REPORT NO. DOT-TSC-UMTA-72-1

OPERATIONAL DIAL - A - RIDE COMPUTER PROGRAM VOLUME 1

Test and Evaluation Report

JUAN F. BELLANTONI
TRANSPORTATION SYSTEMS CENTER
55 BROADWAY
CAMBRIDGE, MA. 02142

transpo 72
U.S. International Transportation Exposition
Dulles International Airport
Washington, D.C.
May 27-June 4, 1972

DEPARTMENT OF TRANSPORTATION • UNITED STATES OF AMERICA

SEPTEMBER 1971
TECHNICAL REPORT

Availability is Unlimited. Document may be Released To the National Technical Information Service, Springfield, Virginia 22151, for Sale to the Public.

Prepared for
DEPARTMENT OF TRANSPORTATION
URBAN MASS TRANSPORTATION ADMINISTRATION
WASHINGTON, D.C. 20590

1. Report No. DOT-TSC-UMTA-72-1	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle OPERATIONAL DIAL-A-RIDE COMPUTER PROGRAM, Test and Evaluation Report VOLUME I	5. Report Date 30 September 1971
	6. Performing Organization Code
7. Author(s) Juan F. Bellantoni	8. Performing Organization Report No.
9. Performing Organization Name and Address Department of Transportation Transportation Systems Center Washington, D.C. 02142	10. Work Unit No. R2718
	11. Contract or Grant No. UM202
12. Sponsoring Agency Name and Address Department of Transportation Urban Mass Transportation Admin. Washington D.C. 20590	13. Type of Report and Period Covered Technical Report, FY71-72 Jan-Sept. 1971
	14. Sponsoring Agency Code

15. Supplementary Notes

This report consists of two volumes. Volume II contains the Test Scenarios and Test Data for the O D-A-R Computer Program.

16. Abstract

This report presents the results of the evaluation of the MIT Urban Systems Laboratory's (USL's) Dial-A-Ride operational computer program. The evaluation was carried out by the Transportation Systems Center (TSC) under PPA UM-02, "Transportation Systems Computer Package", FY'72. The general purpose of the evaluation was to test the Operational Dial-A-Ride (O D-A-R) DOS program against the work statement of November 24, 1970, for extension of the UMTA Grant MASS-MTD-6.

17. Key Words	18. Distribution Statement
Demand Responsive Systems, Dial-A-Ride, Dial-A-Bus, Operational D-A-R, Computer Dispatching	Availability is Unlimited. Document may be Released To the National Technical Information Service, Springfield, Virginia 22151, for Sale to the Public.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	142	

TABLE OF CONTENTS VOLUME I

TABLE OF CONTENTS VOLUME 2

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (CONTINUED)

LIST OF ILLUSTRATIONS (CONTINUED)

LIST OF TABLES

INTRODUCTION

This report presents the results of the evaluation of the MIT Urban Systems Laboratory's (USL's) Dial-a-Ride operational computer program. The evaluation was carried out by the Transportation Systems Center (TSC) under PPA UM-02, "Transportation Systems Computer Package", FY'72. The general purpose of the evaluation was to test the Operational Dial-a-Ride (O D-A-R) DOS program against the work statement of November 24, 1970, for extension of the UMTA Grant MASS-MTD-6. Two other Dial-a-Ride computer programs, the basic and the advanced, were delivered to DOT and evaluated by TSC in a previous report.

The over all project was carried out in nine sub-tasks, delineated in the PPA, as revised 31 March 1971:

1. Receive and test the first version O D-A-R
2. Prepare a preliminary test specification outline
3. Receive and test the second version of O D-A-R
4. Review and approve the acceptance test specification
5. Witness and record data during the acceptance test
6. Review the program documentation
7. Review the backup mode
8. Report on results of 5, 6 and 7 above
9. Analyze and evaluate O D-A-R: report on such work

A report on subtasks 1 and 3 was in the monthly progress reports of February, March and June 1971. This report gives the results of the seven remaining subtasks which have to do directly with the acceptance tests, and is orgainzed in the following segments:

Section 1: Analysis of Work Statement and Design of the Acceptance Test
Section 2: Acceptance Test Results
Section 3: Review of Program Documentation
Section 4: Review of Backup Mode
Section 5: Evaluation

SECTION 1
ANALYSIS OF WORK STATEMENT AND DESIGN OF ACCEPTANCE TEST

The O D-A-R Acceptance Tests, the results of which are given in Volume II and analyzed in the next section, are based on the work statement submitted by MIT/USL on November 24, 1970. As required in the work statement, an acceptance test specification was drawn up by MIT/USL and approved by TSC. The outline provided by TSC and TSC's comments on the draft are given in Appendix B. The final version was agreed upon on June 2, 1971 and is included in Volume II.

1.1 Analysis of Work Statement

Extracts of the work statement, pertinent to the O D-A-R tests, are given in Appendix A. The means by which the Acceptance Test Specification was obtained from the work statement is described below.

Paragraph 3.6.3.3 of the work statement states that the test will demonstrate the "specific capabilities" listed in section 3.4.1. That section, in turn, states:

"Specifically, the Operational DOS System will have the following capabilities:

1. Restart capability
2. Cancellation of a service request
3. Handling of more passengers at a stop than expected
4. Vehicle breakdown procedure (includes re-assignment of passengers)
5. Detection of late vehicles (both for external computation purposes and breakdown suspicion)
6. Priority classes
7. Graphics display
8. Standing requests
9. Automatic billing
10. Provision of hard copy for use in a manual backup system"

Accordingly, the Acceptance Test Specification has a specific test for each of these capabilities. Further, paragraph 3.6.3.3 also states that "These tests will also show how the program accomplishes passenger assignment and vehicle routing in an efficient manner enabling stated pickup and delivery times to be met." Hence a required part of the acceptance tests are the heuristic efficiency and pickup/ delivery constraint scenarios, I.2.2 and I.2.3.

1.2 Design of the Acceptance Test

At least two points should be emphasized regarding the final acceptance test document.

a. The tests cover only those items that the work statement requires to be covered, as requested by MIT/USL.

b. The work statement is essentially non-quantitative. As a result, the acceptance test is a qualitative check in some parts and an attempt at quantification in others. Generally, the list of features is checked qualitatively, but the heuristic efficiency and pickup/delivery times are tested quantitatively.

Each test comprises one or more scenarios. The console inputs and expected outputs are written into each scenario, along with subsidiary I/O, such as high speed printer, graphic display, and card reader/punch.

The complete list of scenarios is:

I.2.2 Heuristic Efficiency

- I.2.2-1 A East-West Distribution Problem #1
- I.2.2-1 B East-West Distribution Problem #2
- I.2.2-2 A Clockwise Distribution Problem #1, version 1
- 1.2.2-2 B Clockwise Distribution Problem #1, version 2
- 1.2.2-2 C Clockwise Distribution Problem #1, version 3
- I.2.2-3 Clockwise Distribution Problem #1
- I.2.2-5 Group-Group Distribution
- I.2.2-6 Two Sector Distribution
- I.2.2-7 Four Sector, Four Vehicle Distribution
- I.2.2-8 FCFS Collection #1
- I.2.2-9 FCFS Collection #2
- I.2.2-11 Branch and Circuit Collection
- I.2.2-12 Diamond-Star Collection Problem
- I.2.2-13 Many-Two test
- I.2.2-14 Simple Many-Many
- I.2.2-15 2-Vehicle Many-Many

I.2.3 Delivery and Pickup Constraints

I.2.3-A	Consistency of Constraints
I.2.3-B	Violation of Constraints

I.2.6 Realistic Case (Cambridge)
I.3.0 Specific Capabilities

I.3.1-A	Restart A - Part of I.2.6
I.3.1-B	Restart B - Part of I.2.6
I.3.2	Cancellation of Service Request
I.3.3	Unexpected Situations
I.3.4	Vehicle Breakdown Procedures
I.3.5	Lateness Detection and Correction
I.3.6	Priority Classes
I.3.7	Graphics - Part of I.2.6
I.3.8	Standing Requests - Part of I.2.6
I.3.9	Automatic Billing - Part of I.2.6
I.3.10	Hard Copy for Manual Backup - Part of I.2.6

The heuristic efficiency test is a series of brief scenarios designed to test the efficiency of the program's vehicle assignments in simple situations. In most of these scenarios, the best routing assignment is obvious by inspection or hand calculation. These scenarios, unfortunately, do not test the heuristic efficiency in the most general many-many case since it is extremely expensive to calculate truly optimal solutions in general cases involving more than five requests.

The delivery and pickup constraint tests (I.2.3) are designed to determine how well the computer program would predict the actual pickup and delivery times. The scenario I.2.3.A was designed to incorporate real street travel time data previously taken by MIT/USL.

The specific capabilites that are tested in scenarios I.3.1 through I.3.10 are those called out in the work statement. Five of these scenarios were combined into a realistic case scenario, I.2.6, so that the interaction among them may be examined.

SECTION 2
RESULTS OF ACCEPTANCE TESTS

The data taken during the acceptance test, May 27 through June 24, are given in Volume II of this report following the scenarios. The first page of each scenario contains a description, statement of purpose, and the expected result. TSC's interpretation of the results are given as follows. (Notation: P indicates that the test was passed; Q indicates that passage of the test was questionable; F indicates that the test was failed. In the scenario diagrams, O indicates an origin and X indicates a destination. In most of the scenarios passengers are entered in the alphabetical order of their name, i.e., Arnie, Bob, Charlie, Dan, etc.).

2.1 Heuristic Efficiency Scenarios*

I.2.2-1A **East-West Distribution Problem** P

O D-A-R minimized the average arrival time by dropping off the third passenger who went to an equally distant destination (see diagram in scenario).

I.2.2-1B **East-West Distribution Problem** F

O D-A-R did not minimize the average arrival time. It delivered Arnie before Bob and Charlie because the program operates on an insertion principle* and therefore will not reassign a tour stop once assigned.

I.2.2-2A **Clockwise Distribution Problem #1** P

The program minimized the average travel time by dropping off as many passengers as possible as soon as possible, i.e., it made a clockwise tour.

I.2.2-2B **Clockwise Distribution Problem #1** F

This is the same test as in the preceding scenario, but with 2 vehicles. The O D-A-R program did not minimize the criterion. Bob should have been assigned to vehicle 2. Once a passenger has been assigned to a vehicle, O D-A-R will not reassign him to another vehicle.**

---

*See Appendix G for a description of the insertion rule and a description of the efficiency criterion used in tests I.2.2.
**See references in Appendix E, Vol II, for a discusion of reassignment.

I.2.2-2C **Clockwise Distribution Problem #1** F

O D-A-R did not minimize the criterion; the deliveries should have been in the reverse order. This also occurred because of the insertion rule (see I.2.2-1B)

I.2.2-3 **Clockwise Distribution Problem #2** P

The program properly delivered the group first and then the other two requests. Also, the two zero trip length requests were properly rejected.

I.2.2-5 **Group-Group Distribution** F

O D-A-R failed to deliver the nearer group first because of the insertion principle (see I.2.2-1B).

I.2.2-6 **Two-Sector Distribution**

In both case A & B, O D-A-R assigned one bus to each section, as expected.

I.2.2-7 **4 Sector, 4 Vehicle Distribution** P

The program dispatched one bus to deliver each of the four groups of passengers, as expected.

I.2.2-8 **FCFS Collection #1** P

Because he called in first, Arnie should be picked up and delivered to the collection point; then Bob is serviced. Both requests require the same amount of travel time. Arnie was serviced first and then Bob, as expected.

I.2.2-9 **FCFS Collection #2** P

D-A-R serviced the customers who had been waiting longer, all other things being equal.

I.2.2-11 **Branch and Circuit Collection** F

The heuristic should produce a circuitous tour or branching tour, depending on which minimizes the criterion (tour time plus total of delivery times) for this two-passenger collection problem.

O-D-A-R made the correct choice in 8 out of the 14 configurations presented to it (Cases A, B F, G, H, J, M, N) and the wrong choice in the other 6 cases (C, D, E, I, K, L).

Much better results are obtained when, in calculating the delivery time for each passenger, 30 seconds is allowed for each prior boarding or alighting but not for his own final alighting. (The total trip time then, includes boarding and alighting times for all passengers but that does not influence the comparison of branch vs. circuit.) With this criterion, the best results agree with the calculated results in all but one case, (1). No explanation could be found for that one discrepancy, despite the best efforts of MIT and TSC.

I.2.2-12 Diamond-Star Collection Problem P

The O D-A-R program assigned the vehicle to pick up and deliver the passengers one at a time: first Arnie then Charlie, Bob, and Dan in that order. The resultant star pattern is superior to any other, including the diamond shaped tour: Arnie, Bob, Charlie, Dan, home.

I.2.2-13 Many-Two Test P

In both case (a) and (b) the heuristic sorted out the seven requests into a group that goes to one destination, and another group going to another destination, and assigned the two buses appropriately. (See Figures 1 and 2.)

I.2.2-14 Simple Many-Many P

The test verified that the heuristic will select the minimum cost tour to service two requests by one bus, given a simple configuration for the origins and destinations.

I.2.2-15 2-Vehicle Many - Many F

In the case (a), the program did not dispatch one vehicle to handle each group. Vehicle 2 should have handled Bob's request (see Figure 3). In case (b), however, very good tours were produced for the same requests (see Figure 4). The only difference between the two cases was

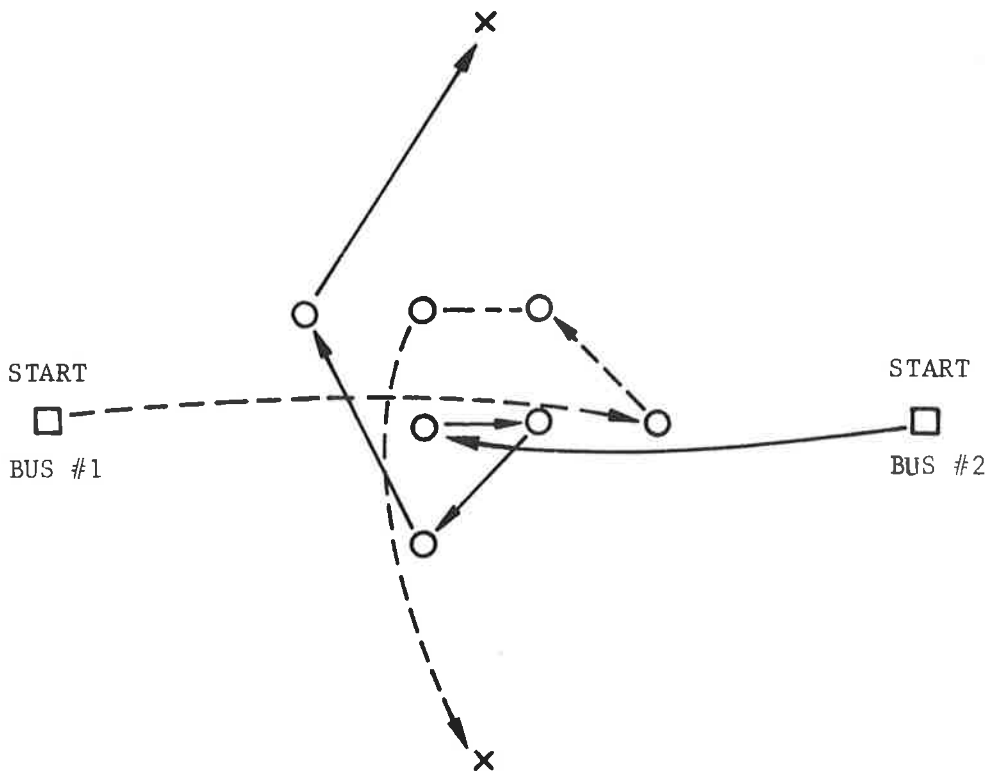


Figure 1. Tours Assignment by O D-A-R
for Scenario I.2.2-13 (a)

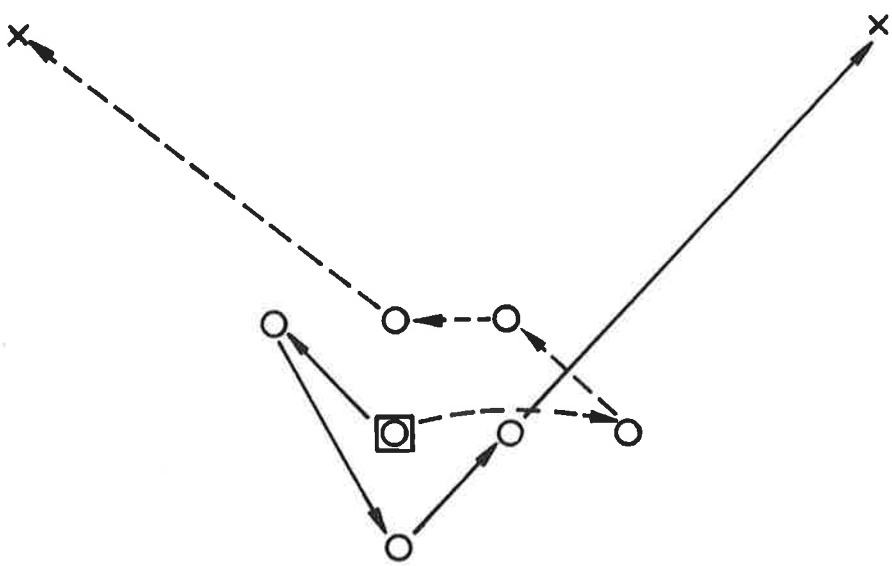


Figure 2.Tours Assignment by O D-A-R
for Scenario I.2.2-13 (b)

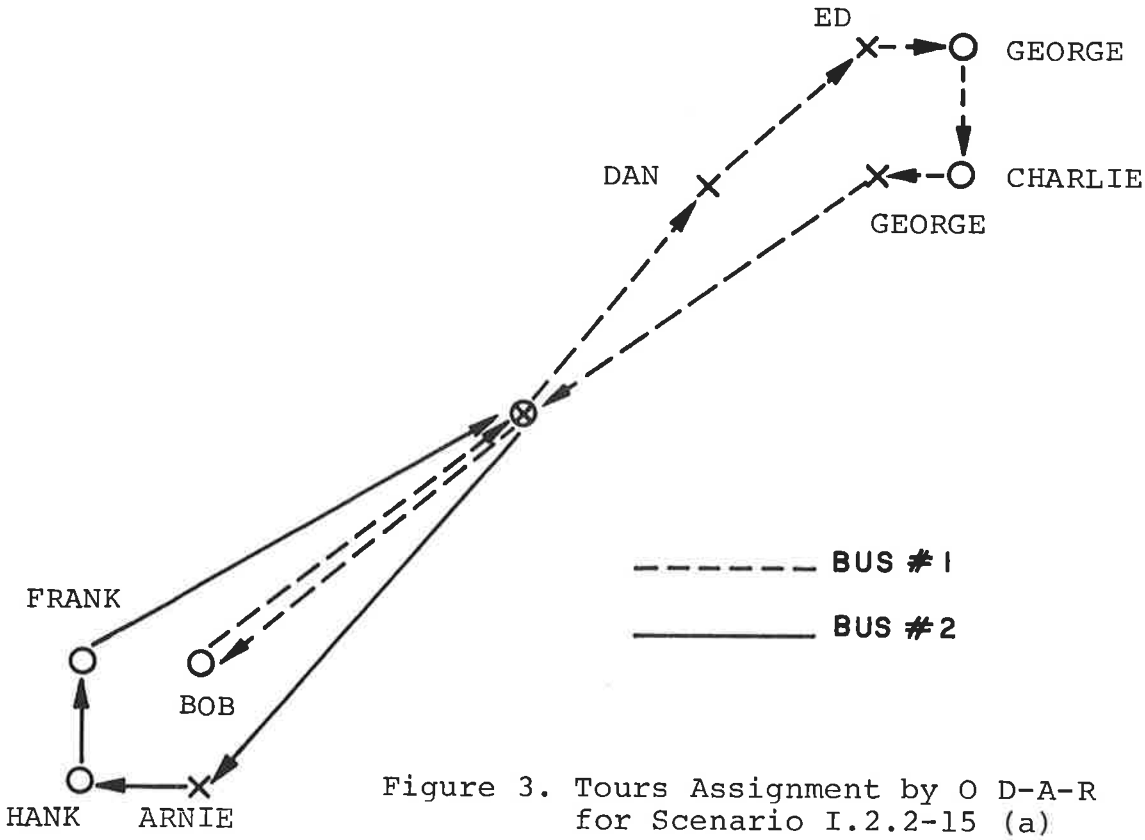


Figure 3. Tours Assignment by O D-A-R
for Scenario I.2.2-15 (a)

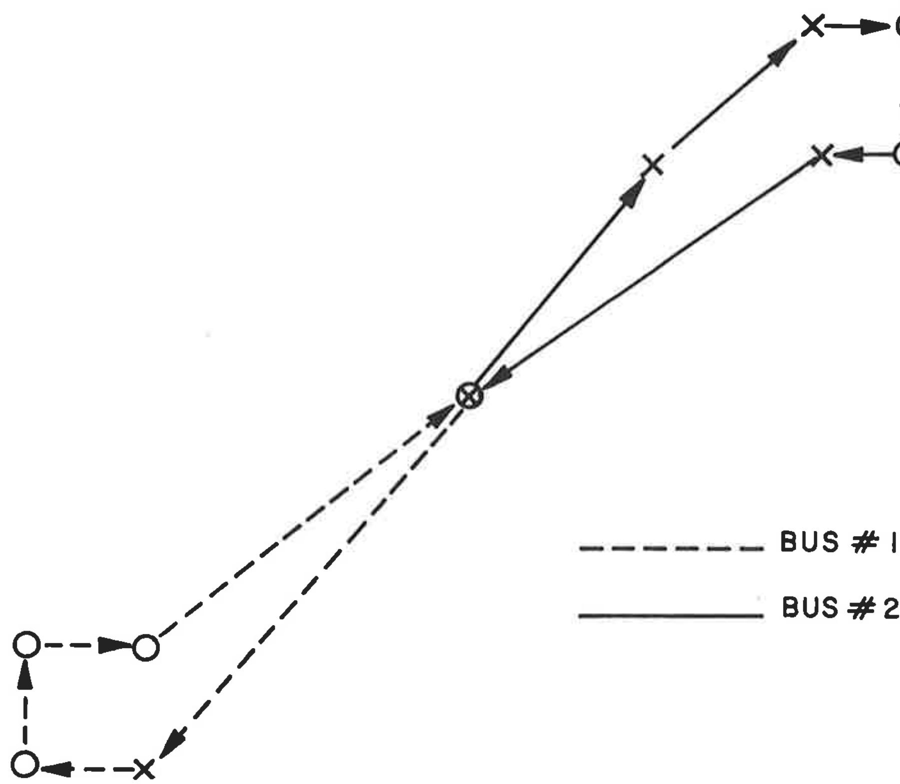


Figure 4. Tours Assigned by O D-A-R
for Scenario I.2.2-15 (b)

the method of locating the vehicles at 50 50 st before the requests were entered: in case (a) the vehicles were entered by the RESTO commands for vehicle 2 and the LOCA command for vehicle 1; in case (b), both vehicles were given dummy passengers to deliver to 50 50 st.

A full explanation of the results of this test could not be determined, but it appears that the prior dummy assignments influenced the tours.

2.2 Pickup and Delivery Constraints

I.2.3-A Consistency F

This scenario was an attempt to verify that the O D-A-R program enables the "stated pickup and delivery times to be met." In order to quantify this requirement, the scenario was designed to compare the pickup and delivery times predicted by the program with those that would be achieved by a vehicle on Cambridge streets. It was planned to employ data previously taken by MIT/USL on Cambridge streets. These data, and criteria of acceptability, were submitted by USL on 23 July 1971. (See Appendix E, Vol. II)

The scenario was constructed on a total of 30 tour links and 2 vehicles. This was considered by TSC to be the minimum sample size on which a conclusion could be based. (See Appendix H)

When the street data were compared to the scenario data, however, it was found that almost half of the tour links in the scenario had no counterparts in the street data. Figure 5 shows the tour links, within the service area of the computer's street map, for which driving data were furnished; Figure 6 shows the tour links of the scenario. The number of useable tour links is about 12, far too few to have any statistical significance.

Moreover, the O D-A-R pickup and delivery time prediction accuracies failed to meet the criteria recommended by USL, even if the vehicle stop times used in the scenario are assumed to be realistic. This is true whether one uses

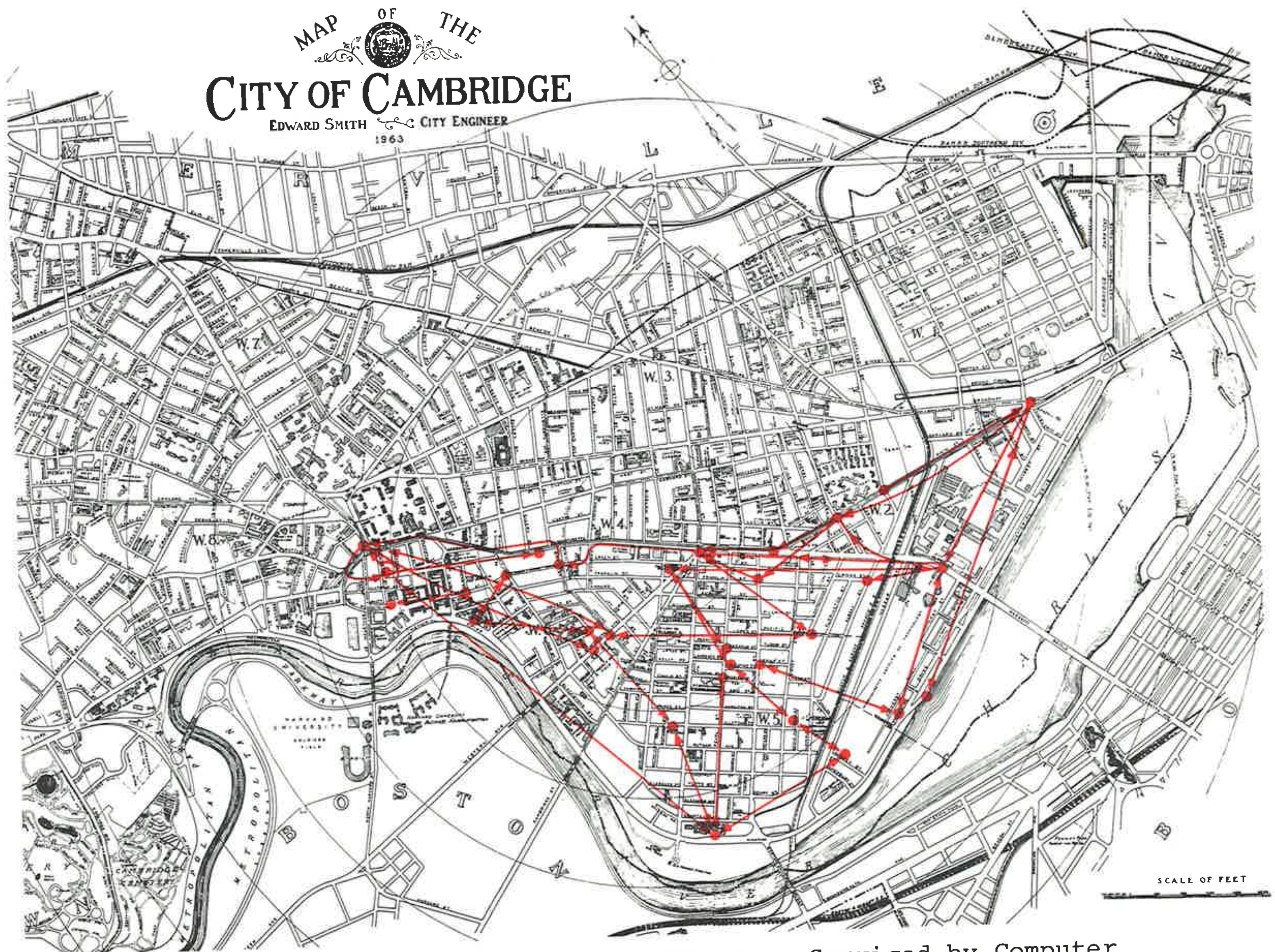


Figure 5. Plot of Trip Lines Within Area Serviced by Computer Map in Scenario I.2.3-A, Corresponding to MIT Travel Time Data of Reference 1, Appendix E, Volume II

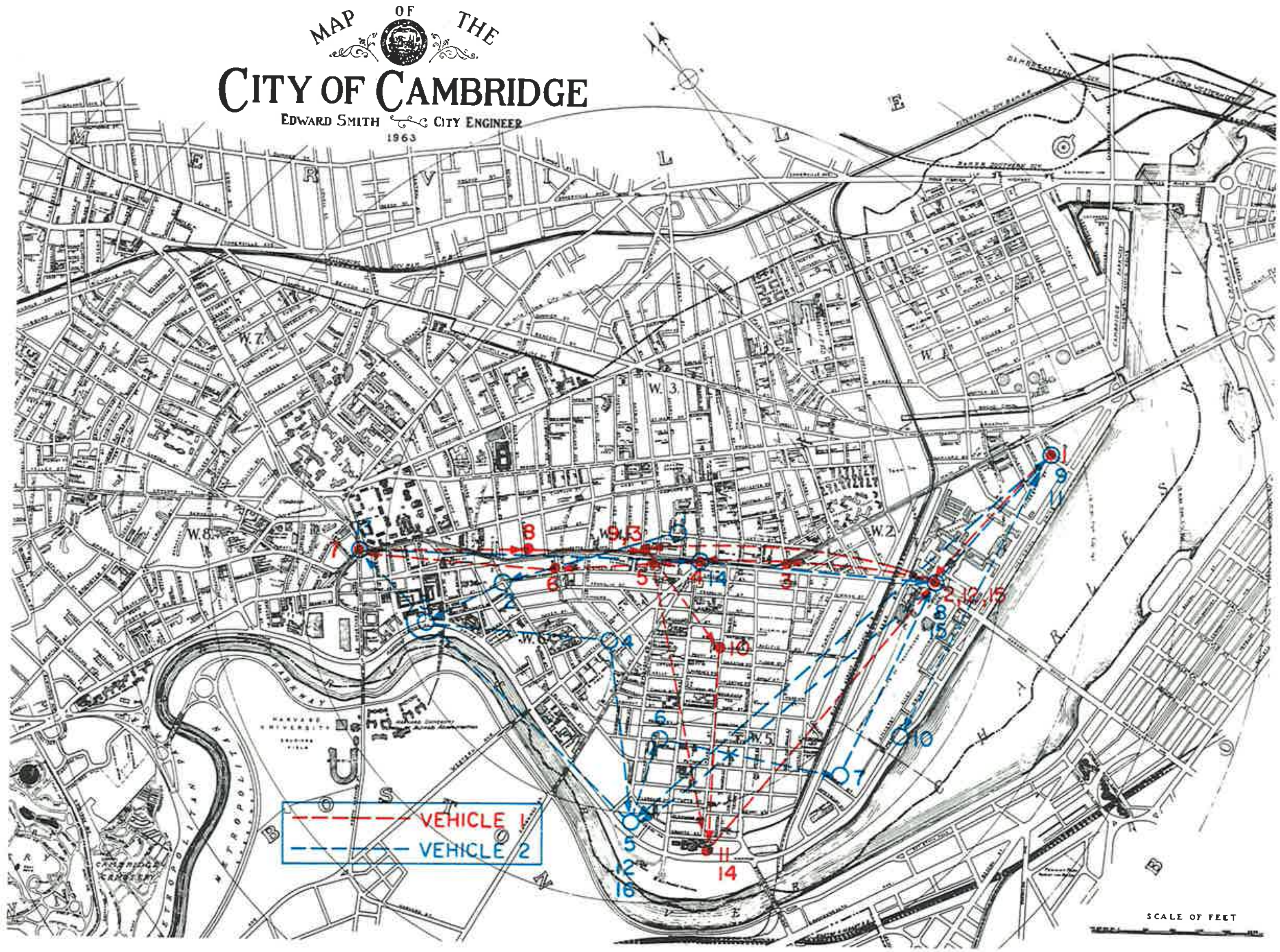


Figure 6. Plot of Trip Requests Entered During Scenario I.2.3-A

the times from the vehicle console sheets (Tables 1 and 2) or from the transaction file (Table 3) produced on the printer. (See Appendix F of Volume II for the transaction file.)

