Analysis of joint trips using C++ in Mobidrive

Pranay Singhi
Analysis of joint trips using C++ in mobidrive

Pranay Singhi
IVT
ETH
CH-8093 Zürich

Telefon: +41-1-633 79 47
Telefax: +41-1-633 10 57
eMail: n8004022@ccs.iitb.ernet.in, axhausen@ivt.baug.ethz.ch

October 2001

Abstract

Models used for the analysis and prediction of urban passenger travel have been based primarily on analysis of choices relative of individual’s trips. Interactions among individuals in a household, and the effect of such interactions on individual activity patterns, is an area that has received limited attention thus far in the activity analysis literature. The German research project Mobidrive, funded by the Federal Ministry of Education and Research, implemented a six-week continuous travel diary, which now yields a unique database for analysing joint trips. The paper here describes the program in C++, which was used for detecting the joint trips within the database.

Keywords

Joint trips analysis, C++ programming, Mobidrive, ETH Zürich – Institut für Verkehrsplanung und Transporttechnik, Strassen- und Eisenbahnbau (IVT)

Preferred citation

1. Introduction

The German research project MobiDrive is an ambitious study to observe and analyse the structure as well as the determinants of temporal aspects in individual travel behaviour. The analysis is mainly based on a six-week travel survey, which was conducted to observe the longitudinal phenomena of behaviour of travel, such as rhythms in the behaviour of the respondents (Axhausen, Zimmermann, Schönfelder, Rindsfüser and Haupt, 2000; PTV AG, Fell, Schönfelder and Axhausen, 2000; Schlich, König, Aschwanden, Kaufmann and Axhausen, 2000).

The project MobiDrive, funded by the German ministry of Research and Education, conducted a six-week continuous travel diary with the aim to analyse the rhythms in the behaviour of the respondents. The survey was performed in the German cities of Karlsruhe and Halle both with about 270,000 inhabitants, in the fall of 1999. A total of 317 persons over 6 years of age in 139 households participated in the main study (Axhausen et al., 2000).

The working paper describes the method used to identify the joint trips made by persons in the same household. The joint trips are analysed using a program written in the C++ programming language. The paper describes the algorithm of the program, the input and the output files, which are used and the variables employed in the program.

2. Joint Trips

Models used for the analysis and prediction of urban passenger travel have been based primarily on analysis of choices (mode, destination, time of travel, etc.) of individuals independent of trip motivation and interactions among multiple travellers.

Interactions among individuals in a household, and the effect of such interactions on individual activity patterns, is an area that has received limited attention thus far in the activity analysis literature. Interactions among individuals might take the form of joint participation in certain activities, such as shopping together or engaging in recreational and social activities together. An example of this would be “serve-passenger” and “escort” activities, in which one
individual oversees the participation of another in activities. Such interactions can lead to
constraints or inter-individual reallocation of tasks that may be very important in individual
activity and travel responses to changes in the transportation or land-use environment.

Joint trips are defined as trips, which are made jointly by individuals in the same household.
The number of people taking part in a joint trip can vary from two to the maximum number of
people in the household.

3. C++ Program Description

The analysis of joint trips in the Mobidrive data was done using C++ as a programming tool
(Lafore, 1991). The tripsum file of the Mobidrive data was used as the primary source refer-
ence for the analysis.

3.1 Input file description:

The input file used was of .dat format and provided the following variables (Axhausen et al.,
2000) (See Figure 1):

1. Cityc: the city code of Halle or Karlsruhe.
2. Studyc: the study code for Main study or the Pre test study.
3. Hhnr: the household number of the person taking the trip.
4. Pnr: the person number in the household.
5. Tdep: the time of departure of the trip taken by the person.
6. Tarr: the time of arrival of the trip taken by the person.
7. Dorr: the day of reporting period of the person taking the trip.
8. Tmmbs: the main mode used in the trip by speed.
Figure 1 Input file

