TJHSST Hallway Simulation 2008-2009 (3rd Rough Draft)

Paul Woods

March 25, 2009

Abstract

The project centers primarily around the creation of a traffic simulation–a simulation tailored specifically to the study of movement around Thomas Jefferson High School for Science and Technology. The project taps into several areas of research, yet the ideas and research can be described as concentrated in the following areas:

- **Traffic Modeling**: This project models traffic.
- **Computer Simulation**: This project is a simulation.
- **Multi-agent Systems**: This project studies a system with multiple agents.
- **Group Dynamics.** This project studies a dynamic group.

Keywords: traffic, simulation, agents, agent, agent-based, modeling, agent-based modeling, groups, group, dynamic, dynamics, group dynamics, multi-agent, multi-agent systems

1 Introduction

1.0.1 Problem Statement

With an increased influx of students each year, the group dynamics of TJHSST's hallway traffic have more likely than not been transformed from any past state. Traditionally, TJHSST accepts 400 additional students each year.

Each student remains at school for approximately four years, so the aggregate of all class sizes has tradionally predictably rounded 1600 students.

However, over the years, the world has experienced a demographic shift. The US student population has increased, and with colleges sorting through the largest applicant pool in the history of the United States of America, TJHSST must now accept 450 students–a number that could very well increase in the future.

In 2006, Olesya Katkova and Peter Riggins began the first projects seeking to understand TJHSST's hallways. Yet, while many students welcome the influx of new potential friends, the recent demographic alteration causes what has been often under the radar phenomenon; previous hallway traffic and group dynamics models have become outdated. If you cannot understand the hallways, how can you understand the school?

1.0.2 Project Aim

Many students travel to and fro class several times a day, yet a person can only be in one place at a time. While many people have been at the school for as long as four years, it is difficult for any individual human being to know both what goes on at in the chorus hall at 10:06 and the Computer Systems Hall at 10:07. Yet, computers are not human beings.

The aim of this project is to write a computer program that can achieve a significant understanding of TJHST hallway movement. The project attempts to do this using group dynamics and by accurately simulating the movement of students throughout the day. The simulation will use path finding techniques to determine each student's course, yet combine this with a collective understanding of assembly and clustering that affects the individual decisions of students.

2 Background

Before creating the experiment, extensive background research was conducting. Below is a quick survey of the research utilized in the creation of the simulation.

2.1 Applied Research

- ***Continuum Crowds*** *by Adrian Trueille, Seth Cooper, Zoran Popovic in conjunction with the University of Washington, Electronic Arts.*

  The project studied how to realistically calculate movement without the use of collision detection. The researchers believed crowds tend to have a sense of which areas will be crowded before they reach those areas, and members of this crowd will adjust based on this knowledge when determining his or her path beforehand. The project was able to successfully develop this simulation; in fact, the researchers concluded that the crowds actually moved more smoothly and realistically when calculating the collisions and movement together, as opposed to separately.

- ***Finding Multi-Constrained Feasible Paths By Using Depth-First Search*** *by Zhenjiang L and J.J. Garcia-Luna-Aceves.*

  The focus of this research was developing a depth first search algorithm that operated with an unusual number of restrictions. The algorithm detailed designs for developing routing systems with many constraints.

- ***Interactice Graphics Using OpenGL and the Graphix Windowing Toolkit*** *by Washington and Lee Universitys Rance D. Necaise.*

  The goal of this project was to analyze the Graphics Wnidowing Toolkit, which combines OpenGL and GLUT into an easier interface in order to make graphics easier for create. The result is to determine how to make graphics easier for students who do not have so much time to spend on graphics. The analysis involved primarily testing the new software, and it worked well for what they wanted to accomplish: finding an easier to use alternative to OpenGL and GLUT.

- ***Performance OpenGL: Platform Independence Techniques*** *by Tom True, Brad Grantham, Bob Kuehne, and Dave Shreiner.*

  The paper contained a section with detailed explanations of errors that appear along with OpenGL commands like glTexImage2d().