P

I.2.3-B Violation

In addition to the pickup and delivery times stated to the passenger, the O D-A-R has internal constraints on estimated waiting time, delivery time and total service time that must be satisfied before a vehicle is assigned. This scenario showed successfully that vehicle 1 would not be assigned to pick up Bob if he was located so that his estimated waiting time was more than the required 4 minutes. In such cases the second vehicle was assigned as expected.

I.3.1 Restart

The restart feature of O D-A-R was tested in scenario I.2.6. A computer failure was simulated (by hanging up the operator's console, which detached DOS from the 360/67) at time 14:22:00. While the backup mode procedures were being carried, all terminals were inactive. The length of the backup period was of intermediate term length (see reference 2., pages 27-28, for the definition of an intermediate term failure).

This length was selected because the corresponding restart procedure is more general than that for a short term failure or that for a long term failure.

At time 15:00:00 the program was restarted by entry of the job control stream given in Appendix J of Volume II. This job stream would ordinarily be entered through a card reader.* These cards indicated to the computer the last stop and the next stop of each vehicle at time of restart. The OSTP command, as explained in section 4, the Backup Mode Review is an undocumented command. In the test its effect

*In the test, because the computer was remotely located, the first part (up to the DOWN entry) was taken from a disk file and the remainder typed in through a console.

TABLE 1
TOUR OF VEHICLE 1 FOR SCENARIO I.2.3-A
CONSTRUCTED FROM CONSOLE SHEETS

ACTUAL TIME * H M S	LOCATION	P/D PASS.	EXPECTED TIME ** H M S	+EARLY -LATE H M S
161643	CARS	P BOB	161715	32
161848	77 Mass Ave.(MIT)	D BOB	162215	327
161856	77 Mass Ave.	P DAN	162140	244
162153	425 Mass. Ave	P ED	162353	200
162449	Central Sq.	P FRANK	162450	1
162617	Post Office	D DAN	162840	223
162839	550 Green St.	D ED	162953	114
163619	Harvard Sq.	D FRANK	163250	-329
163630	Harvard Sq.	P JUDY	163351	-239
164259	1045 Mass. Ave.	D JUDY	163851	-408
164958	City Hall	P KATHY	163949	-1009
165323	20 Lopez St.	D KATHY	164449	-834
165448	Morse School	P OLIVIA	164847	601
170035	77 Mass Ave.	D OLIVIA	165547	-448
170043	77 Mass Ave.	P PAULA	170252	209
170546	City Hall	D PAULA	170952	406
171116	Morse School	P TRICIA	171147	31
171547	CARS	D TRICIA	172147	600
171547	CARS	UNASSIGNED		

* Time vehicle command was entered, as determined from time that dispatch message emerged on vehicle console.

** Predicted on basis of message to passenger on passenger console.

TABLE 2
TOUR OF VEHICLE 2 FOR SCENARIO I.2.3-A
CONSTRUCTED FROM CONSOLE SHEETS

ACTUAL TIME * H M S	LOCATION	P/D	PASS	EXPECTED TIME ** H M S	+EARLY -LATE H M S
161548	YWCA				
161751	20 Putnam	P	ARNIE	161849	58
162018	100 Plympton St.	P	CHARLIE	162019	01
162548	300 Western Ave.	P	GEORGE	162320	-228
162848	705 Memorial Dr.	D	CHARLIE	162819	-29
163150	200 Allston St.	P	HARRY	162939	-211
163517	200 Vassar	D	ARNIE	162649	-828
163527	200 Vassar	D	GEORGE	163420	-107
163717	77 Mass. Ave.	D	HARRY	163739	22
163745	77 Mass. Ave.	P	INGRID	163848	103
164015	CARS	D	INGRID	164348	333
164024	CARS	P	LINDA	164135	111
164327	Joyce Chen	D	LINDA	164935	608
164657	CARS	P	MARTHA	165148	451
164948	705 Memorial Dr.	P	NORA	170116	1128
170316	Harvard Sq.	D	MARTHA	170548	232
170331	Harvard Sq.	D	NORA	171016	645
170341	Harvard Sq.	P	QUENTIN	170252	-49
170349	Harvard Sq.	P	RACHEL	170417	28
171603	Central Sq.	D	QUENTIN	171052	-511
171946	MIT	D	RACHEL	171617	-329
171956	MIT	P	SUSAN	171515	-441
172354	705 Memorial Dr.	D	SUSAN	172415	21

* See note in Table 1.
** See note in Table 1.

TABLE 3
TOUR OF VEHICLE 1 FOR SCENARIO I.2.3-A
CONSTRUCTED FROM TRANSACTION FILE

PASS NAME	P/D	EXPECTED Minutes*	ACTUAL Minutes*	EARLY(+) LATE(-)
BOB	P	3.95	4.55	-.60
	D	6.99	6.65	.34
DAN	P	8.50	6.78	1.72
	D	13.61	14.13	-0.52
ED	P	11.16	9.72	1.44
	D	15.51	16.50	-.99
CHARLIE	P	7.74	8.15	-.41
	D	13.14	16.65	-3.51
ARNIE	P	6.00	5.70	.30
	D	12.43	23.12	-10.69
GEORGE	P	10.84	13.65	-2.81
	D	19.45	23.30	-3.85
FRANK	P	11.74	12.65	-.91
	D	18.52	24.13	-5.60
HARRY	P	16.69	19.68	-2.99
	D	22.83	25.13	-2.30
INGRID	P	25.66	25.60	.06
	D	28.70	28.10	.60
JUDY	P	20.81	24.33	-3.49
	D	23.76	30.82	-7.08
LINDA	P	29.14	28.25	.89
	D	34.78	31.28	3.50
KATHY	P	27.23	37.80	-10.57
	D	29.76	41.22	-11.46
OLIVIA	P	35.93	42.65	-6.72
	D	41.56	48.43	-6.87
MARTHA	P	39.53	34.78	4.75
	D	50.69	51.12	-.43

TABLE 3 (CONTINUED)

PASS NAME	P/D	EXPECTED Minutes*	ACTUAL Minutes*	EARLY(+) LATE(-)
NORA	P	48.61	37.63	10.98
	D	55.78	51.37	4.41
PAULA	P	50.32	48.57	1.75
	D	55.42	53.60	1.81
TRICIA	P	58.82	59.12	-0.30
	D	66.98	63.62	3.36
QUENTIN	P	50.20	51.53	-1.33
	D	55.91	63.90	-7.99
RACHEL	P	51.13	51.67	-.54
	D	61.31	67.62	-6.31
SUSAN	P	62.77	67.78	-5.01
	D	69.41	71.73	-2.32

*Minutes from start of computer run.

was to nullify the entry preceding it which was the last stop entry for vehicle 3, an undesired result. Because of this, the transaction file (Appendix H, Vol. II) does not show vehicle 3 stopping at Harvard Square.

With the minor problem connected with the OSTP command, the program correctly reconstructed the tours of all vehicles as was determined by an examination of the transaction file. Hence it is concluded that the restart was successful.

I.3.2 Cancellation of Service Request P

This scenario was executed successfully as follows: vehicle 14 picked up the first passenger at Kendall Square and breaks down upon arrival at the second pickup, Faculty Club. Vehicle 3 comes to the rescue, but the first passenger cancels out before being picked up. Vehicle 3 is then assigned to pick up only the second passenger at the Faculty Club. (Note that the description, page 1 of the scenario, is in disagreement with the actual test I/O. The description should read, "Immediately cancel the first request," rather than, "Immediately cancel both requests.")

I.3.3 Unexpected Situations P

Although the work statement requires only that the program handle more passengers at a stop than expected, this feature was expanded by USL to include the cases of fewer passengers, or no passengers at all, appearing at the stop. The scenario is based on this expanded feature.

An analysis of the console outputs shows that the program does accommodate more or fewer passengers at a stop than expected. However, there are several aspects of the design of this feature, brought out by the scenario, that require clarification.

1. More passengers may be accommodated at a stop only if their desired destination, priority class and billing requirements are the same as that for one of the previously scheduled passengers at the stop. If these conditions are not met, the extra passengers must telephone the Dial-A-Ride center operation at the passenger console.

2. The anom xxxxyyyy command relates to the subsequent command rather than to the previous one. The scenario was written under the assumption that the anom 14 2 command (page 3 of the scenario, top line) relates to the vehi 14 arrival command two lines prior to it. As a result, the test data are valid but do not correspond to the expected results given on the first page of the scenario.

3. The vehicle delivery message, unlike the pickup message, does not show how many passengers are involved. Such an indication would be helpful as a check in unexpected situations. The program apparently has a minor bug that

caused the message number 255 to be emitted after, rather than before, message number 117. (Note: the 3 in CRS225 is a typographical error not related to the program.)

P

I.3.4 Vehicle Breakdown Procedure

The program successfully reassigned the passenger FIRST to vehicle 16 when vehicle 14 broke down, and reassigned passenger SECOND to vehicle 18 when vehicle 7 broke down. Moreover, the program did not make further assignments to the broken down vehicles.

P

I.3.5 Lateness Detection and Correction

In case (c) O D-A-R assigned Bob to vehicle 1 even though that vehicle had exceeded its expected pickup time for Arnie by more than 3 minutes, the waiting time constraint. It was expected to have assigned vehicle 2 to pick up Bob, as called for in the scenario description. This test, case (c), was rerun with more than a four-minute lateness for vehicle 1, and it then worked as expected, assigning vehicle 2 to Bob's request. The discrepancy was traced to a one-minute quantization in arrival prediction, so that lateness had to exceed 4 minutes (3 minutes waiting time constraint, plus 1 minute quantization) before a lateness is detected.

The program successfully detected lateness of 5 minutes for vehicle 1 and printed out the proper message at the vehicle console.

Two references were submitted, at TSC's request, in support of the thesis that reassignment of passengers waiting for a late vehicle does not improve efficiency (See Appendix E of Volume II). The first of these references does, in TSC's view, support that contention.

P

I.3.6 Priority Classes

A priority class is defined by the constraints that must be met by the vehicle servicing passengers in that class. When Arnie and Bob are both of the same priority, vehicle 1 picks up Arnie and detours to service Bob before delivering him; but when the test is repeated with a higher priority for Arnie than for Bob, Arnie is brought directly to his destination and Bob must wait for another vehicle to be picked up. This results in poorer service for Bob, but better service for Arnie, as is desired.

The realistic case scenario, I.2.6, also contained two priority requests, Mr. Bang and Ralph Nader. Mr. Bang was assigned to vehicle 4, and delivery time was estimated to be 30 minutes from assignment. This was the longest estimated time to delivery of 27 assignments made during the scenario, that were not standing requests, even though the trip was approximately average length. (The next longest estimated delivery time was 24 minutes; delivery time estimates are not available for standing requests.) The reason for such a relatively poor delivery time estimate for a high priority request may be the following: if the algorithm cannot find a vehicle to satisfy the high priority constraints, these constraints are set to infinity, which effectively removes them. An unrealizable priority request is thus treated only on the basis of a higher weight in the selection criterion.

A second priority request, Ralph Nader, was entered into scenario I.2.6 at time 13:50:43. The pickup and delivery times estimated by the program were 5 minutes and 10 minutes. Although the pickup time is only slightly longer than the average (4.08), the predicted delivery time is better than most.

The level of service predicted for the two priority requests may be compared to that for all requests of I.2.6 by reference to Figure 7. This Figure shows that Ralph Nader was delivered more promptly than the other passengers; it also shows that Mr. Bang seems to have been discriminated against. It should be borne in mind that the predicted pickup and delivery times depend on the length of the trip.

In summary it may be said that the scenario I.3.6 demonstrated that O D-A-R does have a priority service feature; in view of the test results of I.2.6 however, the priority service feature must be examined in more detail prior to practical use.

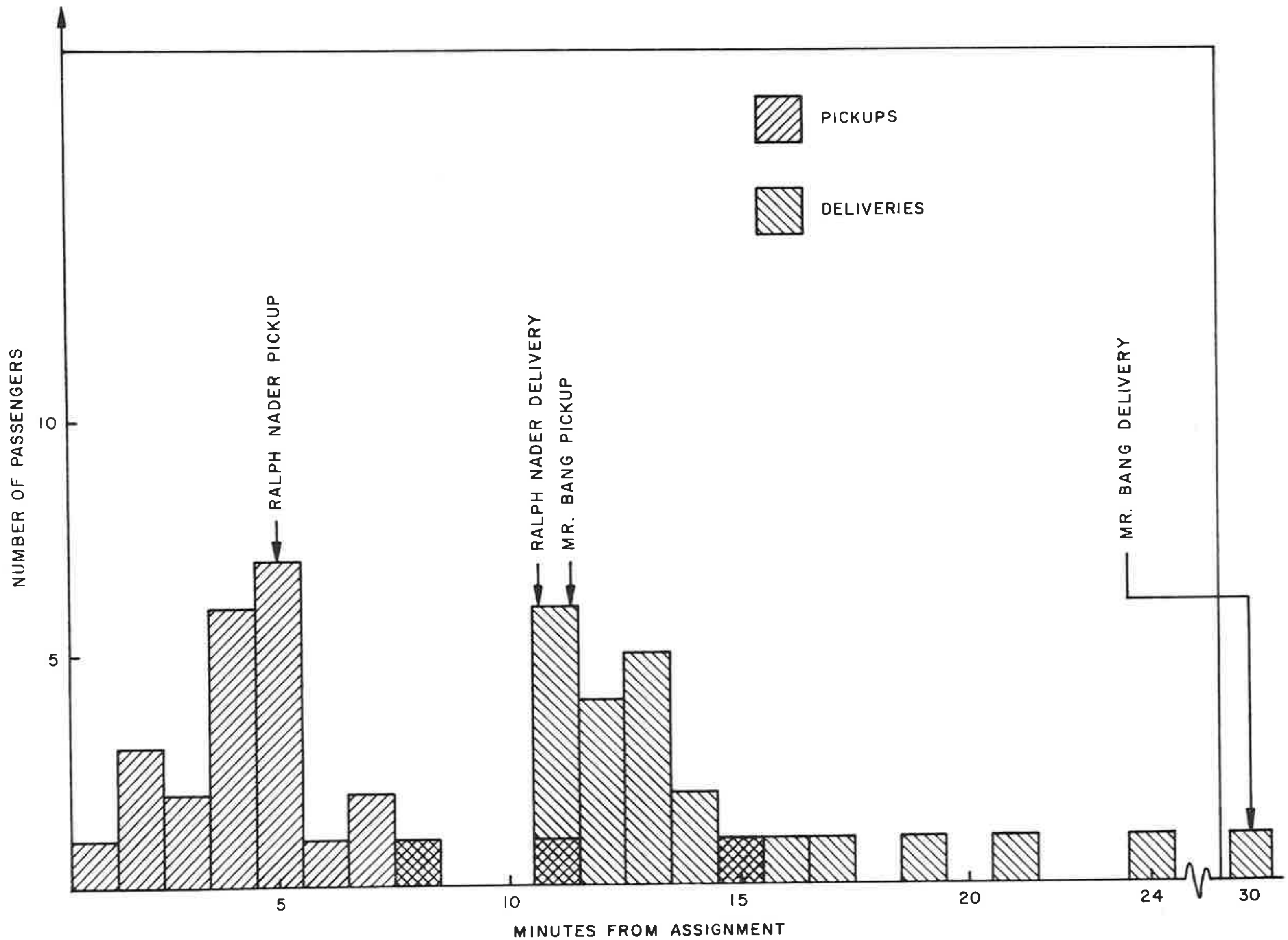


Figure 7. Distributions of Times to Pick up and Delivery Predicted by O D-A-R in Scenario I.2.6

I.3.7 Graphics

Q

The nature of the graphics capability required of the program is described by the work statement, 3.4.1, as one "which will enable supervisors to visually monitor the state of the system, and which can also serve as an educational device for new personnel or for other people interested in observing Dial-a Bus in operation". Appendix E, reference letter 4, Volume II, gives more detail, supplied by MIT/USL, on these functions of the graphics. TSC's analysis of the graphics display is given in the form of 3 questions and corresponding answers.

Question 1: Are the display commands adequate to specify the state of the system?

Yes. The available commands as listed in the User's Manual cover all aspects of demand and capacity that may be of use. For the demand these include desired trips for any passenger, for all who have waited more than a specified time, etc. For capacity, they include vehicle positions and number of passengers, projected tours, and past tours for a single vehicle or for all vehicles.

Question 2: Is the information produced by the display commands effectively presented?

No. Most of the commands were exercised during the realistic case scenario on June 9, 11, and 16 with the following results:

VEHI ALL: This command should display the positions of all the vehicles as well as the number of passengers on each. Figures 8 through 13 are photographs of the ARDS screen during scenario I.2.6. They show six vehicle numbers with passenger contents, next to six dots in an otherwise blank rectangle encompassing the service area (part of Cambridge). Only the most experienced eye can be expected to deduce from this the positions of the vehicles in Cambridge. In fact, it is difficult to judge the positions of the vehicles relative to the X-Y grid, since there are no X-Y scale markings either.

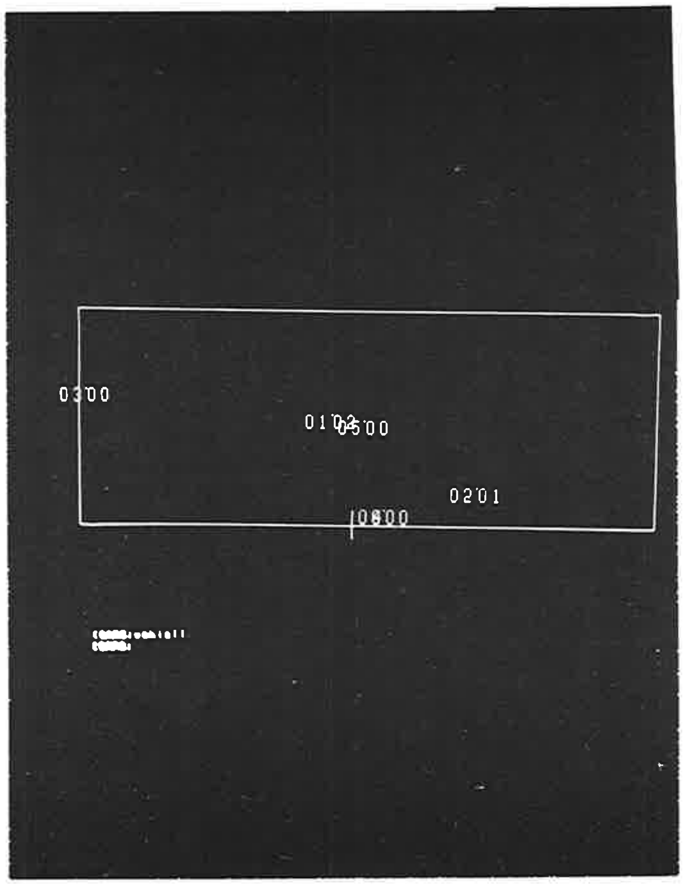


Figure 8. Photo of ARDS Screen After VEHI ALL Command, Taken During Scenario I.2.6, June 16, 1971

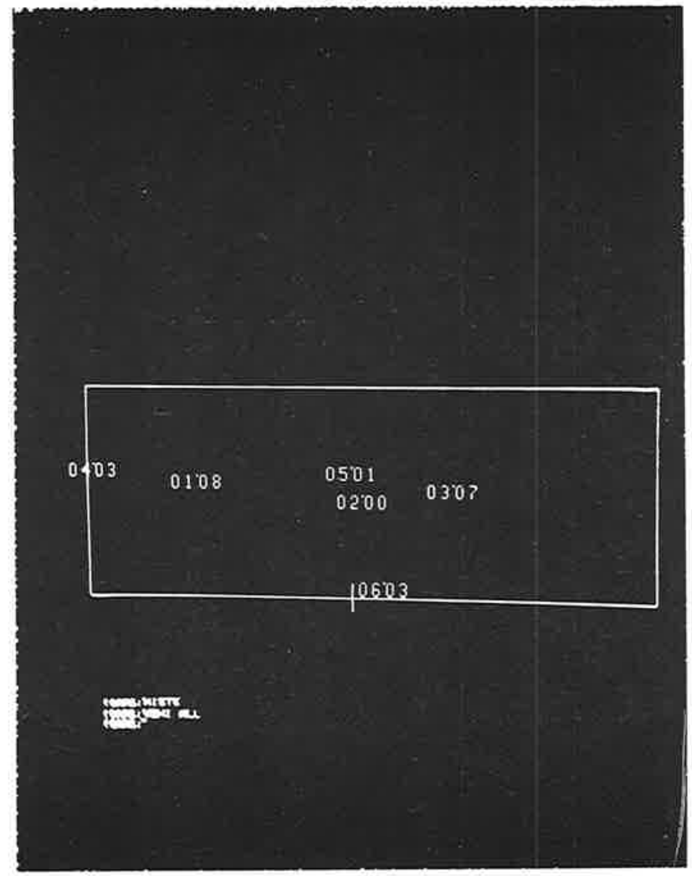


Figure 9. Photo of ARDS Screen After VEHI ALL Command Taken During Scenario I.2.6, June 9, 1971

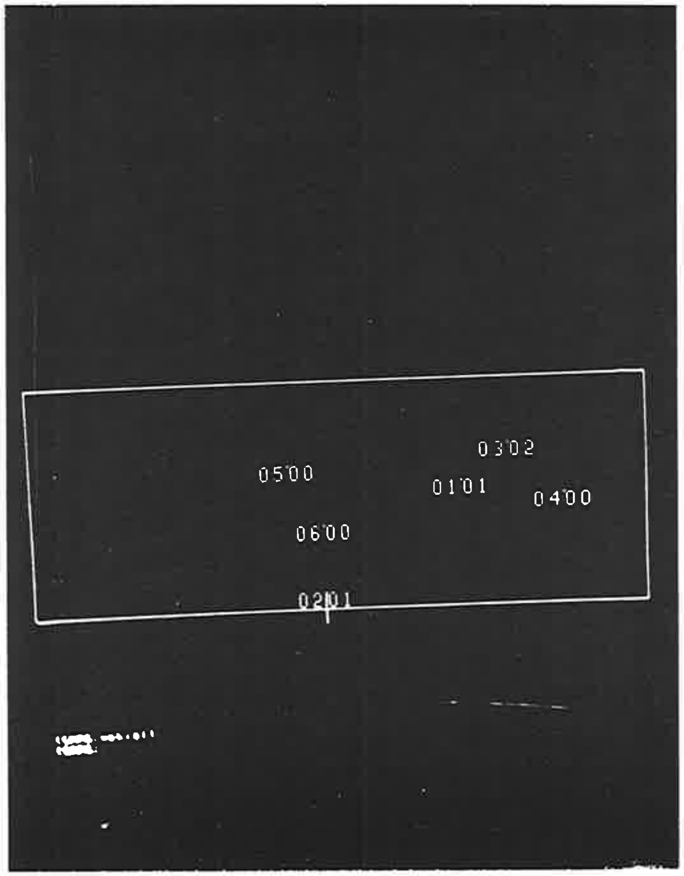


Figure 10. Photo of ARDS Screen After VEHI ALL Command, Taken During Scenario I.2.6, June 11, 1971

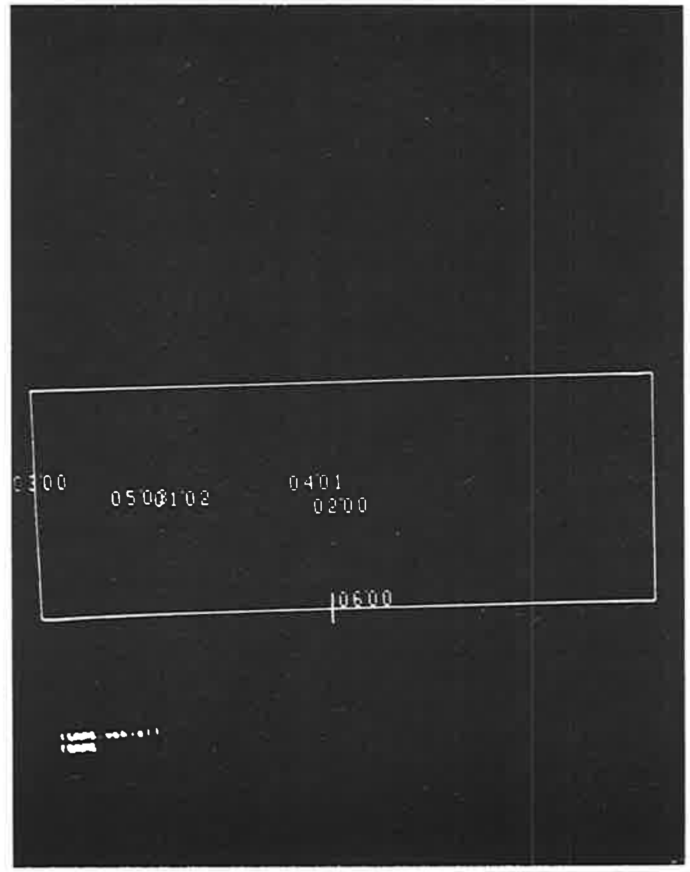


Figure 11. Photo of ARDS Screen After VEHI ALL Command, Taken During Scenario I.2.6, June 11, 1971

