Data Processing Software for an Automatic Data Acquisition System in Mass Transit

A. G. Hobeika, C. E. Nunnally, S. Raju, and P. Anderson

Volumes of raw data collected using the Automatic Data Acquisition System are useless until they can be processed and transformed into meaningful reports and plots. This is done using the data processing software. The software has a matching algorithm that is modular in structure. It uses a simple, systematic, and logical sequence of actions in attempting to match the recorded activities in the raw data file to actual bus stops. The algorithm is tested, validated, and justified by performing sensitivity analyses on data files resembling those obtained from the Automatic Passenger Counter experiment conducted in Roanoke Valley Metro in Virginia. It is found that the algorithm does not change its matching process with changes in the system variables. Thus, a robust algorithm has now been developed to do the matching accurately, removing the need for a radio signpost signal to be embedded to establish locational accuracy, consequently eliminating signpost installation and maintenance worries and costs.

The federal government's increasing cutbacks in operating subsidies to mass transit have forced transit managers to shift their emphasis from capital intensive transit improvements to short-range transit efficiency actions (*1*).

In order to improve efficiency, transit managers need to make intelligent and informed decisions. Such decisions require an extensive, accurate, and continuous flow of data on ridership, fare revenue, schedule adherence, and the like. Establishing this kind of comprehensive and statistically valid data base through labor-intensive manual data collection methods is very expensive. The largest cost item in the manual data collection is the manpower needed to collect on-board data and data on the street (2). Hence, transit agencies are beginning to adopt automated data acquisition techniques. The federal government is also promoting research investigating various aspects related to future widespread implementation of the Automatic Data Acquisition System (ADAS) (*3*).

Automated monitoring programs lend the capability of detecting detailed changes in running and loading characteristics as well as in various levels of service standards (2). Recent research in this field has considerably advanced the state of the art of transit assessment methods. A number of large and small transit agencies have implemented ADAS (*4*). They have now adopted sets of service performance measures and set up systematic evaluation programs. One such example is the Valley Metro bus station at Roanoke, Virginia.

Department of Civil Engineering, Virginia Polytechnic Institute and State University, Blacksburg, Va. 24061.

DESCRIPTION

ADAS is a composition of hardware and associated computer software that work together to yield certain basic reports on routes, schedules, and operations (*5*). The data collected using the hardware give information about the number of passengers boarding and alighting at various stops at various times. The ADAS at Valley Metro in Roanoke, Virginia, is a microprocessor-based system performing four basic steps:

- Data acquisition,
- Data recording and storage,
- Data transfer and data processing, and
- Report generation.

The data acquisition is accomplished using four pairs of infrared sensors fixed to the inner sides of the front and rear doorway of the bus. The sequence of interruption and making of the infrared beams determines the logic for the count generation algorithm to sense the passenger activity as an "on" or "off." The time in 15-sec units and distance in odometer impulses are stamped along with the activities. These are all recorded as a 5-byte-long set of records and stored. The file containing this set of data records is transferred to the processing computer for further processing and report generation.

SYSTEM CONFIGURATION

ADAS consists of four modules interconnected to each other in a daisy-chained fashion and four sensor pairs, each containing two pairs of infrared transmitters and receivers. The four modules are the Odometer/Power Module (OPM), Passenger Count Module (PCM), Data Storage Module (DSM), and Diagnostic Module (DGM). A typical ADAS as in Valley Metro at Roanoke, Virginia, comprises one DGM, one OPM, one DSM, and two PCMs—one each for front and rear doors. The DGM is used to examine certain operational and functional parameters of the system. The OPM, which is fed directly from a 12-v battery, serves two purposes: as a power distribution center to the other modules of the ADAS and as a receiver of odometer impulses. These odometer impulses divided by a division factor give a measure of the distance, which is passed on to the other modules so that appropriate logs are generated. The PCMs monitor the status of the doors via an on/off contact signal and passenger activity via infrared sensor pairs. Door open and close signals and vehicle movement information are used by these modules to generate appropriate logs. The PCMs receive and retain the logs generated by other modules and store them in the order of their arrival, which is the order of their generation or creation. The DSM is the "controller" of the system.

The ADAS provides two different types of communication methods, one for intermodule communication and another for dumping the data file to a computer. Both use a packet-based approach for requests, acknowledgments, and actual data transfers. Communication with the outside world is by means of an RS-232 interface without any handshakes.