- ***A New Approach to Cooperative Pathfinding*** *by Renee Jansen and Nathan Sturtevant.*

  This project explains several alternatives to the A* and IDA* searches. It details how to use cooperative algorithms to find paths in a group situation, when calculating each individual path differently would take a long time, and it explains how weighting movement-determining algorithms can help reduce this time significantly.

2.2 Past Research

Past experiments specificic to TJHSST have also been done.

- ***Traffic Dynamics in Scholastic Environments*** *by Olesya Katkova.*

  This program works on finding a way to map traffic in an TJHSST-esque environment.

- ***Hallway Traffic Simulator*** *by Peter Riggins.*

  The program was designed in Fortran. The purpose of this project was to increase the understanding of TJHSST Hallway traffic and expected student movement tendencies. The analysis was that students effectively moved from place to place. The end result was a program that could be expanded and adapted for several situations.

3 Development Sections

The simulation went through several stages of development. This is the story.

3.1 Phase 1 Develpement

In the beginning, the simulation was nothing more than a small scale model. It was not big, but it was a start. Developed in C programming language, the simulation could only model four rooms and two hallways. But it became bigger and more powerful, until the simulation was ready for testing.

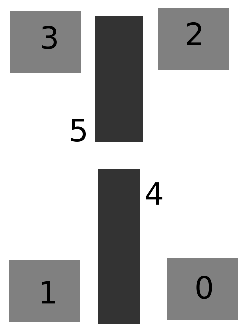

Figure 1: Model mini-building used in simulation phase.

3.2 Phase 1 Testing

The simulation was then tested with the following process:

1. 1) Implementing a preliminary path finding algorithm
2. 2) Outputting the total time of path finding for each hall
3. 3) Analyzing the resulting output

The input the program consisted of the following:

1. 1) A hash containing the information necessary to construct a small-scale model room and hallway layout. The layout constructed using this data is shown in Figure 1.
2. 2) A randomized schedule for each student.
3. 3) Data for initializing input constants,

To test the program in its current state, the simulation started started students in one area and be required to find their way to the designated path. This preliminary simulation assumed the following:

Hall Traffic Given Time:

Hall	10:08	10:11
4	60%	10%
5	40%	60%

Figure 2: Table showing results of Phase 1 Testing.

1. Each room is located approximately three minutes apart from each other.
2. Students will attempt to take the quickest possible path to their destination.
3. Students will stay in their destination once they arrive.
4. Students move at the same speed, regardless of traffic conditions.
5. All students left their initial location at 10:05 and tried to make sure they arrived at their next location by 10:15.

Once the program ran, it output data about where students moved and which hallways were crowded. This data is shown in Figure 2.

Note: At this point, this data was input directly; there was no point in inputting data by reading a file because the map size manageable. The time and processing used to open and read an input file did not make such a method useful.

3.3 Phase 1: Results and Discussion

Every student was able to locate his next period and arrive before the start of the next class (10:05). However, by looking at the data, it was clear that

StudentName	Start	10:08 AM	10:11 AM	Finish
Student0	Area3	Area5	Area2	Area2
Student1	Area1	Area4	Area5	Area3
Student2	Area1	Area4	Area5	Area3
Student3	Area2	Area5	Area4	Area0
Student4	Area1	Area4	Area5	Area2
Student5	Area3	Area5	Area2	Area2

Figure 3: Phase 1 Student Movement Output.

the hallways were not utilized efficiently. For instance, at 10:11, the Hall5 hallway had six times as much traffic as Hall4. Every student was not only able to locate his or her destination, but he or she was able to arrive before there before the start of the next class (10:15).

However, there were some problems. Looking at the data, it was clear that the hallways were not utilized efficiently. For instance, at 10:11, the traffic through Hall5 hallway was 600 percent as much as the congestion through Hall4!

This may have been a result of the fact that students did not take into account traffic when selecting movement. Also, the order of the schools could have been switched so to alleviate the traffic.

The output and results were as expected. The program effectively stores all forms of data, such as student name and hallway connections. These hashes and arrays are all necessary to implement movement algorithms, which will be refined. The Output is shown below.