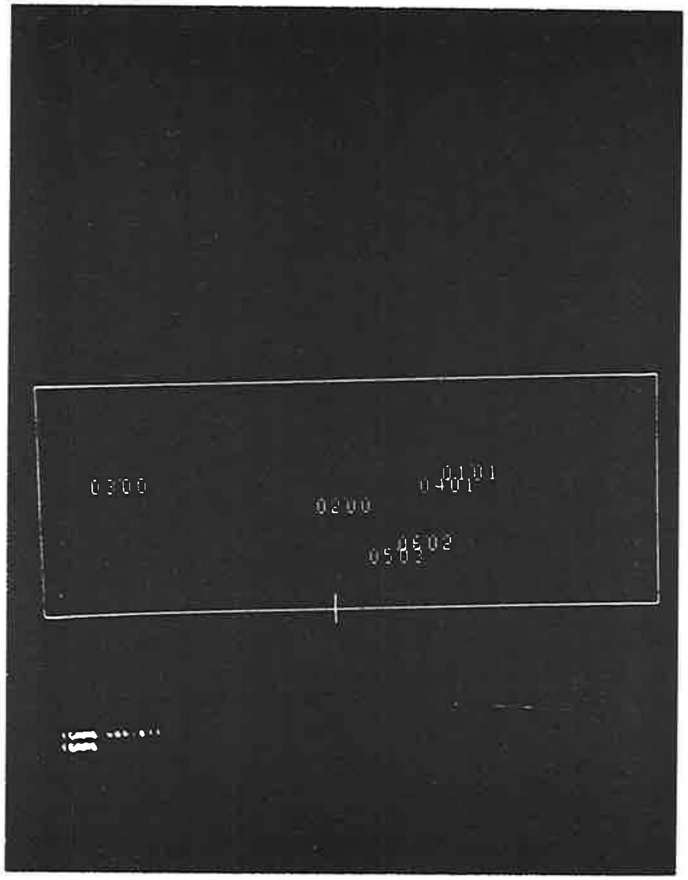


Figure 12. Photo of ARDS Screen After VEHI ALL Command, Taken During Scenario I.2.6 on June 11, 1971

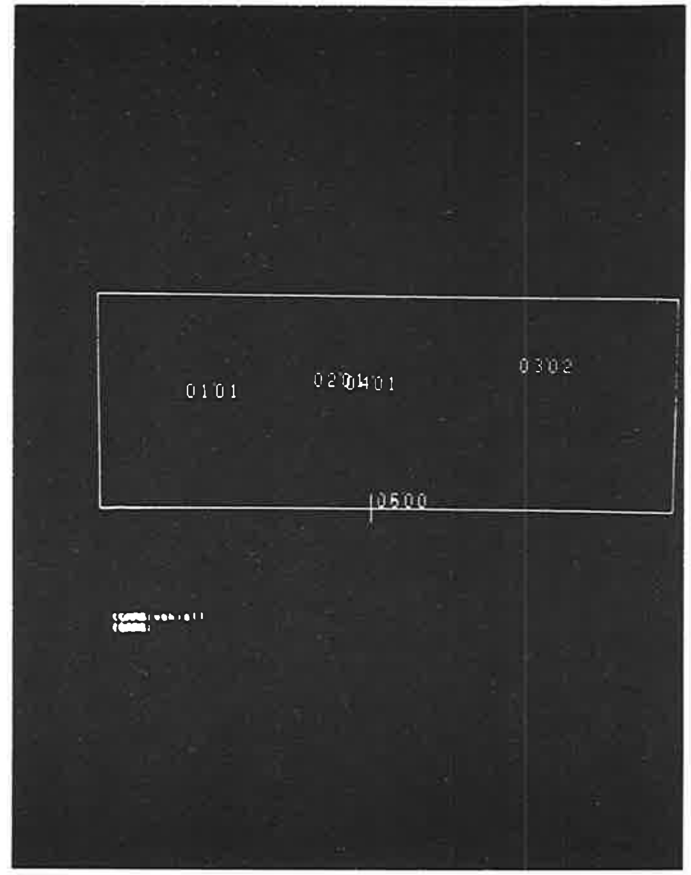


Figure 13. Photo of ARDS Screen After VEHI ALL Command, Taken During Scenario I.2.6 on June 11, 1971

VEHI XX: The tour of vehicle 6 is shown in Figure 14. Once again, no relation to the street network is shown. The vehicle's tour starts at the "06" (present position) and ends by depositing passenger 36 at time 43 minutes. (The characters that do not fit within the right edge of the ARDS screen appear at the left edge, on the next line.) Passenger 36's name is not given on the screen; the supervisor must obtain it by getting the origin and destination of 36 from the backup console and then finding that O/D on the passenger console sheets, which will give the name. The times displayed are minutes from the start of the program, rather than real time. The result of this command for multiple tours is shown in Figures 15 and 16. These photographs illustrate the common problem of overwrite in graphical displays.

PASP (name): The result of giving this command for SRA10 is shown in Figure 17. In this case, the passenger's name appears in the lower left of the screen somewhat blurred. The numbers in the rectangle are the passenger's number and the time at which his request was received by the computer. Clearly, this display can be improved by presenting the passenger's name, origin, destination, and vehicle assignment in large characters above or below the rectangle.

ASSN (name): This command displays the tour of the vehicle assigned to service the named passenger. Figures 18A and 18B show the result of ASSN MAJOR MAJOR being keyed into the display. Without knowing MAJOR MAJOR's passenger number, it is almost impossible to tell which tour leg services him. Even with his number, the displays are difficult to interpret.

PASX NN: This command is not described in the User's Manual. It is apparently equivalent to the WAIT X command, which displays the tours of all customers who have been waiting more than X minutes. The result of giving the PASX 10 command is shown in Figure 19. The overwrite problem is very apparent.

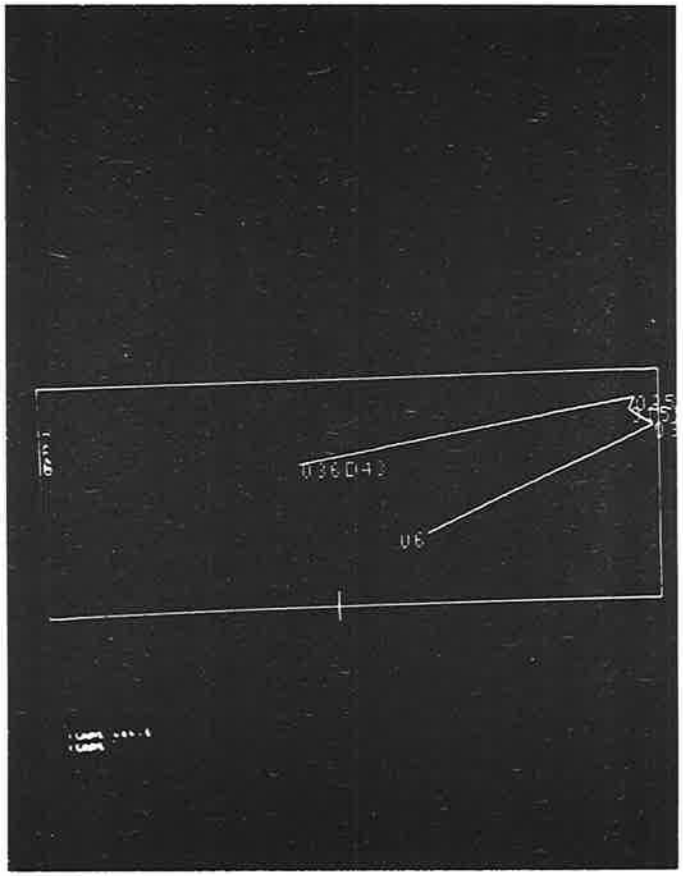


Figure 14. Photo of ARDS Screen After VEHI 6 Command, Taken During Scenario I.2.6 on June 16, 1971

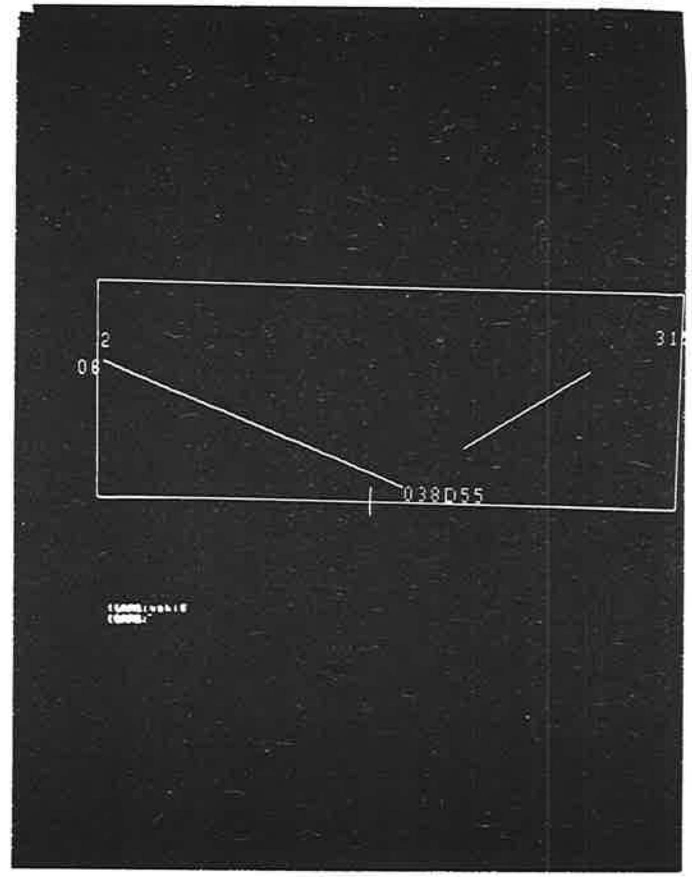


Figure 15. Photo of ARDS Screen After VEHI 6 Command, Taken During Scenario I.2.6 on June 16, 1971

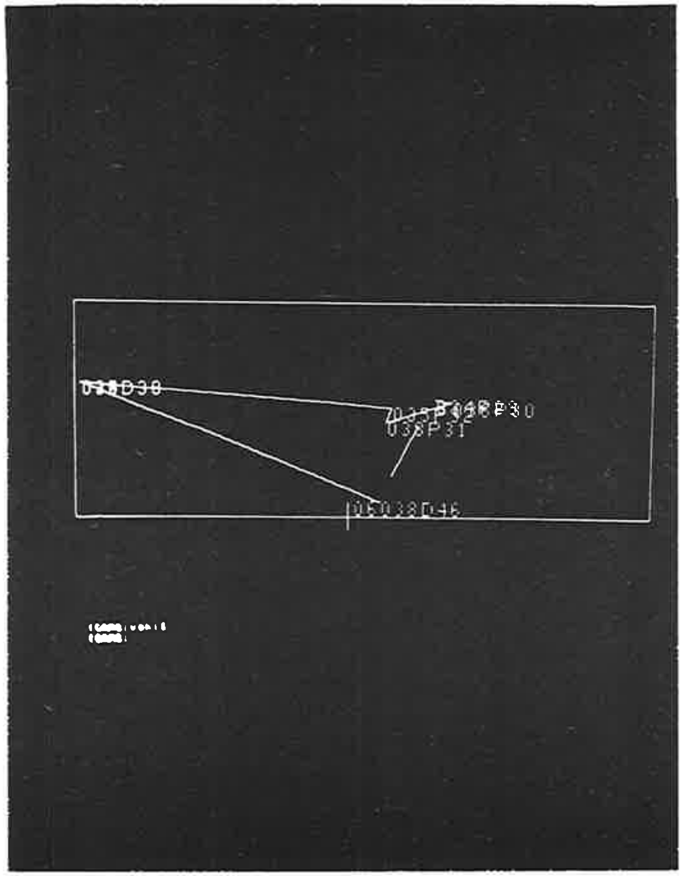


Figure 16. Photo of ARDS Screen After VEHI 6 Command, Taken During Scenario I.2.6 on June 16, 1971

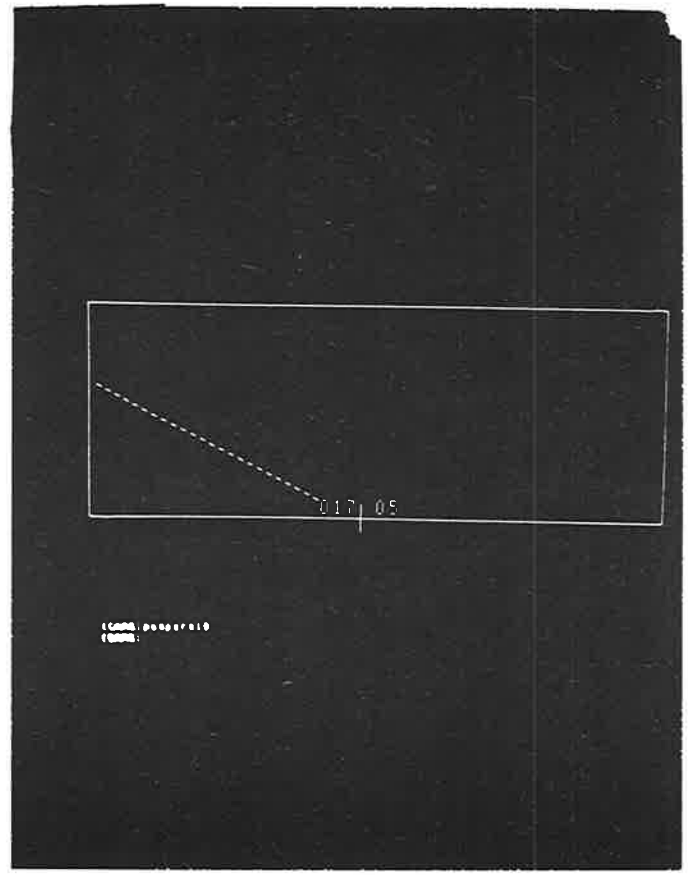
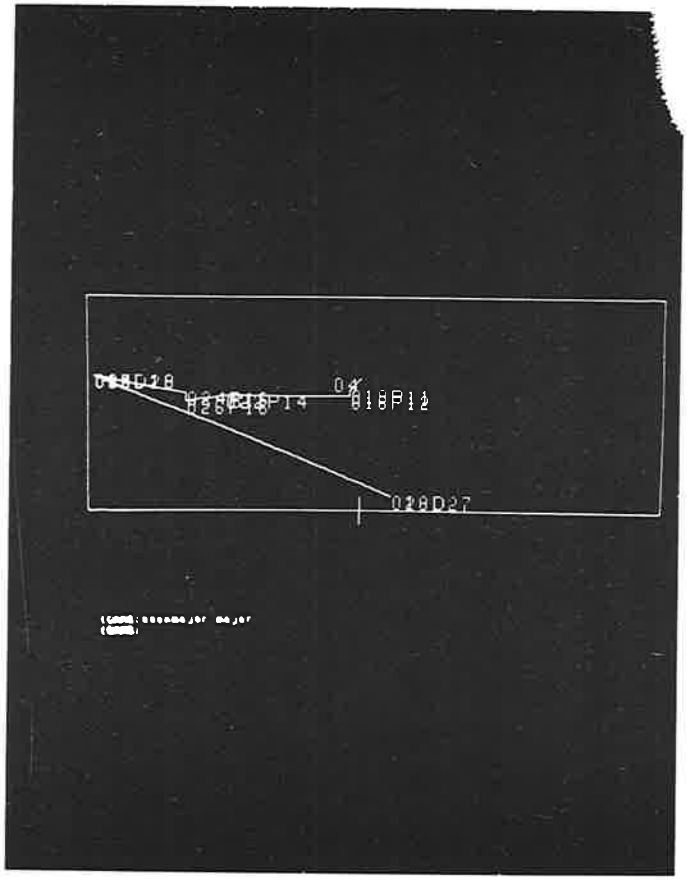


Figure 17. Photo of ARDS Screen After PASP 1ϕ Command, Taken During Scenario I.2.6 on June 11, 1971



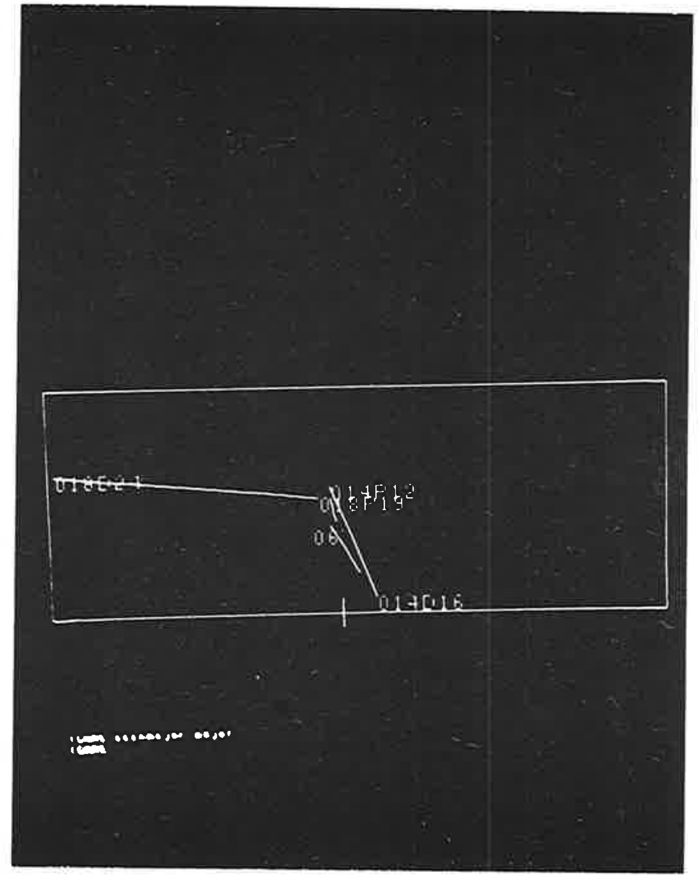


Figure 18. (A and B) Photos of ARDS Screen After ASSN MAJOR MAJOR Command, Taken During Scenario I.2.6 on June 16, 1971

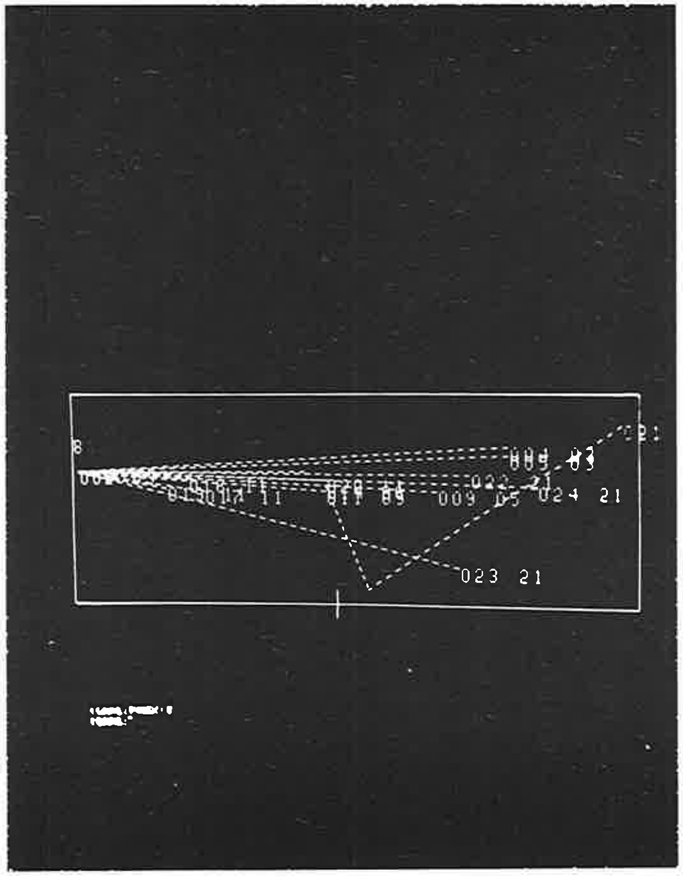


Figure 19. Photo of ARDS Screen After ASSN MAJOR MAJOR Command, Taken During Scenario I.2.6, on June 11, 1971

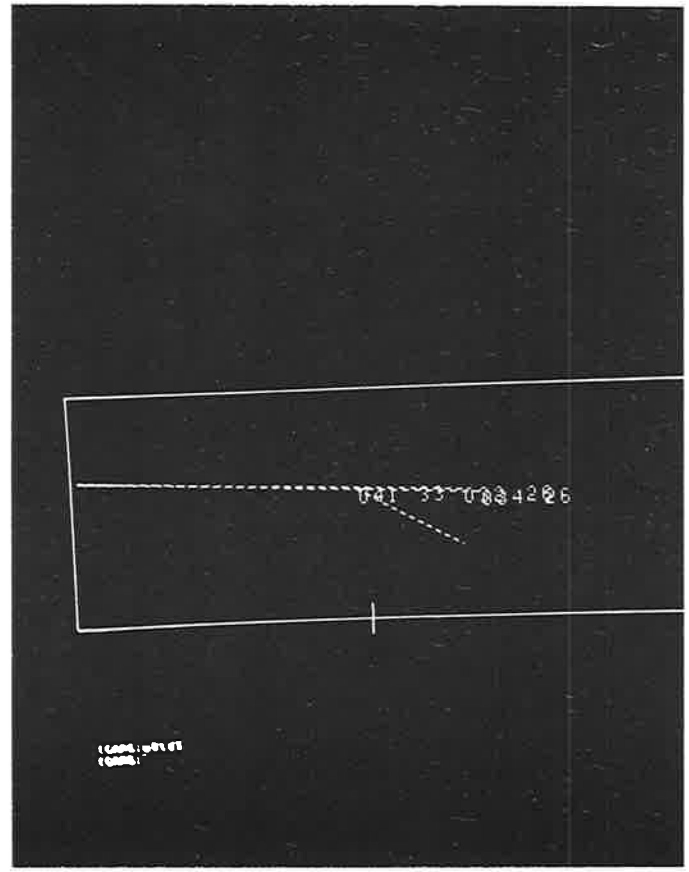


Figure 20. Photo of ARDS Screen After W8TD 5 Command, Taken During Scenario I.2.6 on June 11, 1971

W8TD X: This command displays the tours of all passengers currently being carried who have waited more than X minutes to be picked up. Figures 20 and 21 show the ARDS screen a few seconds after W8TD 5 was keyed in during the realistic case scenario. Once again, passengers are identified only by number on the screen, rather than the name used when the request was entered.

WAIT X: The results of the command WAIT 5, Figures 22 and 23, are similar to those of the W8TD X command. The passenger numbers and non-realtime of estimated pickup are shown. It differs from the W8TD X command in that only passengers currently waiting for pickup are displayed.

HIST X: Figure 24 shows the results of a HIST 6 command. The vehicle is not identified except in the small letters at the lower left. It shows that the vehicle picked up passenger 17 at (non-real) time 17, delivered passenger 10 at time 18, picked up 30 at time 23, etc.

<u>Question 3</u>: <u>Can the graphics serve as an educational device for new personnel and for other people interested in Dial-a-Bus?</u>

Yes. The graphics show, in an impressive fashion, that the computer makes assignments of passengers to vehicles, keeps track of vehicle positions and of customer locations, and is aware of how long each customer is waiting. It also shows how vehicle routes are modified by new requests. Although the graphics cannot fully educate one in the operation of the system, it serves as a good introduction to the O D-A-R.

I.3.8 <u>Standing Requests</u> P

This feature was tested as part of I.2.6. The data verify that the program does allow entry of standing requests, and that those entered were properly handled. A total of 35 standing requests were entered before the beginning of the test, through the input file. They were as follows: (See Volume II, scenario I.2.6, page 114, for detailed list)

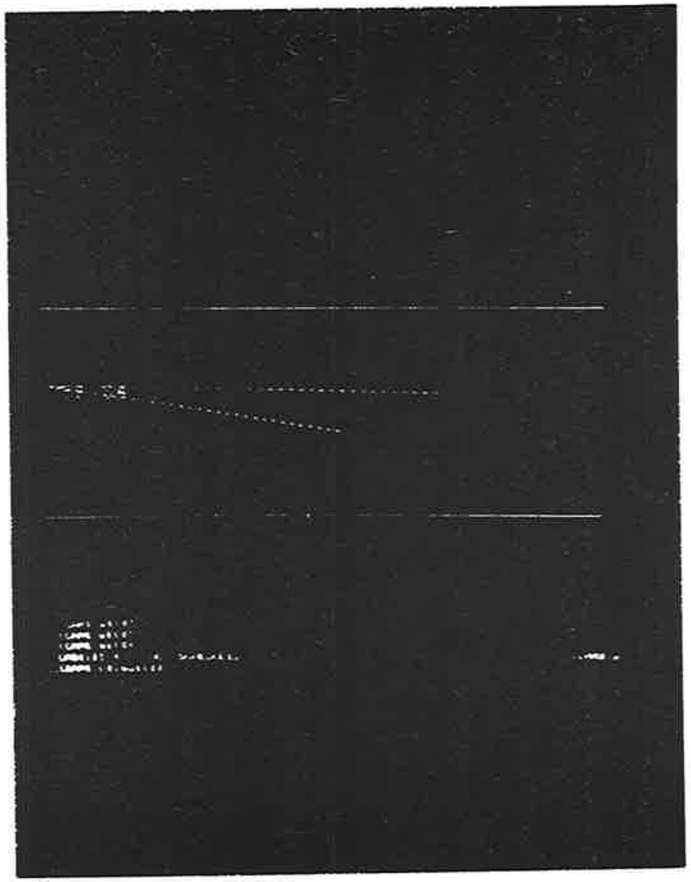


Figure 21. Photo of ARDS Screen After W8TD 2 Command, Taken During Scenario I.2.6, on June 16, 1971

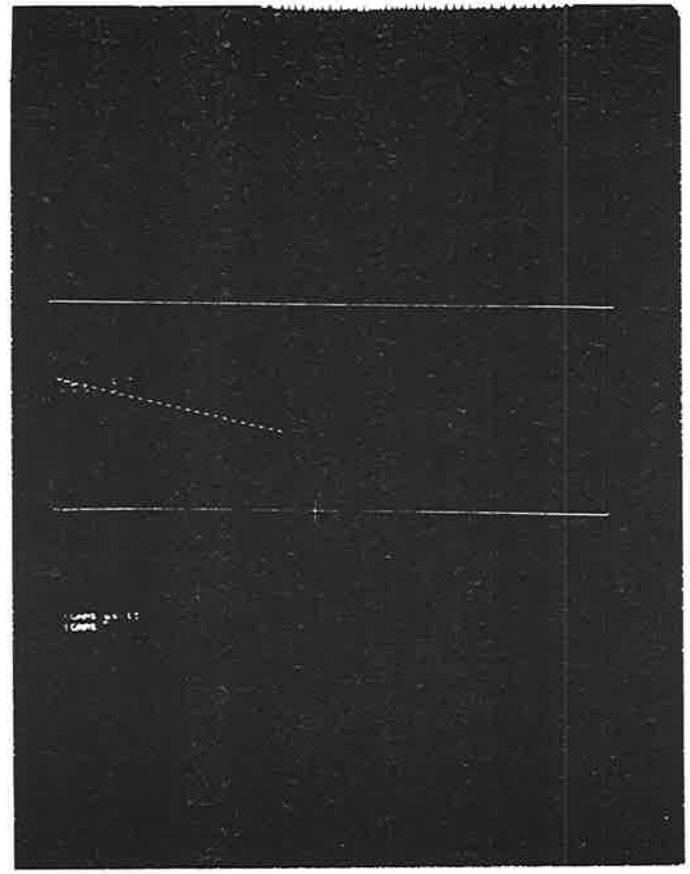


Figure 22. Photo of ARDS Screen After WAIT 5 Command, Taken During Scenario I.2.6, on June 16, 1971

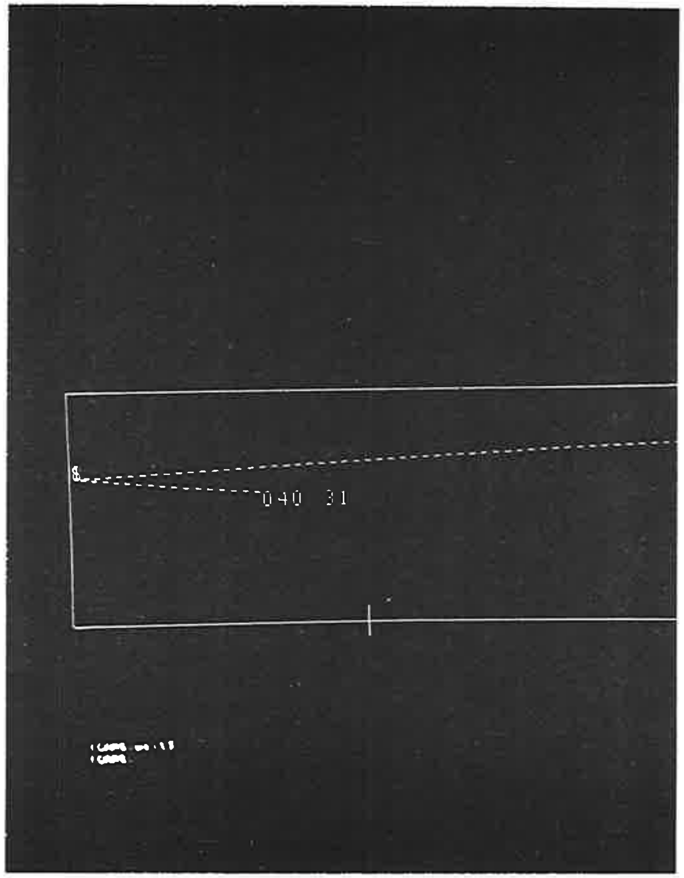


Figure 23. Photo of ARDS Screen After WAIT 5 Command, Taken During Scenario I.2.6, on June 11, 1971

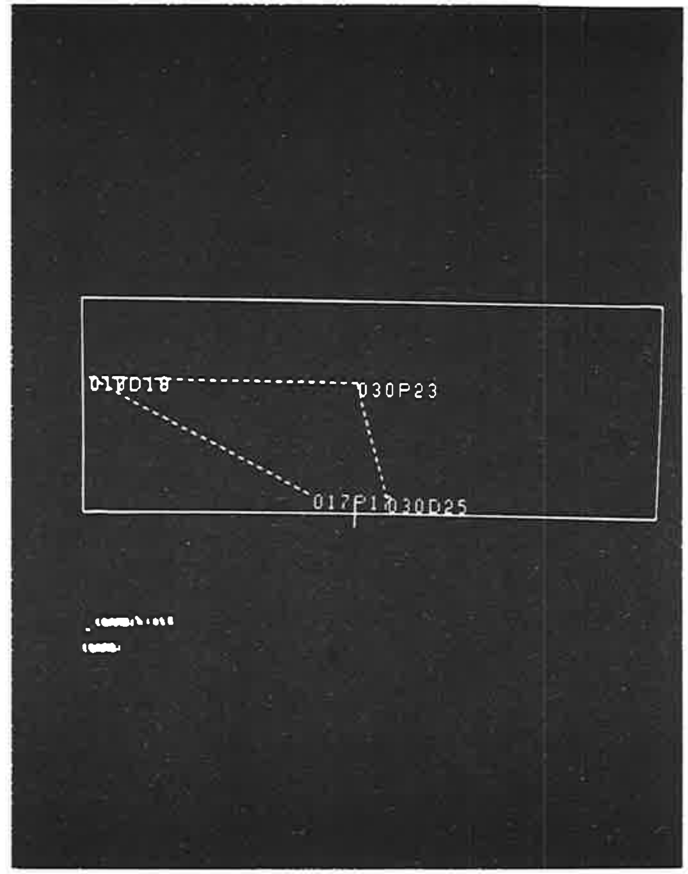


Figure 24. Photo of ARDS Screen After HIST 6 Command, Taken During Scenario I.2.6, on June 16, 1971

Name	Destination	Pick up Time
SRA1 through 6	Harvard Sq.	8:08
SRA7 through 10	Harvard Sq.	8:10
SRA11 through 15	Harvard Sq.	8:15
SR1 through 10	Harvard Sq.	8:25 to 8:35
SMA1 through 10	Morse School	8:05 to 8:30

Where S indicates standing request, R indicates rail station destination, M indicates Morse School destination, and A indicates automatic billing.