DATA PROCESSING SOFTWARE

The file created by the ADAS hardware is a set of 5-byte-long records. The 5 bytes are, respectively,

- The log type,
- The time elapsed in 15-sec units,
- The distance traveled in odometer impulses,
- Passenger ons, and
- Passenger offs.

Whenever an activity occurs, a particular log is created, which is stamped on the PCMs and stored on the DSM. These logs can be created for the following reasons:

- The bus is idle for more than 1 min.
- The counter sensors detect a passenger boarding or alighting (passenger activity).
- The destination signboard is changed.
- The time or distance registers overflow and is reset.

For example, when a passenger activity like passengers getting on or off the bus occurs, a Log Type 5 is created and stored. Similarly, a Log Type 3 is created when the bus idles for a minute, and a Log Type 6 is created when the bus idles for 2 or more min.

The data file generated during the collection period is transferred to the main computer for processing and eventual report generation. The data processing step is a key stage, determining the utility of the ADAS both in terms of present and future software needs of the transit agency. The algorithm for executing this step is built modularly. It employs a simple, systematic, and logical sequence of tasks to do the processing.

Three phases are involved in this software, as shown in Figure 1:

1. Data Integration Phase.
2. Data Matching Phase.
3. Output File Storage Phase.

Data Integration Phase

In the Data Integration Phase, the external file containing all the system information is first retrieved. This information includes the name of the transit company, the base names of the reference files used for matching, the base name of the Section 15 cumulative file, the disk drives where all of the above files are stored, DeltaL, DeltaT, DeltaS2, the number of feet per odometer impulse, and so on.

The reference files used for matching are the distance, schedule, and bus stop name files. The distance reference file includes information about distances between stops within a loop, number of timepoints along the route per loop, the route number, and route name. The schedule reference file includes information like the block number, number of loops of operation, number of timepoints along the route per loop, starting time of the first loop, ending time of the last loop, stop numbers of the timepoints, and times at various scheduled timepoints. The name reference file includes information like the route number, route name, the number of stops per loop, and the names of all the stops along the route.

Next, a program menu is presented and according to the user's choice, the raw data file to be processed is retrieved. Activity records are created. An activity record is any record of the Log Types 3, 5, 6, and 11, representing 1 min idle, passenger activity, 2 or more min idle, and signboard change, respectively. The starting time of the operation of the route is calculated by tracking back from the time of data dumping. The times and distances at which the various activity records occurred are calculated by a process of integration. This is done by adding all the time clicks until the particular record from the starting time is reached. The same is done for the distance. Finally, a count of the number of activity records is generated. This phase is presented in Figure 2.

Data Matching Phase

The Data Matching Phase is the key phase in the processing of the algorithm. In this phase, the reference files—namely, the schedule and distance files—are first retrieved. Then, the potential loops from the activity records are fitted to the reference files. A potential loop is one which passes a twofold test. The first involves matching an activity to Stop Number 1. The first activity record that falls within DeltaT of the scheduled starting time of a loop is taken as the temporary beginning of a potential loop. The second test involves locating the activity record occurring at the end of the loop. This is done by comparing the actual distance corresponding to the various activity records with the reference loop distance; if the difference is within DeltaL, then that activity record is taken as the temporary end of the potential loop. The match ratio for this loop is next calculated using the relation

$$MR = M/[Te(L) - Tb(L)]$$

where

MR = match ratio,
$Te(L)$ = temporary end of the loop L,
$Tb(L)$ = temporary beginning of the loop L, and
M = the number of activity records that have their distance within DeltaS1 of the corresponding reference distance, called "match."

The match ratio reflects the percentage of activities that produced a match out of all the activities. This match ratio is stored. The next activity record that falls within DeltaT of the scheduled time for Stop Number 1 is taken as the temporary beginning of the next potential loop and the end is located as before. Its match ratio is calculated and then compared with that of the previous activity record, and the loop with the higher match ratio is chosen for further comparison. Finally, the loop

