Routing and Scheduling of School Buses by Computer

LAWRENCE D. BODIN

University of Maryland, College Park, Maryland

and

LON BERMAN

Sterling, Virginia

In this paper, a new procedure for the routing and scheduling of school buses is presented. This procedure has been implemented and tested successfully in two school districts. It resulted in about a 20% savings in cost in one of these districts and the transportation of 600 additional students with one extra bus in the second. Also, in this paper, the key aspects of the procedures for the preparation of the data are explained including the ministop concept, a new and innovative method for the coding of the network data. In each section of this paper, the core algorithm is explained, followed by the modifications that were made in attempting to implement the theoretical procedure. In this way, the contract between theory and practice can be shown.

An important question facing many local school boards is how to transport their students to and from their schools in the safest, most economical and most convenient manner. To many communities, student transportation is an expensive, controversial, and important activity. To illustrate the magnitude of the school transportation problem, the annual cost in the State of New York for school bus transportation exceeds $150 million and is increasing at the rate of about 10–15% each year. Although experience has shown that savings of between 5 and 30% can ordinarily be gained by going to a computer-based routing and scheduling system,

Transportation Science
Vol. 13, No. 2, May 1979

0041-1655/79/1302-0113 $01.25

most school districts still route and schedule their buses by hand. However, a school district can receive many side benefits by having a computerized routing and scheduling system outside of the cost savings that may evolve. For example, by including in the data that are collected the appropriate characteristics of the students, additional items of value to the districts such as student identification cards and student transportation lists can be generated.

An automated school bus routing and scheduling system has at least the following three purposes:

1) To minimize the transportation costs to the school district (capital and operational),
2) To minimize the average transportation time of a student (safety of the students), and
3) To have an automated procedure for setting up daily schedules for the buses (convenience to the district).

The system is constrained so that each student must be dropped off at his school or picked up on schedule; no route can be too long and no bus can ever be overloaded.

The term "student transportation costs" requires some discussion. Some of the student transportation costs are borne by the school district while other student transportation costs are subsidized by the state (up to 90% in New York State). However, the aid formula under which the subsidy is dispersed is not necessarily a strict function of how many students are being transported but can also be a function of other factors. As such, these aid formulas can further complicate the problem of minimizing the total student transportation costs for a district. For example, New York State has a formula which subsidizes the district for those riders who live over 1.5 miles from the school. Since the local school districts administer their transportation programs under the existing state aid formula, it is very easy to construct situations where it may be more economical for the local school districts to utilize more buses than would be implied if an overall minimal cost solution would be implemented.

Furthermore, transportation costs for a school district are not necessarily a multiple of the number of buses used by the district. For example, in 1974, the Brentwood School District on Long Island, New York, paid about $11,000 per year for a bus and driver leased for 6 hours a day, $14,000 per year for a bus and driver leased for 11 hours a day, and $15,000 per year for a bus and driver leased for 12 hours a day.

Most papers on the routing and scheduling of school buses concern themselves primarily with the routing problem by itself (BENNETT AND GAZIS,[3] ANGEL et al.,[1] and TRACZ AND NORMAN[11]). NEWTON AND

THOMAS[9] explain a procedure for forming daily schedules for the buses as well as describing a procedure for routing the buses.

In discussing the school bus routing and scheduling system, we assume that the starting and ending times of all the schools in the district are known (the problem of fixing the starting times of the school in order to save buses is discussed later in this section). We further assume that the starting and ending times of the schools can be partitioned into distinct time periods (generally, the case for many districts) and a bus will service at most one route in each time period (eliminating double routing). Under these assumptions, the problem of minimizing the capital cost to the district reduces to the problem of minimizing the maximum number of buses required in any time period. In the time period needing the maximum number of buses, the scheduling component of our system goes to great length to ensure that every bus utilized in a time period prior to the peak time period is used in the peak time period (our scheduling algorithm moves sequentially in time from one time period to the next time period). If possible, therefore, there is no bus idle in a peak time period.

The routes for the buses in the off peak time periods need not be as efficient as the routes in the peak time period. As long as each bus utilized in an off peak time period can be assigned a route during the peak time period, the capital requirements for the system are unchanged. Therefore, routes in the off peak period can be shortened timewise and pick up less students than would ordinarily be prescribed by an "optimal seeking" routing procedure. This implies that more than the minimum number of buses would be utilized in these off peak periods, each route would service less students, but each student on the average would be on a bus a shorter amount of time than with an "optimal seeking" routing procedure.

In a suburban area, a bus generally becomes filled before the time to service the route is exhausted so that the routing problem becomes a student ridership clustering problem which is relatively easy to solve. Moreover, in a suburban area, a bus can service many routes in a day (6–14 for the Brentwood, New York, school district). In these situations, therefore, we have concluded that the scheduling problem is extremely important and the routing problem relatively unimportant when considering the effective utilization of the buses. In a rural area, on the other hand, a bus handles very few routes in a day. Each of these routes tends to be long and the bus's capacity may not become saturated on any of these routes. Hence, in scheduling school buses in a rural area, the routing problem becomes very important.