The above information, plus origin, account number, priority class and number of passengers per request were recorded on punched-cards, as described on page 29 of the User's Manual. They were then processed through the standing request program (SR) which runs independently from the O D-A-R program. The result was two punched cards per request, in A format, that were entered into O D-A-R at initialization time.

Of the 35 standing requests, 21 were picked up and delivered before the (simulated) computer failure; two were picked up before the failure and delivered during it, three were picked up during and delivered after, and one customer (SR2) was picked up before the computer went out of service and delivered after it returned to service.

The service rendered standing requests by O D-A-R may be judged by the closeness with which the stated pickup and delivery times approach the desired, for customers serviced before the computer failure. This is shown in Table 4. The reason that the actual pickup and delivery times are not employed in the comparison is that those used in the scenario are only approximations of what would be achieved by real vehicles and are not representative of actual pickup and delivery times; the projected times, however, are a measure of the software's efficiency in making vehicle assignments.

TABLE 4
DESIRED AND PROJECTED SERVICE TIMES
FOR STANDING REQUESTS IN I.2.6

PICKUP TIMES*

NAME	DESIRED	PROJECTED BY PROGRAM		**PROJECTED TRIP TIME	**PROJECTED DELIVERY TIME	VEHICLE NO.
SRA 1	08	07.18	-.82	7.06	14.24	3
2	08	8.97	.97	8.43	17.40	3
3	08	8.77	.77	9.71	18.48	3
4	08	8.08	.08	10.94	19.02	3
5	08	8.92	.92	6.38	15.30	6
6	08	15.79	7.79	7.18	22.97	2
7	10	10.27	.27	8.24	18.51	3
8	10	9.83	-.17	4.97	14.80	4
9	10	10.91	.91	5.33	16.24	4
10	10	10.15	.15	6.29	16.44	6
12	15	14.86	-.14	3.78	18.64	4
14	15	16.01	1.01	3.64	19.65	4
SMA 1	05	5.19	.19	4.80	10.09	6
2	05	4.47	-.53	6.81	11.28	2
3	05	8.68	3.68	7.30	15.98	5
4	10	9.63	.37	4.97	14.60	2
5	10	12.00	2.00	5.21	17.21	1
6	15	16.03	1.03	11.73	27.76	4
7	15	16.92	1.92	3.52	20.44	5
8	20	24.72	4.72	3.17	27.89	2
9	25	25.39	.39	4.28	29.67	6

*Minutes from start of test, as given in the scenario sheets for desired, and in the transaction file for projected.
**As given in the transaction file.

An inspection of Table 4 shows that the projected pickup times are, indeed, close to the desired pickup times: 96.5% are within 6 minutes and 87% are within 3 minutes. These results should be compared with those of scenario I.2.3.A; it will be found that the service projected for standing requests is considerably better than that given to telephoned requests, at least for pickups. From the table, one also may infer how effective the algorithm may be in delivering commuters to a train; for if all the SRA requests of Table 4 wished to catch a train at Harvard Square (their common destination) at, 23 minutes after start, their average time from projected pickup to train time would be 12.19 minutes, of which an average of 6.81 minutes is actual travel time.

It should be born in mind that the results of Table 4 are based on projected pickup and delivery times; the connection with actual street times was not established by scenario I.2.3.A, q.v.. Moreover, for reasons that are not apparent, the table shows different projected delivery times at Harvard Square for passengers who should arrive together, according to the dispatching messages on the vehicle console and the tours on the backup console. The difference in projected delivery times is 4.85 minutes for vehicle 4 (passengers SRA8, 9, 12, 14) and 3.32 minutes for vehicle 2 (passengers SMA2, 4). The explanation for these differences was not available at the time this report was prepared.

One more result having to do with standing requests was obtained from scenario I.2.6. At the start of the test, each of the six vehicles was immediately assigned to a pickup. Figure 25 shows the positions of the vehicles (squares) and of their assigned pickups (circles) at the time of assignment. It appears that the vehicles have been assigned deliberately to make the most distant, rather than the closest, pickup. The reason is that the algorithm attempts to minimize the difference between the expected pickup time and the pickup time specified in the standing request. Since the standing requests are considered for assignment at 4.5 minutes before the requested pickup time,

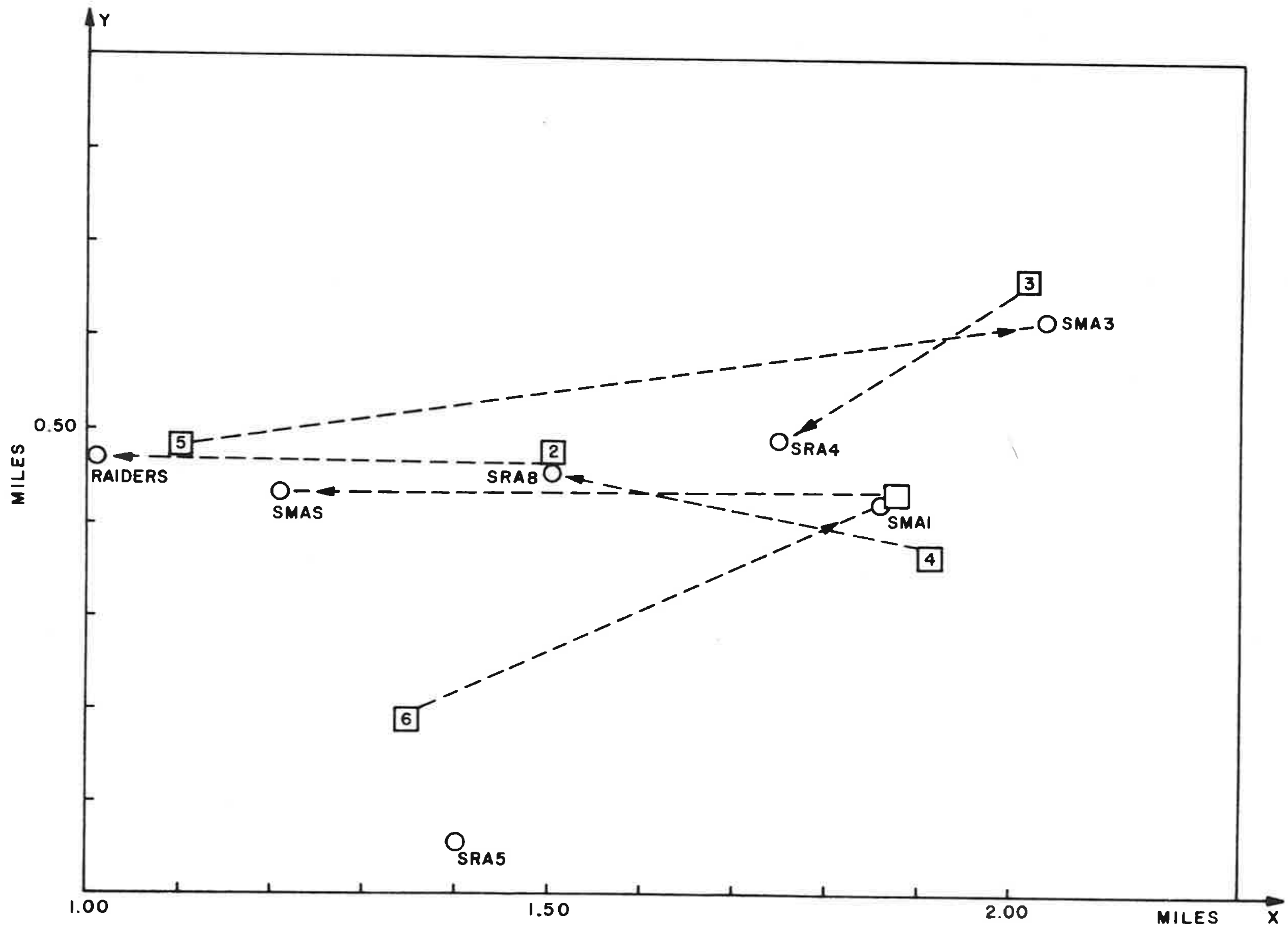


Figure 25. Position of Vehicles and Their Assigned Pickups Shortly after Start of Scenario I.2.6

the initial assignments tend to make the first leg of each tour equal to 4.5 minutes. Although the startup situation may be improved by changing the 4.5 minutes to, say, 1.0 minute (via input ATOTLC, line 8 of input parameter file) this may produce poorer results than the 4.5 value after the initial conditions have been superseded by dynamic routing conditions.

In examining the standing request feature, several improvements seemed to suggest themselves:

1. The pickup time may be entered more conveniently in local time, rather than minutes from start of the program. The start of the program is variable and would have to be inserted each day.

2. It would be convenient if the standing request input card included the days on which service is required, since some people will prefer not to be picked up on, say, Saturday or Sunday or Holidays. This would avoid assembling a new standing request deck each morning.

3. The latest desired time of drop-off is of concern to a customer who wishes to meet a train or to get to school or a job on time: it should be on the standing request card. The present program has no way of assigning vehicles so as to make such deliveries. The proper time to make a pickup, so as to assure on-time delivery, must be decided by the customer and/or the operator who takes the request. There is no way to guarantee that the algorithm will achieve a desired delivery time, since it is not known to the program. An adequately early pickup must be selected and relied upon in each case.

I.3.9 Automatic Billing Q

The O D-A-R program does not produce bills automatically; it produces two IBM punched cards for each transaction, containing information to be used in a billing program. This billing program is to be supplied by the community in which the D-A-R system is installed.

The billing cards are mixed in with all the other transaction cards provided by the program. They cannot be interpreted by inspection; the information must be extracted by the billing program, which must also separate the billing cards from the other transaction cards.

Although they have not produced a billing program, MIT/USL has provided stand-alone programs to construct and to update a customer account file. (See Volume IV of the Operational DOS Program Description , pp. 437-447).[6] An independent evaluation of this program, and of the transaction cards, was performed by contractor to TSC. This evaluation is given in Appendix D.

The transaction cards produced during scenario I.2.6 included all automatically billed trips. SRA 1 through 15, SMA 1 through 10, SR 1 through 10, as well as Roger Williams, Ralph Nader, Raiders, and Mrs. Uhaul. It did not produce a transaction card for billing maid 3, who had not been deposited by the end of the scenario. These transaction cards were read in and interpreted by a separate program (not a deliverable item) that produced the full transaction file given in Appendices G and H of Volume II.

The transaction cards from scenario I.2.6 were examined and the following points were noted:

a. No cards are produced for customers who were both picked up and dropped off during manual backup.

b. The date of the transaction is not included on the card; also, the time is minutes from start of the computer, rather than real time of day.

c. Standing requests are not indicated on the cards, unless the "standard trip" number is used for this purpose. Hence it is not possible to give them special rates (i.e. lower by the month, charges for no-show, fixed weekly rate). (Note that "standard trips" are an undocumented feature of automatic billing, Vol. IV, ODOS Program Description).

d. The documentation of the transaction cards is inadequate and inaccurate (See Documentation Review, Section 3 of this report). Several other aspects of the automatic billing feature are apparent from scenario I.2.6.

e. Fraud checking is done only through the account number. The data of the customer file are not available to the operators to assist in checking whether the customer is legitimately using the account. But photographic identification may be preferable to prevent use of stolen cards. However, the program does successfully execute a consistency check on the inserted account numbers (i.e. acct. 1030365 invalid, but 1020365 valid during scenario I.2.6).

f. Pickup and delivery zones were all zero; the documentation does not explain how a non-zero zone can be recorded on the transaction card. The street map file has no provision for zones.

g. The "standard trip" numbers produced during I.2.6 and printed in the transaction file seem to make no sense whatever.

h. The transaction cards cannot be interpreted on a standard keypunch and hence cannot be inspected visually. Such visual inspection can be helpful in detecting errors in the program, errors in the card, operator or passenger errors, or in informing the supervisor of recent trips. The information contained on the present transaction cards, plus the date, trip number and other useful information can easily be punched in readable form on two cards, as follows:

ITEM	FORMAT
Transaction code*	I2
Date of request	A20
Time of request	I6
Origin address	A20
Origin coordinates	I6
Origin zone	I1

* including standing request information, yes or no

ITEM	FORMAT
Estimated pickup time	I6
Actual pickup time	I6
Destination address	A20
Destination zone	I1
Destination coordinates	I6
Estimated delivery time	I6
Actual delivery time	I6
Passenger account number	I8
Vehicle number	I2
Number of passengers	I2
Priority class	I1
Trip number for that day	I4
Spaces for legibility 31	31X
	160

Because TSC was aware early in the project that the automatic billing feature would not provide bills, the scenario was designed to test only that the data (in this case, on the transaction cards) needed for automatic billing are supplied. The scenario outputs indicate the lack of such information as the date of the trip whether or not it was a standing request, as well as any information whatsoever for trips encompassed by the manual backup period.

I.3.10 Hard Copy for Manual Backup P

This scenario, incorporated into I.2.6, tested whether the hard copy provided is adequate for manual backup. At the time of (simulated) computer failure, the backup console showed the most recent projected tours for the vehicles (see test data, Volume II). The tours are given for vehicle 4, as an example, in the following way:

```
142151   CRS0400   04   039   042   039   042
```

where the first 6 digits are the time of day, the CRSXXXX is the message number and the rest of the line indicates that vehicle 4 is to pick up passenger 39, pick up passenger 42, drop off passenger 39, drop off passenger 42. The addresses are given by passenger number on the hard copy prior to the latest tour, e.g.:

142104 CRS0410 039 01 100 MEMORIAL DR. HARVARD SQ.

which gives the origin and destination of the one passenger number 39. These data are adequate to inform the dispatcher of the current contents of each vehicle, of its position at the next stop, of its projected stops.

The hard copy does not show passenger names. These must be obtained from the operator consoles, by matching addresses, or from the system status dump. This dump is obtained by the supervisor typing STAT on his console. Because he cannot do so just after the computer has failed, the last available status dump many not contain the latest assignments. Such, indeed, was the case for scenario I.2.6; demand number 42 in the sample above does not appear on the latest status dump (Appendix I, Volume II).

I.2.6 Realistic Case

Most of the results of this scenario are reported under the specific capabilities which were encompassed by it, i.e., restart, graphics, standing requests, automatic billing, hard copy for manual backup. Some additional information regarding algorithm efficiency was also obtained and is presented here.

Figure 26 shows the tours assigned by the algorithm up to the computer failure. The efficiency of these assignments, taken as a whole, cannot be readily checked because of the large number of possibilities. Also, the requested pickup times for the standing requests are varied, further complicating an evaluation.

Considering these complications, it can be said only that the tours *appear* to be sensible on a vehicle-by-vehicle basis. Only one assignment, that of nurse Duckett, vehicle 3, is questionable: her pickup should have been inserted between those of SRA2 and SRA7 rather than between SRA3 and SRA2, as can be seen from Figure 26.

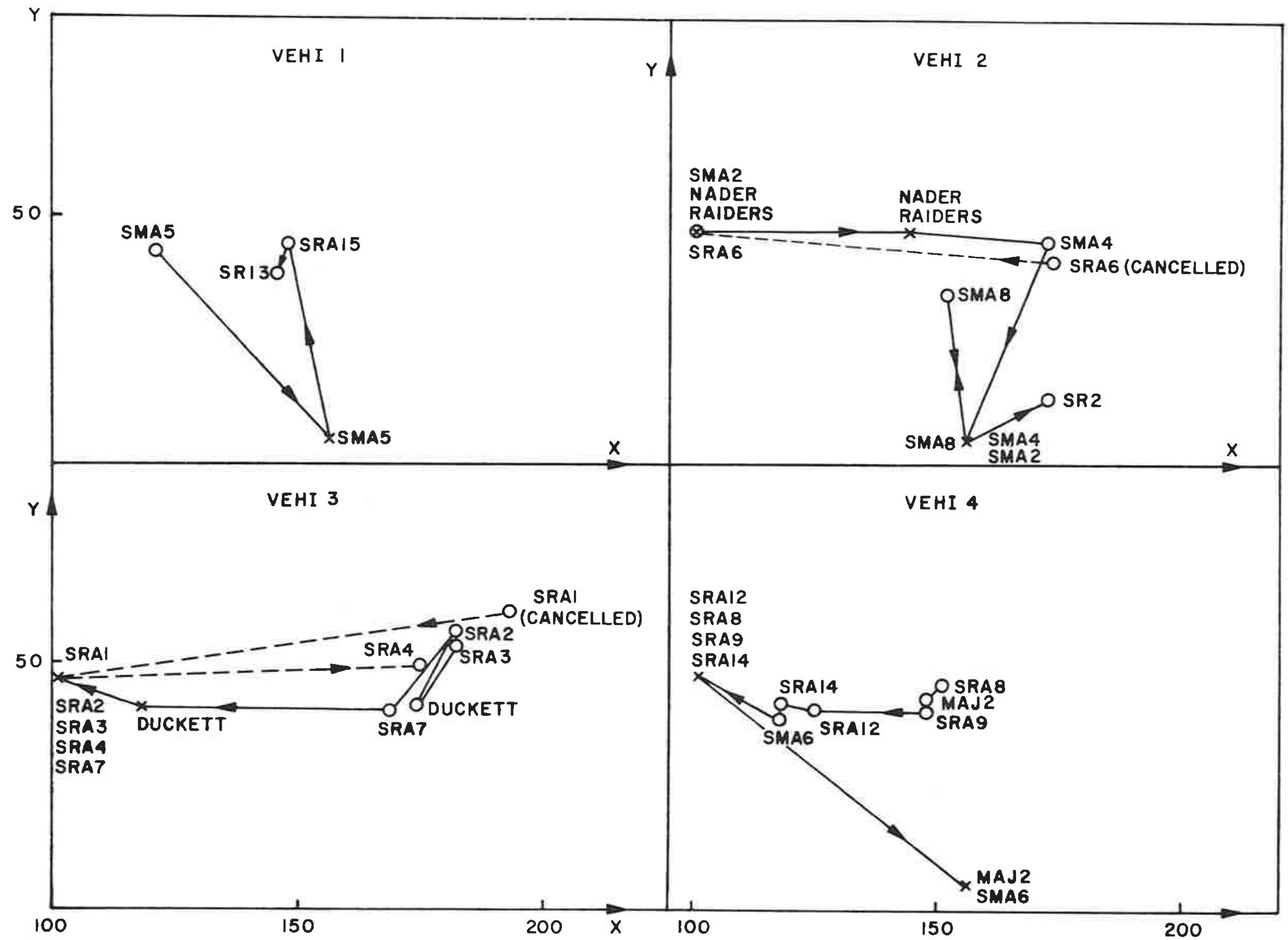


Figure 26 Tours of the Vehicles in Scenario I.2.6 before Simulated Computer Failure (Sheet 1 of 2)

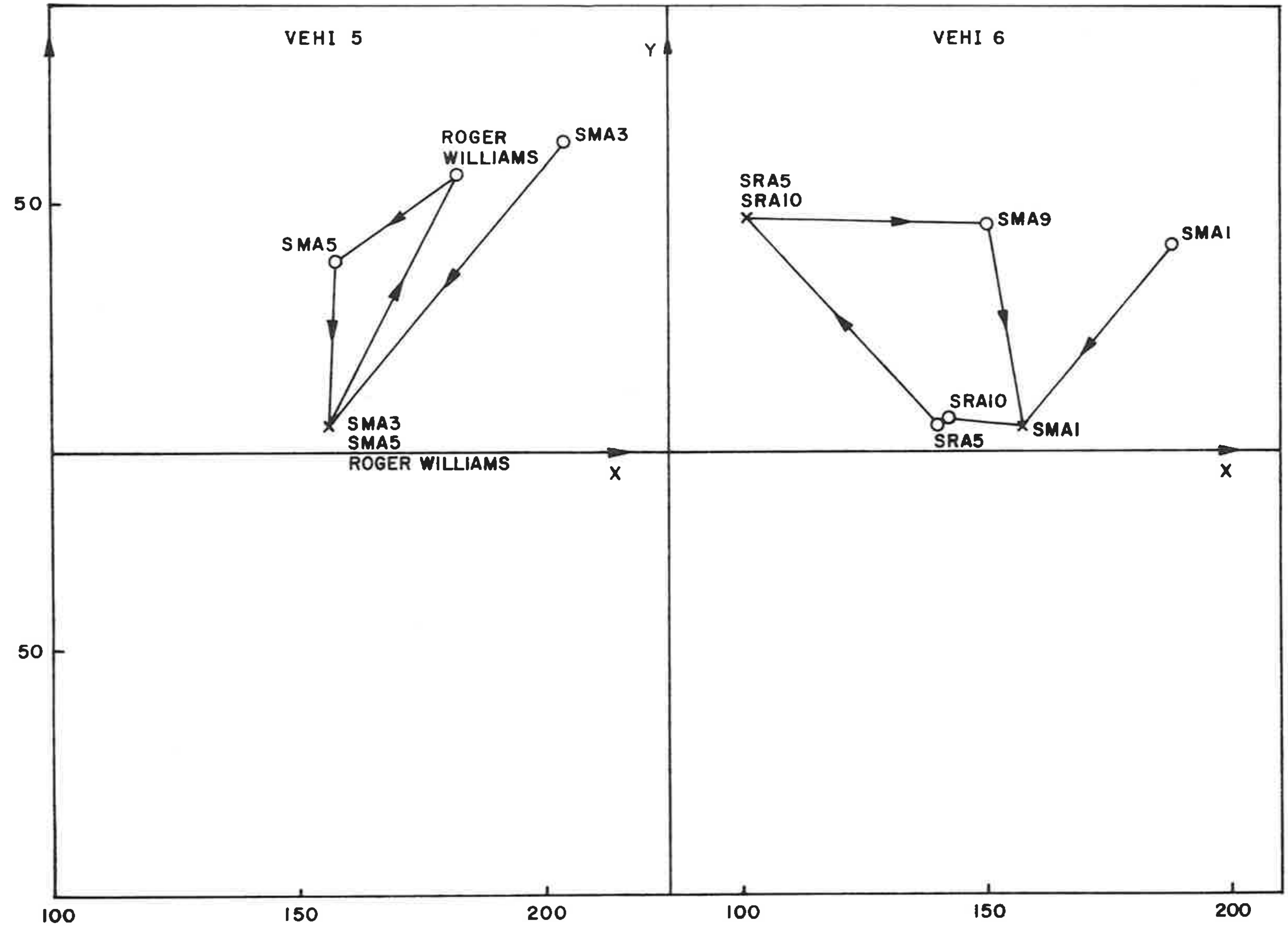


Figure 26. Tours of the Vehicles in Scenario I.2.6 before Simulated Computer Failure (Sheet 2 of 2)

SECTION 3
REVIEW OF PROGRAM DOCUMENTATION

The four items of documentation required by the work statement, and their scheduled delivery dates are as follows:

Item	Scheduled Delivery Date
1. Acceptance Test Specification	28 February* to 30 April**
2. User's Manual	31 May*
3. Program Description	31 May*
4. Manual Backup System Handbook	31 May*

Before reviewing these documents, one should first discuss their purpose, as given in the work statement. The pertinent statements are on page A-2 of Appendix A, repeated here for convenience:

> "This program, as it will be delivered and documented, could be supplied to potential Dial-A-Bus system operators and would then provide them with the necessary computerized dispatching capability."
>
> on page A-3:
>
> "With the documentation to be delivered with this program, as described in the next section, this program truly represents a complete package which could be furnished to locations desiring to implement a complete thirty-vehicle Dial-A-Bus system."

*As given in the work statement.
**Extension recommended by TSC.

on page A-5:

"The Operational DOS Program has been designed to operate a twenty-to-thirty-vehicle Dial-A-Bus system in any locality. All this required was to code the street network and insert it into the program in accordance with the instructions contained in the documentation supplied with the program."

finally, on page A-5:

"With its flexibility, the Operational DOS Program and its related documentation provides an excellent capability for any locality desiring to operate a Dial-A-Bus system to be able to do so with a minimum of additional support which could be obtained from any of a number of computer-oriented organizations throughout the country."

The general purpose of the documentation, then, is: to allow communities to use the D-A-R program with minimum additional programmer support. One support area, that of coding the street network, is explicitly mentioned. It seems reasonable to measure the effectiveness of the documentation by the amount of effort required to setup and operate the program in a typical community, with the documentation that was delivered. This is the point of view taken in the following reviews.

3.1 Acceptance Test Specification

The final Acceptance Test Specification covered only those items explicitly called for in the work statement. There are several other tests that TSC believes to be helpful in ascertaining the effectiveness of the operation of O-D-A-R in a real situation. These tests, as given in the original outline submitted by TSC on February 1, 1971 (see Appendix B), included the following:

a. Response Times: How long will it take for the passenger or vehicle console to receive a reply from the computer, once a message has been typed in? This information is valuable in deciding whether a time-shared computer can be used for what is essentially a real-time computer program. An excessive response time can invalidate the time-share approach and require a dedicated machine, which is much more costly.

b. Realistic Cases other than Cambridge: For reasons given in TSC's review of 30 March (Appendix B), the part of Cambridge employed in I.2.6 does not present some of the important features of a typical Dial-A-Ride community. In particular it does not exhibit the strong sectionization produced by railroads and highways going through the service area, nor does it have lakes or rivers; its street names are of a uniform format. In order to explore these potential problems, TSC drafted a scenario based on Haddonfield N.J. and encoded the street map. Unfortunately, it was not possible to include this scenario in the Acceptance Test Specification.

c. Failure Protect Provisions: Although the O D-A-R software cannot be made to prevent human errors or machine malfunctions, it may be possible to minimize the program's susceptibility to such occurrences. For example, it was found that hitting the carriage return key at the end of a line caused the teletype to shut down. Software that prevented such a shutdown would increase the convenience to the user and reduce the community's dependence on programmer support.

d. System Capacity: The major improvement afforded by O D-A-R over the advanced program is the employment of up to 10 terminals and a number of buses (20 to 30) limited only by the backup capability (see work statement Appendix A, section 3.1.3 and 3.4.1). Hence it seems desirable that a test be included to verify that O D-A-R does indeed operate efficiently with 10 terminals and 30 vehicles. Such a scenario (I.2.8) was drafted by TSC (see letter of 30 March, Appendix B).