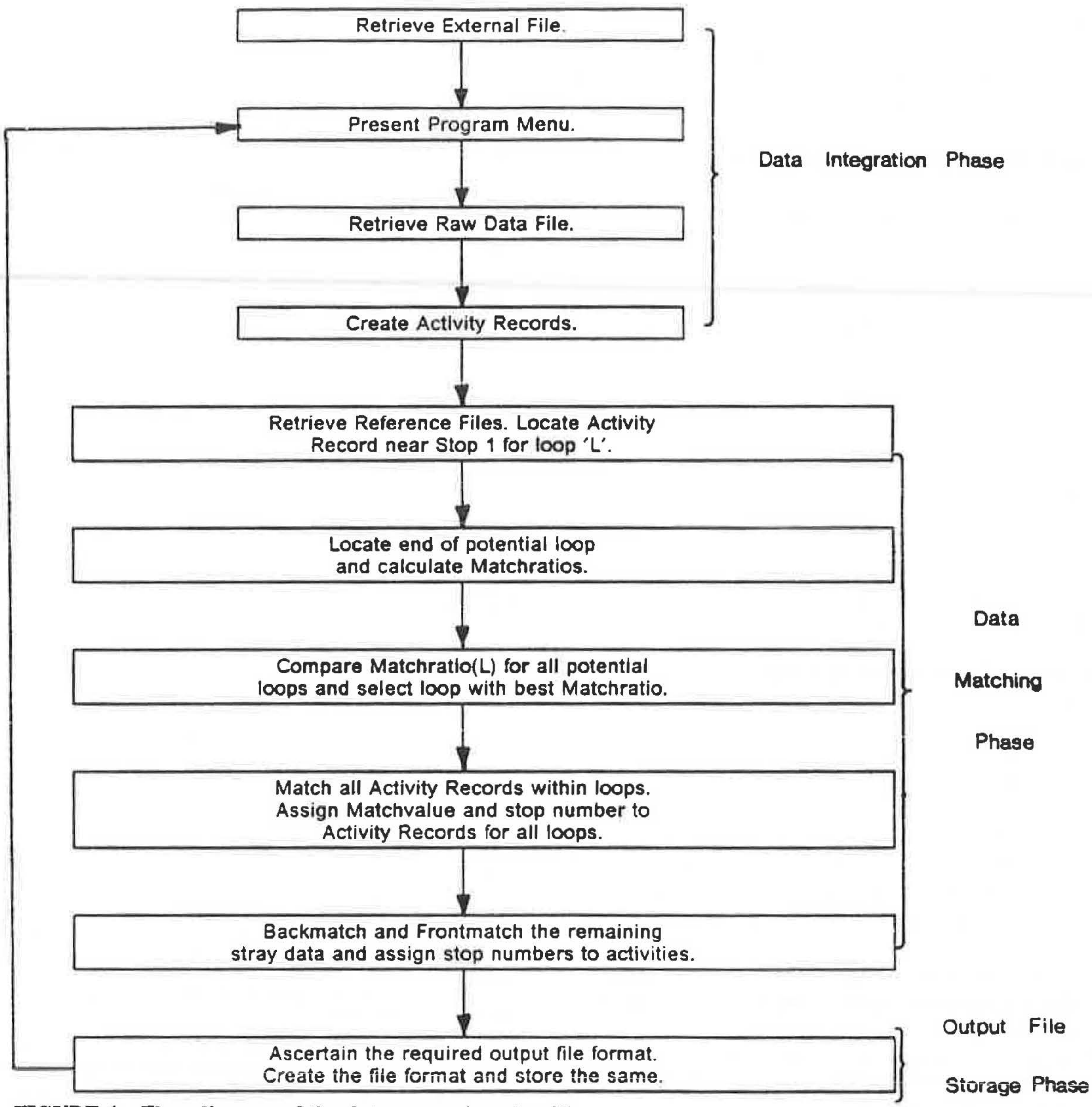

FIGURE 1 Flow diagram of the data processing algorithm.

with the highest match ratio is chosen as the good loop. The beginning and end of the potential loop with the highest match ratio become the beginning and end of the good loop. All the good loops for all the activity records are located in the same manner. The activities within each of the good loops are matched, by comparing the distances of each of the activity records with the reference distances for various stops. If the difference is within DeltaS2, then a match value indicating an excellent match and the stop number from the reference file are assigned. If the difference is within DeltaS1, then a match value indicating a good match and the stop number from the reference file are assigned. However, if the difference is beyond DeltaS1, then a match value indicating a poor match along with the stop number giving the closest correspondence are assigned. This is done for all activity records within all good loops. The next stage involves matching the activity records that fall outside the good loops. These stray data are matched using two techniques: backmatching and frontmatching. In the backmatching technique, the stray data occurring before the beginning of the first good loop are matched, by comparing the difference between the actual and reference distance with DeltaS1. The matching is carried backward from the beginning of the good loop. In the frontmatching technique, the data occurring after the end of the last good loop are matched using the same procedure except that the matching is done forward. Thus, in this way, matching is attempted for all activity records. The flow diagram for this phase is shown in Figure 3.