<table>
<thead>
<tr>
<th>Cityc</th>
<th>studyc</th>
<th>hhnr</th>
<th>pnr</th>
<th>tdep</th>
<th>tarr</th>
<th>dorp</th>
<th>tmmb</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>26100</td>
<td>27600</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>48600</td>
<td>50400</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>54000</td>
<td>56100</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>61200</td>
<td>63300</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>26400</td>
<td>27900</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>47400</td>
<td>48900</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>52200</td>
<td>53100</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>61200</td>
<td>62400</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>29700</td>
<td>30600</td>
<td>3</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>33300</td>
<td>34200</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>36000</td>
<td>39600</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>61200</td>
<td>64800</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>34200</td>
<td>35100</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>42300</td>
<td>43200</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>50400</td>
<td>54000</td>
<td>4</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>68400</td>
<td>72000</td>
<td>4</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>42300</td>
<td>43200</td>
<td>6</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>45900</td>
<td>46800</td>
<td>6</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>54000</td>
<td>55200</td>
<td>7</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>64800</td>
<td>66000</td>
<td>7</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>54000</td>
<td>56100</td>
<td>1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>61200</td>
<td>63300</td>
<td>1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>52200</td>
<td>53100</td>
<td>2</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>61200</td>
<td>62400</td>
<td>2</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>34200</td>
<td>35100</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>42300</td>
<td>43200</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>50400</td>
<td>54000</td>
<td>4</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>68400</td>
<td>72000</td>
<td>4</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>41400</td>
<td>43200</td>
<td>5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>46800</td>
<td>48600</td>
<td>5</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>42300</td>
<td>43200</td>
<td>6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>45900</td>
<td>46800</td>
<td>6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>54000</td>
<td>55200</td>
<td>7</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>64800</td>
<td>66000</td>
<td>7</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

3.2 Algorithm of the program

The algorithm of the program proceeds as follows:
The program reads the input data from the input file as described before.

The program creates two classes by the name info and set. The class info has the information about a particular trip from the t_sum file which includes the study code, city code, household number, person number, day of reporting, departure and arrival times of the trip, trip main mode by speed. The info class also has objects for indicating whether the trip is joint trip, number of persons in the trip and the person numbers. The class set identifies the joint trip and stores the location of the position of the two trips in the array in the objects x and y. The set also stores the persons taking part in the trip by their person number.

The program then identifies the plausible joint trips by using the criteria that two trips are a joint trip if they have the same study code, same city code, same household number, same day of reporting, same trip main mode by speed and having different in the trips. The essential criteria after all this satisfied is that the difference in arrival times and departures times should be less than five minutes as reported by the person. The interval was chosen after trying for intervals of zero, five minutes, ten minutes. The maximum number of joint trips were identified with the interval of five minutes.

The program then merges the joint trips analysed by the above criteria. For example if person 1 and person 2 have a joint trip and at the same time person 2 and person 3 have a joint trip then this means that all the three have made a joint trip. The program merges the joint trips for five, four and three persons. The process of merging the trips for four and five persons is more complicated. The logic used is that if each of them has either made a joint trip with the any of the remaining using the above criteria then it is a joint trip by all the members.

After merging the trips now the array of trip has the person number of the people taking part in the joint trip in the objects per, per2, per3, per4, per5 respectively storing the person numbers. The default value of zero is stored in case that is not a join trip.

The output of the program is stored in a .txt format as described in the next section. The output is actually added to the array of trips with extra variables to indicate whether it is a joint trip or not, the total number of persons taking part in the joint trip, the person number of each individual in the joint trip and the corrected departure and arrival times taking into consideration the effect of the joint trip.

3.3 Description of the output file

The output file is of the form of .txt file and it has the following variables in it (See Figure 2):

1. no: which denotes the trip number
2. studyc: the study code of Main study or the Pre test study.
3. cityc: the city code of Halle or Karlshrue.
4. hhnr: the household number of the person taking the trip.
5. pnr: the person number in the household.
6. jtrip: is an indicator of whether it is a joint trip or not. 1 represents that the trip is a joint trip whereas 0 represents that the trip is not a joint trip.
7. nop: the total number of persons taking part in the joint trip.
8. p1: indicates whether the person number 1 in the household is taking part in the joint trip or not. 1 indicates he takes part in the joint trip whereas 0 indicates that he is not in the joint trip.
9. p2: indicates whether the person number 2 in the household is taking part in the joint trip or not. 1 indicates he takes part in the joint trip whereas 0 indicates that he is not in the joint trip.
10. p3: indicates whether the person number 3 in the household is taking part in the joint trip or not. 1 indicates he takes part in the joint trip whereas 0 indicates that he is not in the joint trip.
11. p4: indicates whether the person number 4 in the household is taking part in the joint trip or not. 1 indicates he takes part in the joint trip whereas 0 indicates that he is not in the joint trip.
12. p5: indicates whether the person number 5 in the household is taking part in the joint trip or not. 1 indicates he takes part in the joint trip whereas 0 indicates that he is not in the joint trip.
13. Tdep: gives us the corrected time of the departure of the each individual in the trip. In case of a joint trip the departure times are corrected by considering the mean of the departure times of all the individuals taking part in the joint trip.
14. Tarr: gives us the corrected time of the arrival of the each individual in the trip. In case of a joint trip the departure times are corrected by considering the mean of the arrival times of all the individuals taking part in the joint trip.
### 4. Results

