## Indiana University Bloomington

CSCI B 565

Data Mining

# IU Bus Route Optimization
# Group Name: B-Miners

*Authors:*
Kaushik Roy
Sujith Shivaprakash
Suhas Jagdish

*Professor:*
Dr. Dalkilic

December 18, 2015

**Abstract**

The goal of this project is to optimize the transportation time of buses across different stops for a particular Route, while ensuring that time constraints are met and passenger counts are within bus capacity.

We asses the effect of weather and passenger counts on the current route system. These effects include variations in time of arrival at a particular bus stop, variations in travel time between any two bus stops and dwell times at these stops.

We then formulate an algorithm to compute new bus routes to improve the efficiency of the existing IU bus system. The pros and cons of the proposed method are also discussed.

# Contents

4 | 1 Introduction

5 | 2 Problem Description

2.1 Presentation of case study 5
2.2 Mathematical formulation 5

6 | 3 Data Description

3.1 Existing model 6
3.2 Proposed model 7

7 | 4 Computational Implementation

4.1 Part 1 - Dwell Times and time variance 7
4.2 Calculating Average Passenger Counts per Route, per Day, given normal or abnormal Weather 8
4.3 New Routes computation 8

9 | 5 Visualizations

5.1 Variance between actual and schedule times for all the routes 9
5.2 Dwell times at a stop at different intervals of the day 10
5.3 Passenger counts at a stop at different intervals of the day 11
5.4 Average time taken at different stops 12

13 | 6 Proposed Changes

6.1 Modified route map for A and X 14
6.2 Route map for E 15

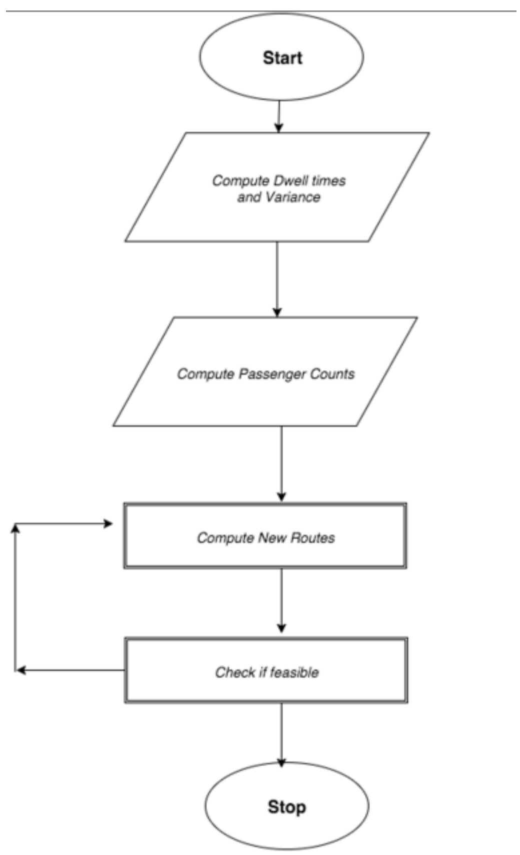
16

7
Concluding remarks

16 8
References

# 1 Introduction

- Due to the current passenger density and erratic weather patterns, the travel times between stops are unpredictable and this has lead to lowered passenger satisfaction. This calls for a need to optimize the current transport system.
- About IU Bus System : Following are the architecture details of database system that was used , along with the details of the data :
- Database used : Access
- We had been provided with IU Bus data for the timespan from Aug 2014 to Mar 2015. We were also provided weather data for the same timespan.
- Tables present : Interval data 201415, Route ID, Schedule Data, Stop ID, Weather Data, Work Record.




- Flow:

- Databases used : MySQL, MongoDB.
- Languages used : R.
- Softwares and Systems used : Revolution R, Karst, Big Red
- Information about the data : We have total 66 stops in total, 26 buses running and 4 routes.

# 2 Problem Description

Due to the current passenger density and erratic weather patterns, the travel times between stops are unpredictable and this has lead to lowered passenger satisfaction. This calls for a need to optimize the current transport system.