Output File Storage Phase

In the Output File Storage Phase, the processed data are converted to an output file format and stored. This output format depends on the needs of the present and future management software of the transit company. It is observed that this is dependent on the size, location, and management strategies of the transit agency. However, because of the modular nature of the algorithm, excellent flexibility and transferability potential

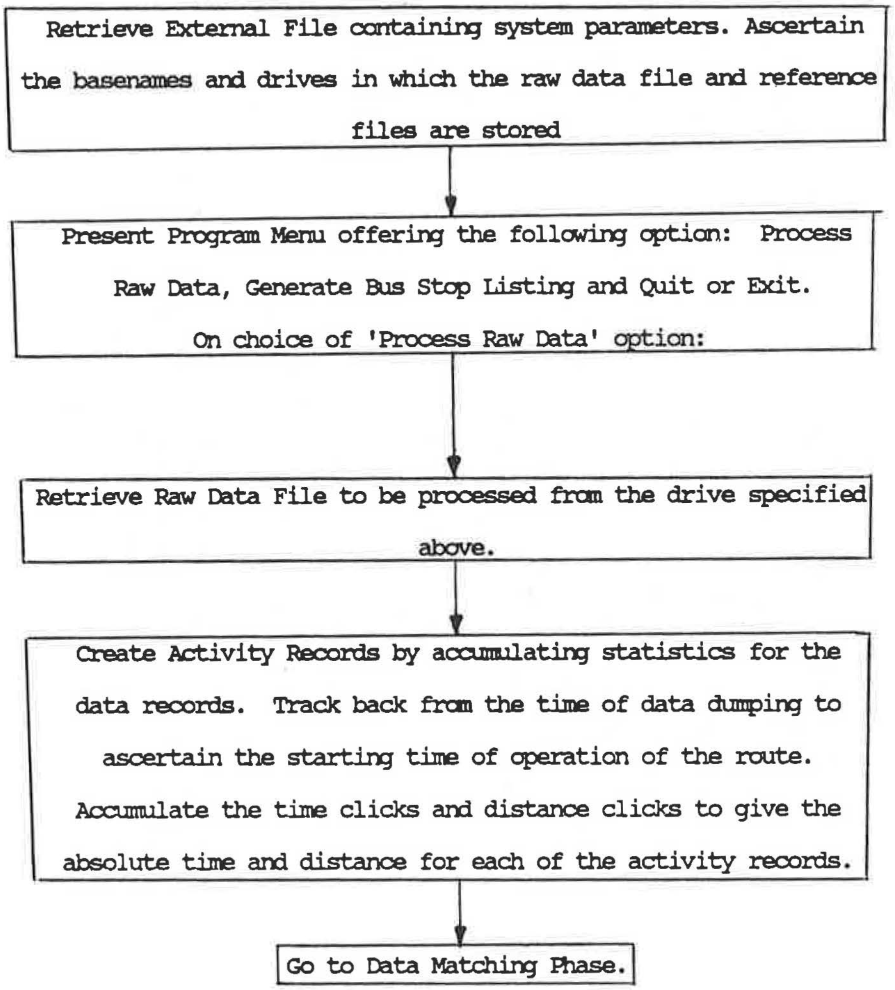

FIGURE 2 Flow diagram of the Data Integration Phase.

exist. This phase is represented by the flow diagram shown in Figure 4.

SENSITIVITY ANALYSIS

Any algorithm must be tested, validated, and justified before its final adoption. For doing so, a hypothetical test route was designed, and the algorithm was tested for various values of the system parameters. The results obtained, in all situations, were in conjunction with what was conceptualized; hence, it was proven that the algorithm did the matching exactly. The program was also run on real-life data, and the results obtained were very encouraging.

A test route with 12 stops and several test data files having one or more loops of data were created and the program run on them. This was done so that the matching could be traced and checked manually. It was found that the algorithm matched as intended. Also, the distances and times in the data files resembled in magnitude those in reality. The manual checking of the matching process on the real-life data file could not be done because of the large number of stops per loop and the voluminous data that are matched. However, it is worth noting that the distances in the distance reference file play a crucial part in determining the quality and quantity of matching. In other words, the amount and accuracy of matching depend upon the distances recorded in the distance reference file to a large extent.

The Roanoke Valley Metro bus equipped with ADAS collected data throughout the day on a normal operating schedule. The algorithm was used for matching activities in all types of service: peak, off-peak, midday, night, etc., and it performed as intended.