3.4 Phase 2 Development

Phase two of the program was tested and implemented in the C programming language. The testing process consisted of the following steps:

1. Running the path-finding algorithm algorithm
2. Students will attempt to take the quickest possible path to their destination.
3. Outputting program data to a text file
4. Reviewing the output to test feasibility and realism

Students:
jon
sally
hill
karel
jim
greg
megan
joe
sarah

Figure 4: Phase 2 name input.

This new version of the program runs much more quickly and efficient than the old code.

This new program ads several new components and functions. OpenGL has been added as a critical component. Testing has been performed, and while the the visualization did not work at this stage, OpenGL does print the initial screen and window successfully.

The program could possibly receive information from alien sources in later stages of development, though the program currently receives input from itself throughout the code execution. The current focus is not on efficiency or storage, but rather on functionality. In later stages, the entire school will be mapped out, not just a small previously input sample.

One other key change is that hallways and rooms are now stored separately as different entities, which was not the case with the Phase 1 program.

The input the program receives consists of several things. Some of the most important pieces of data are listed below:

1. Information about hallways

2. Information about rooms

3. Student name data

4. Student schedule data

A sample of the program output is shown on following pages.

RoomNumber:	RoomName	Hall 1	Hall 2	Hall 3	Hall4
0		0	0	0	0
1	r1230	2	3	0	0
2	r1240	1	0	0	0
3	r1250	4	2	0	0

Figure 5: Phase 2 Room/Hall Input.

HallNumber:	Room0	Room1	Room2	Room3	Room4	Room5
0	1	3	0	0	0	0
1	2	0	0	0	0	0
2	1	3	0	0	0	0
3	1	0	0	0	0	0
4	3	0	0	0	0	0

Figure 6: Phase 2 Room/Hall Output Based on Transposed Input.

Student Num	Name	Event0	Event1	Event2	Event3
0		0	0	0	0
1	jon	0	1	2	1
2	sally	0	2	3	2
3	hill	0	2	1	1
4	karel	0	2	3	2
5	jim	0	3	2	3
6	greg	0	2	1	1
7	megan	0	2	2	3
8	joe	0	3	1	1
9	sarah	0	3	3	3

Figure 7: Phase 2 Student Scheduling Data

HallNum	hall0	hall1	hall2	hall3
Hall0:	0	0	0	0
Hall1:	2	4	0	0
Hall2:	1	3	0	0
Hall3:	2	0	0	0
Hall4:	1	0	0	0

Figure 8: Hall Data Input.

Event	Time
0	0
1	905
2	915

Figure 9: Time of Events Data.

3.5 Phase 2 Organization

The new project code is organized differently than the Phase 1 version. Several layers of organization include hallways, rooms, and students. This new organization allows printing functions to be separated from other functions, though functions can still be called as a result of code written near the top of the program. This new organization is necessary, as the code has reached the point where it spans a length of over eight hundred lines. Confusion and coding inefficiency made this new organization necessary.

The new simulation works with regard to student data, hall data, room data, and other forms of crucial data. Most of this data is stored in hashes, and has been tested extensively several times. While these areas of the program work fine, there are still areas that are under construction. One such area is the graphics department.

The code currently has the necessary frameworks for graphics. However, because the graphics do not work. Because data can be analyzed in text form, analyzing and testing the code is still easy, and such processes of testing have been and still are fully functioning. While the first phase code works well with regards to path determination in a very small and limited area, the second phase requires an enhanced path determination algorithm. While such an algorithm is in development and improving, the algorithm falls short of providing the quick, accurate, and realistic student movement patterns the code is intended to render. Because of this, this algorithm is another area that will be worked on extensively.

3.6 Phase 3: Development

For phase 3, the simulation simulated over an electronic version of the TJHSST Hallway Map and the graphics component was put on a total freeze. This map including the hallway numbers used in the simulation is shown. The program was modified to calculate output student path data correctly, and an OpenOffice document was created which imports the data (and can update it as the code outputs new data). The table in this document is included below.

The simulation program works with 800 students (as opposed to 1600) due to memory issues, though this will likely be tackled in a later phase. The table accounts for this by multiplying each value by 2. The OpenOffice spreadsheet graphs teh of the output, which is also included on the following pages. The output was analyzed to make sure it made sense this analysis is included below.