The program was used to identify joint trips from the t_sum file and the total number of joint trips identified was 11517 out of the 52273 trips made. The distribution of the joint trips with
respect to the number of people taking part in the joint trips is shown in the Table 1 and Figure 3:

Table 1: Frequency distribution of trips with respect to the number of persons

<table>
<thead>
<tr>
<th>Number of persons</th>
<th>Frequency</th>
<th>Share [%]</th>
<th>Cumulative percent [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40756</td>
<td>78.00</td>
<td>78.00</td>
</tr>
<tr>
<td>2</td>
<td>9074</td>
<td>17.40</td>
<td>95.30</td>
</tr>
<tr>
<td>3</td>
<td>1806</td>
<td>3.50</td>
<td>98.80</td>
</tr>
<tr>
<td>4</td>
<td>592</td>
<td>1.10</td>
<td>99.90</td>
</tr>
<tr>
<td>5</td>
<td>45</td>
<td>0.10</td>
<td>100.00</td>
</tr>
<tr>
<td>Total</td>
<td>52273</td>
<td>100.00</td>
<td>100.00</td>
</tr>
</tbody>
</table>

Figure 3 Number of persons participating

5 Acknowledgements

The author gratefully acknowledges the support of Mr. K. W. Axhausen, Mr. S Schönfelder, and Mr. A König for their constant guidance and help in analysing and understanding the data and preparing the report.
6 Literature


Appendix A: Source Code of the program

// program to identify joint trips for household upto five persons
#include <iomanip.h>       // header files
#include <iostream.h>
#include <fstream.h>
#include <math.h>
class info               // class of info which has the information
{                        // of the trip made from the input file
    public:
        int hhnr;
        long tarr;
        long tdep;
        int pnr;
        int dorp;
        int tmmbn;
        int jtrip;
        int cityc;
        int studyc;
        int nop;
        int per1;
        int per2;
        int per3;
        int per4;
        int per5;
    }
const int MAX = 186;     // MAX is the size of the number of trips
struct set               // structure which is temporarily used in
                        // the program to identify joint trips.
{
    int x;
    int y;
    int per;
    int per2;
    int per3;
    int per4;
    int per5;
}

void main()              // main program
{
    info arr[MAX];      // creating array of class info
    int i,j,k,h, per2, per3, per4, per5,tper, tdep, tarr;
    set trip[MAX],a,b,c,d,e,f;

    for(i=0;i<MAX;i++)   // initializing the array
    {
        arr[i].hhnr=0;

arr[i].pnr=0;
arr[i].tarr=0;
arr[i].tdep=0;
arr[i].dorp=0;
arr[i].tmmbs=0;
arr[i].cityc=0;
arr[i].jtrip=0;
arr[i].studyc=0;
arr[i].nop=1;
arr[i].per1=0;
arr[i].per2=0;
arr[i].per3=0;
arr[i].per4=0;
arr[i].per5=0;
trip[i].x=0;
trip[i].y=0;
trip[i].per=0;
trip[i].per2=0;
trip[i].per3=0;
trip[i].per4=0;
trip[i].per5=0;
}
k=0;
ifstream inFile; // file input output
ofstream outFile;
inFile.open("hh50.dat",ios::in); // reading the file