A sensitivity analysis on the algorithm was performed to check the variation in the matching process with changes in the system parameters. This involved recognizing the parameters affecting the matching process and the roles played by these parameters during the various steps in matching. The system parameters are DeltaL, DeltaT, DeltaS1, and DeltaS2.

The steps in the matching process from the point of view of performing sensitivity analysis are

1. Location of the beginning of a potential loop.
2. Location of the end of a potential loop.
3. Calculation of the match ratio for the potential loop.
4. Determination of the best match ratio and, consequently, the good loop.
5. Determination of all good loops.
6. Matching within good loops.
7. Matching outside good loops using backmatching and frontmatching techniques.

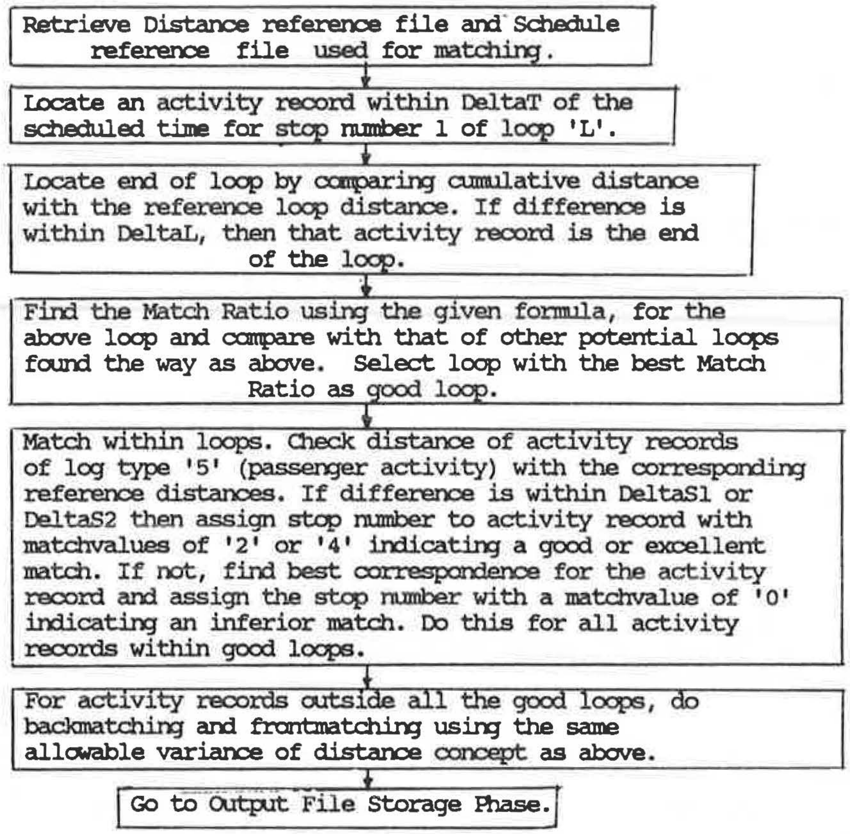

FIGURE 3 Flow diagram of the Data Matching Phase.

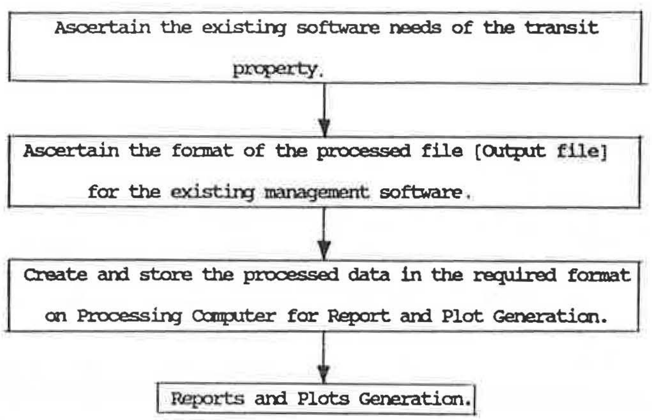

FIGURE 4 Flow diagram of the Output File Storage Phase.

The roles played by the various parameters and their effect on the matching process are detailed below.

• *DeltaT*: This is involved only in the location of the beginning of a potential loop (i.e., Step 1). Increasing the values of the parameter may lead to a greater number of iterations of the algorithm. However, it is to be so chosen that it would account for the worst possible delay at the beginning of a loop of the particular route. A sensitivity analysis using a real-life data file shows that DeltaT does not affect the overall matching process. This is illustrated in Table 1.

