.25 |  |
| 2 | Friday | Friday 9 AM | MD | Baltimore, MD | 1.5 |  |
| 3 | Tuesday | Tuesday 9 AM | NY | Albany, NY | 8.0 |  |
| 3 | Wednesday | Wednesday 9 AM | CT | Hartford, CT | 2.0 |  |
| 3 | Thursday | Thursday 9 AM | MA | Boston, MA | 1.75 |  |
| 3 | Friday | Friday 9 AM | ME | Augusta, ME | 2.75 |  |
| 4 | Tuesday | Tuesday 9 AM | AR | Little Rock, AR | 8.0 |  |
| 4 | Wednesday | Wednesday 9 AM | CO | Denver, CO | 5.25 |  |
| 4 | Thursday | Thursday 9 AM | Envision | Denver, CO | 0.0 |  |
| 4 | Friday | Friday 9 AM | WY | Cheyenne, WY | 2.0 |  |
| 5 | Tuesday | Tuesday 9 AM | AZ | Phoenix, AZ | 8.0 |  |
| 5 | Wednesday | Wednesday 9 AM | STC | Phoenix, AZ | 0.0 |  |
| 5 | Thursday | Thursday 9 AM | UT | Salt Lake City, UT | 4.5 |  |
| 5 | Friday | Friday 9 AM | ID | Boise, ID | 4.25 |  |
| 6 | Tuesday | Tuesday 9 AM | RI | Providence, RI | 8.0 |  |
| 6 | Wednesday | Wednesday 9 AM | NH | Concord, NH | 2.0 |  |
| 6 | Thursday | Thursday 9 AM | VT | Montpelier, VT | 2.0 |  |
| 6 | Friday | Friday 9 AM | VA | Richmond, VA | 6.25 |  |
| 7 | Tuesday | Tuesday 9 AM | MS | Jackson, MS | 8.0 |  |
| 7 | Wednesday | Wednesday 9 AM | LA | New Orleans, LA | 3.5 |  |
| 7 | Thursday | Thursday 9 AM | KS | Kansas City, KS | 5.0 |  |
| 7 | Friday | Friday 9 AM | MO | Jefferson City, MO | 3.0 |  |
| 8 | Tuesday | Tuesday 9 AM | FL | Tallahassee, FL | 8.0 |  |
| 8 | Wednesday | Wednesday 9 AM | GA | Atlanta, GA | 4.5 |  |
| 8 | Thursday | Thursday 9 AM | KY | Frankfort, KY | 4.25 |  |
| 8 | Friday | Friday 9 AM | WV | Charlston, WV | 3.0 |  |
| 9 | Tuesday | Tuesday 9 AM | WI | Madison, WI | 8.0 |  |
| 9 | Wednesday | Wednesday 9 AM | MN | St Paul, MN | 5.0 |  |
| 9 | Thursday | Thursday 9 AM | IA | De Moines, IA | 4.5 |  |
| 9 | Friday | Friday 9 AM | NE | Lincoln, NE | 3.5 |  |
| 10 | Tuesday | Tuesday 9 AM | SC | Columbia, SC | 8.0 |  |
| 10 | Wednesday | Wednesday 9 AM | NC | Raleigh, NC | 3.75 |  |
| 10 | Thursday | Thursday 9 AM | TN | Nashville, TN | 4.25 |  |
| 10 | Friday | Friday 9 AM | AL | Montgomery, AL | 5.0 |  |
| 11 | Tuesday | Tuesday 9 AM | MI | Lansing MI | 8.0 |  |
| 11 | Wednesday | Wednesday 9 AM | OH | Columbus, OH | 5.25 |  |
| 11 | Thursday | Thursday 9 AM | IN | Indianapolis, IN | 3.5 |  |
| 11 | Friday | Friday 9 AM | IL | Springfield, IL | 4.25 |  |
| 12 | Tuesday | Tuesday 9 AM | AK | Anchorage, AK | 8.0 |  |
| 12 | Wednesday | Wednesday 9 AM | WA | Seattle, WA | 6.5 |  |
| 12 | Thursday | Thursday 9 AM | CA | Richmond, CA | 5.5 |  |
| 12 | Friday | Friday 9 AM | OR | Portland, OR | 5.25 |  |
| 13 | Tuesday | Tuesday 9 AM | MT | Helena, MT | 8.0 |  |
| 13 | Wednesday | Wednesday 10 AM | ND | Bismark, ND | 7.25 |  |
| 13 | Thursday | Thursday 9 AM | SD | Pierre, SD | 6.5 |  |
| 13 | Friday | Friday 10 AM | OK | Oklahoma City, OK | 7.5 |  |
| 14 | Tuesday | Tuesday 9 AM | NM | Santa Fe, NM | 8.0 |  |
| 14 | Wednesday | Wednesday 9 AM | TX | Austin, TX | 6.5 |  |
| 14 | Thursday | Thursday 9 AM | SDIR | San Diego, CA | 5.75 |  |
| 14 | Friday | Friday 12 AM | HI | Honolulu, HI | 9.25 |  |
| 15 | Tuesday | Tuesday 9 AM | PR | San Juan, PR | 8.0 |  |

# Lesson Learned and Moving Forward

What was most surprising is that writing this algorithm took very little time and was tolerant of noise and imperfections. Going forward we can see making the following improvements and changes to match more closely what might be needed:

* Add in AIRA staff who might travel and their travel times to the different locations so trips could be built for specific staff.
* Add preference select for staff to indicate sites they do or don’t want to visit and allow sites to indicate staff they would prefer to visit.
* Set optimization on an actual calendar of available dates (for both staff and sites)
* Give preference to visiting sites when weather is less likely to disrupt plans.
* Improve model for travel to include both flying and driving times. (Currently it’s a mix of both, but we decide at the beginning which it will be for a particular leg.) It might be more efficient to have driving segments together so a car is rented for longer periods of time and this reduces hassle for the traveler.

All of these changes can be done in the current model, all that is required is to collect more data, improve the model and create an evaluation criteria that matches what we are optimizing for.