## 2.1 Presentation of case study

- The IU bus system started in 1966, consists of five fixed routes that connect outer campus to the campus core. The routes are A,B,E,X. Campus bus operates from 7:30AM to midnight Monday-Thursday,7:30AM to 3:00AM on Friday, 10:00AM to 3:00AM on Saturday and 12:PM until 10:00PM on Sunday. Buses are accessible to all riders. The buses are also equipped with GPS rackers, allowing patrons to access real time data on bus locations and next arrival times.
- The purpose of the system is to encourage the reduction of employee and student commuting single occupancy vehicles and incentivize the use of public transport.
- It was focused on information such as how students who live off-campus currently commute to campus.
- The system suffers from unpredictable variations in times of arrival vs scheduled times causing inconvenience to passengers.
- Also, due to the erratic weather in Bloomington, Indiana bus breakdowns and other unprecedented bus events occur.
- Therefore, there is a need to construct a more efficient bus route system.

## 2.2 Mathematical formulation

1: First we construct a weighted un directed complete graph of nodes (representing bus stops) called **G (V,E)** with the distance between two given stops as edge weights.
2: Let set R $\in$ shortest path between each node graph **G**.
3: Considering each $\mathbf{v} \in \mathbf{V}$ as a source s , obtain shortest paths to destination nodes d (all other vertices in V)

4: Let **A(i) = average passenger count at Stop i**
5: For every path s to d through intermediate nodes $i_1$, $i_2$, **.....**, $i_n$ calculate the average passenger count(APC) define below:
**(A(s) + A($i_1$) + A($i_2$) + .... + A($i_n$) + A(d))/(n+2)**
▷ n + 2 because there are n intermediate nodes plus a source and a destination.
6: Check if APC <= Bus capacity, set path s–>d as a potential route.
7: Choose 4 paths $s_i$ –> $d_i$ from G, such that **APC** is least.
8: Check new route effectiveness by calculating time taken T between $s_i$ and $d_i$ as (time at destination - time at start).
**if** *(T - sum($D_i$) = sum(time between any two stops))* **then**

assign the route.
**else** Goto Step 7

# 3 Data Description

This section shows the existing and proposed model and discusses the pros and cons of both.

## 3.1 Existing model

- **A Route**: goes through major stops:
  Stadium->wells->3rd&Jordan->IMU->Stadium
- **B Route:** goes through major stops:
  Fisher Court->Jordan&10th-¿3rd&Jordan->Fisher Court
- **E Route:** goes through major stops:
  Evermann->3rd&Jordan->IMU,wells->Evermann
- **X Route:** goes through major stops:
  Stadium->IMU->Stadium
  - **Pros**
    1. Sufficiently large amount of data has been maintained with meaningful attribute names.
    2. It spans the IU campus quite well and contains adequate number of buses.
  - **Cons**
    1. Very high traffic zones at ?hotspots? not dealt with effectively. (Stadium, Jordan Hall, Wells Library, 3rd and Jordan) express routes can be expanded (Not very meaningful at the moment). All the data related to the system is maintained in an access database.

2. Highly inconsistent and conflicting attribute values leading to very distorted analysis. No efficient data model to store the bus system entities.

### 3.2 Proposed model

- As outlined in the previous section, the existing system suffers from a proper data model due to which effective data analysis cannot be performed.
- First step in building our model is to design an effective data management system to store all the related bus system entities. We have used Mongo and MySQL for storing and querying bus data. We were able to derive some valuable insights from the data, using which we devised an algorithm to compute new set of routes based on various factors such as passenger count, variance between stops and the dwell times.
- The algorithm as discussed in section 2.2 eliminates a route from the current model and employs a new model with only 3 routes.

## 4 Computational Implementation

The following were the sequence of computational steps taken that resulted in the proposed system.