for(i=0;i<MAX;i++)
{
    inFile >> arr[i].cityc >> arr[i].studyc >> arr[i].hhnr >> arr[i].pnr >>
    arr[i].tdep >> arr[i].tarr >> arr[i].dorp >>
    arr[i].tmmbs ;
    cout << arr[i].hhnr << " " << arr[i].pnr << " " <<arr[i].tdep <<"
" << arr[i].tarr << " " << arr[i].dorp << "\n" ;
}
for(i=0;i<MAX;i++) // for converting passenger and driver as same
{
    if(arr[i].tmmbs==5)
    {
        arr[i].tmmbs=4;
    }
}
for(i=0;i<MAX;i++) // conditions to be satisfied for joint trip
{
    for(j=i;j<MAX;j++)
    {
        if(arr[i].studyc==arr[j].studyc) // same study code
            if(arr[i].cityc==arr[j].city) // same city
                if (arr[i].hhnr==arr[j].hhnr) // same household
if (arr[i].dorp==arr[j].dorp) // same day of
    // reporting period
if(arr[i].pnr!=arr[j].pnr) // different persons
if(arr[i].tmmbs==arr[j].tmmbs) // same mode
if((abs(arr[i].tdep-arr[j].tdep)<300)&&(abs(arr[i].tarr-arr[j].tarr)<300))
    // difference in departure and arrival times of 5 min
    {  
        trip[k].x=i;
        trip[k].y=j;
        trip[k].per=arr[i].pnr;
        trip[k].per2=arr[j].pnr;
        k++;
    }
    // plausible joint trips stored in trip set
for(i=0;i<MAX;i++) // converting joint trips for three
    // persons
{
    if(trip[i].y!=0)
    {
        if(trip[i].x==trip[i+1].x)
        {
            a.x=trip[i].y;
            a.y=trip[i+1].y;
            for(h=i;h<MAX;h++)
            {
                if(a.x==trip[h].x && a.y==trip[h].y)
                {
                    trip[i].per3=arr[trip[h].y].pnr;
                    trip[i+1].per3=-1;
                    trip[h].per3=-1;
                }
            }
        }
    }
}
for(i=0;i<MAX;i++) // converting joint trips for four persons......
{
    if(trip[i].x==trip[i+1].x && trip[i].x==trip[i+2].x)
    {
        a.x=trip[i].y;
        a.y=trip[i+1].y;
        b.x=trip[i].y;
        b.y=trip[i+2].y;
        c.x=trip[i+1].y;
        c.y=trip[i+2].y;
        for(h=i;h<MAX;h++)
        {
            if(a.x==trip[h].x && a.y==trip[h].y)
            {
                per2=2;
                per3=3;
for(i=0;i<MAX;i++) // converting joint trips for five persons
{
    if(trip[i].x==trip[i+1].x&&trip[i].x==trip[i+2].x&&trip[i].x==trip[i+3].x)
    {
        a.x=trip[i].y;
        a.y=trip[i+1].y;
        b.x=trip[i].y;
        b.y=trip[i+2].y;
    }
}
Analysis of joint trips using C++ in Mobidrive

```
c.x=trip[i].y;
c.y=trip[i+3].y;
d.x=trip[i+1].y;
d.y=trip[i+2].y;
e.x=trip[i+1].y;
e.y=trip[i+3].y;
f.x=trip[i+2].y;
f.y=trip[i+3].y;

for(h=i;h<MAX;h++)
{
    if(a.x==trip[h].x && a.y==trip[h].y)
    {
        pper2=2;
        pper3=3;
        trip[i+1].per4=-1;
        trip[i+1].per3=-1;
        trip[i+1].per5=-1;
        trip[h].per4=-1;
        trip[h].per3=-1;
        trip[h].per5=-1;
    }
    if(b.x==trip[h].x && b.y==trip[h].y)
    {
        pper2=2;
        pper4=4;
        trip[i+2].per4=-1;
        trip[i+2].per3=-1;
        trip[i+2].per5=-1;
        trip[h].per4=-1;
        trip[h].per3=-1;
        trip[h].per5=-1;
    }
    if(c.x==trip[h].x && c.y==trip[h].y)
    {
        pper2=2;
        pper5=5;
        trip[i+3].per4=-1;
        trip[i+3].per3=-1;
        trip[i+3].per5=-1;
        trip[h].per4=-1;
        trip[h].per3=-1;
        trip[h].per5=-1;
    }
    if(d.x==trip[h].x && d.y==trip[h].y)
    {
        pper3=3;
        pper4=4;
        trip[h].per4=-1;
        trip[h].per3=-1;
        trip[h].per5=-1;
        trip[i].per2=0;
    }
    if(e.x==trip[h].x && e.y==trip[h].y)
    {
```
pper3=3;  
pper5=5;  
trip[h].per4=-1;  
trip[h].per3=-1;  
trip[h].per5=-1;  
trip[i].per2=0;  
}  
if(f.x==trip[h].x && f.y==trip[h].y)  
{  
pper4=4;  
pper5=5;  
tper=arr[trip[h].y].pnr;  
trip[h].per4=-1;  
trip[h].per3=-1;  
trip[h].per5=-1;  
trip[i].per2=0;  
trip[i].per3=0;  
}  
}

