K. Huang, K. Marrero, E. Radler, J. Rivera

Northwestern University's SafeRide Services
IEMS 310 – Operations Research Project

I. Introduction

SafeRide is a service provided to members of the Northwestern community as a safe and free alternative to walking alone after dark. Every night of the week, drivers (student employees) provide rides to and from destinations in and around Northwestern's Evanston campus. However, SafeRide is not a taxi, it is a safety service meant to be used in conjunction with other strategies such as traveling in groups and using Northwestern's shuttle services.

On weeknights from 9 PM to 2:30 AM, SafeRide dispatchers sends **six** cars to numerous pickup locations, and delivers a maximum of **four** people per pick up. Almost every student receives the route they've requested, but sometimes SafeRide cannot accommodate students in certain situations like getting picked up from a bar, traveling for a distance shorter than two blocks, or needing medical transport. On average, SafeRide requests around 197 rides per weeknight, ranging from one to four passengers per ride. Dispatchers are constantly trying to find the best way to route SafeRide drivers from pick up to pick up in order to minimize the amount of time a driver travels to pick up a student and the amount of time a driver is without a passenger in their vehicle. Unfortunately, this optimization often gets set aside because of the huge time constraint dispatchers face when accepting students' requests.

II. Scenarios

Our first optimization scenario considers a typical weeknight at SafeRide, where six drivers are already on duty and are currently stationed at the following locations:

1. Barnes and Noble
2. Zeta Beta Tau Fraternity
3. Pratt Ct. and Foster St.
4. University Library
5. Tech Parking Lot
6. Willard Residential College

A dispatcher has a recently received ten requests from students to be picked up at the following locations:

1. Regenstein Hall
2. Bobb Hall
3. Zeta Tau Alpha Sorority
4. Sigma Alpha Epsilon Fraternity
5. University Library
6. Bobb Hall (2)
7. Norris University Center
8. Allison Hall
9. Ford Bldg
10. Elder Hall

The constraints of this optimization problem include the specific distances drivers are away from each of the students' requested pickups, and that SafeRide vehicles are limited to transporting four people at a time, meaning dispatchers can pair student requests together for one driver. Our goal is to optimize the use of each SafeRide driver based on their current location by minimizing the time it takes the drivers to pick up all ten caller groups.

The second scenario is an extension of the first scenario, and involves the same number of SafeRide drivers (6) and the same number of caller groups (10). The drivers, however, are now required to pick up the caller group and then drop them off at their desired destination before picking up the next group. The dispatcher received ten calls with the following pickup and drop-off locations:

1. Regenstein to Judson/Dempster
2. Bobb to Nevin's
3. Zeta to Phi Delta Theta
4. SAE to Tridelt
5. Library to 812 Forest
6. Bobb to BK
7. Norris to Ridge/Noyes
8. Allison to Tech Library
9. Ford to Shepard
10. Elder to CVS

This optimization scenario also includes the constraint that each driver is at a specific distance away from their first or second pickup. There no longer is a constraint of capacity within each vehicle because the driver is required to pick up and drop off one caller group before moving to the next one. It's expected that some drivers will need to pick up only one caller group and that some will have to pick up two caller groups, based on the number of drivers versus the number of requests. As with the first scenario, our goal is to minimize the sum time of all of the drivers' pickups and drop-offs.

III. Analysis

a. Scenario 1

Applying the aforementioned constraints, a linear program was created in Excel and using Solver generated the minimum possible sum amount of time required to pick up all 10 students with the six vehicles, 11.3 minutes. The linear program is shown below. Refer to appendix B for a complete key defining each variable:

$$\min \sum_{i=1}^{6} (T1_i + T2_i)$$

$$\text{s.t.} \quad T1_i = \sum_{j=1}^{10} a_{ji} * x_{ji}$$

$$T2_i = \begin{cases} \sum_{j=1}^{10} b_{ji} * y_{ji} & T1_i = 0 \\ \sum_{j=1}^{10} a_{ji} * y_{ji} & T1_i > 0 \end{cases}$$

$[B]_i = C*[X]_i$ *Note: brackets denote that they are column vectors, not scalar values

$$\sum_{j=1}^{10} x_{ji} \leq 1 \qquad \sum_{j=1}^{10} y_{ji} \leq 1 \qquad \sum_{i=1}^{6} x_{ji} + y_{ji} = 1$$

$$\sum_{j=1}^{10} (p_{ji}*x_{ji} + p_{ji}*y_{ji}) \leq 4$$

$$0 \leq x_{ji} \leq 1 \qquad 0 \leq y_{ji} \leq 1$$

The program utilized the calculated times found between each of the starting locations and the ten desired pick up locations, as well as the times for travel between the pick up locations. The program assumes that only two rounds of pick-ups will occur, such that no car will travel to more than two locations before dropping off its passengers, and also sets rider capacities for each pick up location to:

Pick Up Location	Capacity
Regenstein Hall	3
Bobb Hall	2
Zeta Tau Alpha Sorority	1
Sigma Alpha Epsilon Fraternity	1
University Library	1
Bobb Hall (2)	2
Norris University Center	1
Allison Hall	2
Ford Bldg	1
Elder Hall	3

The results show that the quickest options involve the drivers originating at ZBT Fraternity, University Library, Tech Parking Lot, and Willard Residential College stopping at two locations. The second pick-ups then start from the location of either the first pick-up or their original location.