### 4.1 Part 1 - Dwell Times and time variance

- In the Interval data table provided, we appended additional columns called Scheduled time,Main Route, Day and Weather
- The table already contained columns representing record ID, From, To, time, BusID, RouteID and date time.
- The 'record ID' was used as a identifier. The 'From' column indicated from which stop the bus was traveling, the 'To' column indicated to which stop the bus was traveling, the 'time' contained how much time it took to reach stop 'To' from stop 'From'.
- Also, when 'From' and 'To' were the same value, the 'time' part of 'date time' indicated time of arrival at stop 'To'.
- The 'BusID' and 'RouteID' together identified a particular bus on a given date.
- First, we identified 'Main Route' for each bus as one the values in $A1, A2, A3, A4, A5, A6, A7$ for A Route
- $B1, B2, B3, B4, B5$ for B Route, $X1, X2$ for X Route and $E1, E2, E3$ for E Route

- We do this by matching 'BusID' and 'RouteID' in the Work Record table. We also obtained 'Scheduled time' using the 'stop ID' match from the Schedule Data table.
- We obtained the 'Weather' on a given date by matching the 'date time' field with the 'EDT' field in the Weather table and extracting the 'Event' field.
- Finally, we obtain the Day of week as the 'Day' column from the Schedule Data.
- Now, to compute dwell time per stop we simply extract all 'time' values when 'From' = 'To'.
- and to compute variance we take difference between Actual time in the 'time' part of the 'date time' field and the 'Scheduled time' column value, when 'From' $\neq$'To'

### 4.2 Calculating Average Passenger Counts per Route, per Day, given normal or abnormal Weather

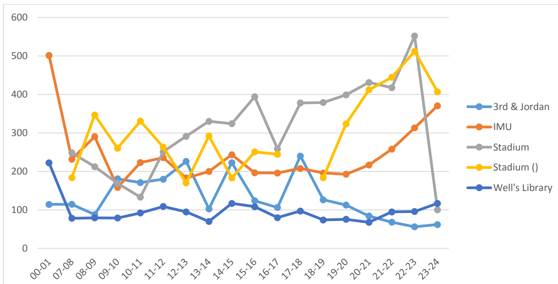
- We first figure out which stop names the times in the Raw tables (ARaw, BRaw,..,etc) by matching 'Stop ID' and 'Stop' from Schedule Data table with the times. These tables were present in the RidershipSchedule sheets.
- Next we add Columns 'Day' and 'Weather' to the Pivot tables (APivot, BPivot,..,etc) from the augmented interval data table by matching dates.
- Next we filter the augmented pivot tables on 'Weather' and 'Day' as required.
- For example, 'Weather'='Normal' and 'Day'='Monday' (All weather values other than normal were considered abnormal).
- Next, the passenger counts are averaged over respective stops.
- These counts are plotted by time ranging across all time values in the augmented APivot table.

### 4.3 New Routes computation

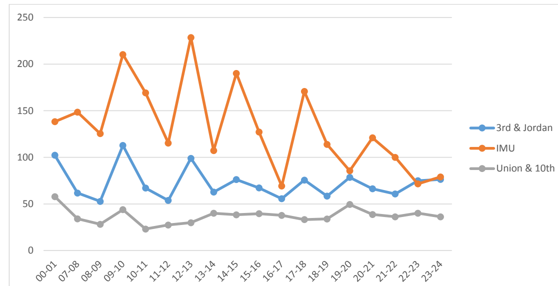
Finally, using the above information, the mathematical formulation as mentioned above is used to compute new routes. Examples for visualizations of compiled data is provided in the next section. The visualizations for rest of the routes are attached in the zip file.

# 5 Visualizations

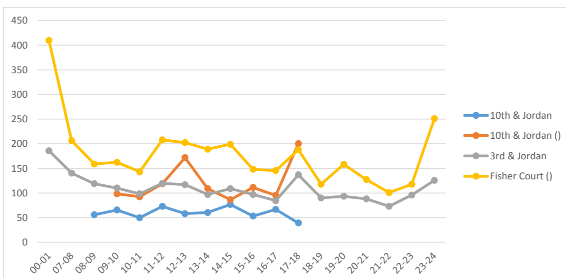
## 5.1 Variance between actual and schedule times for all the routes