if(pper2==2 && pper3==3 && pper4==4 && pper5==5) // checks if it is a joint trip for five persons  
{  
trip[i].per5=tper;  
trip[i].per2=2;  
trip[i].per3=3;  
trip[i+1].per4=-1;  
trip[i+1].per3=-1;  
trip[i+1].per5=-1;  
trip[i+2].per4=-1;  
trip[i+2].per3=-1;  
trip[i+2].per5=-1;  
trip[i+3].per4=-1;  
trip[i+3].per3=-1;  
trip[i+3].per5=-1;  
}  
}

cout << "\n";  
for(i=0;i<MAX;i++)  
{  
if (trip[i].per !=0 && trip[i].per2 !=0) // to identify whether it //it is a joint trip or not  
{  
arr[trip[i].x].jtrip=1;  
arr[trip[i].y].jtrip=1;  
if(trip[i].per5>0) // output for joint trip // of five persons  
{  
}
\[ tdep = \frac{\text{arr}[\text{trip}[i].x].tdep + \text{arr}[\text{trip}[i].y].tdep + \text{arr}[\text{trip}[i+1].y].tdep + \text{arr}[\text{trip}[i+2].y].tdep + \text{arr}[\text{trip}[i+3].y].tdep}{5}; \]

\[ tarr = \frac{\text{arr}[\text{trip}[i].x].tarr + \text{arr}[\text{trip}[i].y].tarr + \text{arr}[\text{trip}[i+1].y].tarr + \text{arr}[\text{trip}[i+2].y].tarr + \text{arr}[\text{trip}[i+3].y].tdep}{5}; \]

\[ \text{arr}[\text{trip}[i].x].\text{nop}=5; \text{arr}[\text{trip}[i].x].\text{per1}=1; \]
\[ \text{arr}[\text{trip}[i].x].\text{per2}=1; \text{arr}[\text{trip}[i].x].\text{per3}=1; \text{arr}[\text{trip}[i].x].\text{per4}=1; \]
\[ \text{arr}[\text{trip}[i].x].\text{per5}=1; \text{arr}[\text{trip}[i].x].\text{tdep}=tdep; \text{arr}[\text{trip}[i].x].\text{tarr}=tarr; \]

\[ \text{arr}[\text{trip}[i].y].\text{nop}=5; \text{arr}[\text{trip}[i].y].\text{per1}=1; \]
\[ \text{arr}[\text{trip}[i].y].\text{per2}=1; \text{arr}[\text{trip}[i].y].\text{per3}=1; \text{arr}[\text{trip}[i].y].\text{per4}=1; \]
\[ \text{arr}[\text{trip}[i].y].\text{per5}=1; \text{arr}[\text{trip}[i].y].\text{tdep}=tdep; \text{arr}[\text{trip}[i].y].\text{tarr}=tarr; \]

\[ \text{arr}[\text{trip}[i+1].y].\text{nop}=5; \text{arr}[\text{trip}[i+1].y].\text{per1}=1; \]
\[ \text{arr}[\text{trip}[i+1].y].\text{per2}=1; \text{arr}[\text{trip}[i+1].y].\text{per3}=1; \text{arr}[\text{trip}[i+1].y].\text{per4}=1; \]
\[ \text{arr}[\text{trip}[i+1].y].\text{per5}=1; \text{arr}[\text{trip}[i+1].y].\text{tdep}=tdep; \text{arr}[\text{trip}[i+1].y].\text{tarr}=tarr; \]

\[ \text{arr}[\text{trip}[i+2].y].\text{nop}=5; \text{arr}[\text{trip}[i+2].y].\text{per1}=1; \]
\[ \text{arr}[\text{trip}[i+2].y].\text{per2}=1; \text{arr}[\text{trip}[i+2].y].\text{per3}=1; \text{arr}[\text{trip}[i+2].y].\text{per4}=1; \]
\[ \text{arr}[\text{trip}[i+2].y].\text{per5}=1; \text{arr}[\text{trip}[i+2].y].\text{tdep}=tdep; \text{arr}[\text{trip}[i+2].y].\text{tarr}=tarr; \]