In conclusion, then, it may be said that although the Acceptance Test Specification satisfied the requirements of the work statement, TSC believes that the inclusion of tests such as described above would have enhanced the utility of the program to the receiving community.

3.2 User's Manual

The User's Manual is the primary contact of the system operator with the software. The work statement (Appendix F1) requires that it contain a written description of all commands, adequate to operate the program in any environment, and information on setting the program up on a computer.

The Table of Contents of the User's Manual is contained in Appendix E. It comprises, for the most part, a thorough description of the operation of each of the six types of terminals used in the Dial-A-Ride program. For example, all the commands that may be put into, and all the messages that may be received from the vehicle terminal are thoroughly explained in Chapter 2. The same is true of the passenger messages, which are described in Chapter 3, and the supervisor's messages, handled in Chapter 4, etc.

The User's Manual also contains a complete specification of the input parameters. But, in the absence of any description of the various features of O D-A-R; and how they are related to the over-all dispatching system, this list of options tends to raise as many questions as it answers, e.g., what is R POOL, line 14 used for? What are the objective functions? Where is the transaction information described?

Offsetting the excellent presentation of the terminal messages, however, are some defects that cannot be classed as insignificant. These defects can, for the most part, be traced to the assumption behind the structure of the User's Manual, namely, that the operation of O D-A-R is the sum of the operations carried out at the six terminals.

a. Inadequate Initialization Procedures: Although Chapter 1 contains instructions on how to log in any one of the terminals, it is only from Chapter 7 which contains instructions on the computer terminal, that one may infer the order in which the terminals may be logged in. Moreover, no instructions whatever are given for setting up the job control preceding the input parameters of the input file; Chapter 10.1 gives only a sample for running the program on a 360/50 (see Appendix E). Obviously, a 360 systems programmer is required to setup the job control properly for any machine.

b. No Overview of the O D-A-R Operation: There is no description of what the program does or of the relations among the terminals. The introduction gives the user no information as to why each terminal is needed or when it is required or not required. There are no examples of normal operation.

c. Inadequate Instructions for Setting up the Street Address File: Chapter 11 gives only "a list of allowable addresses for CARSDOS". (See Appendix E).

The list is that for a portion of Cambridge. It does not describe how to encode any other town or how to handle addresses such as 55 Esplanade, 1070 50th St., S.W., or 3 Shady Grove, La., if, indeed, they can be handled at all. Nor is it stated in the User's Manual where in the system setup process the cards (if that is what is used) are to be inserted. No information is given to relate the units of x, y to units of the input file, or to the speed of the vehicle.

d. Inadequate Treatment of How to Set Up Standing Requests: Chapter 3, dealing with the passenger terminal, contains the instructions for handling standing requests.

It is stated that they must be in chronological order at the end of the input file. It also states that they are the result of a pre-processing program described in the program description, but does not give the name of the subroutine, or any page reference. Search through the index of the program description reveals a program, SR, which writes out a standing request in A format on unit 4. But, since no format or description of the output data is given, one cannot know whether they come out in chronological order, or indeed, what output device should be assigned as unit 4 without reading the program listings. Thus, a basic task that any operator should be able to perform (it must be done every day, see results of scenario I.3.8, section I.) requires intimate knowledge of the computer program.

The writeup in the User's Manual concludes with the observation that the passenger console is not used at all in the handling of standing requests.

e. Inadequate Description of Priority Classes: Although the method of inserting a priority request is described in detail under the passenger terminal writeup, no reference is made there to the constraints that define the priority classes, or to the description of the constraints in the input file.

f. Automatic Billing Setup Description: The billing feature is described in Appendix A of Volume IV of the Program Description; it properly belongs in the User's Manual.

The description of the automatic billing feature setup fails to convey several key pieces of information.

First, that the customer record programs are unconnected to the O D-A-R program, except by a program to be written by the user; second, the format given on page 446 for the automatic billing cards is incorrect, as indicated in Appendix D; third, the meaning and use of standard trips, passenger condition codes, and zones on page 446 are not explained; fourth, the units are not given for the coordinates and times of the billing record cards.

In conjunction with the review of this manual all commands contained therein have been exercised and the results are as follows.

All those commands are operable that are required for functioning according to the work statement. However, one supervisory function, the ability to change the average speed of the vehicles in the system, is inoperable and its use causes a main task interrupt, thus stopping CARDOS. This speed change command might be employed to meet changing traffic or weather conditions but is not needed to meet the requirements of the work statement.

A detailed summary of all the features is contained in Appendix C.

In summary, it may be said that the User's Manual provides an adequate description of the operation of all of the terminals, but fails to instruct the reader adequately in what the program's features are and how to set them up on the computer. In addition to the above, the User's Manual contains no general description or diagram of the functioning of the program, terminals, and personnel; it does not describe the assignment algorithm or mechanism, nor is there adequate treatment of such features as dispatching points, transaction information, continuous vehicle location (Is it used?), map approximations, extra-vehicular time, the standing request parameters, or zero length trips. Considerable effort must be expended by the user community or their software contractor to fill in these numerous details. Alternately, extensive consultation with the personnel who wrote the program would be required to setup and operate it.

3.3 Program Description

The work statement specifies that:

"A written description will be provided which will enable UMTA or other organizations to understand and modify the program. In particular, flow charts of all programs accompanied by brief written descriptions including definitions of all important variables will be provided as well as the procedures necessary to set up details of a new street network to implement operation in a specific locality."

These requirements will be discussed in turn:

a. UMTA (as represented, in this case, by TSC) found that the documentation of individual subroutines was extensive and of great assistance in understanding or modifying those individual subroutines. However, TSC believes that the absence of any over-all flow charts of any type is a serious detriment to understanding and modifying the program. An indication of the complexity of the program may be gained from Figure 27. This diagram was constructed by TSC, in an attempt to understand the O D-A-R, from essentially the same information* as is contained in the program description. Such complexity, TSC believes, calls for one or more over-all information flow charts to aid in understanding the interrelation among subroutines, the points of input and output, and the points of contact between the processing and communications portions of the program and between those portions and the DOS.

b. In order to test whether organizations other than UMTA or TSC are able to understand and modify the program, TSC presented the documentation for review to an in-house software and data processing contractor, Service Technology Corporation. The review, prepared by one of their experienced staff members, is contained in Appendix F.

c. The work statement requirement for flow charts and written description of all programs is met fully. This part of the documentation is thorough and at such a level of detail as to facilitate understanding. These charts, when used with the annotated program listings, represent the best possible form of documentation of subroutines, in TSC's opinion. It is to be noted that the ARDS display software is

*The information was obtained from previously delivered documentation for the basic program, updated by consultation with MIT/USL during the course of the project.

a proprietary item furnished by Adage Ambilog, Inc. and consequently the subroutines are referenced but not documented.

One shortcoming of this part of the documentation, however, is the absence of any classification of the subroutines except that of alphabetic order.

d. Documentation of the procedures to set up the street map file is contained in the User's Manual and is reviewed above.

3.4 Manual Backup System Handbook

Review of this portion of the documentation is contained in the next section.

NOTES TO FIGURE 27

In order to follow a dispatch message or request through the flow chart of Figure 27, one must first identify the code letters used. They are:

1. Initialization (I) - This indicates a routine that performs setup of parameters, zeroing of core for a variable(s), or preparation for some later job, such as a housekeeping chore.

2. Display (D) - This pertains to the ARDS display routines - any phase.

3. Supervisor (S) - Usually a control of some sort. The subroutine checks to see that the proper steps are performed in demands required of its subroutines.

4. Passenger (P) - These subroutines deal with some phase of the passenger requests. They deal with pickup, delivery, status, number of passengers, etc.

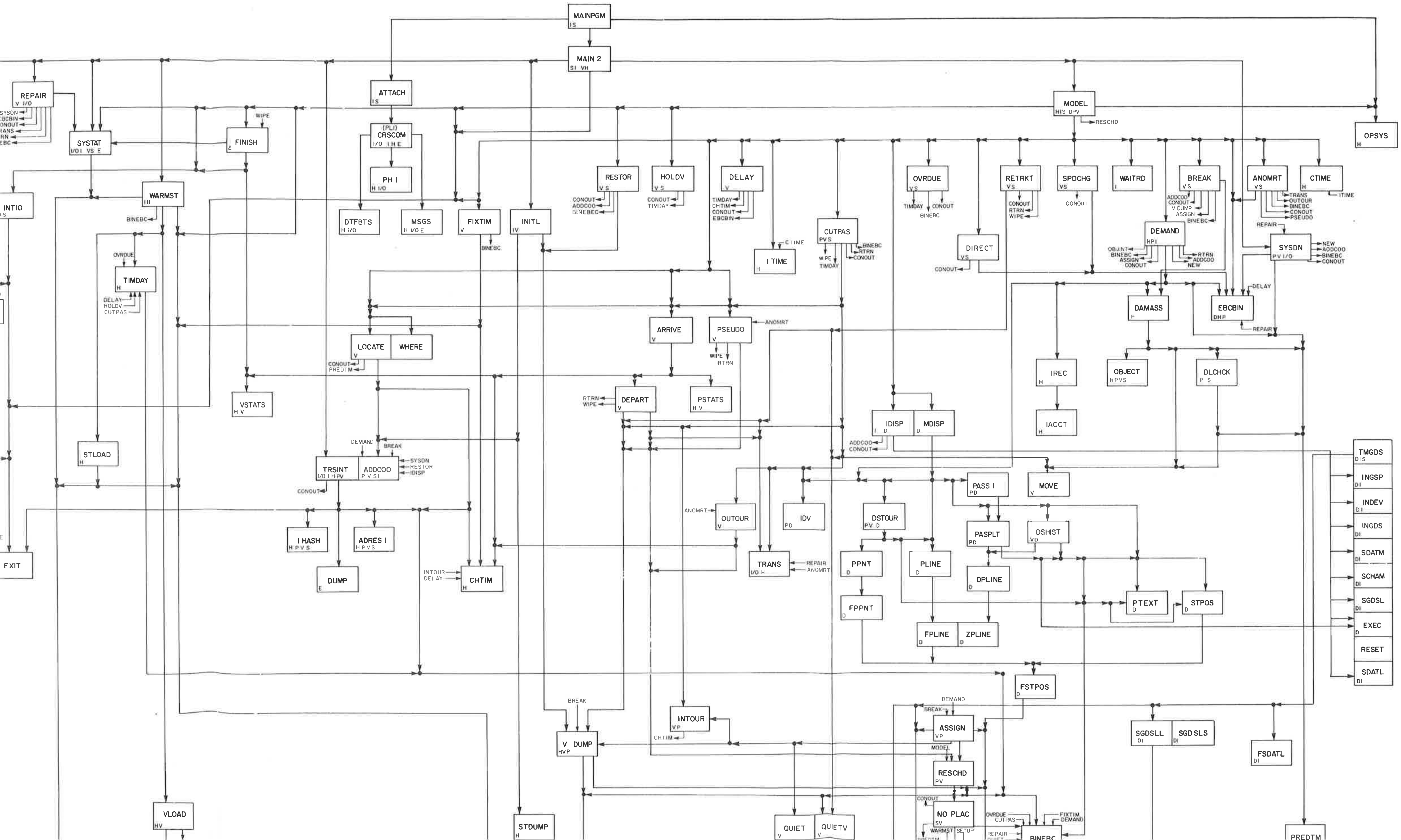
4. Vehicle (V) - These subroutines control vehicle status of some sort. They deal with all phases of vehicle status including assignment, present position, speed, number of passengers, next stop, time of pickup, etc.

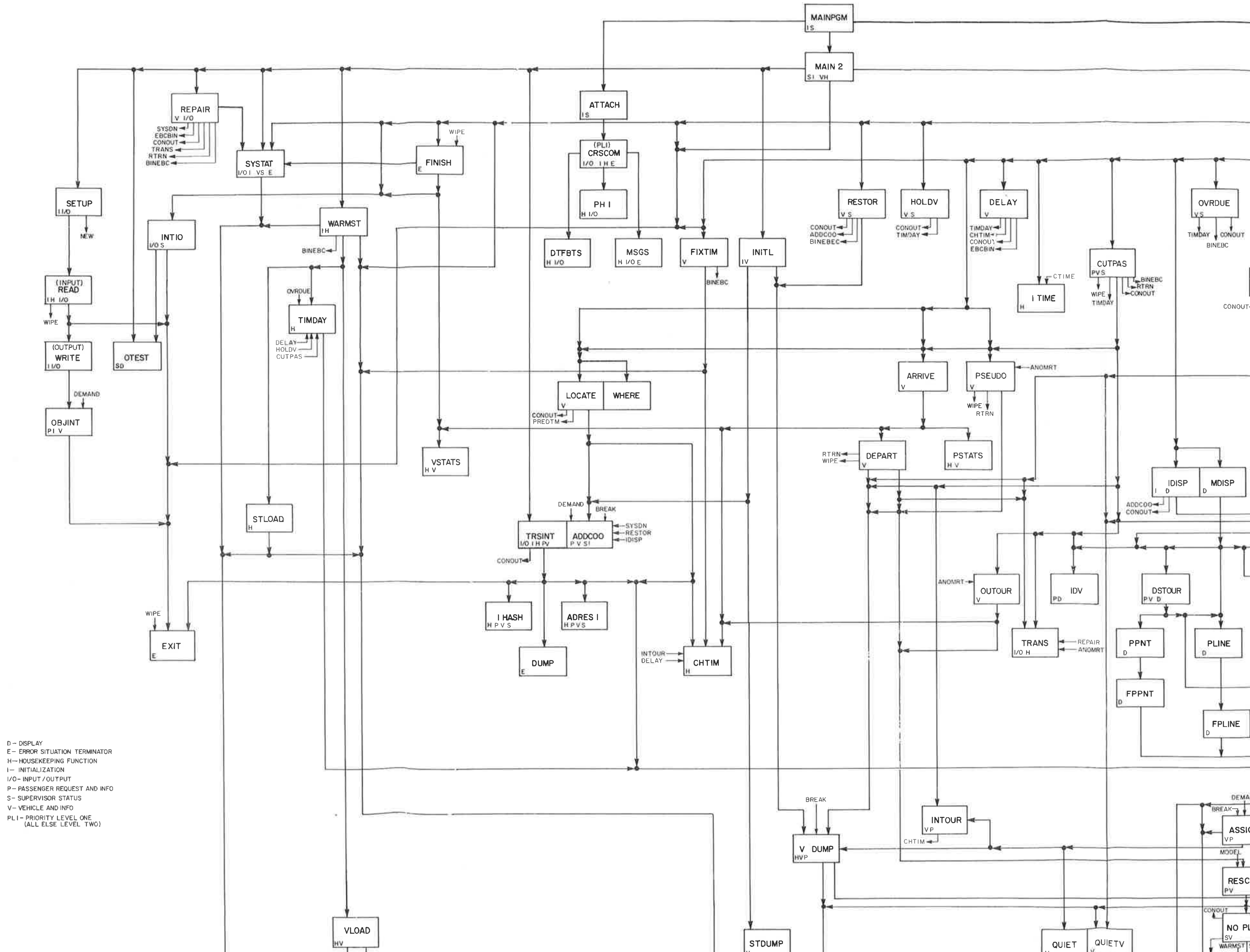
6. Error Situation or Terminator (E) - Exit routines of this type are of two classes. Either they are a normal terminator or abnormal dump and error situation. The dumping is under supervisor control and specific parameters are saved on the output to help identify causes of the error.

7. Housekeeping (H) - Subroutines of this nature perform basic parameter setup and routine chores that check or increment key control variables.

8. Input/Output (I/O) - These subroutines deal with reading and writing of data. It is not used here in the graphic sense, only for reading data (cards, magnetic tape, or disc terminal requests) or writing data (punch cards, magnetic tape or disc, line printer output, or terminal information).

   There is some overlap in the identification (initialization can at times be considered housekeeping and housekeeping supervisory) but the label serves to identify the primary purpose of the subroutine and clarify the ones of greatest importance.


MAINPGM
MAIN 2
REPAIR
SYSTAT
FINISH
ATTACH
(PLI) CRSCOM
PH I
WARMST
INTIO
DTFBTS
MSGS
FIXTIM
INITL
TIMDAY
RESTOR
HOLDV
DELAY
OVRDUE
RETRKT
SPDCHG
WAITRD
BREAK
ANOMRT
CTIME
MODEL
OPSYS
CUTPAS
I TIME
DIRECT
DEMAND
SYSDN
DAMASS
EBCBIN
LOCATE
WHERE
ARRIVE
PSEUDO
VSTATS
DEPART
PSTATS
IREC
IACCT
OBJECT
DLCHCK
IDISP
MDISP
STLOAD
TRSINT
ADDCOO
I HASH
ADRES I
DUMP
EXIT
CHTIM
OUTOUR
IDV
TRANS
DSTOUR
PASS I
MOVE
PASPLT
DSHIST
PPNT
PLINE
DPLINE
FPPNT
FPLINE
ZPLINE
PTEXT
STPOS
TMGDS
INGSP
INDEV
INGDS
SDATM
SCHAM
SGDSL
EXEC
RESET
SDATL
FSTPOS
INTOUR
V DUMP
ASSIGN
RESCHD
SGDSLL
SGDSLS
FSDATL
NO PLAC
VLOAD
STDUMP
QUIET
QUIETV
BINEBC
PREDTM


MAINPGM
MAIN 2
ATTACH
(PLI) CRSCOM
PH I
DTFBTS
MSGS
FIXTIM
INITL
REPAIR
SYSTAT
FINISH
SETUP
INTIO
WARMST
TIMDAY
(INPUT) READ
(OUTPUT) WRITE
OTEST
OBJINT
STLOAD
VSTATS
LOCATE
WHERE
TRSINT
ADDCOO
I HASH
ADRES I
DUMP
CHTIM
EXIT
VLOAD
STDUMP
V DUMP
RESTOR
HOLDV
DELAY
I TIME
CUTPAS
OVRDUE
ARRIVE
PSEUDO
DEPART
PSTATS
OUTOUR
IDV
TRANS
IDISP
MDISP
DSTOUR
PPNT
FPPNT
PLINE
FPLINE
INTOUR
ASSIGN
RESCHD
QUIET
QUIETV
D– DISPLAY
E– ERROR SITUATION TERMINATOR
H– HOUSEKEEPING FUNCTION
I– INITIALIZATION
I/O– INPUT/OUTPUT
P– PASSENGER REQUEST AND INFO
S– SUPERVISOR STATUS
V– VEHICLE AND INFO
PLI– PRIORITY LEVEL ONE (ALL ELSE LEVEL TWO)

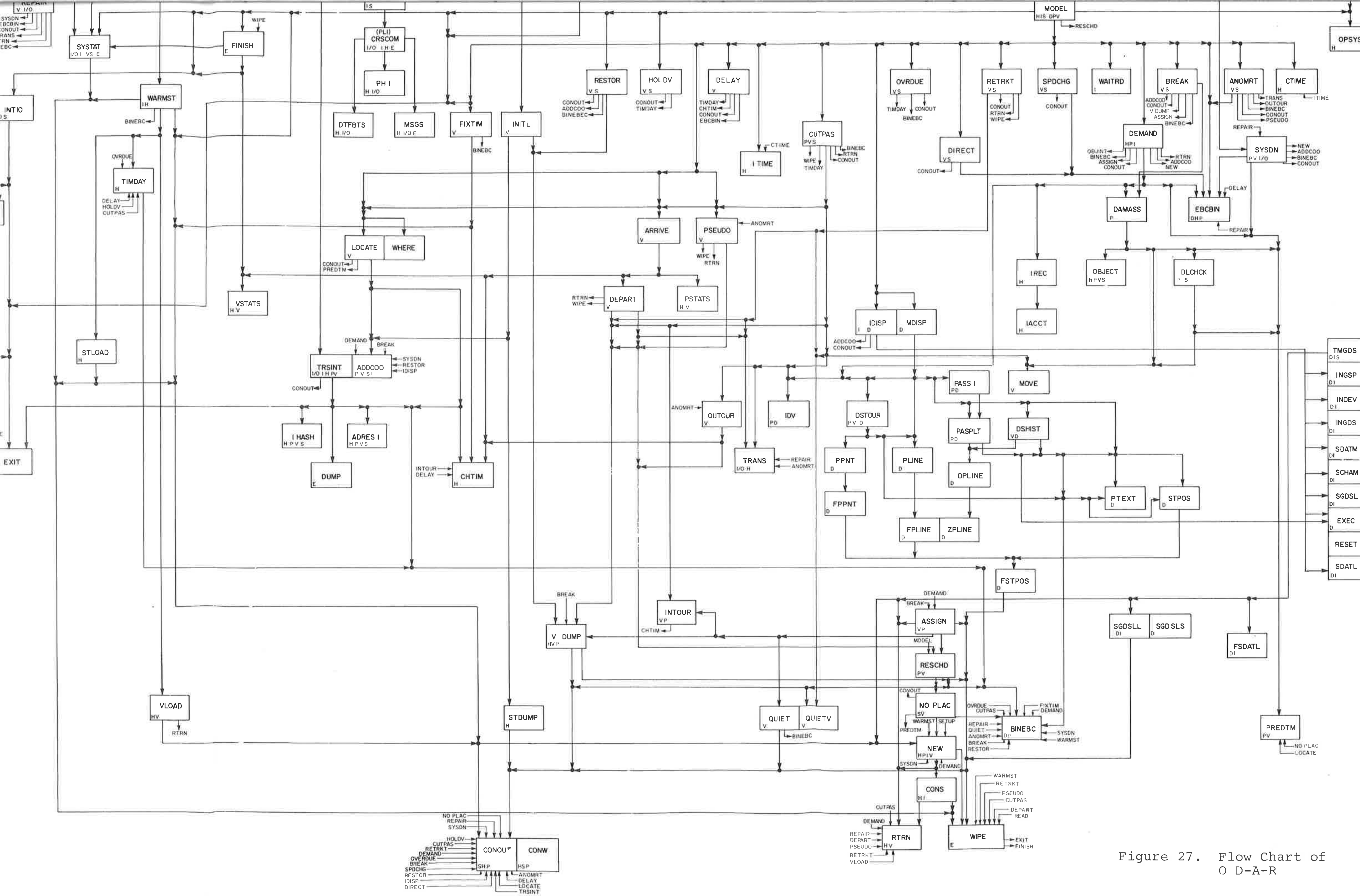
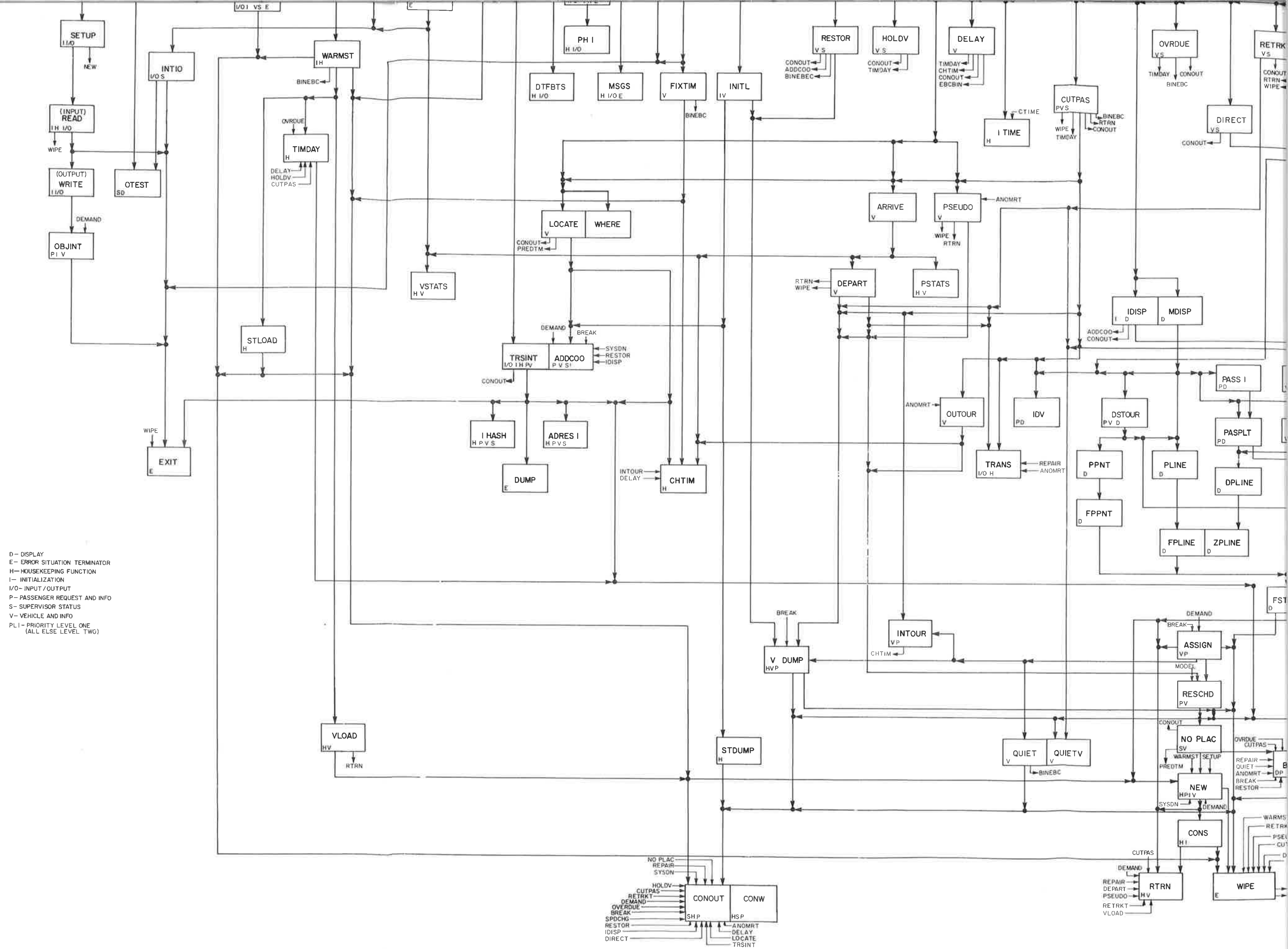


Figure 27. Flow Chart of O D-A-R



SETUP
INTIO
WARMST
PH I
RESTOR
HOLDV
DELAY
OVRDUE
RETRK
(INPUT) READ
DTFBTS
MSGS
FIXTIM
INITL
I TIME
CUTPAS
DIRECT
TIMDAY
(OUTPUT) WRITE
OTEST
OBJINT
LOCATE
WHERE
ARRIVE
PSEUDO
VSTATS
DEPART
PSTATS
IDISP
MDISP
STLOAD
TRSINT
ADDCOO
EXIT
I HASH
ADRES I
DUMP
CHTIM
OUTOUR
IDV
TRANS
DSTOUR
PASS I
PASPLT
PPNT
PLINE
DPLINE
FPPNT
FPLINE
ZPLINE
INTOUR
V DUMP
ASSIGN
RESCHD
VLOAD
STDUMP
QUIET
QUIETV
NO PLAC
NEW
CONS
CONOUT
CONW
RTRN
WIPE
D– DISPLAY
E– ERROR SITUATION TERMINATOR
H– HOUSEKEEPING FUNCTION
I– INITIALIZATION
I/O– INPUT/OUTPUT
P– PASSENGER REQUEST AND INFO
S– SUPERVISOR STATUS
V– VEHICLE AND INFO
PLI– PRIORITY LEVEL ONE (ALL ELSE LEVEL TWO)

SECTION 4
REVIEW OF BACKUP MODE

The manual backup handbook outlines a procedure to continue Dial-A-Bus operations in the event of computer failure. The method is based on a decentralization of decision making devoid of an automatic computer backup system. The decision making responsibility is placed on the vehicle drivers thus negating any need for additional personnel to handle manual operation. The handbook states that the scheme employed is currently being used successfully, in certain taxi-dispatching operations, but gives no references.

The Manual Backup Handbook is divided into the two main functions for the drivers, dispatchers and telephone operators:

1. The preparation for and operation of manual backup.

2. Procedures for returning to computer operation (restart).

In the event of computer failure certain material should be available for manual backup operation. As the vehicle drivers require not just their next stop but their complete projected tour, the dispatcher is provided with a backup terminal. He constructs, from the backup terminal output, the vehicle's future tour list and relays it to the driver. In addition, the control center must have files in which to place cards containing trip information for new requests received during manual operation for each vehicle.