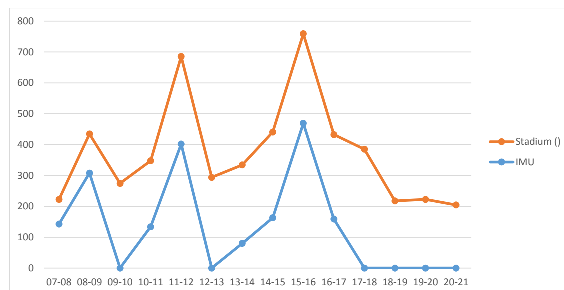

**Route A**: Variance (in seconds) at different times across all the stops




**Route E**: Variance (in seconds) at different times across all the stops



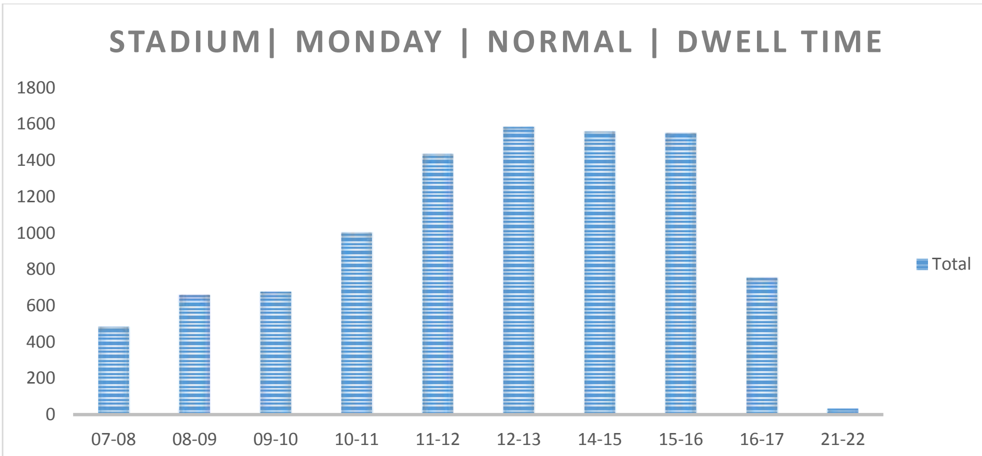

**Route B**: Variance (in seconds) at different times across all the stops



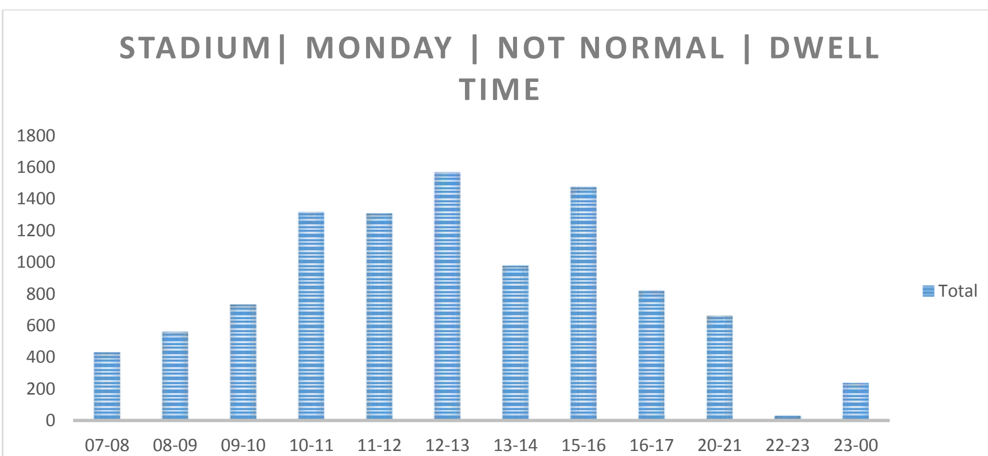

**Route X**: Variance (in seconds) at different times across all the stops

## 5.2 Dwell times at a stop at different intervals of the day

Route A: Dwell times (in seconds) for Stadium at different times on a NORMAL day.