\[ \text{arr}[\text{trip}[i+3].y].\text{nop}=5; \text{arr}[\text{trip}[i+3].y].\text{per1}=1; \]
\[ \text{arr}[\text{trip}[i+3].y].\text{per2}=1; \text{arr}[\text{trip}[i+3].y].\text{per3}=1; \text{arr}[\text{trip}[i+3].y].\text{per4}=1; \]
\[ \text{arr}[\text{trip}[i+3].y].\text{per5}=1; \text{arr}[\text{trip}[i+3].y].\text{tdep}=tdep; \text{arr}[\text{trip}[i+3].y].\text{tarr}=tarr; \]

\}

else
{
\{
\text{tdep}=(\text{arr}[\text{trip}[i].x].\text{tdep} + \text{arr}[\text{trip}[i].y].\text{tdep} + \text{arr}[\text{trip}[i+1].y].\text{tdep} + \text{arr}[\text{trip}[i+2].y].\text{tdep} + \text{arr}[\text{trip}[i+3].y].\text{tdep})/4;
\text{tarr}=(\text{arr}[\text{trip}[i].x].\text{tarr} + \text{arr}[\text{trip}[i].y].\text{tarr} + \text{arr}[\text{trip}[i+1].y].\text{tarr} + \text{arr}[\text{trip}[i+2].y].\text{tarr})/4;
\text{arr}[\text{trip}[i].x].\text{nop}=4; \text{arr}[\text{trip}[i].x].\text{tdep}=tdep; \text{arr}[\text{trip}[i].x].\text{tarr}=tarr;
\text{arr}[\text{trip}[i].y].\text{nop}=4; \text{arr}[\text{trip}[i].y].\text{tdep}=tdep; \text{arr}[\text{trip}[i].y].\text{tarr}=tarr;
\text{arr}[\text{trip}[i+1].y].\text{nop}=4; \text{arr}[\text{trip}[i+1].y].\text{tdep}=tdep; \text{arr}[\text{trip}[i+1].y].\text{tarr}=tarr;
\text{arr}[\text{trip}[i+2].y].\text{nop}=4; \text{arr}[\text{trip}[i+2].y].\text{tdep}=tdep; \text{arr}[\text{trip}[i+2].y].\text{tarr}=tarr;
\text{arr}[\text{trip}[i+3].y].\text{nop}=4; \text{arr}[\text{trip}[i+3].y].\text{tdep}=tdep; \text{arr}[\text{trip}[i+3].y].\text{tarr}=tarr;
\}
\}
if(trip[i].per==1)
{
    arr[trip[i].x].per1=1; arr[trip[i].y].per1=1;
    arr[trip[i+1].y].per1=1; arr[trip[i+2].y].per1=1;
}

if(trip[i].per2==2)
{
    arr[trip[i].x].per2=1;
    arr[trip[i].y].per2=1; arr[trip[i+1].y].per2=1; arr[trip[i+2].y].per2=1;
    arr[trip[i].x].per3=1; arr[trip[i].y].per3=1; arr[trip[i+1].y].per3=1; arr[trip[i+2].y].per3=1;
    arr[trip[i].x].per4=1; arr[trip[i].y].per4=1; arr[trip[i+1].y].per4=1; arr[trip[i+2].y].per4=1;
}
else
{
    arr[trip[i].x].per3=1; arr[trip[i].y].per3=1;
    arr[trip[i+1].y].per3=1; arr[trip[i+2].y].per3=1;
    arr[trip[i].x].per4=1; arr[trip[i].y].per4=1;
    arr[trip[i+1].y].per4=1; arr[trip[i+2].y].per4=1;
    arr[trip[i].x].per5=1; arr[trip[i].y].per5=1;
    arr[trip[i+1].y].per5=1; arr[trip[i+2].y].per5=1;
}