TABLE 1 VARIATION OF THE MATCHING PROCESS WITH DELTAT

	Step						
DeltaT	1	2	3	4	5	6	7
00	Y	Y	Y	Y	Y	Y	Y
04	P	P	P	P	P	P	P
08	P	P	P	P	P	P	P
12	P	P	P	P	P	P	P
16	N	N	N	N	N	N	N
20	N	N	N	N	N	N	N
24	N	N	N	N	N	N	N
28	N	N	N	N	N	N	N
32	B	A	S	L	I	N	E
36	N	N	N	N	N	N	N
40	N	N	N	N	N	N	N
44	N	N	N	N	N	N	N
48	N	N	N	N	N	N	N

NOTE: DeltaL = 35, DeltaS1 = 6, DeltaS2 = 3. N = No change. P = Partly changed (5 to 10 percent). Y = Change > 10 percent. BASLINE = Baseline for comparison.

• *DeltaL*: This is involved only in Step 2 (i.e., location of the end of a potential loop). There is absolutely no change in the matching process with a change in DeltaL. This is shown in Table 2.

• *DeltaS1*: This is involved in a number of steps of the matching process, namely, Steps 3, 4, 5, 6, and 7 (i.e., calculation of match ratio, determination of a good loop, determination of all good loops, matching within the good loops, and matching outside the good loops, respectively).

TABLE 2 VARIATION OF THE MATCHING PROCESS WITH DELTAL

	Step						
DeltaL	1	2	3	4	5	6	7
00	N	N	N	N	N	N	N
05	N	N	N	N	N	N	N
10	N	N	N	N	N	N	N
15	N	N	N	N	N	N	N
20	N	N	N	N	N	N	N
25	B	A	S	L	I	N	E
30	N	N	N	N	N	N	N
35	N	N	N	N	N	N	N
40	N	N	N	N	N	N	N
45	N	N	N	N	N	N	N

NOTE: DeltaT = 32, DeltaS1 = 6, DeltaS3 = 3. N = No change. P = Partly changed. BASLINE = Baseline for comparison.

Hence, as far as the overall matching process goes, this parameter plays a crucial role and determines the actual precision of the matching process. The value for this parameter has to be carefully selected so that most of the unforeseen circumstances (like the bus stopping away from a stop due to one or more vehicles parked there, etc.) are taken care of. Also, the value of this parameter must be less than the distance between any two stops. The variation in the matching process with change in DeltaS1 is indicated in Table 3. A mild effect is observed in Steps 6 and 7, but the overall matching process remains essentially the same.

TABLE 3 VARIATION OF THE MATCHING PROCESS WITH DELTAS1

	Step						
DeltaS1	1	2	3	4	5	6	7
4	N	N	N	N	N	N	N
5	N	N	N	N	N	N	N
6	B	A	S	L	I	N	E
7	N	N	N	N	N	N	N
8	N	N	N	N	N	N	N
9	N	N	N	N	N	P	P
10	N	N	N	N	N	P	P

NOTE: DeltaL = 35, DeltaT = 40, DeltaS2 = 3. N = No change. P = Partly changed. BASLINE = Baseline for comparison.

• *DeltaS2*: This is involved only in Step 6—matching within good loops with a high precision. As indicated in Table 4, there is practically no change in the matching process with a change in DeltaS2. More activity records were matched when the allowable variance for precise matching was increased. The change or difference was negligible. In other words, the additional number of activity records got matched.

Thus, from the sensitivity analysis, it can be concluded that the algorithm does not change appreciably in its overall matching process with changes in the system parameters, indicating that a robust and reliable routine is now available for use with the ADAS hardware.

It should be recognized that there are a few implicit assumptions in the algorithm. The bus using the ADAS is assumed to be operating on a fixed route. The data are assumed to be collected for a greater-than-one-loop length of the route. The bus is assumed to be operating on the normal route at normal schedule. Any changes in the route must be incorporated into the reference files before matching. In addition, the operator is assumed not to allow boarding and alighting of passengers between stops along the route. These assumptions are valid, however, since events disrupting them are rare.

TABLE 4 VARIATION OF THE MATCHING PROCESS WITH DELTAS2

		Step						
DeltaS1	DeltaS2	1	2	3	4	5	6	7
6	3	B	A	S	L	I	N	E
	4	N	N	N	N	N	N*	N*
	5	N	N	N	N	N	N*	N*
7	3	B	A	S	L	I	N	E
	4	N	N	N	N	N	N*	N*
	5	N	N	N	N	N	N*	N*
8	3	B	A	S	L	I	N	E
	4	N	N	N	N	N	N*	N*
	5	N	N	N	N	N	N*	N*

NOTE: DeltaL = 35, DeltaT = 36. N = No change. N* = Additional matches. BASLINE = Baseline for comparison.