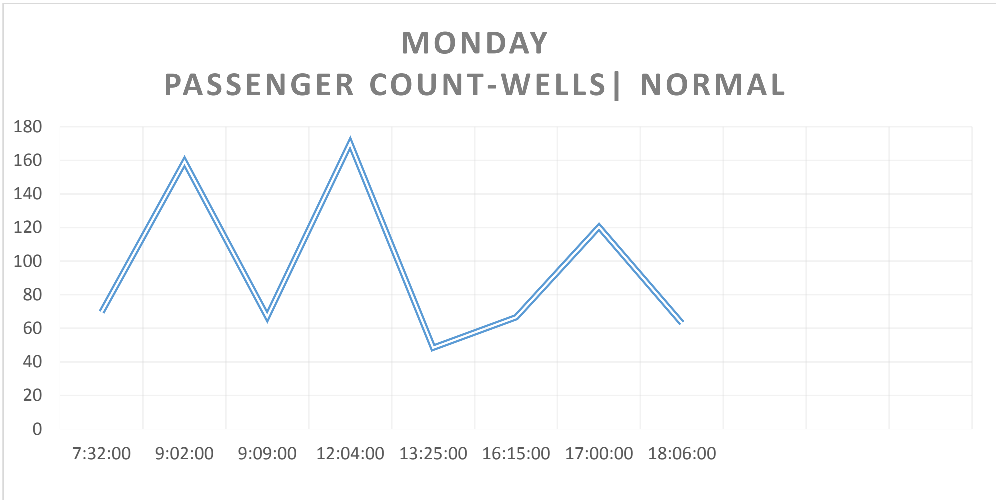



Route A: Dwell times (in seconds) for Stadium at different times on a NOT NORMAL day.

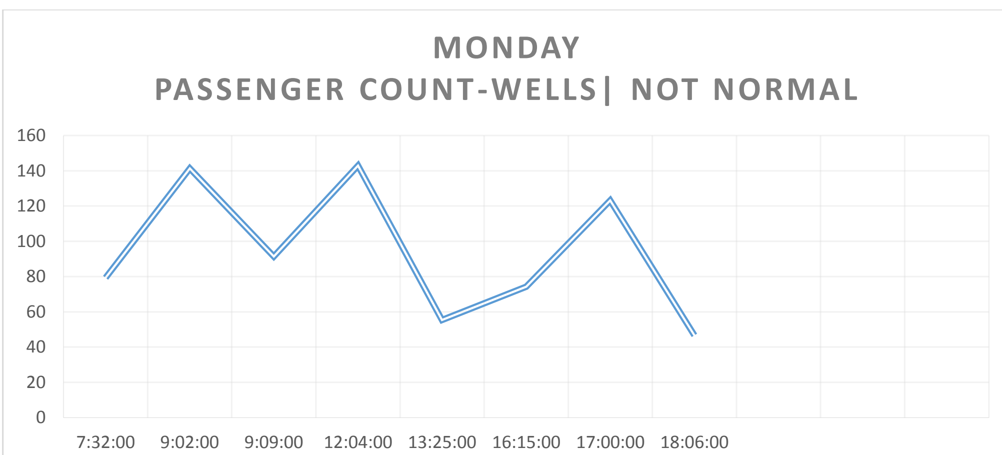


## 5.3 Passenger counts at a stop at different intervals of the day

Route A: Passenger counts at different times at Wells Library during NORMAL weather conditions

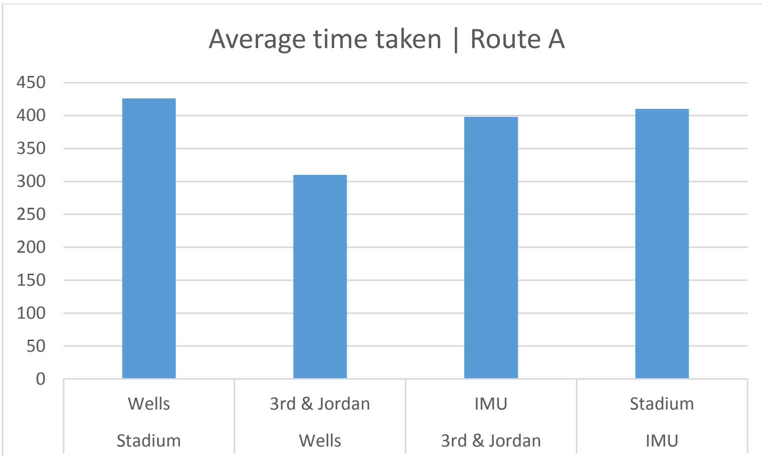



Route A: Passenger counts at different times at Wells Library during NOT NORMAL weather conditions.



