For office use only	Team Control Number	For office use only
T1 _____	**0000**	F1 _____
T2 _____		F2 _____
T3 _____	Problem Chosen	F3 _____
T4 _____	**B**	F4 _____

2017
MCM/ICM
Summary Sheet

The LaTeX Template for MCM Version v6.2.1

Summary

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Keywords: keyword1; keyword2

Contents

1 Introduction

1.1 Background

Lewis Mumford, a famous sociologist and literary critic, once said in a metaphorical manner, "Adding highway lanes to deal with traffic congestion is like loosening your belt to cure obesity." Fortunately, he did not experience the worse congestion around today's highway toll plaza.

Currently, with roaring number of vehicles, rising construction costs and constrained available areas, traffic jam becomes more and more serious but future toll-plaza construction opportunities are limited to improve this situation markedly. Figure 1 shows the congestion in the toll plaza near Tappan Zee Bridge.

Figure 1: Toll plaza congestion

Subject to the constraints referred above, neither increasing highway lanes nor building more tollbooths seems practical enough to relieve traffic jam around a toll plaza nowadays, particularly for some heavily-traveled roads such as the Garden State Parkway, New Jersey. Therefore, looking for some innovative design improvements on the geometric parameters of the extent toll plaza is an effective solution.

1.2 Restatement of the Problem

In this paper, we are required to explore if there is a better-than-ever toll plaza model with specific shape, size, and merging pattern. In this model, the prerequisite is that vehicles fan in from B tollbooth egress lanes down to L (B>L) lanes of traffic (i.e., the number of both tollbooths and the lanes after merging are fixed).

We aim to construct a model that can optimize the arrangement according to the following conditions.

- Enhance the capability of the accident prevention(A).
- Maximize the throughput(T).
- Minimize the cost of the land and road construction(C).

Through our analysis, we determine if there are better solutions than any toll plaza in common use. Afterwards, the performance of our solution in light and heavy traffic and other various situations along with corresponding sensitivity analysis is discussed.

References

[1] D. E. KNUTH The TEXbook the American Mathematical Society and Addison-Wesley Publishing Company , 1984-1986.

[2] Lamport, Leslie, LaTeX: " A Document Preparation System ", Addison-Wesley Publishing Company, 1986.

Appendices

Appendix A First appendix

Here are simulation programmes we used in our model as follow.

Input matlab source:

```
function [t,seat,aisle]=OI6Sim(n,target,seated)
pab=rand(1,n);
for i=1:n
    if pab(i)<0.4
       aisleTime(i)=0;
    else
        aisleTime(i)=trirnd(3.2,7.1,38.7);
    end
end
```

Appendix B Second appendix

some more text **Input C++ source:**

```
//============================================================================
// Name        : Sudoku.cpp
// Author      : wzlf11
// Version     : a.0
// Copyright   : Your copyright notice
// Description : Sudoku in C++.
//============================================================================

#include <iostream>
#include <cstdlib>
#include <ctime>

using namespace std;

int table[9][9];

int main() {

    for(int i = 0; i < 9; i++){
        table[0][i] = i + 1;
    }
```

```
    srand((unsigned int)time(NULL));

    shuffle((int *)&table[0], 9);

    while(!put_line(1))
    {
        shuffle((int *)&table[0], 9);
    }

    for(int x = 0; x < 9; x++){
        for(int y = 0; y < 9; y++){
            cout << table[x][y] << " ";
        }

        cout << endl;
    }

    return 0;
}
```