From the results of these tests it has been concluded that there is no need for radio signposts along the routes for establishing locational accuracy. In this system, the algorithm is robust and can lock in and match the activities to stops. It is believed that the elimination of the use of radio signposts would result in savings of up to $2,000 per bus and eliminate associated installation and maintenance costs and worries.

The software, which is available as a package along with the complete hardware, is written in BASICA and runs on any IBM PC or compatible computer with a BASIC compiler.

DATA UTILITY

Once the raw data file is processed and the output file is stored in a particular format, several reports and plots can be generated. These reports include the

- Timepoint Profile Report,
- Section 15 Daily Report,
- Section 15 Annual Report,
- Summary Report, and
- Bus Stop Listing Report.

The plots include the

- Passenger Load Plot,
- Route Demand Plot,
- Route Evaluation Plot, and
- Schedule Evaluation Plot.

In addition, any reports and plots specifically needed by a transit agency can be developed depending on the needs.

A sample output of the Bus Stop Listing Report from Roanoke Valley Metro is shown in Figure 5. This listing provides general information such as the block number, route number, route name, and date of creation of the file. Also provided is crucial information on the locations and times of passenger activity, precision of the activity indicated by match value (*M*), distance and time the activity occurred (Dist and Depart), number of passengers boarding and alighting at each

ROANOKE VALLEY METRO BUS STOP LISTING PREPARED 01-01-1980 PAGE 3
BLOCK # 63 ROUTE # 5 MELROSE/VIRGINIA HGTS. 10-21-86

STOP#	STOP NAME	M	DIST	DEPART	SCHED	DEV	ONS	OFFS	PASS
		0	65.62	13:41: 0			0	1	0
151	17TH/LOUDON	0	65.94	13:43:15			1	0	1
156	22ND/LOUDON	4	66.06	13:45:15			1	0	2
181	GUN CLUB/MELROSE	0	69.32	14: 0:30			1	0	3
		0	71.04	14: 6: 0			0	1	2
		0	71.81	14: 9:45			3	2	2
1	RKE.SALEM PLAZA	4	72.90	14:18: 0	14:15: 0	- 3: 0	0	1	1
8	MELROSE/35TH	4	73.62	14:20:45			1	0	2
18	MELROSE/LAFAYETTE	4	74.54	14:23:45	14:22: 0	- 1:45	1	0	3
20	MELROSE/24TH	0	77.45	14:35:15			0	1	2
47	WOOLWORTH'S	4	77.46	14:37:15			0	1	1
48	JEFF/CHURCH	4	77.60	14:41:15	14:40: 0	- 1:15	4	2	3
		0	77.72	14:42:30			0	3	0
48	JEFF/CHURCH	0	79.05	14:49:15	14:40: 0	- 9:15	0	1	0
		0	79.67	14:51: 0			0	1	0
66	MAIDEN LANE/DUNMORE	0	80.43	14:54:15			0	1	0
		0	80.52	14:54:45			1	0	1
76	MEMORIAL/2229	0	80.97	14:56: 0			0	1	0
90	MEMORIAL/WESTOVER	0	81.98	15: 0:15			0	1	0
		0	83.00	15: 3:30			1	0	1
105	HAMPTON/DENNISTON	0	83.32	15: 5:45			1	0	2
		0	84.27	15: 8:30			0	1	1
116	MAIN/FERDINAND	0	85.59	15:14:45			0	1	0
		0	85.63	15:15:45			3	0	3
130	CHURCH/2ND	0	85.78	15:17:45			2	0	5
		0	86.02	15:20: 0			1	0	6
135	RORER/5TH	0	87.37	15:25: 0			0	1	5
		0	89.27	15:30:45			0	2	3
		0	89.38	15:31:15			0	1	2
169	35TH/MELROSE	0	89.76	15:32:15			0	1	1
		0	90.41	15:34: 0			0	1	0
1	RKE.SALEM PLAZA	4	90.63	15:43:30	15:40: 0	- 3:30	3	1	2
17	MELROSE/FOREST PARK	2	92.14	15:47:45			1	0	3
37	SALEM/9TH	4	94.22	15:53:30			1	0	4
		0	95.18	16: 0:15			2	3	3
47	WOOLWORTH'S	4	95.19	16: 5: 0			2	0	5
48	JEFF/CHURCH	2	95.31	16: 7: 0	16: 5: 0	- 2: 0	2	1	6
		0	95.33	16: 7:30			1	0	7

FIGURE 5 Sample output of Bus Stop Listing Report at Roanoke Valley Metro in Virginia.

stop (Ons and Offs), and passenger load on the bus at each stop (Pass). Also, at timepoints, the scheduled time (Sched) and deviations (Dev) of the bus at these stops are indicated. The schedule deviations at particular timepoints along the route serve as indicators of the quality of design of the schedules and aid in the proper design and maintenance of routes and schedules. This increases the efficiency and reliability of service of various routes, especially when schedule adherence is critically regarded by riders.