## 5.4 Average time taken at different stops

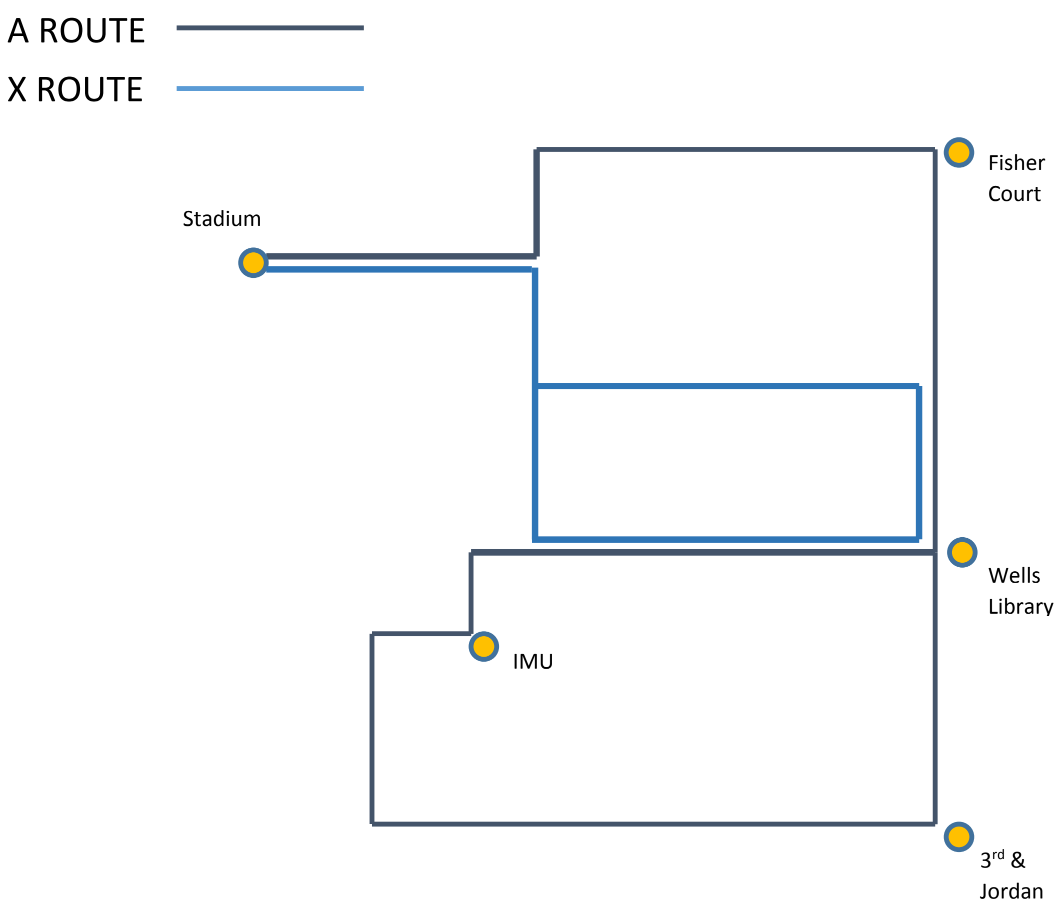
Route A: Average time taken at different stops