3.7 Phase 3: Results

Top 5 Most crowded hallways in the simulation:

1. 2 (The 2nd floor intersection of hallways near the backside of the school. This is the location of a 4-way intersection (hallways in three direction and stairs leading to the 1st floor).

2. 14 (The 1st floor intersection near the front of the school.)

3. 22 (The Junior Lounge, which intersects 3 hallways, an exit, and a pathway leading to trailers).

4. 23 (The 1st floor hallway in which many physics classes are taken)

5. 16 (The hallway leading to the Millennium Courtyard).

Top 5 Least crowded hallways in the simulation:

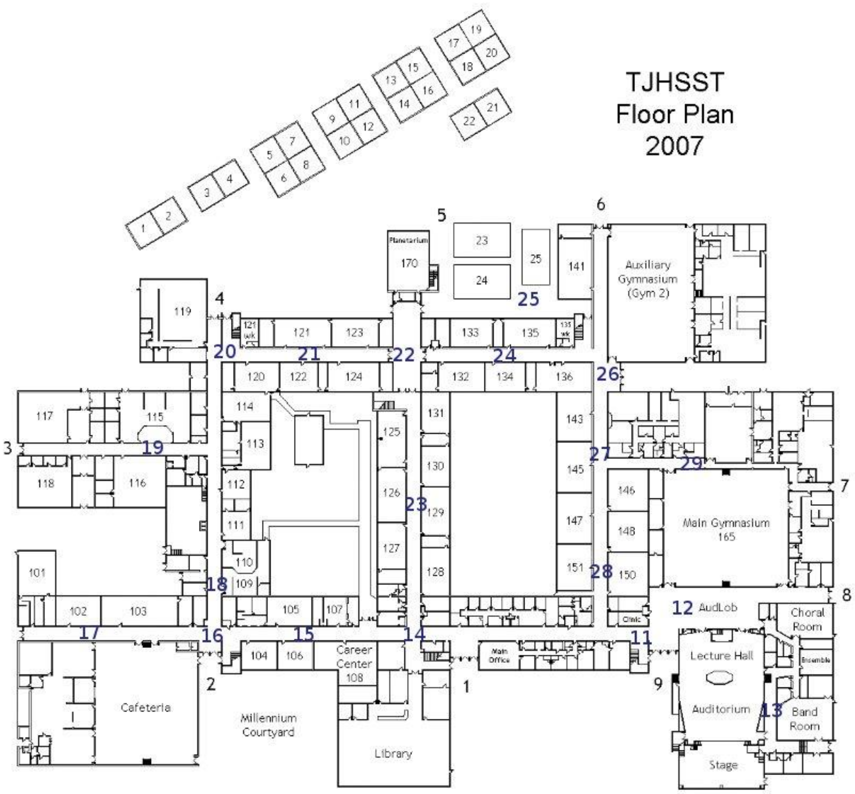

Figure 10: Model mini-building used in simulation phase.

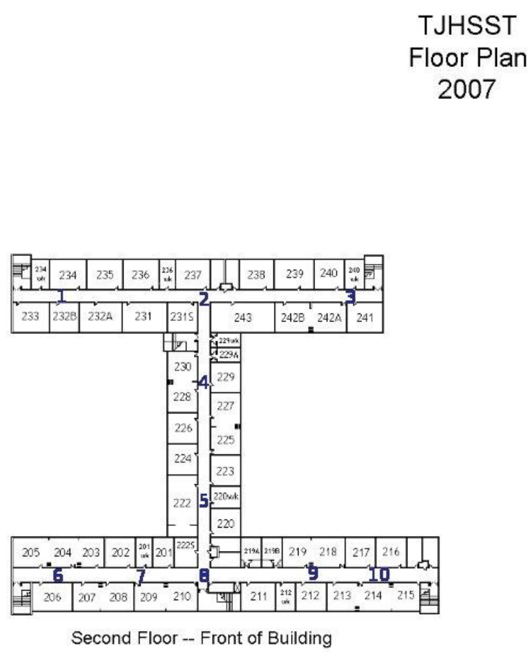

Figure 11: Table showing results of initial experiment.