if(trip[i].per==2)
{
    arr[trip[i].x].per2=1; arr[trip[i].y].per2=1;
    arr[trip[i+1].y].per2=1; arr[trip[i+2].y].per2=1;
    arr[trip[i].x].per3=1; arr[trip[i].y].per3=1;
    arr[trip[i+1].y].per3=1; arr[trip[i+2].y].per3=1;
    arr[trip[i].x].per4=1; arr[trip[i].y].per4=1;
    arr[trip[i+1].y].per4=1; arr[trip[i+2].y].per4=1;
    arr[trip[i].x].per5=1; arr[trip[i].y].per5=1;
    arr[trip[i+1].y].per5=1; arr[trip[i+2].y].per5=1;
}

if(trip[i].per3>0 && trip[i].per4<=0) // output for
    //joint trips of 3 persons
{
    tdep=(arr[trip[i].x].tdep + arr[trip[i].y].tdep +
    arr[trip[i+1].y].tdep)/3;
    tarr=(arr[trip[i].x].tarr + arr[trip[i].y].tdep +
    arr[trip[i+1].y].tdep)/3;
    arr[trip[i].x].nop=3 ; arr[trip[i].x].tdep=tdep ;
    arr[trip[i].x].tarr=tarr ;
    arr[trip[i].y].nop=3 ; arr[trip[i].y].tdep=tdep ;
    arr[trip[i].y].tarr=tarr ;
    arr[trip[i+1].y].nop=3 ; arr[trip[i+1].y].tdep=tdep ;
    arr[trip[i+1].y].tarr=tarr ;
    if(trip[i].per==1)
    {
        arr[trip[i].x].per1=1 ; arr[trip[i].y].per1=1 ;
    }
    arr[trip[i+1].y].per1=1;
if(trip[i].per2==2)  
    arr[trip[i].x].per2=1; arr[trip[i+1].y].per2=1;  
    if(trip[i].per3==3)  
        arr[trip[i].x].per3=1; arr[trip[i+1].y].per3=1; 
    if(trip[i].per3==4)  
        arr[trip[i].x].per4=1; arr[trip[i+1].y].per4=1; 
    if(trip[i].per3==5)  
        arr[trip[i].x].per5=1; arr[trip[i+1].y].per5=1;  
}

if(trip[i].per2==3)  
    arr[trip[i].x].per3=1; arr[trip[i+1].y].per3=1;  
    if(trip[i].per3==4)  
        arr[trip[i].x].per4=1; arr[trip[i+1].y].per4=1; 
    else  
        arr[trip[i].x].per5=1; arr[trip[i+1].y].per5=1;  
}

if(trip[i].per==2)  
    arr[trip[i].x].per2=1; arr[trip[i+1].y].per2=1;  
    if(trip[i].per2==3)  
        arr[trip[i].x].per3=1; arr[trip[i+1].y].per3=1;  
    if(trip[i].per3==4)  
        arr[trip[i].x].per4=1; arr[trip[i+1].y].per4=1; 
    else  
        arr[trip[i].x].per5=1; arr[trip[i+1].y].per5=1;  
}

if(trip[i].per==3)  
    arr[trip[i].x].per3=1; arr[trip[i+1].y].per3=1;  
    if(trip[i].per3==4)  
        arr[trip[i].x].per4=1; arr[trip[i+1].y].per4=1; 
    else  
        arr[trip[i].x].per5=1; arr[trip[i+1].y].per5=1;  

Analysis of joint trips using C++ in Mobidrive

October 2001

arr[trip[i].x].per5=1; arr[trip[i+1].y].per5=1;
}
if(trip[i].per2==4)
{
    arr[trip[i].x].per4=1; arr[trip[i].y].per4=1;
    arr[trip[i+1].y].per4=1;
    arr[trip[i].x].per5=1; arr[trip[i].y].per5=1;
    arr[trip[i+1].y].per5=1;
}
if(trip[i].per==3)
{
    arr[trip[i].x].per3=1; arr[trip[i].y].per3=1;
    arr[trip[i+1].y].per3=1;
    arr[trip[i].x].per4=1; arr[trip[i].y].per4=1;
    arr[trip[i+1].y].per4=1;
    arr[trip[i].x].per5=1; arr[trip[i].y].per5=1;
    arr[trip[i+1].y].per5=1;
}
if(trip[i].per2>0 && trip[i].per3==0) // output /for two persons
{
    tdep = (arr[trip[i].x].tdep + arr[trip[i].y].tdep)/2;
    tarr = (arr[trip[i].x].tarr + arr[trip[i].y].tdep)/2;
    arr[trip[i].x].nop=2; arr[trip[i].y].nop=2;
    arr[trip[i].x].tdep=tdep; arr[trip[i].y].tdep=tdep;
    arr[trip[i].x].tarr=tarr; arr[trip[i].y].tarr=tarr;
    if(trip[i].per==1)
    {
        arr[trip[i].x].per1=1;
        arr[trip[i].y].per1=1;
    }
    if(trip[i].per2==2)
    {
        arr[trip[i].x].per2=1;
        arr[trip[i].y].per2=1;
    }
    if(trip[i].per2==3)
    {
        arr[trip[i].x].per3=1;
        arr[trip[i].y].per3=1;
    }
    if(trip[i].per2==4)
    {
        arr[trip[i].x].per4=1;
        arr[trip[i].y].per4=1;
    }
    if(trip[i].per2==5)
    {