Once the drivers have received their projected tours from the dispatcher, manual operation may commence. The telephone operator receives the request and punches two information cards, one for the origin and one for the destination. These cards are handed to the dispatcher who broadcasts the requests to the bus drivers who in turn mentally compute the additional time required to handle the new request. Then the dispatcher will query the drivers to determine which vehicle should handle the request. Having assigned the passenger to a vehicle, the dispatcher places the cards in that vehicle's file and awaits pickup or dropoff information from the vehicle or another request from the operator. As each new request enters the system or as each vehicle makes a stop, the card files and/or pre-computer failure tour lists are updated.

The purpose of burdening the drivers with the task of passenger assignment is fourfold:

1. The driver can quickly handle requests due to his familiarity with the service area and the human ability to recognize patterns.

2. Each driver has the responsibility to order his stops to suit traffic and road conditions and his own individuality.

3. Each driver is responsible for only a few passengers, thus he can introduce service constraints on an individual passenger basis.

4. Manual backup provides the drivers with the very important feeling of participation in the control of the system.

Besides the participation of the drivers in making assignment decisions, they must also keep the dispatcher informed of their stops in order that tour list and card files can be updated. The accurate keeping of files and tour lists is necessary for a smooth and rapid transition from manual backup to normal computer operation.

Restart is provided for by two types of files:

1. The backup files: These files maintained during manual backup, contain present vehicle tour information.

2. Dumping files: These disk files contain pertinent parts of core for use in reconstructing the original system in the event of a computer failure.

These files are employed in the 3 methods of restart available with CARDOS. The three procedures, as outlined in the Manual Backup Handbook are as follows:

1. Short term failure (warmstart) - Warmstart is attempted prior to manual backup operation before any new requests have been handled or any vehicles have completed a stop. Warmstart procedure uses only the dumping files; should this fail, manual backup preparations begin.

2. Long term failure (dead start) - A dead start is attempted when all original outstanding requests have been handled during manual backup. Thus, the backup files contain all the vehicle stop information required, as all data dumped during normal operation is outdated.

3. Intermediate term failure (down start) - Between short and long term failure is where some of the original requests have been serviced. Restart requires a combination of input from the backup files, information relating to stops planned during manual backup and the dumping files, and information on the original vehicle stops.

Each job stream for the three types of restart consist of a collection of pre-prepared cards and request cards made during manual backup by the telephone operators and placed in the card files. Sample job control streams are contained in Appendix A of the handbook.

As the spatial relationship between operation, dispatcher, supervisor and hardware are of considerable importance during manual backup, the handbook provides a brief discussion on and sample layout of the control center.

The scheme for manual backup appears to be relatively simple and a viable solution to the problem of computer failure. However, the system as it is presented in the handbook raises several questions which must be answered before manual operation can function reliably.

1. Automatic billing - No provisions are outlined to handle automatic billing requests during manual backup.

2. Standing Requests - How are the new standing requests handled during manual backup if at all?

3. Cancellation - No procedures are given to allow for proper transaction cards to be punched for cancellation of trips for passengers on the vehicle or for automatically billed customers prior to service.

4. Vehicle breakdown - No procedures are outlined to handle vehicle breakdowns. It is assumed that new origin cards are punched and combined with the destination cards to handle the reassignment problem. These cards along with any other outstanding requests are given to the dispatcher to handle in the normal new request way.

5. Backup terminal deficiencies - As in the event of vehicle breakdown and for cancellation of a service request, the backup terminals output is not updated; incorrect information about the vehicles' list of future stops may be relayed at the start of manual

backup to the vehicle drivers.

6. Trip cancellation - Should a trip cancellation come in during manual backup to a telephone operator, there are no provisions in the handbook to make any association between the name given by the caller and the unique number assigned to his request as an identifier.

7. Priority classes - There are no provisions for more than one level of service.

8. Fare collection - The manual does not indicate who has responsibility to compute the trip fare.

9. Driver participation - The entire manual backup scheme depends on the driver's ability to estimate the additional time required to make a proposed stop for all possible insertions of that stop in his present tour. This is a computation analogous to the optimization algorithm executed by the computer. The handbook states on page 1 that "The system is currently being used in certain taxi-dispatching operations, and is of proven reliability and feasibility." Since no reference is given for this statement, TSC was unable to verify it; such verification is strongly urged before O D-A-R is deployed.

10. Specific deficiencies of the Handbook - The following items have been noted to be lacking or in error.

    a. The handbook mentions on page twenty that the telephone operator types in the name of a passenger requesting service on a card. However, as the manual states later, the operator assigns a unique demand ID instead of a name.

    b. The sample punched cards on page 20 show incorrect column specification; the column definition contained on page 21 should be followed.

    c. The one control card omitted from the handbook is the TIME command; this card should follow DOWN and DEAD in the input decks for down-start and dead-start. The format is TIME XXXX where XXXX is the length of time in minutes that the computer system has been down.

d. Another undocumented control card, OSTP, appears in the example job control for a down-start on page 42. This control card indicates that the last completed stop for that vehicle is on the vehicle's original tour list. However, this control card can be omitted from the input deck.

e. The example job control for dead-start, as shown on page 41, indicates that all pickups and deliveries have been completed and all vehicles are unassigned. (This would not normally be the case). Control cards and trip cards should follow as shown in the example for down-start except for the five cards following DOWN.

f. The rules on pages 29-30 for the usage of the NSTP and FSTP cards are unnecessarily complicated. The NSTP card should be replaced by one that states LAST STOP IS ON ORIGINAL TOUR for one that states FIRST STOP IS ON ORIGINAL TOUR, or both. The FSTP card is unnecessary. Also, there should be no need for the dispatcher to sort pickups from deliveries; the computer can do so much more reliably.

TABLE 5
SUMMARY OF HEURISTIC EFFICIENCY TESTS, I.2.2

NUMBER	NAME AND TYPE	NO. VEHICLES*	NO. STOPS
	PASSED		
1A	East-West Distribution, One/Many	1	3
2A	Clockwise Distribution #1, One/Many	1	8
3	Clockwise Distribution #2, One/Many	1	4
6	Two-Sector Distribution, One/Many	2	9
7	4 Sector, 4 Vehicle Distribution, One/Many	4	11
8	FCFS Collection #1, One/Many	1	3
9	FCFS Collection #2, Many/One	1	5
12	Diamond-Star Collection, Many/One	1	5
13	Many-Two Test, Many/Two	2	9
14	Simple Many-Many, Many/Many	1	4
	FAILED		
1B	East-West Distribution, One/Many	1	3
2B	Clockwise Distribution #1, One/Many	2	8
2C	Clockwise Distribution #1, One/Many	1	8
5	Group-Group Distribution, One/Many	1	9
11	Branch and Circuit Collection, Two/One	1	3
15	2-Vehicle Many/Many,	2	10

*Number of vehicles involved in dispatching.

SECTION 5
EVALUATION

5.1 Summary

The preceding results of the acceptance test and documentation review may be summarized as follows.

a. Heuristic Efficiency. These tests were designed to determine the efficiency of the assignment algorithm in situations that would be obvious to a human dispatcher. They involved from 1 to 4 vehicles and from 3 to 11 stops. The summary given in Table 5 shows that 10 passed the test and 6 failed. The success or failure of the algorithm in these tests does not seem to be connected with the number of stops, or with whether the problem was: distribution or collection, many/one, one/many, many/few or few/many. It seems rather to be connected with the order in which the requests are inserted. In most cases (1B, 2B, 2C, 5) the failures are attributable to the insertion principle or the no-reassignment rule of the algorithm, both of which make the results dependent on the order of the requests. In one case, (15), failure seems to be connected with the method by which the vehicles are brought to the starting point; and in another case (11), the complete explanation could not be determined by TSC or MIT.

   Scenario I.2.6 seemed to indicate that the algorithm performed well in making assignments in complicated situations. Due to the complexity of the situation, however, it is extremely difficult to determine the best assignments for scenario I.2.6 so that the efficiency of the algorithm in this case can be surmised, but not proved.

b. Constraint Consistency and Violation. These scenarios were designed to determine whether the assignments allowed the stated pickup and delivery times to be met. Due to lacunae in the street time data, and to other factors discussed in Section II, the tests did not establish this. The tests did establish, however, that the assignments do not violate the internal constraints of the program.

c. Specific Capabilities. Tests of the ten capabilities listed in the work statement were made with these results:

- Restart (passed)- The program was successfully restarted after a simulated failure, with only minor difficulty.

- Cancellation of a Service Request (passed) - This feature operated perfectly.

- Unexpected Situations (passed) - These tests showed that the program can accommodate an increase or decrease in the number of passengers associated with a previously assigned pickup, including no passengers showing at all.

- Vehicle Breakdown (passed) - The required capability was successfully tested.

- Lateness Detection and Correction (passed) - Lateness detection was found to operate, but with an unexpected one minute quantization in the detection time. Lateness correction was restricted to making no further assignments to a vehicle that had been detected late. Re-assignment of waiting customers had been found to be impractical by previous research.

- Priority Classes (passed) - Scenario I.3.6 verified that the program has the required feature, but I.2.6 indicates that it may be of little use in practice.

- Graphics (questionable) - Although adequate information was chosen to be displayed, it was so presented as to be difficult to interpret and use. It is effective for demonstrations only.

- Automatic Billing (questionable) - The transaction cards produced lacked some of the basic information required for billing.

- Standing Requests (passed) - The program made assignments of the vehicles to pick up standing requests close to the desired times. Although this feature operated properly, TSC believes that its design should be improved.

- Hard Copy for Manual Backup (passed) - The hard copy of latest tours was adequate for manual backup, although the means of determining the passengers' names was inconvenient.

d. Acceptance Test Specification. The tests for heuristic efficiency and constraint consistency/violation were designed to be quantitative; those for specific capabilities were largely qualitative. Some tests that TSC considers basic to the proper functioning of the program were not included.

e. User's Manual. The User's Manual included excellent descriptions of all commands, as required by the work statement. But it does not meet the other two requirements of the work statement, namely, that:

   1. The information would allow UMTA or other organizations to operate the program in any desired environment and that

   2. The manual would provide information on setting the program up on a computer.

f. Program Description. The documentation of individual subroutines and source listing comments were found to be very thorough, as was the non-graphic descriptions of the data structure, program structure and control flow. But the absence of any over-all flow charts, subroutine classification, description of the algorithm, or of the various features, make it very difficult for UMTA or other organizations to understand and modify the program as required by the work statement.

g. Manual Backup Handbook (backup mode). The basic scheme is plausible, but its feasibility can be determined only by empirical information not presently available to TSC. The handbook does not treat adequately the backup mode for: automatic billing, standing requests, cancellations, vehicle breakdowns, priority classes, fare computation. Also, several minor deficiencies and errors were noted.

5.2 Additional Observations

In addition to the information obtained from specific tests, several observations were made by TSC during the course of the project. They are reported here for the information of potential users of the O D-A-R program.

a. The final version of the O D-A-R program is well debugged in comparison to the first two versions and in relation to most programs of its size and complexity. TSC's experience in many days of operating the program revealed only one command that did not function (the SPEE command, which is not vital) and a few minor format problems (ex: reversal of the sequence of messages in the test for unexpected situations).

b. The overall DOS-ODAR software system was found to be subject to failures. TSC and MIT experienced numerous system failures, i.e., cases in which the program was disconnected from the computer or rendered inactive in such a way as to require the restart procedures to be enacted before continuing. This occured several times after, as well as before, the acceptance tests were conducted but not during any acceptance test. The exact causes of these failures were not traced or traceable in most cases. The causes fall into three categories, however: erroneous inputs; machine, operator, or software problems at the time-share facility; O D-A-R system software problems. Although many cases of system failure can be avoided when more experience is gained, it seems that the Manual Backup Mode will be a vital part of the O D-A-R.

c. About 50 hours of systems programmer effort were required to set up the O D-A-R at the two facilities, dedicated 360/50 and time-share 360/67. Continual programmer attendance at the terminals was required to start, run and restart the program. In addition frequent consultation of the MIT/USL personnel who constructed the program was required for both systems setup and to handle special problems during running. It is to be expected that these three types of support (system programmer, operating programmer, MIT/USL consultation) would be required to set up and initiate D-A-R service at a demonstration site. After about six months of break-in, these support personnel could be reduced to operating programmer, plus occasional systems programmer and MIT consultation time. A full year should be allowed before the dispatcher and telephone operators can be expected to carry out routine operation.

d. The time it takes the O D-A-R program to start printing out an assignment, after the operator has typed in the request and hit the return key* was measured by TSC in separate tests, with the results of Figure 28. This Figure shows that, with about 50 - 60 other time-share users, the O D-A-R response time increases from about two seconds to eight seconds as the number of demands in the system increases from 0 to 150. Such response times are probably acceptable to most customers. The data also showed that the program's data pool was filled at 184 demands, at which point no further demands could be entered.

The response time data of Figure 28 was corroborated on September 28, when responses of less than four seconds were obtained with 60 passengers in the system and about 50 time-share users.

e. A rough estimate of the computer cost for running the O D-A-R on a 360/67 via time share terminal was obtained from data taken on two separate days. A simple model for the total cost, K, of a day's operation of length T hrs. is:

$$K = S + \bar{N}_C TC + N(f + \alpha\bar{N})$$

where S = cost to set-up O D-A-R in the morning

$\bar{N}_c$ = average number of consoles used during the day

N = total number of passengers handled

f = fixed CPU cost per request, including dispatching

$\bar{N}\alpha$ = CPU cost per request when there are $\bar{N}$ other passengers in the system, excluding cost f

* A Control - S is used instead of the Return, in practice, because hitting the Return key causes the O D-A-R systems to stop completely, requiring a restart.

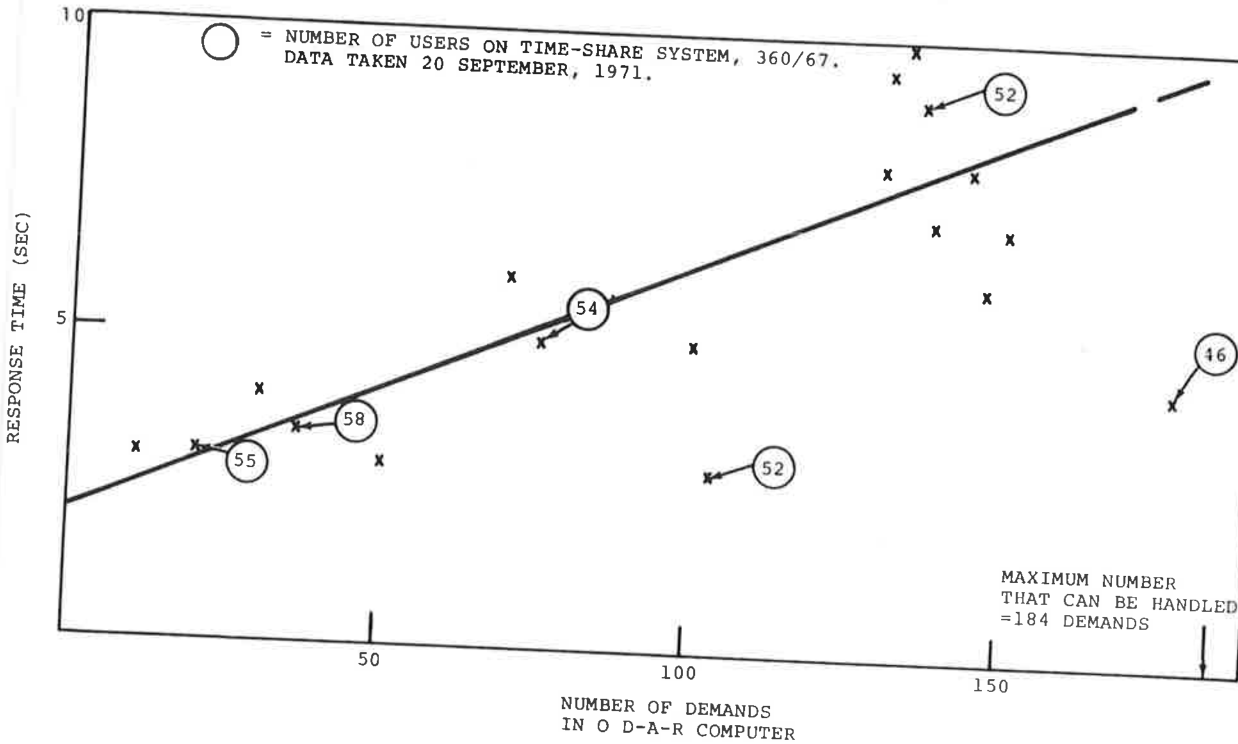


Figure 28. O D-A-R Response Time Vs. Number of Active Demands

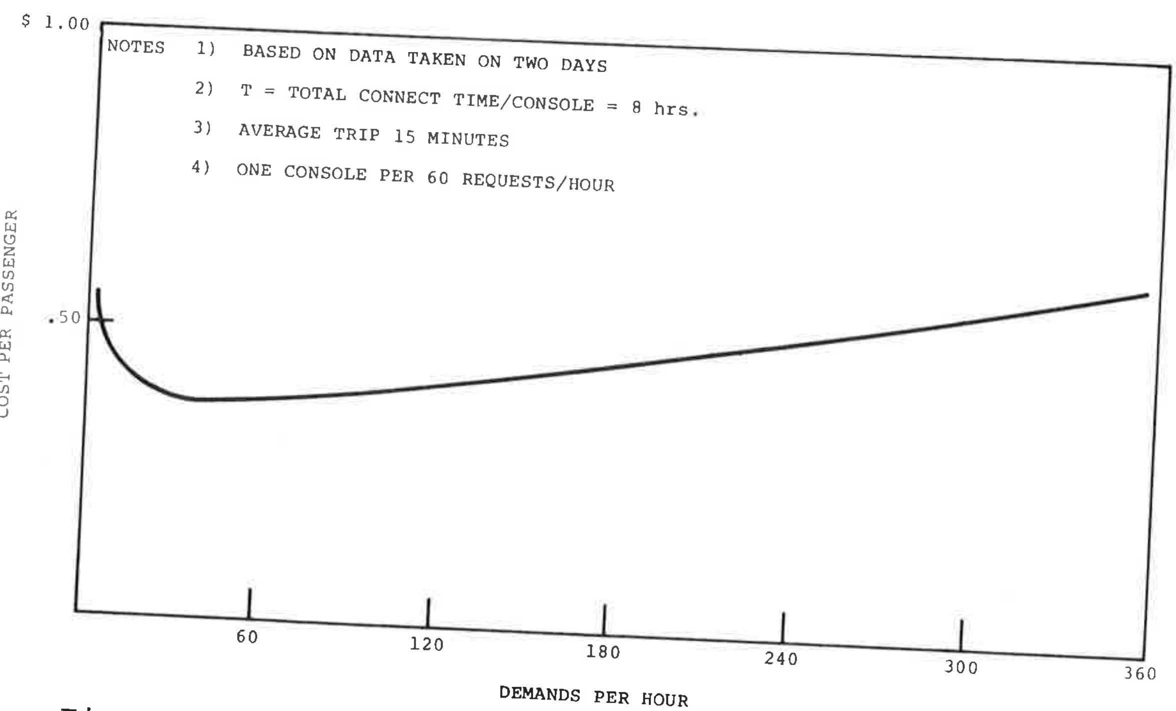


Figure 29. Approximate Data Processing Cost/Passenger for 360/67.

$\bar{N}$ = average number of passengers in the system

C = cost per terminal per hour connect time

Data taken on two occasions yielded the following:

		28 Sept. data	29 Sept. data
$\bar{N}$	=	30	50
N	=	60	100
K	=	45.99	68.50
T	=	.75	.95
$\bar{N}_c$	=	3	3
C	=	10.00	10.00

This gives two simultaneous equations for three unknowns, s, f, α:

$$45.99 = S + 3(.75)(10.) + 60(f + \alpha 30)$$

$$68.50 = S + 3(.95)(10.) + 100(f + \alpha 50)$$

Although the third data point required to solve for S, f, d, could not be obtained, one may assume that CPU time is proportional to the straight line fitted to the response times shown in Figure 28 for about 50 - 60 users. This gives $\alpha/f = 3/140$, and one then has f = .148, α = .00315, and S = \$9.40. Then

$$K = \$9.40 + \bar{N}_c TC + N(.148 + .00315\bar{N})$$

The cost K/N per passenger is plotted in Figure 29 vs. demands/ hr., R, where N = RT, and with $\bar{N}_C$, the number of consoles approximated as R/60. The number $\bar{N}$ in the system is assumed to be R/4 (average trip of 15 min.). Figure 29 thus is a plot of the total of the four components of cost:

$$\frac{K}{N} = \quad .15 \text{ fixed CPU cost/request}$$

$$+.17 \text{ terminal connect cost/request}$$

$$+9.40/RT \text{ system start cost/request}$$

$$+.00079R \text{ CPU cost/request, due to processing demand list.}$$

This function is relatively insensitive to T if T $\geq$ about 8 hours. The largest components of the computer cost are the fixed CPU cost required to accept and process each request, and the terminal connect time. The system start up cost is negligible compared to these. The processing cost proportional to the number of requests in the system becomes significant above 100 requests per hour.

5.3 Conclusions

The objective of the present project was to test and to evaluate the O D-A-R program against the Work Statement. The specific requirements of the work statement were tested and the results may be stated as follows:

Heuristic Efficiency: The algorithm appears to make efficient assignments for a large number of demands, but, because of the insertion principle and no-reassignment rule, it does not perform as well for a small number of demands, particularly for all orders of inserting the demands.

Features: All required features worked satisfactorily except for automatic billing and graphics.

Documentation: The documentation was excellent on details but lacking adequate information for the over-all system.

Back-Up Mode: The scheme is plausible but in need of more experimental evaluation. The Handbook lacks details needed for handling the many features during back-up.

In addition the following observations were made relative to running the program:

Operating Considerations:

The program runs well and is relatively well debugged

The overall O D-A-R, time-share, DOS System was subject to failures.

Successful operation requires a D-O-S System programmer, an applications programmer, and assistance from MIT/USL personnel familiar with the program.

The response time is good.

The computer costs are significant.

Having made the above specific tests and observations, it remains to answer the question: "Does O D-A-R achieve its general purpose, as given in the Work Statement?" The Work Statemen says on page A-2, Appendix A:

"This program, as it will be delivered and documented, could be supplied to potential Dial-A-Bus system operators and would then provide them with the necessary computerized dispatching capability."

On page A-3:

"With the documentation to be delivered with this program, as described in the next section, this program truly represents a complete package which could be furnished to locations desiring to implement a complete thirty-vehicle Dial-A-Bus system."

On page A-5:

"The Operational DOS Program has been designed to operate a twenty-to-thirty-vehicle Dial-A-Bus system in any locality. All this required was to code the street network and insert it into the program in accordance with the instructions contained in the documentation supplied with the program."

Finally, on page A-5:

"With its flexibility, the Operational DOS Program and its related documentation provides an excellent capability for any locality desiring to operate a Dial-A-Bus system to be able to do so with a minimum of additional support which could be obtained from any of a number of computer-oriented organizations throughout the country."

Considering the deficiencies in documentation, in the Back-up Handbook, and the second and third operating considerations listed, it is TSC's opinion that the O D-A-R program is not fully successful in achieving its stated objectives. Its greatest strengths lie in a good algorithm for large demand, numerous well-functioning features, (except for two), and efficient, relatively "buggless" programming. Its greatest weaknesses lie in the completeness of the "package" supplied. The deficiencies summarized in this section, and described in previous sections, would have to be remedied in order to realize fully the work statement objectives quoted. Without such modifications, however, the O D-A-R program may still serve as a good base from which a demonstration program can be launched provided adequate support personnel, as detailed in this section, were available.

APPENDIX A

EXTRACTS FROM WORK STATEMENT

DIAL-A-BUS SYSTEM COMPUTER PROGRAMS

November 24, 1970

Those portions of the work statement dealing with the operational program are contained in the following pages.

3.0 WORK DESCRIPTION

3.1 General

The following three related operational computer dispatching programs for use in the Dial-A-Bus environment will be developed:

1. The Basic Program - This program will serve as a planning tool. It will be used to pratest Dial-A-Bus in a particular environment, allowing tailoring of the Dial-A-Bus system to fit into the given conditions in an efficient fashion.

2. The Advanced Program - This program, which may be considered as a development of the Basic program, can be used to operate a small Dial-A-Bus system of seven to ten vehicles in a given community on an IBM System 360, Model 67, time-sharing computer. The advanced program will accept a stream of actual customer demands, make routing decisions and output messages to the appropriate vehicles giving their destinations.

3. The Operational DOS Program - This program is an improvement of the advanced program which can be used to operate a twenty-to-thirty-vehicle Dial-A-Bus system in an urban community on one of a variety of IBM System 360 computers under the Disc Operating Systems (DOS). It is not constrained to the Model 67 time-sharing machine under the CP/CMS operating system. In addition, this program will be throughly tested prior to delivery and will have a graphics display capability to enable a supervisor to see a display of the actual state of the system during operation. This program will be able to operate with multiple input terminals whereas the advanced program is limited to a single input terminal. This program, as it will be delivered and documented, could be supplied to potential Dial-A-Bus system operators and would then provide them with the necessary computerized dispatching capability. The size of the Dial-A-Bus system which could be operated with this program is limited by the capability of the back-up system needed to take over operation

in the event of a computer malfunction.

All three programs will be delivered to UMTA in the form of source decks, object decks, and complete commented source listings together with a user's manual for each program. A complete program description, including flow charts, will be prepared and delivered for the basic program and the Operational DOS Program and in addition, an acceptance test specification will be delivered for the Operational DOS Program. This specification will be the basis for the performance of an acceptance test on that program prior to delivery.

3.2 The Basic Program

3.3 The Advanced Program

3.4 The Operational DOS Program

3.4.1 General Description - The Operation DOS Program is an improvement over the advance program in order to eliminate the need to operate on an IBM 360/67 machine with a single input terminal. This program could be used to operate a Dial-A-Bus system of twenty to thirty vehicles. In fact, the size limitation for a Dial-A-Bus system using this program would be determined by the back-up capability for the scheduling function rather than the program itself. (Since the program still utilizes only a single computer, there is no automatic computerized backup.) The Operational DOS Program is capable of being operated on any IBM System 360 Model 40 or better computer. It operates under a Disc Operating System, a standard supported IBM System.

With the documentation to be delivered with this program, as described in the next section, this program truly represents a complete package which could be furnished to locations desiring to implement a complete thirty-Vehicle Dial-A-Bus system. To adapt the program to a locality, the street network of that locality must be inserted into the program.

The Operational DOS Program will have a graphics capability utilizing and ARDS display unit which will enable supervisors to visually monitor the state of the system, and which can also serve as an educational device for new personnel or for other people interested in observing a Dial-A-Bus system in operation. The system will also accept standing requests for service from customers as well as provide for automatic billing.

The program will be debugged and, prior to delivery, will undergo an acceptance test conducted in accordance with an acceptance test specification delivered under the documentation requirements of this work statement.

Specifically, the Operational DOS system will have the following capabilities:

1. Restart capability
2. Cancellation of a service request
3. Handling of more passengers at a stop than expected
4. Vehicle breakdown procedure (includes re-assignment of passengers)
5. Detection of late vehicles (both for internal computation purposes and breakdown suspicion)
6. Priority classes
7. Graphics display
8. Standing requests
9. Automatic billing
10. Provision of hard copy for use in a manual backup system

3.4.2. Information Flow - The information flow in the Operational DOS Program is the same as that for the advanced program which is described in paragraph 3.3.2. The difference in this area is that the Operational DOS System can be operated with multiple input terminals such as an IBM 1050 or IBM 2740, whereas the advanced program was limited to a single terminal.

3.4.3. Implementation - The Operational DOS Program has been designed to operate a twenty-to-thirty-vehicle Dial-A-Bus system in any locality. All this required was to code the street network and to insert it into the program in accordance with the instructions contained in the documentation supplied with the program.

The program operates under the Disc Operating System and can operate on the following model IBM System 360 computers: Model 40, Model 50, Model 65, Model 67, and Model 75; and the following IBM 370 computers: Model 145, Model 155, and Model 165. The computers can be on-site or at a remote location, and the operation can be time-shared or dedicated.

On the larger machine configurations (Model 65 or above) more efficient operation would be achieved using a partition of the computer running under the OS/360 rather than DOS. While not specifically covered in this contract, conversion of the program from DOS to OS is considered a minor programming job.

With its flexibility, the Operational DOS Program and its related documentation provides an excellent capability for any locality desiring to operate a Dial-A-Bus system to be able to do so with a minimum of additional support which could be obtained from any of a number of computer-oriented organizations throughout the country.

In addition to the program, to avoid a system shutdown, the locality should provide for a manual backup system in the event of computer malfunction. A system compatible with this computer program will be developed under Item 3.5 below.