1. 29 (The hallway leading to one of the entrances to the boys locker room and the back door entrance to the main gym).
2. 25 (The pathway to the section of trailers located between Dr. Acios room and the Junior lounge).
3. 17 (The hallway leading to and fro the cafeteria).
4. 13 (The hallway leading to the band and chorus rooms).
5. 19 (The hallway leading to the Systems Lab).

3.8 Phase 3: Analysis

Most of the hallways identified the two Top 5 categories matched personal experiences from real life. However, two in particular (hallways 17 and 16) do not match individual experiences. The reason why the movement through hall 17 is underestimated is likely that the program treats each room equally when determining student schedules, and it also does not take into account the beginning and end of the day. Since one of the primary reasons for moving through hallway 17 is going to and fro buses and cars and going toward the cafeteria and senior lounge (whose sizes far exceed that of the average classroom), the simulation underestimates the traffic. To fix this, the simulation can take into account class sizes when determing student schedules.

The simulation likely overestimates the traffic through hall 16. Though, personal experiences do not match real life, a real-life test can be performed to be sure of this. The reason, though, that it may underestimate the traffic is by overestimating the level of traffic to to rooms that are often reached through hall 16 (Rooms 101, 102, 103, 109, 110, 111, 112, 104, 105, 106, and Millennium Courtyard Exit). In addition, one major flaw of the current program is that it treats bathrooms as classrooms, so hallway 16 treats the girls and boys bathrooms as a single 30-student room in which students take classes in. This use, however, never has and hopefully never will match real-life.

While the simulation underestimates traffic to the cafeteria, which must also be reached through hall 16), the overestimates of the other rooms more likely than not would balance this out, especially since the Cafeteria and

Time (minutes) / Hall Number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29		
0	82	26	94	92	44	74	50	16	82	34	24	48	70	48	54	24	68	92	42	24	94	38	112	80	42	40	18	66	20	1598	
1	22	98	18	24	52	48	30	82	16	60	62	68	18	108	4	196	0	42	36	84	6	144	34	6	6	142	56	46	18	1526	
2	88	102	132	56	12	146	0	104	50	22	104	18	34	80	30	62	0	62	16	14	42	16	110	12	26	60	24	54	14	1490	
3	42	136	30	54	28	56	16	60	50	40	22	34	16	192	4	140	0	22	42	14	14	72	64	8	6	146	38	80	26	1452	
4	40	124	140	88	2	74	14	70	94	18	90	16	14	58	32	52	0	80	10	6	14	78	152	8	12	46	24	50	16	1422	
5	56	182	28	22	24	18	12	100	38	52	56	14	2	204	20	84	0	24	4	60	14	122	92	2	4	110	32	24	10	1410	
6	82	102	76	82	28	32	12	38	78	42	48	2	4	164	70	10	0	10	2	60	36	94	204	38	0	36	18	32	6	1406	
7	70	162	64	22	62	8	14	86	24	42	84	4	0	172	76	44	0	40	2	42	42	172	64	14	6	34	16	36	0	1402	
8	22	214	20	22	20	6	22	150	26	26	144	46	2	74	10	118	0	70	0	76	12	174	14	16	2	62	38	16	0	1402	
9	84	36	86	72	40	42	52	20	76	50	28	62	46	32	38	10	68	84	60	32	58	22	62	76	32	42	14	68	8	1400	
Sum	588	1182	688	534	312	504	222	726	534	386	662	312	206	1132	338	740	136	526	214	412	332	932	908	260	136	718	278	472	118		14508
																														14508	

Figure 12: Table showing results of experiment.