CONCLUSIONS

Easy-to-implement and inexpensive data processing software for ADAS is now available for bus transit.

The proven software capabilities eliminate the need for radio signposts for establishing locational accuracy, which translates into savings of associated installation and maintenance costs and less worry.

The software package is modular in nature, allowing for further enhancement. The potential for software development in this area is tremendous, and the described algorithm promotes this flexibility.

The software is interactive and menu-driven, giving the user an opportunity to understand and visualize the sequence of the matching process.

The output file format can be easily changed, depending on the requirements of the transit company, because of the structured approach used in the algorithm.

The software can be run on any IBM-PC or compatible computer. Thus, the ADAS can fit in simply without any investment for additional data processing machines.

With the present ADAS, transit agencies can now establish a comprehensive data base to monitor various routes and schedules. This allows managers to exercise better system surveillance and make better decisions.

A great potential for developing statistically valid demand models for each of the routes is created, which aids in system planning.

Finally, the system requires very little maintenance. It is advisable to have an employee of the transit agency involved in the installation and implementation of the system, however, so that someone in the agency knows how the system hardware operates. A detailed manual for software support is available. The fear of change and the hesitancy to use new technology for data processing, report generation, and so on is one of the biggest hindrances to widespread implementation of ADAS. With stronger marketing efforts, better software support, and more structured information dissemination, this problem can be overcome.

GLOSSARY (*6*)

Loop: the spatial distribution of the route starting from a point or stop and returning to it.

Good loop: any loop whose start-up time is within DeltaT of the scheduled starting time and whose match ratio is the highest.

Match ratio: the percentage of activities that produces a match out of all the activities.

Match: when an activity record's distance from the previous activity record falls within DeltaS1 (a small distance value) of the corresponding reference distance.

Match value: a number or value given to an activity record to represent the precision of the match.

Timepoints: stops along the route that have scheduled times attached to them.

DeltaL: the allowable distance variation in the number of odometer impulses for determining when the matching routine should end.

DeltaT: the allowable time variation in the number of time clicks for determining the beginning of a loop.

DeltaS1 and DeltaS2: the allowable distance variations in the number of odometer impulses for determining the outer and inner limits of a positive match indicating a good match and an excellent match, respectively.

ACKNOWLEDGMENTS

The authors want to thank the staff of Valley Metro at Roanoke, Virginia, for their support and enthusiasm during the development of this project. The authors also acknowledge the support of the Virginia Department of Transportation.

REFERENCES

1. L. E. Deibel and B. Zumwalt. *A Modular Approach to On-Board Automatic Data Collection Systems.* Final Report. MITRE Corporation, Aug. 1984.
2. Multisystems, Inc. *An Assessment of Automatic Passenger Counters.* Report DOT-I-82-43. U.S. Department of Transportation, Sept. 1982.
3. P. J. Poirier and V. J. Hobbs. *Automated Passenger Counter Systems: Synopsis of Working Group Meeting.* Report DOT-TSC-UMTA-83-23. Transportation Systems Center, Office of Technology Applications, U.S. Department of Transportation, Cambridge, Mass., March 1982, pp. 3–5.
4. Multisystems, Inc. *Transit Data Collection Design Manual.* Report DOT-I-85-38. Transportation Systems Center, U.S. Department of Transportation, Cambridge, Mass., 1985.
5. R. Srinath and A. G. Hobeika. *Application of Automatic Data Acquisition Systems in Mass Transit.* M.S. thesis. Virginia Polytechnic Institute and State University, Blacksburg, Aug. 1987.
6. P. Anderson and A. G. Hobeika. *Development and Implementation of an Automatic Passenger Counter System at a Small Transit Company.* M.S. thesis. Virginia Polytechnic Institute and State University, Blacksburg, Aug. 1986.

Publication of this paper sponsored by Committee on Transit Management and Performance.