3.5 Manual Backup System

To provide the capability of conducting a complete twenty-to-thirty-vehicle Dial-A-Bus system demonstration using the Operational DOS Program a manual backup system will be developed concurrently with the completion of the computer program. This development will include the necessary programming effort to be included in the Operational DOS Program to provide hard copy data on the state of the system at all times to serve as the input to the manual backup system. The manual backup system will be designed to allow a Dial-A-Bus system to continue to function, at a lower efficiency, in the event of a computer malfunction. This system will be documented in a handbook for use by potential Dial-A-Bus operating agencies.

3.6 Documentation

3.6.3 The Operational DOS Program - Documentation for the operational DOS program will include the following:

3.6.3.1 User's Manual - A user's manual will be provided which will include written description of all commands. With this information, UMTA or other organizations could operate the program in any desired environment. The manual also provides information on setting the program up on a computer, preparatory to operation.

3.6.3.2 Program Description - A written description will be provided which will enable UMTA or other organizations to understand and modify the program. In particular, flow charts of all programs accompanied by brief written descriptions including definition of all important variables will be provided as well as the procedures necessary to set up details of a new street network to implement operation in a specific locality.

3.6.3.3 Acceptance Test Specification - An acceptance test specification will be submitted to UMTA which will contain tests to demonstrate all of the specific capabilities of the computer program listed in section 3.4.1. The tests will also show how the program accomplishes passenger assignment and vehicle routing in an efficient manner enabling stated pickup and delivery times to be met. UMTA will provide comments on said specification within thirty days of receipt. MIT and UMTA will mutually agree upon this document prior to the start of the acceptance test.

3.6.4 Manual Backup System

3.6.4.1 Handbook - A handbook will be provided which describes a manual backup system to be used in conjunction with the Operation DOS Program to conduct a complete Dial-A-Bus system demonstration. This handbook will contain instructions on the establishment and operation of the manual backup system for use by the operating organization.

3.7 Acceptance Test-Operational DOS Program

To assure that the Operational DOS Program is debugged, checked out, and satisfactory for operational usage, an acceptance test of this program will be conducted prior to program delivery. This test will be conducted for UMTA personnel in accordance with the Acceptance Test Specification written under Task 3.6.3.3.

4.0 DELIVERY SCHEDULE

Items called for in Section 3, "Work Description", will be delivered to UMTA as follows:

Item	Delivery Date
COMPUTER PROGRAMS	
- The Basic Program	July 6, 1970
- The Advance Program	July 6, 1970
- The Operational DOS Program*	May 31, 1971

* This program will be delivered after Acceptance Testing during the week of May 25 through May 29.

Item	Delivery Date
DOCUMENTATION	
- Basic Program User's Manual	July 30, 1970
- Basic Program Description	November 30, 1970
- Advanced Program User's Manual	July 30, 1970
- Operational DOS Program Acceptance Test Specification	February 28, 1971
- Operational DOS Program User's Manual	May 31, 1971

- Operational DOS Program Description	May 31, 1971
- Manual Backup System Handbook	May 31, 1971

In addition to the above, the following will be provided to transportation Systems Center personnel designated by UMTA:

- Operational DOS Program debugged without graphics, restart, automatic billing or standing requests capability or documentation	December 15, 1970
- Operational DOS Program without automatic billing or standing requests capability or documentation	February 28, 1971

APPENDIX B

OUTLINE OF ACCEPTANCE TEST SPECIFICATION
February 1, 1971

COMMENTS ON DRAFT OF ACCEPTANCE TEST
March 30, 1971

U.S. DEPARTMENT OF TRANSPORTATION

TRANSPORTATION SYSTEMS CENTER
55 BROADWAY
CAMBRIDGE, MASSACHUSETTS 02142

DATE: February 1, 1971

REPLY TO ATTN OF: Juan F. Bellantoni/PDA

SUBJECT:

TO: Mr. Edwin H. Porter, Jr.
Massachusetts Institute of Technology
Building E40
Amherst & Wadsworth Streets
Cambridge, Mass. 02139

Dear Edwin,

According to UMTA's request and our own project plans for the DAR acceptance tests, I am sending you herewith an outline of our acceptance procedure. This outline is for USL's guidance in drawing up the draft of the ODAR computer program Acceptance Test document, item I of the outline. The draft is to be submitted 28 February.

Since the Operational DAR program is designed to run on any 360 model 40 or larger, we are making arrangements for USL to run the acceptance tests on the MITRE Corporation (Bedford) 360/50 and on a 360/67 (rented from Lincoln Laboratory or Interactive Data Corporation). In both cases, we hope to have 10 terminals at STC. I hope that USL finds these arrangements suitable for the test, and to that end we have been in consultation with Trevor Higgonet. His assistance in our visit to MITRE is appreciated.

Our experience with the DAR program so far has led me to believe that a meaningful test will require time labeling of the console inputs and outputs beyond what is now provided. We would like to explore with Trevor ways to accomplish this with a minimum of labour.

Finally, let me add that the accompanying outline does not contain detailed steps for testing the various features; I hope to work with you and Nigel Wilson in constructing the actual procedures.

Sincerely,

Juan F. Bellantoni

cc:
PDA/C. Toye
PD/F.Tung
PDA/P. Bushueff

INTRODUCTION

The Acceptance Procedures here outlined are intended to test the Operational Dial-A-Ride (ODAR) Software to be developed by the MIT Urban Systems Laboratory. The purpose of the tests is to ascertain the performance of the ODAR deliverables relative to the Work Statement prepared by USL, dated November 24, 1970.

The Procedures are divided into three general areas, covering the computer program itself, the documentation, and the back-up mode.

I ODAR COMPUTER PROGRAM ACCEPTANCE TESTS (25 May through 29 May)

I.1 Equipment and Location of Tests

This section will specify the hardware and systems software required for the test, as well as number and type of personnel, location of hardware, type and amount of data to be taken, method of recording data, method of interpreting data, and criteria for acceptance.

I.2 General Capabilities

These tests will ascertain that the program achieves the objectives stated in the Work Statement p. 21. Each test will be given in the form of one or more scenarios. A scenario is made up of

1. An input file (FT02 file)
2. Start-up and initialization procedures.
3. Chronological lists of operator and dispatcher inputs, including time and method of insertion, and input device.

4. A chronological list of all other inputs, as in 3., for machine operator, supervisor, test supervisor, etc.
5. A chronological list of computer outputs, including debug graphics, and test data recording.
6. A scenario description, giving the rationale behind the particular test, including diagrams.
7. A street map file.

The following scenarios will be executed:

I.2.1 Graphics Test

This test will cover:

a. Display of the state of the system for supervisory personnel: location of vehicles, projected and completed tours for each vehicle, number and location of passengers, etc.
b. Display of operating and personnel statistics, and customer data (such as standing requests).

I.2.2 Efficiency of Heuristic

Waiting, transit and total time constraints shall be removed and one objective function used for all tests (presumably 4). Vehicle capacity shall be set at a high value. One priority class shall be used. A series of scenarios will be constructed, of increasingly complex geometry and increasing number of buses. The secenarios will cover many-many, one-many, many-one, many-several, and several-many arrangements. In each case the most efficient route (according to the objective function) shall be pre-calculated for comparison with the actual output.

I.2.3 Delivery and Pick-up Time Constraints

These tests shall determine that delivery and pick-up time constraints are not violated. Specifically, they shall determine that

a. Pick-up and delivery times conveyed to the customer are consistent with the mean speed, and projected tour of the bus assigned, as modified due to reported lateness or earliness. Consistency shall be defined quantitatively in the above context and data gathered on it.

b. The extent of the detour that a projected tour will accommodate, without violating pick-up and delivery constraints, agrees with the allowable slack in those constraints.

I.2.4 Bus Capacity Constraints

These tests shall determine that bus capacity is not exceeded in any circumstances, except reassignment due to vehicle breakdown or system overload.

I.2.5 Response Times

These tests will determine the time elapse between the completion of a service request insertion and the printout of the vehicle assignment and customer pick-up messages, as a function of the number of O-D's in the system and of the number of vehicles in the system. They shall also determine the effect, if any, of standing requests on the response time for requests.

The response times for priority requests, request cancellation, detection of late vehicles and vehicle breakdown shall also be determined by scenarios similar to that for request response time.

All response time tests will be run under both dedicated and time-share conditions and results compared.

I.2.6 Realistics Cases

Whereas tests I.2.1 through I.2.5 are to be contrived situations, based on simple geometric arrangements of demands within a simple grid street network, the present test will check performance in a more realistic street network, with a realistic combination of program features operating simultaneously. The situation chosen should cover a typical commuting town (say, Winchester, Mass.) with subscription many-one service to the rail depot superimposed on many-several school, hospital, and shopping demands, with many-many service in the background. A realistic number of buses, demands, and constraints should be employed. As many of the general capabilities tested in I.2 and of the specific capabilities of section I.3 should be tested.

Output data should include tour routes and schedules, pick-up and drop-off time errors, information flow through operators, dispatchers and supervisor, as well as diagnostic and graphic outputs.

I.2.7 Failure Protect Provisions

These test shall determine the extent to which the software provides protection and/or recovery from failures such as

a. Erroneous inputs, including invalid or incorrect commands and mistruck keys on the consoles.

b. Hardware failures such as a single disk failure, single or multiple console failures.

The scenarios will validate that the maximum dispatching capability is maintained, consistent with available hardware and personnel.

I.3 Specific Capabilities

The specific capabilities called for on p. 22 of the Work Statement referenced in the INTRODUCTION will be tested by scenario as described under I.2.

I.3.1 Restart

The test of I.2.6 should be interrupted and restarted at several points, following the standard restart procedures. The length of the interruption should be varied over the entire range permitted by the restart procedures, and data taken on the time required to re-achieve full operation.

I.3.2 Cancellation of Service Requests

This test shall determine the effect of cancellation when bus capacity has been reached, when reassignment due to breakdown occurs, and when more than the expected number of passengers board at a pick-up.

I.3.3 Handling of more passengers at a stop than requested

These tests shall include more than one destination and various priority classes for the extra passengers. The effect on service time and pick-up times will be recorded, allowing realistic constraints to be in force.

I.3.4 Vehicle Breakdown Procedure

Simultaneous breakdown of more than one vehicle will be tested, as well as the case of inadequate seats on the remaining vehicles. Assignment for stranded passengers will be compared to assignment for similarly located new demands in terms of waiting time. The case in which more than one bus is required to handle the reassigned passengers will be tested.

I.3.5 Detection of Late Vehicles

These scenarios will test the proper incorporation of vehicle lateness, as predicted by the vehicle driver, into future vehicle routings and pick-up/delivery times. The detection of late vehicles in the event of communication breakdown will be tested as well.

I.3.6 Priority Classes

Tests shall be devised to measure the difference in service rendered to different priority customers. A method of determining this difference will be devised, such as recording a run with several priority 2 passengers, and then recording the same run with one of the passengers changed to priority 1. Similarly a set of priority 1 demands may be re-run with one demand changed to priority 2.

I.3.7 Graphics Display

(See I.2.1.)

I.3.8 Standing Requests

The treatment of standing requests will be determined by comparing the computer response to a demand sequence with and without standing requests. Tests will also be devised to check the interaction of standing requests with restart, cancellations, vehicle breakdown, and bus capacity.

I.3.9 Automatic Billing

The automatic billing feature will be tested under the following conditions:

a. All automatic billing
b. Mixed automatic billing and fare collection
c. Mixed priority classes, plus fare collection
d. Vehicle breakdown

e. Cancellation of telephone request

f. Cancellation of standing request

g. Standing requests mixed with telephone requests

h. More passengers at a stop than requested

The method of verifying requests for automatic billing will be tested. Output will be checked for accuracy and completeness of information as well as for proper format.

I.3.10 Hard Copy for Manual Backup

The contents and format of this printout will be specified; frequency of printout will be speicfied and tested for; accuracy of printout will be tested by comparison with dispatcher, operator and supervisor I/O records, as well as with graphics output.

II DOCUMENTATION ACCEPTANCE PROCEDURE

This procedure will be supplied by the Transportation Systems Center.

III BACK-UP MODE REVIEW

This procedure will be supplied by the Transportation Systems Center.

U.S. DEPARTMENT OF TRANSPORTATION

TRANSPORTATION SYSTEMS CENTER
55 BROADWAY
CAMBRIDGE, MASSACHUSETTS 02142

DATE: March 30, 1971

REPLY TO ATTN OF: Juan F. Bellantoni/PDA

SUBJECT:

Mr. Charles Broxmeyer
Urban Mass Transportation Administration
Department of Transportation
400 Seventh Street, S. W.
Washington, D. C. 20590

Dear Mr. Broxmeyer,

The "Operational DOS Program Acceptance Test Specification" draft submitted to you by MIT/USL on 2 March, 1971 has been reviewed by TSC. Our comments and suggestions are contained in Attachments 1 through 6. Many of these comments were discussed with D. Roos, E. Porter, N. Wilson, and T. Higonnet of USL in a meeting with F. Tung, P. Bushueff and myself on 18 March.

A summary of the scenarios submitted in the draft follows (numbering corresponds to reference 2):

Require Minor Revision

I.2.4	Vehicle Capacity Constraints
I.3.2	Cancellation of Service Requests
I.3.3	Unscheduled Requests
I.3.4	Vehicle Breakdown Procedure

Postponed by Previous Agreement		Date Due
*I.2.2	Efficiency of Heuristic	31 March
I.2.7.b	Failure Protect Software (Machine)	31 March
I.3.1	Restart	31 March
I.3.8	Standing Requests	30 April
I.3.9	Automatic Billing	30 April
I.3.9	Automatic Billing (description only)	31 March

Not Yet Submitted

I.2.3.a	Delivery and Pick-up Constraints (Consistency)
I.2.5	Response Times
I.2.7.a	Failure Protect Software (Human)
I.3.10	Hard Copy for Manual Backup

Require Major Revision

I.3.7/I.2.1	Graphics
*I.2.3.b	Delivery and Pick-up Constraints (Violation)
I.2.6	Realistic Cases
*I.3.5	Lateness Detection and Correction
I.3.6	Priority Classes

Not in Original Outline, But Should Have Been

*I.2.8	System Capacity Tests

*In order to expedite the approval process TSC has prepared drafts of those scenarios marked by an asterisk. Copies are included among the attachments.

Juan F. Bellantoni, Chief
Analysis and Computation Branch

cc:
MIT/USL
D. Roos
E. Porter
T. Higonnet (w/attachments)
N. Wilson

TSC
PD/F. Tung
PDA/P. Bushueff (w/attachments)
SA/F. Hassler

References:

1) Work Statement, Dial-A-Bus System Computer Programs, MIT/USL, November 24, 1970.

2) Acceptance Procedure Outline, accompanying letter of 1 February 1971, J. Bellantoni/TSC to E. Porter, MIT/USL.

3) Letter, 19 March 1971, J. Bellantoni/TSC to E. Porter, MIT/USL.

4) Letter, 17 February 1971, J. Bellantoni/TSC to C. Broxmeyer/UMTA.

Attachment 1

Review of Acceptance Test Draft
Submitted by MIT/USL for the Operational
Dial-A-Ride Computer Dispatching Program
on 2 March 1971

General Comments and Suggestions

1. The test scenarios should be put into a uniform format that serves not only as a test specification but also as a test record. A sample made up by TSC is attached (Attachment 2). Several copies were provided to MIT/USL at the preliminary review meeting on 18 March. This format references

    a) the (FT02) setup file
    b) initialization procedure
    c) hardware configuration
    d) a street map file

   all of which should be included as appendices to the set of completed scenarios. It should be noted that several different FT02 and street map files may be required. (It will not be necessary to provide one set of appendices for each scenario, as suggested in the TSC outline, reference 2.)

2. It is not clear from the Acceptance Test draft, or from the March 1 version of the Operational DOS Dial-A-Ride User's Guide, how an address such as 400 Seventh Street, S. W. is to be encoded. Neither is it clear how to encode, say, 55 Broadway or 21 5th Avenue. As suggested to MIT/USL at the meeting on 18 March, such addresses should be allowed for. T. Higonnet/USL is presently looking into the situation, and has reported that 55 Broadway and 21 5th Avenue may be acceptable as is to the ODAR program, but there may be difficulties with 400 Seventh Street, S. W.

3. MIT/USL has agreed to use a single heuristic for all tests. The one selected, and concurred with by USL on 18 March, is objective function 4. This heuristic minimizes the sum of the delivery times of all passengers, plus total trip time, on an insertion basis.

4. It should be noted that the capabilities tested in scenarios I.2.4, I.2.5, I.2.6, I.2.7, I.2.8 are not called out explicitly in the MIT Work Statement (reference 1). But TSC believes them

to be so fundamental to an operational Dial-A-Ride program that the intent of the work statement, and the specific capabilities called for in it, cannot be truly validated without successful execution of these scenarios.

5. All scenarios should be numbered according to the original outline (reference 2) to reduce confusion. The order of running the tests need not correspond to those numbers.

6. For many scenarios, the detailed input/output listing on p.2 will be obtained by actual computer run before the acceptance test is made. In such cases, the acceptance test will merely verify the known result. It is obviously advantageous to pretest as many scenarios as possible in that manner.

Specific Comments and Suggestions

The scenario numbers in parentheses refer to the MIT/USL Acceptance Test Draft. All others refer to the outline (reference 2).

I.1 (I.1) The complete hardware configuration, as transmitted to TSC, should be put in an Appendix to the Acceptance Test Specification. It should be stated that TSC will provide the hardware for the tests, including an ARDS and up to ten 2741 and/or KSR33/35 terminals, as required by specific tests; and that the terminal configuration and telecommunication software is subject to MIT approval. This section should also call out some means of presenting for human consumption the data readable by computer, so that such data may be incorporated into the test records.

Finally, it should be stated in this section that the means of interpreting the data and criteria of acceptance will be given for each test in its scenario sheets, with the general requirement that each scenario be run without interruption to its end, and that the software not be modified after the Acceptance Tests start.

I.2.1 (I.2.1) Graphics: It is assumed here that "graphics" refers not to the display of alphanumeric information on a CRT, so much as to the display of 2-dimensional line drawings (no shades) on a CRT. In order for this test to make sense it is necessary to state what information the graphics presents, and why. The Work Statement says that the ARDS will "enable supervisors to visually monitor the state of the system" (p.21). TSC suggests that the items that constitute the "state of the system" should be enumerated and those amenable to graphical display should be tested for.

The scenario, as it now stands, does not make clear, under "Purpose," what display functions are being tested. MIT/USL has been requested (reference 3) to provide a brief written description of the function(s) of the ODAR graphics to aid us in reviewing this scenario.

Even without a list of features to be tested, some comments may be made on the graphics capability.

1) The graphics capability should be tested in as realistic a situation as possible because there is no way to predict its utility in a complex situation from simple cases. Hence this scenario should be one of the realistic cases I.2.6 modified by the addition of the graphics output.

2) TSC will provide photographic recording of the output; the type of the output; and the points in the scenario at which the record is taken, should be called out.

I.2.2 Efficiency of Heuristic: By prior arrangement (reference 4), the detailed output lists for this set of scenarios need not be submitted before 31 March. Also by prior arrangement, TSC will provide some of the scenarios. Herewith are drafts of fifteem (15) heuristic efficiency scenarios (Attachment 3). They operate on a grid street network, also provided herewith (Attachment 4). This is a revision of the original grid network made up by TSC , and is identical to the network listing given to MIT/USL at the 18 March meeting.

The scenario drafts of Attachment 3 are complete except for references and actual console I/O lists. MIT's comments and suggestions on these scenarios are solicited.

I.2.3 Delivery and Pick-up Time Constraints:

a) Consistency. The draft does not contain a scenario corresponding to this test. As discussed with USL, it is desired to check the computed pick-up and delivery times with what would be accomplished by a vehicle moving at the assumed mean speed over the actual street network. Hence the scenario really checks the computer's internal approximation to the street network. If the network is a grid, and O-D's are spread randomly over it, then the mean error is foreseen immediately to be $\bar{\bar{\epsilon}}$,

$$\bar{\epsilon} = \frac{2}{\pi}\int_0^{\pi/2}\left[1 - |\sin\theta| - |\cos\theta|\right]d\theta$$

$$= 1 - 4/\pi$$

$$= -.27$$

or 27% of the calculated transit time. Rather than verify this (for example, by setting FUDGE to 1.27 and obtaining zero average error over the grid), a more practical test should be made. Hence it is recommended that a sufficient number (say, 20 or more) O-D's be entered from Test Town #3 (this is Cambridge, see I.2.6 below) and the pick-up and delivery times be checked against one or more feasible routes taken from the map, using the same mean vehicle speed as the program. The O-D's should be scattered about the town, and it is suggested that the scenario submitted for the Realistic Case (I.2.6) be adapted. Post-test reduction should produce the variance of the error and the value of FUDGE that brings the mean to zero.

b) Constraint Violation (I.2.3.a). This scenario should be written for the grid (Attachment 4) so that distances may be selected and calculated more easily. To facilitate re-writing, the scenario has been sketched out for a grid (Attachment 5). MIT/USL will find it relatively easy to check and complete the scenario.

I.2.4 (I.2.4.a) Vehicle Capacity Constraints: This scenario is well constructed as is and needs only to be put into the proper format.

I.2.5 Response Times: MIT/USL did not submit a scenario for this test. Discussion with T. Higonnet of USL brought out the advantages of combining this test with I.2.8, discussed below.

I.2.6 (I.2.6) Realistic Cases: The scenario submitted tests out the simultaneous, or nearly simultaneous, operation of the following features: priority classes, unscheduled requests, cancellations, graphics display, vehicle breakdown, lateness detection, and restart. It does so in a many-many environment. Standing requests and automatic billing are not included in the scenario. It is recommended that they be included so that this scenario can serve for tests I.3.8 and I.3.9 as well. Also, provision should be made for using the output data of this scenario for test I.2.3.a, as mentioned therein.

Notwithstanding the above, the scenario has several shortcomings, as follows:

1) The scenario submitted is based on the same section of Cambridge that was used for all other scenarios submitted, and in fact, for almost all of the tests performed on the ODAR/DOS program so far. TSC fears that continued use of this street file will preclude encounter with many of the practical problems that the program is intended to handle. For example, this section of Cambridge is almost uniformly connected by streets (except for the area along Memorial Drive) and does not exhibit the strong sectorization caused by railroads and/or highways in many commuting towns that Dial-A-Ride is aimed at. Such sectorization can profoundly affect the dispatching strategy and efficiency. Also, the area is relatively free of natural boundaries such as rivers, lakes, parks, and forests that are common in practical cases. Finally, the adequacy of the encoding scheme itself should be tried out on more than one town.

2) The scenario does not test out the program in other than the many-many case. In particular, the less sophisticated, but much more common

cases of many-one, one-many, many-several, and fixed route plus diversions are far more likely to occur in practice (e.g., Mansfield, Toronto, Rochester, Haddonfield and Peoria). These are special cases of many-many only for truly optimal algorithms; for sub-optimal algorithms (heuristics), they are essentially different problems and must be treated as such.

3) The scenario, as submitted, services some twenty passengers with thirteen buses in about 30 minutes, an unrealistic ratio. The interesting cases (home-work, home-recreation, etc.) involve from 10 to 20 demands/hour/vehicle. Also, the scenario does not specify how to obtain such data as average service level, average wait time, average delivery error from the "separate program" it mentions.

For the reasons given in 1), 2), and 3) above, it is recommended that the scenario submitted for Realistic Cases be modified as follows:

1) Standing requests and automatic billing should be added.

2) Predicted delivery times be outputted, and compared with map route times, as suggested under I.2.3.a.

3) The number of vehicles should be reduced to, say, six and the number of passengers increased to at least sixty (including standing requests) per hour.

4) The output data and how it is to be gathered should be specified in more detail, including graphic outputs.

5) Realistic Case scenarios should be written for other towns. They should include many-one, one-many and fixed route plus diversions.

In line with the last recommendation, TSC will provide a street map file and scenario draft for many-one-many service in Haddonfield, New Jersey (Test Town #2) and the same for fixed route plus diversion in Mansfield, Ohio (Test Town #1). These are in preparation now. Although

USL has indicated, through T. Higonnet, that the program is not designed to cover fixed route plus diversions, it is hoped that the availability of the map file and scenario outline will provide the occasion for creative adaptation of the program's capabilities by the USL personnel most familiar with it.

I.2.7 Failure Protect Software: The portion dealing with protection from human errors, I.2.7.a, has not yet been submitted; the portion dealing with machine errors, I.2.7.b, is scheduled for 31 March.

I.2.8 (I.2.4.b) System Capacity and Response: This scenario, which was not in the original outline but should have been, has been drafted by TSC and is included here (Attachment 6). The reduction in available core in the MIT version may not be necessary.

I.3.1 Restart: This scenario is due 31 March 1971. It should be based on I.2.6.

I.3.2 (I.3.2) Cancellation of Service Requests: The draft submitted by MIT should be adequate when put in the format.

I.3.3 (I.3.3) Unscheduled Requests (Anomaly): This scenario is well constructed. It is suggested, however, that unscheduled deliveries, be included since this is, in TSC's opinion, a desirable feature and one that is already in the program. The latter fact was brought out by MIT in the meeting on 18 March.

I.3.4 (I.3.4) Vehicle Breakdown Procedure: The idea behind the scenario is acceptable. However, many ramifications suggest themselves, such as two rescue vehicles required to carry all the stranded passengers, breakdown and repair in quick succession, and the method of informing the supervisor. It was agreed at the 18 March meeting that TSC would write additional scenarios for such cases as time permits.

I.3.5 (I.2.3.b) Lateness Detection and Correction Test: Although apparently suitable when reviewed at the 18 March meeting, reflection shows this scenario needs extensive revision. It should be rewritten for the grid network. The purpose of the grid here is to correlate waiting time constraints and delivery time constraints with distances travelled over the grid by the vehicle, assuming uniform vehicle speed. The scenario has been sketched out by TSC (Attachment 7).

I.3.6 (I.3.6) Priority Classes: It was clarified by USL at the 18 March meeting that priority classes are distinguished by different constraints, as on page 1 of the scenario sheets. Since this is so, it is necessary to state the constraints corresponding to each priority class and to test that the program makes the proper distinctions. Moreover, the constraints are specified in minutes of time; for the assumed mean vehicle speed this is equivalent to distances. Therefore the servicing of priority requests is dependent on distances and the entire scenario should be done on a grid. In rewriting this scenario, USL should insert the priority two (higher) request between the priority one (lower) demands, as well as on the end, as in their submittal. In fact, scenario I.2.3.b (I.2.3.a) may easily be modified to this end by running it with arnie and bob priority one, and then changing arnie to priority two.

I.3.7 Graphics Display: This test is the same as I.2.1. It should be included in the Realistic Cases, Test Town #2, which is the only complete town of the three. It is advisable to make the graphics an option in I.2.6, rather than to make other aspects of the scenario depend on them unnecessarily.

I.3.8 Standing Requests: This scenario is to be submitted 30 April. It is advisable to base it on I.2.6, by adding standing requests.

I.3.9 Automatic Billing: The same comments apply here as for I.3.8.

I.3.10 Hard Copy for Manual Backup: MIT/USL has not yet submitted a test scenario for this.