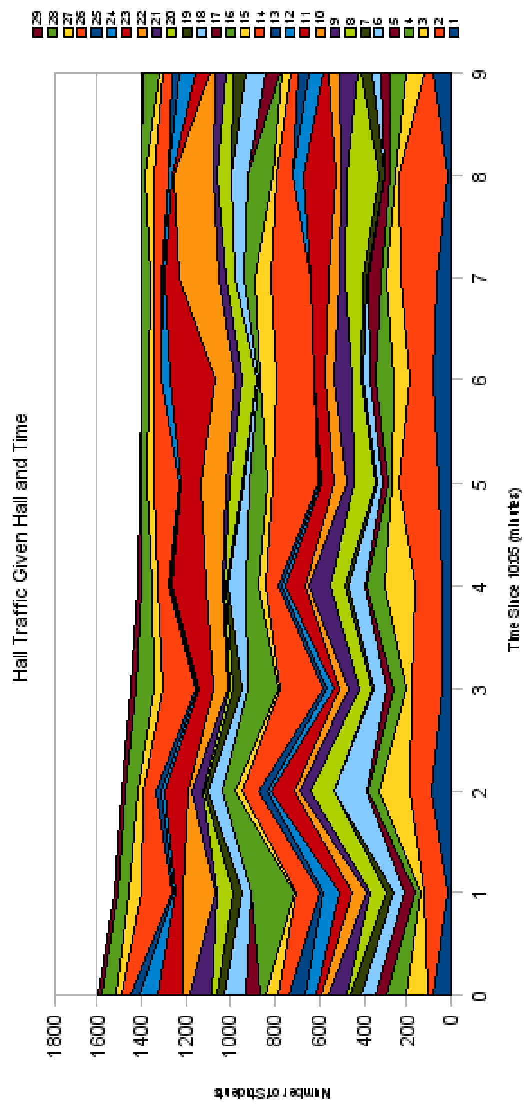

Figure 13: Graph of results.

Senior Lounge are only goals during select periods of the day. To fix these models, the next simulation must contain student class sizes and more realistic scheduling, eliminating the treatment of bathrooms, exits, and gyms as equals to normal classrooms.

On top of that, while the number of students in the hallway decreases over time (as evidenced by the graph), the decrease likely does not mimic real-life. This is because the program only requires that students find a path, but not necessarily the most realistic path, to their classes. The only constraint is that they must reach their destination in time. The model also does not account that students will go out of their way to reach certain destinations, linger in a certain hall talking to their friends, or leave class early and arrive to class late. Accounting for this will more likely than not increase the accuracy of the results.

4 The Future

The following are goals for the fourth quarter:

1. Fix the student scheduling algorithm to remove bathrooms and exits as classrooms.

2. Give each room a class-size attribute to improve the accuracy of the results.

3. Adjust the path finding algorithm to make students take past crowding experiences into account.

4. Compare the program output with data collected from real-life.

5. Perfect the path finding algorithm and set up real-life scenerios, time permitting?

No one knows what the future may hold. Yet, more likely than not, progress will be made. Olesya Katkova and Peter Riggins are long gone, and this project cannot continue forever. Yet, while this particular project may be nearing its close, another project may very well take its place. And it just may be the case that the students of TJ will never stop trying to understand the hallways.

References

[1] Olivia Kotkova. "Traffic Dynamics in Scholastic Environments"

[2] Peter Riggins. "Hallway Traffic Simulator"

[3] Adrian Trueille, Seth Cooper, Zoran Popovi "Continuum Crowds"

[4] Zhenjiang L. J.J, Garcia-Luna-Aceves. "Finding Multi-Constrained Feasible Paths By Using Depth-First Search"

[5] Tom True, Brad Grantham, Bob Kuehne, and Dave Shreiner. "Performance OpenGL: Platform Independence Techniques"

[6] Rance D. Necaise. Computer Science Department Washington and Lee University. "Interactive Graphics Using OpenGL and the Graphix Windowing Toolkit"

[7] Encarta. "Group Dynamics"

[8] CreatingMinds. "Creative Quotes and Quotations On Dreaming"

[9] Zhenjiang L. J.J, Renee Jansen, Nathan Sturtevant. "A New Approach to Cooperative Pathfinding"

Special thanks to Mr. Latimer, Mr. Torbert, Chris Beachem, Masood Malekghassemi, Joseph Lattin, Avnish Samraat Goel and others who have helping with the project, whether by reading papers, helping coding, or helping preparing presentations.