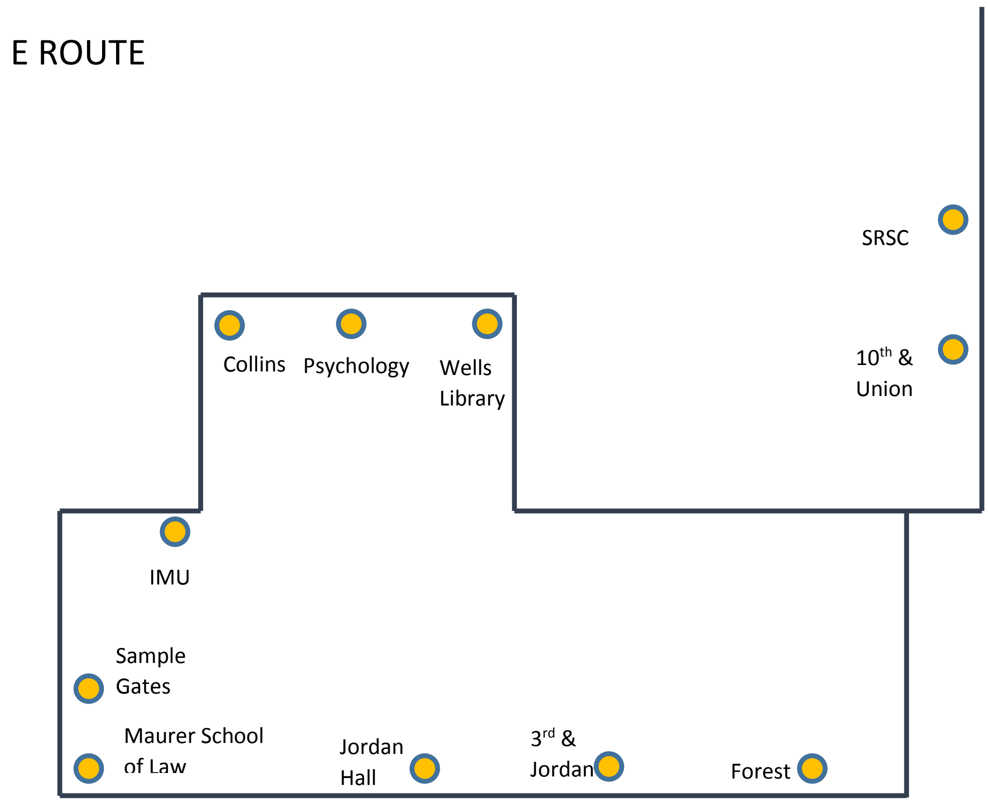
# 6 Proposed Changes

- The proposed system employs an efficient data management system using MySQL and Mongo DB. This is the first step in modelling the bus data.
- Our algorithm resulted in an effective transit system which consists of only 3 routes - A, E and X. New routes with updated stops are shown below -
  - Route A: Stadium –> Fischer court –> Wells Library –> 3rd & Jordan –> IMU –> Stadium
  - Route E: Evermann –> Union & 10th –> Willkie –> 3rd & Jordan –> IMU –> Wells Library –> Evermann
  - Route X: Stadium –> Wells Library –> Stadium
  - Route B: Eliminated
- With our proposed model, there exists only 3 routes and more number of buses for each route can be employed to increase the frequencies of the buses.
- The new route map for A, E and X is as shown below -

### 6.1 Modified route map for A and X



## 6.2 Route map for E



# 7 Concluding remarks

- We have investigated the short comings of the existing system due to factors such as passenger congestion and weather patterns and proposed a new more efficient system.
- The advantages of our system are firstly, the algorithm only requires computation of dwell times and passenger count for updated data, the process of which has been outlined
- So it is not time constrained. Also, the augmented data storage structures make these computations very easy. Furthermore, the mathematical method used does not require any additional information to compute the new routes.
- The proposed Routes have decreased average passenger count per route and leave more buses for the system to employ at all times owing to the elimination of a route.

# 8 References


1. http://www.iubus.indiana.edu/campus_bus/info/New/IUB%20TDM%20Report%20-%20Final.pdf
2. Toshioka, K.; Kawamura, T.; Sugahara, K., "Web Application to Generate Route Bus Timetables," in Internet and Web Applications and Services, 2008. ICIW '08. Third International Conference on , vol., no., pp.109-114, 8-13 June 2008
3. Baizhan Shi; Chunyan Zhao; Yong Zhang, "Route optimization for bus dispatching based on improved ant colony algorithm," in Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on , vol.2, no., pp.807-810, 26-28 Feb. 2010
4. Yikui Mo; Jun Deng; Jingyuan Wang, "An Improvement Route Generation Algorithm for Bus Network Design," in Power Electronics and Intelligent Transportation System, 2008. PEITS '08. Workshop on , vol., no., pp.199-202, 2-3 Aug. 2008