APPENDIX C

SUMMARY OF FEATURES CONTAINED IN

THE OPERATIONAL DOS PROGRAM USER'S MANUAL

Features		Status	Comments
A.	Customer Request Information input by operator		
1.	self identification(e.g. name or identification code)		
1a.	group traveling	A*	input at the passenger terminal
1b.	priority requests	A	input using PRIR command
1c.	automatic billing specification	A	input using ACCT and followed by a seven digit no. see report on automatic billing program.
2.	origin location	A	input at passenger terminal
3.	destination location	A	input at passenger terminal
*4.	abandonment of a passenger request	A	request cancelled using the QUIT command.
5.	zero trip length	A	will not accept a destination where the trip length equals zero.
6.	passenger cancellation	A	trip and bus assignment cancelled at the passenger terminal using the CNCL command
B.	Computer output information transmitted by operator to customer		
1.	acknowledgement of service request and estimated pick-		printed output message gives the name

*A indicates the command was available. NA indicates the command was not available.

	ups and delivery times	A	of passenger
2.	vehicle identification number		estimated pickup time, estimated drop off time and the number of the vehicle the passenger if assigned to
3.	coordinate determination	A	any terminal can find out the coordinates of a given address by using the WHR command
C.	Information generated by CARDOS sent to a vehicle terminal		
1.	vehicle commands(e.g.	A	the vehicle commands appear according to vehicle number at a vehicle dispatching terminal directed by the supervisor (see directing vehicle messages).
2.	location of next vehicle stop	A	output at the vehicle terminal
3.	number of passengers to be collected or discharged	A	output at vehicle terminal for pick-ups only, not for drop offs
4.	passenger identification	A	identification output according to the name given as input at the passenger terminal
5.	fare(s)if stop is a pickup point and fare is to be collected	NA	the fares are not printed at any terminal
6.	notification of vehicle lateness	A	Cardos automatically prints messages for vehicles overdue at

		five minute intervals
D. Information received by the dispatcher from the vehicle		
1. arrival of vehicle	A	the VEHI command is used to inform CARDOS of a vehicle arrival. Following this input, a message is output informing the dispatcher of the vehicle's next assigned stop
2. Eroneous vehicle specification	A	the RETR command can be used to withdraw a priviously invalid VEHI command and correct the vehicle's tour
3. repeat next stop for vehicle	A	the REPE command will repeat the last assignment for the specified vehicle
4. specify a vehicles location	A	the LOCA command updates CARDOS with the new vehicle position
5. phasing a vehicle out of service	A	the HOLD command prohibits any new passengers to be assigned to that vehicle
6. breakdown of a vehicle	A	the BREA command breaks that vehicle down, making it unavailable for future service(seel.d.7), and reassigns any passengers on the vehicle tour
7. restoring a vehicle to	A	the RSTO command

	service		restores to service a vehicle that had been previously phased out of service or broken down
8.	anomaly at a pickup	A	the ANOM command informs CARDOS that the number of passengers at a pickup is other than that expected.
9.	cancellation of a passenger in a vehicle	A	the CNCL command informs CARDOS that a passenger wishes to terminate his trip on a vehicle
E.	Supervisory functions input at the supervisor terminal		
1.	statistics	A	the STAT command causes a full statistical output, as described in section 12 of the User's Guide, to be printed on the high speed printer. The SYST command causes a system status, as described in section 12 of the User's Guide, to printed on the high speed printer.
2.	monitor switch setting	A	the MONI command causes trace information, useful in the event of a system crash, to be printed on the high speed printer. The NOMO command causes a stop to trace information printing.
3.	changing average speed	NA	the SPEE command

		should change the average speed of the vehicles in the system but it is not operable. When the command is given CARDOS fails due to a main task termination
4. detection of overdue vehicles	A	the OVER command provides the supervisor with information as to which vehicles are due to arrive at their next stop within the specified time.
5. system backup messages	A	messages are written at the supervisor terminal to indicate when the system backup capacity is being exceeded
6. directing vehicle messages	A	the DRCT command will cause vehicle dispatching messages to be printed at any terminal directed. This command is used to assign certain vehicles to a number of vehicle dispatchers or in the case of terminal failure
7. restart	A	messages concerning the outcome of an attempt at restart are printed at the supervisor terminal
8. hanging up a terminal	A	the HANG command will disconnect the specified terminal (see DIAL command)

9. Dialing up a terminal	A	the DIAL command enables an operator to reconnect a terminal that has been disconnected using the HANG command. However, if the terminal identification is not known to CARDOS the HANG and DIAL command will not work for that terminal(see failure protect)
10. vehicle terminal backup	A	the supervisor can execute any vehicle commands should the need arise.
F. Graphics features performed at the Graphic terminal		
1. general commands	A	the ERAS command will clear the display screen and then display the outline rectangle of the street map
2. passenger commands	A	the passenger commands consist of displaying the desired trips of the passenger. The WAIT command displays the desired trips of all passengers who have waited more than the specified time. The W8TD command causes the desired trips of all those passengers on buses who waited more than a specified time to be picked up.

3. vehicle commands	A	the vehicle commands give a complete vehicle tour. The VEHI ALL command shows the last reported position of all the vehicles. The VEHI command shows the projected tour of the vehicle assigned to that passenger specified. The HIST command displays tour history for that vehicle.
4. Supervisor backup feature	A	All the supervisory commands' can be input at the graphics terminal.
G. The backup feature Output at the Backup	A	no command can be input at the backup terminal.
1. passenger trip information	A	as each request is received a line of information connecting the demand number, number of passengers origin, and destination is printed at the backup terminal.
2. vehicle tours	A	as a tour change is made(a request added or a passenger dropped off) an updated tour list is printed. However, these messages do not occur for a vehicle broken down or when a CNCL request is received. Also the same messages are printed out twice when a new

request destination
to be added to the
bus tour.

H. Failure Protection Feature

CARDOS has certain failure protect features which are the HANG and DIAL commands for terminal failure, input data checking to prohibit invalid commands and values to be accepted and the RETR command to withdraw a previously entered command. These features provide for possible failures due to human error, whereby the restart procedure provides for computer hardware failure. In the event of terminal hardware failure, certain features can be assumed by other terminals and output messages directed to other terminals (see DRCT command).

If extra terminals are available, they may be connected. However, during the initial start up procedure, until the terminal type has been accepted and stored by CARDOS, any terminal failure will cause that part (or telephone line) to be unaccessable for future reconnection to CARDOS.

I. Computer Terminal

Inputs to the computer terminal control the operation of the computer under the constraints of DOS.

In addition to the handling of the input job stream, certain questions regarding the use of graphics and the number of terminals to be used must be provided at the computer console.

Error messages generated by CARDOS appear at this terminal as does the DOS error messages.

APPENDIX D

STC-DOT-TSC-297-71-876

EVALUATION OF THE AUTOMATIC
BILLING FEATURE OF THE
O-D-A-R PROGRAM

Prepared by

Scott Moffatt

Service Technology Corporation
Cambridge, MA 02142

August, 1971

Prepared for
B. Paul Bushueff

DOT Transportation Systems Center
Cambridge, MA 02142

1. CONCEPTS & PURPOSE

The intent of the automatic billing feature of O D-A-R was to provide a means by which the computer program could monitor its own usage, and provide information which would allow the installation to bill its users automatically. This was accomplished only in part, with the rest of the work left to the installation.

O-D-A-R punches offline (at the time of passenger dropoff), a "transaction" card(s) which contains various information such as origin, destination, account number, etc. These cards are to be read by a separate program (i.e., not part of O D-A-R) and the information contained therein used to create a bill to be sent to the user. This separate program which constitutes most of the actual billing procedure does not exist. It is considered to be installation-dependent and will have to be written by each installation itself.

2. THE TRANSACTION CARDS

The format for the transaction cards punched by O D-A-R given in the appendix of Volume IV is in error.

The cards are punched two per transaction, and they are punched in "A" or alphameric character format whether they are real numbers, integer numbers or true characters. Therefore, if they are read all in "A" format the proper information will be entered into core and may then be handled by the program as integers, etc. The proper format is shown in Table 6.

The information present on these cards seens to be sufficient for most accounting purposes. Any additional pieces of data could be added easily due to the uniform formatting.*

---

* [However, the description of the card formats on page 446 of Volume IV of the documentation places the transaction code in the wrong position on the card, and omits the expected pickup and delivery times that are punched on the cards. Also, judging from the meaningless standard trip numbers punched, standard trips are not supported by CARDOS, although standing requests can be processed] B.P.B.

TABLE 6. TRANSACTION CARD FORMAT

ENTITY	MODE	INPUT FORMAT	SUGGESTED OUTPUT FORMAT	NO. OF BYTES (CHAR. ON CARD)	CARD COLUMN
CARD #1					
(1) Transaction Code	Integer	A4	I10	4	1-4
(2) Passenger Name	(ALPHA)	5A4	5A4	20	5-24
(3) Time of Request	Real	A4	F10.3	4	25-28
(4) rigin Address	(ALPHA)	5A4	5A4	20	29-48
(5) Origin Coord	Integer	A4	I10	4	49-52
(6) Origin Zone(currently not used)	Integer	A4	I10	4	53-56
(7) Est. of Pick-up Time mode at demand assign. time	Real	A4	F10.3	4	57-60
(8) Actual Pick-up Time	Real	A4	F10.3	4	61-64
CARD #2					
(1) Destination Address	(ALPHA)	5A4		20	1-20
(2) Destination Coord.	Integer	A4	I10	4	21-24
(3) Destination Zone (currently not used)	Integer	A4	I10	4	25-28
(4) Est. of Del. Time mode at demand assign. time	Real	A4	F10.3	4	29-32
(5) Actual Delivery Time	Real	A4	F10.3	4	33-36
(6) Passenger Acct. Number	Integer	A4	I10	4	37-40
(7) Passenger Priority Class	Integer	A4	I10	4	41-44
(8) No. of Pass. in Group	Integer	A4	I10	4	45-48
(9) Veh. Assign. Number	Integer	A4	I10	4	49-52
(10.)Stan. Trip Number	Real	A4	F10.3	4	53-56

3. THE CUSTOMER ACCOUNT FILE

Two programs, each consisting of a set of similar subroutines, do exist, however, which allow the installation to create, update and maintain a customer account file.

The first of these is called ADDC and is used either to create the file initially, or to add new customers to an existing file. The input data to this program (customers name, address, credit rating, etc.) must be punched onto data cards. ADDC will assign the new customer an account number according to a relatively sophisticated check digit technique, and will load initialization data into the customer's record in the file.

The second program is called CONTROL and is used to alter, update, or delete currently existing customer data. Its input is in the form of a series of simple commands which allow the installation personnel to alter the information about any customer in the account file.

These two programs take care of the customer accounting information. The program which uses this information to generate bills is the same one discussed in Section 1 above which must yet be written for each installation. A combination of the data from the O D-A-R transaction cards and the customer account file should be enough to enable bills to be generated by this program.

It is at this point several bugs exist in the current system. According to the automatic billing documentation both ADDC and CONTROL offer the ability to input standard trip information nto the customer file from cards. In both programs they require the coordinates for the origin and destination of the standard trip to be punched in packed format (i.e., both coordinates in one word) and attempt to read them as a four-character integer. Since the packing technique used involves multiplying one of the coordinate to the result, this "packed" integer will invariably be greater than four characters long.

Clearly it was meant to be read as four EBCDIC characters which would be converted internally to an integer when needed. This would necessitate the following procedure to punch the input cards:

1. The packed number would be computed from the original x - y coordinates,

2. The 32-bit integer thus formed would be examined in four 8-bit bytes and each byte would be checked to see

what EBCDIC character corresponded to that particular 8-bit configuration,

3. The four characters would then be punched on the card in the appropriate columne

Of course, since most of these would be unprintable characters, they would have to be multi-punched one at a time. To avoid all this obvious nonsense, the programs ADDC and CONTROL should be modified to read the coordinates in separately, and then perform the packing operation internally. Such a program modification should be easy to do and would certainly eliminate the clumsy method now used.*

4. SYSTEM TESTS

ADDC was found to operate quite successfully without the standard trip feature. However one must remember to initialize the disk file area prior to running the program.

The customer file having been created CONTROL was successfully employed to update the file. Control output a thorough listing of all changes made should any accounting trace be required.

These evaluations were made froma detailed examination of the source listings and the operation of the two programs provided. Even so it should be kept in mind that a complete system test would require the inclusion of the third program yet to be written.

---

* [Also the computer programs ADDC and CONTROL have not been written to accept standard trips should the coordinates be input correctly. Therefore, the documentation is in error in regards to standard trip data lacks procedures for packing the origin and destination coordinates, and is incorrect in the format of the transaction cards punched (the proper format being required for the creation of the missing automatic billing program).] B.P.B.

5. FINAL COMMENTS

With the exception of the transaction cards and the account number generation algorithm, there appears to be no automatic billing feature. The crux of such a program is not written, and the two file maintenance routines that are in existence are not sufficient as they now stand. An estimat of 80 t0 120 man-hours would be required to write and debug the main billing program itself to turn out finished bills.

As for the documentation on the "automatic" billing feature, it ranges from vague to incorrect. With these ideas in mind, it is suggested that this billing feature not be considered as a positive factor for the O D-A-R system.

APPENDIX E

EXCERPTS FROM THE

O D-A-R USER'S MANUAL

TABLE OF CONTENTS

TABLE OF CONTENTS (CONTINUED)

10. SAMPLE JOB CONTROL AND INPUT FILE DESCRIPTION

10.1 Sample Job Control

This job control is suitable for running CARSDOS on an IBM 360/50. It assumes that the necessary file definitions have been placed in the standard label area (see IBM DOS publications for further details).

```
// JOB RUN THE CARS SYSTEM
// ASSGN SYS002,X'00C'
// ASSGN SYS005,X'00E'
// ASSGN SYS006,X'130'
// ASSGN SYS007,X'130'
// ASSGN SYS008,X'130'
// ASSGN SYS003,X'130'
// ASSGN SYS004,X'130'
// DLBL IJSYS03,'CARS DUMPING FILE1',99/365
// EXTENT SYS003,234079,1,,3720,60
// DLBL IJSYS04,'CARS DUMPING FILE2',99/365
// EXTENT SYS004,234079,1,,3760,60
// ASSGN SYS010,X'032'
// ASSGN SYS011,X'033'
// ASSGN SYS012,X'034'
// PAUSE BEFORE RUNNING THE SYSTEM
// EXEC SUPR000
COLD
DATA STARTS HERE
01    5  00.0
   20   20   20    20    20   20
DOS TEST RUN
 2 3    4  000 0 1
 0  20  8  4.0 .019  0. 999. 00.0
 0.50 1.20  4.0
 2 0.5 0.5
 10. 1.5   5.  1.5  15.
  5. 1.3   5.  1.3  10.
  0.0  0.0  25  0.0
 1 1 1   0.   0.   15.
 0 03412
MIT
12 CENTRAL SQ
CARS
```

```
B.U. BRIDGE
14 PLYMPTON ST
365 WESTERN AV
MORSE SCHOOL
700 MEMORIAL DR
600 GREEN ST
23 CENTRAL SQ
83 PLEASANT ST
50 WINTHROP ST
3 MT. AUBURN ST
27 HAYWARD ST
132 EPIE ST
FRONT ST
29 BOW ST
12 PETERS ST
PILGRIM ST
7 LOPEZ ST
/*
/&
```

11 LIST OF ALLOWABLE ADDRESSES

This is a list of the allowable addresses for CARSDOS. Because of a particularity of the address-to-coordinate file construction program, street names are easy to find and verify but not place names. For place names, the first two characters of the name will be found in the TYPE column and the rest of the name in the STREET NAME column. The entire list is alphabetical by STREET NAME.

All addresses with a street number between those in the LO and HI columns are considered to be at a point with co-ordinates given in the X and Y columns.

Two fictitious addresses, LOWER LEFT and UPPER RIGHT which are given the co-ordinates of the lower left and upper right hand points of the service area must appear in the street file.

STREET NAME	TYPE	LO	HI	X	Y	
BRIDGE	BU	000	999	191	36	
ACORN	ST	6	27	156	14	
ALBANY	ST	10	10	182	53	
ALBANY	ST	50	91	180	47	
ALBANY	ST	115	140	177	42	
ALBANY	ST	143	195	174	36	
ALBANY	ST	224	298	171	27	
ALLSTON	ST	57	99	161	19	
ALLSTON	ST	137	171	156	19	
ALLSTON	ST	198	240	151	19	
AMES	ST	1	5	199	54	
AMES	ST	12	50	182	56	
AMESBURY	ST	000	999	172	11	
ANGLIM	ST	000	999	169	21	
ARROW	ST	1	23	115	47	U
ATHENS	ST	9	31	116	42	U
AUBURN	ST	100	135	158	38	
AUBURN	ST	146	174	153	38	
AUDREY	ST	000	999	173	13	
B SHOP	SU	000	999	175	44	
BANKS	ST	7	49	119	43	D
.	.	.	.	.		
.	.	.	.	.		
.	.	.	.	.		
.	.	.	.	.		
YCE CHEN	JO	000	999	180	18	

<u>NOTE</u>: The User's Manual contains the complete list of addresses and coordinates. They cover that portion of Cambridge bounded by Main Street, Massachusetts Avenue; Bolyston Street and Memorial Drive (See Figure 5).

APPENDIX F

STC-DOT-TSC-297-71-875

EVALUATION OF THE O-D-A-R PROGRAM DOCUMENTATION

Prepared by

Scott Moffatt

Service Technology Corporation
Cambridge, MA. 02142

August, 1971

Prepared for

Paul Bushueff

DOT Transportation Systems Center
Cambridge, MA. 02142

TABLE OF CONTENTS

1. SCOPE OF THE EVALUATION

The following documentation of the O-D-A-R (Operational Dial-A-Ride) program developed by the MIT Urbans Systems Laboratory was included in this evaluation:

(a) system design, data structure, and description of variables as described in Volume I of the Operational DOS Program Description;

(b) individual subroutine documentation contained in Volumes II, III, and IV of the Operational DOS Program Documentation;

(c) the O-D-A-R Users Manual; and

(d) commented source listings of the O-D-A-R subroutines.

These items will be evaluated individually in the following sections, with final overall comments at the end.

2. SYSTEM DESIGN AND PROGRAM STRUCTURE

This section of the O-D-A-R documentation leaves the most to be desired. A rather sketchy account of the system neglects to detail the most important part of the program, i.e., the actual assignment algorithm. What the user really desires to know is what criteria are used to select a particular vehicle to satisfy a particular demand. The document spends a great deal of time talking about satisfying links constraints, tour constraints, demand constraints, but never says <u>what</u> these constraints are!

The discussion of the modular design of O-D-A-R suggests that such a structure was chosen for the purpose of easy program modification. It goes on to point out which routines should be altered if a new demand assignment algorithm were desired. No mention is made, however, of the current algorithm and one assumes that the user is expected to figure it out from looking at the source listings themselves. This would be a long and tedious process.

3. DATA STRUCTURE AND DESCRIPTION OF VARIABLES

The data structure of O-D-A-R is fairly well documented. The concept of the data pool is explained well and the access techniques used are clear. A rather lengthy description of the variables in all the commons and data blocks provides a convenient reference to anyone who is attempting to modify the program. Unfortunately, many of the variable descriptions involve the use of terms which are not defined elsewhere in the documentation (e.g., probability seeds, objective function, composite vehicle, etc.).

The statistical variables are listed but no mathematical equations showing how they are computed are given.

4. INDIVIDUAL SUBROUTINE DOCUMENTATION

This part of the documentation is also done moderately well. Each subroutine has a brief description of what other routines and variables and commons are used by the subroutine itself. Also a capsule summary of what the routine does is provided. A flowchart then follows which details information flow.

Once the name of the routine desired is known, one may obtain an excellent idea of its structure and purpose. However, the absence of some kind of cross-reference system or index makes figuring out what routine one needs almost a hopeless task.

5. USERS MANUAL

The O-D-A-R User's Manual provides a thorough explanation of all commands, responses and error messages which refer to the various terminals used by the system. However, no real examples are given as to how to use the system in the interactive mode. A sample batch run setup is included but no sample 360/67 runs are shown. In other words, if one were placed in front of a terminal and told that it was a vehicle terminal, or a passenger terminal, etc., this manual would provide one with a good description of what one could or could not do.

On the other hand, no information relating the terminals to one another is given, i.e., no overall feel for what the system is doing is conveyed to the user. Therefore, the manual does not satisfy the basic requirement of any users manual, in that it alone is insufficient information to operate the system.

6. COMMENTED SOURCE LISTINGS

The commenting in the O-D-A-R source listings is excellent. Detailed logic flow may be followed easily using the comments provided. Indeed, one is somewhat overwhelmed by the sheer bulk of the comments which are at least 50% of the source listing. This is by far the most superior portion of the system documentation.

7. SUMMARY COMMENTS

An overall view of the O-D-A-R program is missing from its documentation. The impression received from reading through these documents is that the more trivial concepts (e.g., data structure, variable descriptions) are presented in great detail, while the more meaningful analyses like the demand-assignment algorithm are all but ignored. The total absence of any mathematical equations describing the algorithm being used is striking considering the apparent degree of sophistication required by such a system.

O-D-A-R may be used to test a particular geographical area for various demands and traffic conditions, etc. However, it is not as well suited to be used as a tool for experimenting with various assignment algorithms as the documentation's description of the O-D-A-R modularity leads one to believe.

The documentation is not self-supporting; that is to say, a thorough demonstration of the system by MIT would have to be presented before it could be run by the average user. This limits the flexibility of the program and ties it somewhat to the organization which originally wrote it.

The mechanics of modifying O-D-A-R subroutines are made simple by its modularity. However, the lack of an overall system cross-reference makes it nearly impossible to figure out which of the more than 100 subroutines to modify for any given program function.

APPENDIX G

THE O D-A-R ASSIGNMENT ALGORITHM

The O D-A-R Assignment Algorithm

A thorough discussion of the assignment algorithms developed on project CARS is given in USL TR-70-13 "Scheduling Algorithms for a Dial-A-Ride Systems," MIT/USL, March 1971, chapter 4. That chapter gives the general philosophy, and very likely some of the equations, used in O D-A-R. But the chapter is very broad and makes no reference of the actual algorithm programmed and delivered for the O D-A-R program.

Another source of information regarding the assignment algorithm is volume I of the program description. The write-up given on pp 9-11, under "Program Structure", does not describe the constraints employed, although it refers to "link constraints" etc. frequently. Unless an extensive and tedious search is made through the subroutine descriptions, this reference cannot be used as a description of the algorithm.

Finally, the User's Manual may be referred to. But, as pointed out in the documentation review, Section 3 of Volume I of this report, the User's Manual does not contain any description of the algorithm, heuristic or criterion.

Faced with this situation, the following description was composed by TSC, using all available information, including consultation with the MIT personnel most familiar with the program. It describes criterion 4, used for the Efficiency Scenarios I.2.2, to the best of TSC's knowledge:

1. The computer maintains a provisional future tour for each vehicle, as shown in Figure G-1, including waiting time, travel time and total time constraints for each passenger of the form shown in the scenarios:

   Waiting Time $\leq X$ (minutes)

   Travel Time $\leq Y_1 D + Y_Z$ (minutes)

   Total Time $\leq Z_1 D + Z_Z$ (minutes)

   where D is the time to go from origin to his destination on a straight line at the vehicle nominal speed, and X, Y_1, Y_2, Z_1, Z_2 are numbers determined by the program from the priority class of the passenger. Given the time of each passenger's request, the program can determine the latest times at which each stop on each tour can be made without violating any one of the three constraints for any passenger.

2. When a new request is entered, the new origin and destination are "inserted" by trial into each link of each vehicle as shown in Figure G-2, to determine whether any vehicle can service the request without violating any of the stop times of (1). All possible insertions are attempted for all vehicles. It should be noted that re-ordering the existing stops is not allowed by the algorithm. Possible insertions that do not violate any constraints, including those of the new customer, will be termed "feasible" insertions.

3. If there are any feasible insertions, only those are considered; if no feasible insertions are found consideration is given to the "infeasible" ones. Selection is made by picking the insertions that minimize Z.

$$Z = N_1 e_1 + N_2 e_2 + (e_1 + e_2)$$

where

e_1 = increase in tour time required by insertion of the new origin.

e_2 = increase in tour time for insertion of the new destination.

N_1 = number of deliveries on the original tour after the new origin.

N_2 = number of deliveries on the original tour after the new destination.

It should be noted that the results of scenario I.2.2-11 and discussions with MIT/USL suggest that the program includes 30 seconds for a passenger delivery in the above criterion calculation, to be added to the delivery time of all passengers that are delivered after him, but not to his own delivery time, plus 30 seconds for a passenger pickup, to be added to the delivery time of all passengers following him including his own delivery time.

4. Once a pair of insertions has been selected, the expected stop times are adjusted for the entire projected vehicle tour involved.

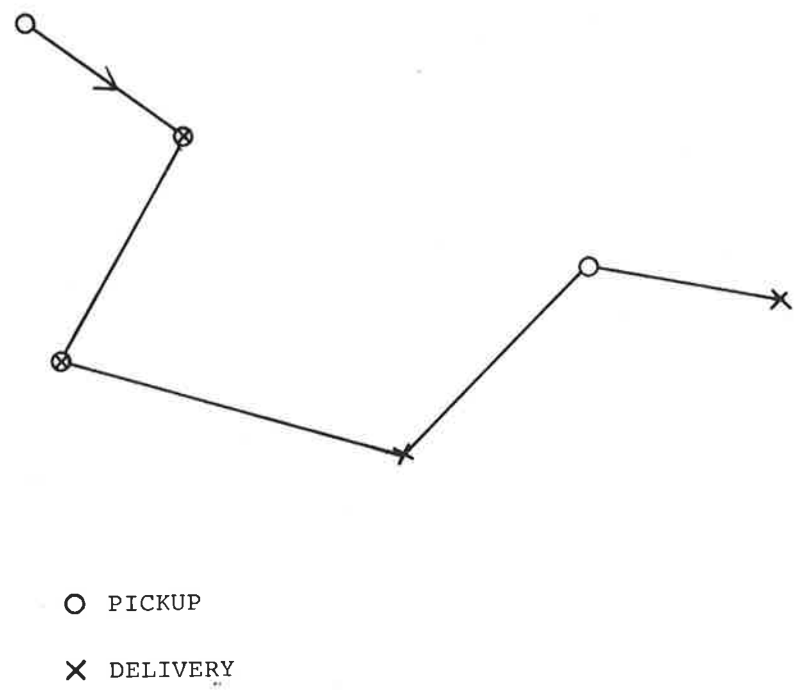


Figure G-1 Typical Provisional Vehicle Tour.

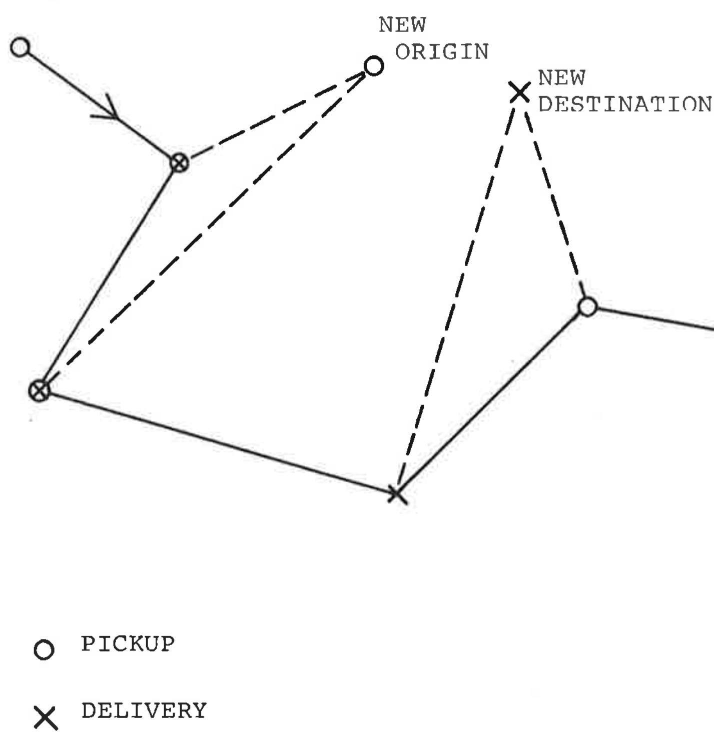


Figure G-2 Typical Trial Insertion for New Demand.

APPENDIX H

MINIMUM SAMPLE SIZE FOR SCENARIO 1.2.3-A

MINIMUM SAMPLE SIZE FOR SCENARIO I.2.3-A

The criteria for this scenario are of the form "x percent of the passengers will be picked up or delivered within y minutes of the stated time." It is desired to estimate the sample size needed to establish the value of x, with reasonable confidence.

In order to facilitate calculation, the problem will be formulated in terms of p, the probability of the pickup or delivery time error magnitude being greater than the stated tolerance. The question will be put: "If p is the probability of any one pickup (or delivery) being out of tolerance, what is the likelihood of the data showing that exactly n out of N pickups (or deliveries) are out of tolerance? This question can be answered directly by calculating B(N,n;p), the binomial probability

$$N!\ p^{n}\ (1-p)^{N-n}/n!\ (N-n)!$$

As an example, suppose 10 stops have occurred and only one is out of tolerance. It may be supposed from these data that p is .10. But this is not the only possible explanation for the data, as the following table shows:

Value of p	Likelihood of observing n = 1 for the given p
.05	.32
.10	.39
.15	.35
.20	.27
.25	.19
.30	.12
.35	.07
.40	.04

On the other hand, if the sample size N is increased to 20 stops, and the number of out-of-tolerance stops is in the same proportion (n=2) then the table is as follows:

Value of p	Likelihood of observing n = 2 for the given p
.05	.189
.10	.285
.15	.230
.20	.137
.25	.067

.30	.028
.35	.010
.40	.003

The range of p that may have produced the given result with reasonable likelihood (liklihood greater than .1, say) has here been reduced from the previous table, but still covers the span of $p \leq 0.05$ to $p = 0.20$.

Finally, if 30 observations are taken and 3 are out of tolerance, then the table appears as follows:

Value of p	Likelihood of Observing n = 3 for the given p
.05	.121
.10	.236
.20	.079
.30	.007
.40	.0003

Therefore, 30 observations will reduce the "likely" range of p to .10 or less, if 3 of the 30 observations are out of tolerance. Although they are not given here, the tables for n = 1, 2, 3, 4 show a similar reduction in the range of likely p's as N increases from 10 to 20 to 30. Hence a value of $N \geq 30$ will assure that the range of likely values of p is small enough (say, under .10) to draw useful inferences about its value.