Originating Location	Pick Up Location 1	Pick Up Location 2	Total Travel Time
Barnes and Noble	-	Regenstein	3 min
ZBT Fraternity	SAE	Ford	1.3 min
Pratt Ct. and Foster St	-	Library	2.1 min
University Library	Norris	Elder	1.6 min
Tech Parking Lot	Bobb	Bobb	0.2 min
Willard Residential College	Zeta	Allison	3.1 min

b. Scenario 2

Using the different scenario description, a second linear program was created in Excel and using Solver generated the minimum possible sum amount of time required to pick up and drop off all 10 students with the six vehicles, 42.8 minutes. The linear program is shown below:

$$\min \sum_{i=1}^{6} (T1_i + T2_i)$$

$$\text{s.t.} \qquad T1_i = \sum_{j=1}^{10} a_{ji} * x_{ji}$$

$$T2_i = \begin{cases} \sum_{j=1}^{10} b_{ji} * y_{ji} & T1_i = 0 \\ \sum_{j=1}^{10} a_{ji} * y_{ji} & T1_i > 0 \end{cases}$$

$B]_i = C*[X]_i$ *Note: brackets denote that they are column vectors, not scalar values

$$\sum_{j=1}^{10} x_{ji} \leq 1 \qquad \sum_{j=1}^{10} y_{ji} \leq 1 \qquad \sum_{i=1}^{6} x_{ji} + y_{ji} = 1$$

$$0 \leq x_{ji} \leq 1 \qquad 0 \leq y_{ji} \leq 1$$

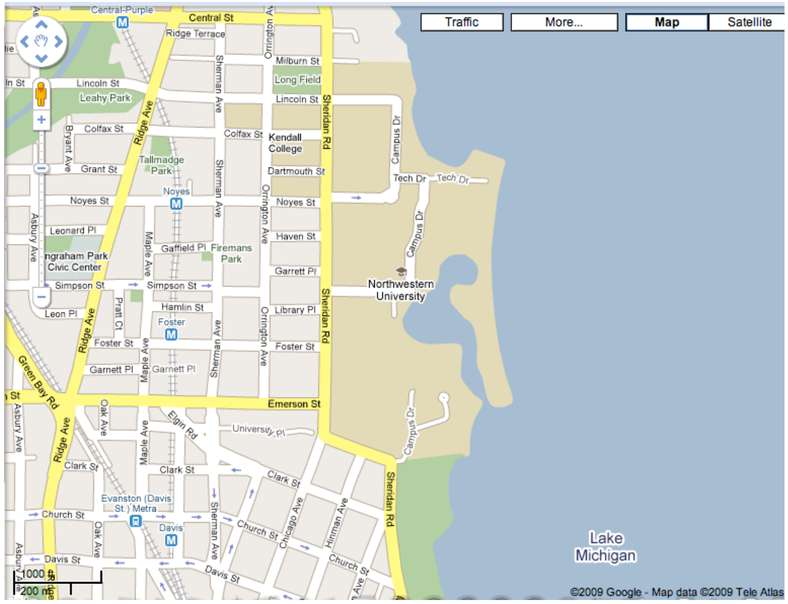
This scenario differs from the first in that for those drivers who need to complete a second pick up, their originating location for the second rider will be the drop-off for the first rider. This scenario aimed to minimize the sum amount of time that all the drivers will be in transit, either picking up a rider, or driving them to their drop-off location. The program utilized the calculated times found between each of the starting locations and the ten desired pick up locations, as well as the times between the ten desired drop-off locations and the other desired pick-up locations (excluding the pick-up that preceded the drop-off). The new constraints also adjusted the linear program, such that there is no longer a capacity for each location but rather that a driver must complete a route before picking up a new passenger and beginning a new route.
Similar to Scenario 1, four drivers will complete two rounds of pick-ups. The results show that it is quickest for the drivers originating at Barnes & Noble, University Library, Tech Parking Lot, and Willard to pick up two rounds of passengers. Those that do not have passengers in the first round leave from their originating location to pick up their second round passengers.

