

# An Algorithm for In-Situ Bus Scheduling and Passenger State Update

Yung-Fu Chen and Alan Weide

The Ohio State University, Columbus OH 43221, USA  
`chen.6655@osu.edu`, `weide.3@osu.edu`

## 1 Introduction

Your final project report should include: 1. A write-up describing

(a) the design rationale of the module(s) you developed, along with the algorithm you have designed/used and a detailed API

(b) instructions and illustration of use, in terms of a README file and/or user guide,

(c) how your implementation has been tested,

(d) performance of your implementation,

(e) assumptions or limitations of your design,

(f) any omissions or errors in your implementation, and

(g) any bugs or weaknesses you have found in the algorithm/protocol. (Only if some of these do not apply to your project should you omit them.)

2. Program sources and executable (including any test programs). Credit will be reserved for the quality of documentation of your programs.

3. Script of a session showing the program under test.

### 1.1 Problem Specification

### 1.2 Bus Stop Functions

## 2 Algorithm

### 2.1 Request Assignment Algorithm

## 3 Implementation

### 3.1 .NOW notes vs. Emulator

### 3.2 Software Architecture

### 3.3 Testing and Evaluation

### 3.4 Limitations, Challenges, and Omissions

- Request Generation
- Search for nearby buses: BFS vs. Broadcast
- No time synchronization—instead, use Emulator’s absolute time, which is based on the host’s time, which is the same for all nodes

### **3.5   Using the Smart-Bus Application**

## **4   Conclusion**

## A Source Code