if (trip[i].per == 2) {
    arr[trip[i].x].per2 = 1;
    arr[trip[i].y].per2 = 1;
    if (trip[i].per2 == 3) {
        arr[trip[i].x].per3 = 1;
        arr[trip[i].y].per3 = 1;
    }
    if (trip[i].per2 == 4) {
        arr[trip[i].x].per4 = 1;
        arr[trip[i].y].per4 = 1;
    }
    if (trip[i].per2 == 5) {
        arr[trip[i].x].per5 = 1;
        arr[trip[i].y].per5 = 1;
    }
}
if (trip[i].per == 3) {
    arr[trip[i].x].per3 = 1;
    arr[trip[i].y].per3 = 1;
    if (trip[i].per2 == 4) {
        arr[trip[i].x].per4 = 1;
        arr[trip[i].y].per4 = 1;
    }
    if (trip[i].per2 == 5) {
        arr[trip[i].x].per5 = 1;
        arr[trip[i].y].per5 = 1;
    }
}
if (trip[i].per == 4) {
    arr[trip[i].x].per4 = 1;
    arr[trip[i].y].per4 = 1;
    arr[trip[i].x].per5 = 1;
    arr[trip[i].y].per5 = 1;
}
outFile.open("hh50m.txt", ios::out); // sending the output to the file
```cpp
 outFile << setw(4) << "no." << setw(8) << "studyc" << setw(10) << "cityc"
 << setw(8) << "hhnr" << setw(8) << "pnr" << setw(8) << "jtrip" << setw(6)
 << "nop" << setw(6) << "p1" << setw(6) << "p2" << setw(6) << "p3" << setw(6)
 << "p4" << setw(6) << "p5" << setw(10) << "tdep" << setw(10) << "tarr" << "\n";

 for(i=0;i<MAX;i++)
 {
     outFile << setw(4) << i+1 << setw(8) << arr[i].studyc << setw(10) << arr[i].cityc
 << setw(8) << arr[i].hhnr << setw(8) << arr[i].pnr << setw(8)
 << arr[i].jtrip << setw(6) << arr[i].nop << setw(6) << arr[i].per1 << setw(6)
 << arr[i].per2 << setw(6) << arr[i].per3 << setw(6) << arr[i].per4
 << setw(6) << arr[i].per5 << setw(10) << arr[i].tdep << setw(10) << arr[i].tarr << "\n" ;
 }
```

Eine vollständige Liste der Berichte kann vom Institut angefordert werden:

IVT ETHZ
ETH Hönggerberg (HIL)
CH - 8093 Zürich

Telephon: +41 (0)1 633 31 05
Telefax: +41 (0)1 633 10 57
E-Mail: hotz@ivt.baug.ethz.ch
WWW: www.ivt.baug.ethz.ch

Der Katalog kann auch abgerufen werden:

http://www.ivt.baug.ethz.ch/veroeffent_arbeitsbericht.html

The Working Papers Traffic and Spatial Planning are intended for the quick dissemination of the results of the members and guests of the Institut. Their content is the sole responsibility of the authors.

A complete catalog of the papers can be obtained from:

IVT ETHZ
ETH Hönggerberg (HIL)
CH - 8093 Zürich

Telephon: +41 (0)1 633 31 05
Telefax: +41 (0)1 633 10 57
E-Mail: hotz@ivt.baug.ethz.ch
WWW: www.ivt.baug.ethz.ch

The catalog can also be obtained from:

http://www.ivt.baug.ethz.ch/veroeffent_arbeitsbericht.html