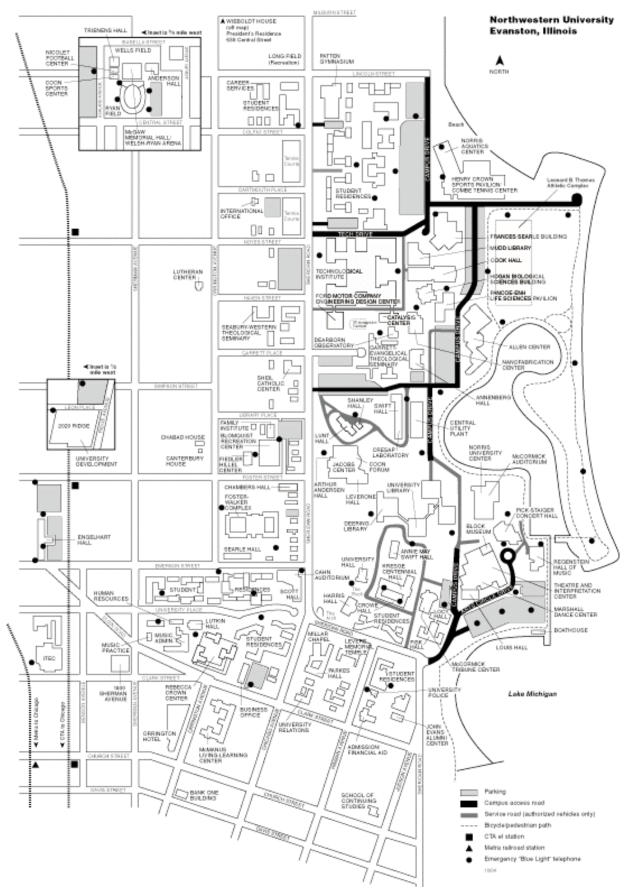
Originating Location	Pick Up Location 1	Pick Up Location 2	Total Travel Time (min)
Barnes and Noble	Allison	Bobb	5.8
ZBT Fraternity	-	Library	6.7
Pratt Ct. and Foster St	-	Norris	5.4
University Library	Bobb	Regenstein	9.5
Tech Parking Lot	Elder	Ford	8.2
Willard Residential College	ZTA	SAE	7.2

IV. Conclusions

This optimization problem provides an example of a delegation that the dispatcher would have to repeat over and over throughout the night as new calls come in. The originating, pick-up, and drop-off locations provide a wide breath of stops that potential drivers would have to make, and these programs would provide a feasible solution to making the process of assigning drivers to passengers more efficient. Passengers would have to be assigned in batches however, because simply adding new passengers to the equation could change the current plans already established, so after the batch is complete the last drop-off location would become the new originating location. This is an extremely complex problem to solve, dealing with collection of data and unforeseeable circumstances (such as on the spot changes of drop-offs and number of passengers). Scenario 2 allows for the addition of passengers because one route must be completed before another can begin, however there is currently no way to predict when a passenger will ask to change destinations. Perhaps records could be kept for how often this happens and a probability for this occurrence could be determined. Creating a system in which these changes could automatically update the system, from the car, could be a potential solution. Another solution could be changing the way calls are received and the way dispatchers communicate and delegate to the drivers. One option could be that dispatchers receive calls and then immediately enter them into the system. The computer than generates the batches from the information received from all the dispatchers, and then sends out text messages to the callers to alert them of the arrival times of their drivers. Then, either another set of dispatchers or the computer communicates with the drivers to alert them of their next destination. The computer would use a similar program to the one created in scenario 2, and would use the destinations of the riders as the starting point for the next rider before they went to pick them up. The SafeRide service is a valuable program on Northwestern's campus, but perhaps there is a better way for it to operate to make it more efficient. This project demonstrated how complex the issue can become, but perhaps simplifying the way data is collected could help the program to run more smoothly, so as to increase the number of students it can help in a single night.

Appendix A: Maps of Locations





Appendix B: Linear Program Key

Scenario 1:

i = driver number i
j = caller number j

$T1_i$ = Time it takes driver i to make his/her route for the first pick-up
$T2_i$ = Time it takes driver i to make his/her route for the second pick-up

a_{ji} = component cell in matrix A. Located at row j and column i.
A = time matrix for the first pick-up:

Pick ups	**1**	**2**	**3**	**4**	**5**	**6**
Regenstein	3.0	3.1	4.8	3.1	2.7	3.4
Bobb	3.2	0.6	3.2	1.3	0.2	2.2
Zeta	1.5	3.4	2.4	3.7	2.6	2.1
SAE	3.5	0.8	3.7	1.2	0.3	2.8
Library	2.1	1.5	2.1	0.0	1.3	2.4
Bobb	3.2	0.6	3.2	1.3	0.2	2.2
Norris	2.2	1.6	2.2	0.0	1.2	2.3
Allison	1.0	2.1	2.6	1.1	1.5	1.5
Ford	2.0	1.7	2.4	0.9	0.1	2.6
Elder	3.7	0.2	3.5	1.8	0.4	2.9

b_{ji} = component cell in matrix B. Located at row j and column i.
C = time matrix that shows time it takes to get from one pick-up location to another:

	Regenstein	Bobb	Zeta	SAE	Library	Bobb	Norris	Allison	Ford	Elder
Regenstein	0.0	2.5	3.2	2.4	3.0	2.5	3.1	1.1	2.6	3.1
Bobb	2.5	0.0	2.2	0.1	1.2	0.0	1.4	1.6	0.5	0.7
Zeta	3.2	2.2	0.0	3.2	2.4	2.2	2.4	1.0	2.7	3.1
SAE	2.4	0.1	3.2	0.0	1.4	0.1	1.4	1.7	0.5	0.2
Library	3.0	1.2	2.4	1.4	0.0	1.2	0.0	1.2	0.9	1.3
Bobb	2.5	0.0	2.2	0.1	1.2	0.0	1.4	1.6	0.5	0.7
Norris	3.1	1.4	2.4	1.4	0.0	1.4	0.0	1.1	0.9	1.6
Allison	1.1	1.6	1.0	1.7	1.2	1.6	1.1	0.0	1.5	2.0
Ford	2.6	0.5	2.7	0.5	0.9	0.5	0.9	1.5	0.0	0.6
Elder	3.1	0.7	3.1	0.2	1.3	0.7	1.6	2.0	0.6	0.0

X = First pick-up matrix.
Y = Second pick-up matrix.
x_{ji} = component cell in matrix X. Located at row j and column i.
y_{ji} = component cell in matrix Y. Located at row j and column i.
p_{ji} = component cell in matrix P. Located at row j and column i.

$[B]_i$ = Column i in matrix B.

$[X]_i$ = Column i in matrix X.

P = person capacity matrix:

Pick ups	1	2	3	4	5	6
Regenstein	3	3	3	3	3	3
Bobb	2	2	2	2	2	2
Zeta	1	1	1	1	1	1
SAE	1	1	1	1	1	1
Library	1	1	1	1	1	1
Bobb	2	2	2	2	2	2
Norris	1	1	1	1	1	1
Allison	2	2	2	2	2	2
Ford	1	1	1	1	1	1
Elder	3	3	3	3	3	3

<u>Scenario 2:</u>

The key is exactly the same as scenario 1, except matrix A and matrix C have different values to account for the fact that the driver must drop off its passengers at the destination location, which takes additional time.

A = Time matrix for first pick-up. This is the time it takes for driver, starting from initial location, to get to pick-up location and then to passenger's drop-off location.

Pick ups	1	2	3	4	5	6
Regenstein	6.5	6.6	8.3	6.6	6.3	6.9
Bobb	6.2	3.6	6.2	4.3	3.2	5.2
Zeta	3.0	4.9	3.9	5.2	4.1	3.6
SAE	6.9	4.2	7.1	4.6	3.7	6.2
Library	7.3	6.7	7.3	5.2	6.5	7.6
Bobb	5.8	3.2	5.8	3.9	2.8	4.8
Norris	5.4	4.8	5.4	3.2	4.4	5.5
Allison	2.3	3.4	3.9	2.4	2.8	2.8
Ford	4.3	4.0	4.7	3.2	2.4	4.9
Elder	7.5	4.0	7.3	5.6	4.2	6.7

C = Time matrix for pick-up two. This shows the time it takes for driver to get to second pick-up location from a drop-off location and drop off second passenger(s).

	Judson/ Dempster	Nevin's	Phi Delt	Tridelt	812 Forest	Burger King	Ridge/ Noyes	Tech Library	Shepard	CVS
Regenstein	7.0	7.3	6.6	6.5	8.0	5.6	6.8	5.8	6.7	6.9
Bobb	6.2	6.0	3.5	6.6	8.5	5.6	5.4	3.5	6.5	6.6
Zeta	5.1	4.6	3.0	3.0	5.8	2.8	3.7	2.9	2.7	2.9
SAE	6.5	6.6	3.6	6.8	9.1	6.1	5.5	4.1	7.0	6.9
Library	9.3	8.9	6.5	8.4	10.4	6.6	8.4	6.7	8.6	8.5
Bobb	5.8	5.6	3.1	6.2	8.1	5.2	4.0	3.1	6.1	6.2

Norris	7.3	6.9	4.5	6.4	8.4	5.6	6.4	4.7	6.6	6.5
Allison	4.6	3.7	2.8	2.4	6.0	1.7	4.8	2.6	2.3	2.4
Ford	5.4	5.9	3.0	4.6	7.5	4.8	4.3	2.6	4.6	4.0
Elder	8.0	7.7	4.3	7.1	9.6	6.6	6.0	4.6	7.1	7.6

Appendix C: Excel program (separate file)