As noted previously, the key factor in reducing the cost of most school bus routing and scheduling situations is the selection of the proper starting and ending times of the schools. If the starting and ending times of the school district can be changed to reduce the number of students requiring transportation in the peak time period, then a reduction in the number of buses needed in the peak time periods can be realized (assuming a fixed bus capacity for each school). We generally converted the number of students requiring transportation at each school into a set of routes using the procedure described in the routing portion of this paper. We then adjusted the starting and ending times of the schools in order to minimize the maximum number of routes required in any time period.

As an example, if time period 1 requires 20 buses and time period 2 needs 40 buses, then a minimum of 40 buses will be used in the District. If, however, the starting and ending times of the schools are changed so that 32 buses are needed in time period 1 and 28 in time period 2, then the minimum number of buses required is 32—a savings of 8 buses. Of course, the scheduling component may find a set of schedules for the buses requiring more than 32 buses. However, if capital cost minimization is a prime objective, it is better to start off with a lower bound of 32 than a lower bound of 40. Professor Jacques Ferland and one of his students at the University of Montreal have had considerable success in implementing a procedure for optimally selecting the starting and ending times of the schools. Because of political consideration within the district we studied, we determined the school starting and ending times manually.

The school bus routing and scheduling system described in this paper has been implemented on both an NCR Century 200 computer with a 8K 32 bit word memory and an IBM 370/155 computer. The system has four basic components—an input component, a routing component, a scheduling component, and an output component as well as the school starting and ending time component described above. In the ensuing sections, these four components are described in detail. Suffice it to say at this point that this system has reduced student transportation costs and has avoided the problem of students overloading the capacity of the bus.

1. MINISTOPS AND THE INPUT COMPONENT

THE MOST time-consuming, tedious and in many cases costly aspect of the routing and scheduling of school buses is the generation and processing of the input data. The data have to be prepared so that both the set of bus stops for a school and the number of students to be picked up at each bus stop for the school can be found quickly, easily, and with a minimum amount of additional data preparation. To achieve this objective, we introduce the idea of the ministop.

A ministop is a location in the district which can be used as a bus stop. All the potential bus stops in the district become the set of ministops for the district and this set of ministops becomes the set of nodes in a directed network $G = [N, A]$. For each node $i \in N$, the branch $(i, j) \in A$ says that ministop j is directly accessible from ministop i along the road network without having to go through any other ministop. Each student in the district is assigned to the ministop closest to his home.

Although it may seem that the ministop idea requires a lot of extra data preparation, we found that, on an annual basis, the ministop idea is effective and requires less work than assigning each student to an actual bus stop each year. The key idea behind the use of ministops is that the ministop a student is assigned to remains unchanged regardless of whether the student attends an elementary school, a junior high school or high school as long as he does not move. Knowing the ministop that a student is assigned to, the actual bus stop that the student is to be assigned to can change from one year to the next and this bus stop assignment is found automatically in the input component.

Given the ministops, the entire district can be treated as one network and different bus stop locations for a school can be analyzed with almost no additional data preparation once the network is generated. Thus, the routing and scheduling of the buses can be carried out easily while taking into account such factors as the number of students attending the schools, the additions and deletions of schools from the district, the redistricting of the school boundaries, and the changes in bus stop locations. Designing a flexible system allows different bus stop alternatives for a school to be tried quickly and economically. Without the ministops, each bus stop configuration for a school would require both the coding of a new network and the determination of the number of students to be picked up at each stop. Both of these data bases are tedious to prepare manually and both data bases can contain data preparation errors.

Since the student lists for the school districts were already on a computer, the two major problems that occurred in setting up the data for this problem were the following:

1) The ministop network was coded incorrectly, and
2) A student was assigned to an incorrect ministop.

To check if the ministop network was in error, the length of the branch from node i to node j (node j coded adjacent to node i) was compared to the Euclidean and Manhattan distances from node i to node j. If the length of the branch did not fall between these two distances, an error message was printed out. This error check picked out most errors with this set of data. To determine if the ministop assigned to a student was in error, the distance from the student assigned ministop to the student assigned bus stop was computed. If this distance exceeded a tolerance value fixed in advance, then in error message was printed out. Again, this error check found most errors in the student data (and virtually all gross errors).

Given the ministop network G, *the matrix of shortest travel times* between all pairs of ministops (nodes in G) can be computed. This matrix of shortest travel times is then stored on tape or disk for use by the remainder of the system. To use this matrix for a particular school, the user only has to specify a list of the bus stops for the school (a subset of the ministops) and read in the matrix of shortest travel times between ministops. The user can then generate the shortest travel time between all pairs of bus stops for a school by reading the rows of the matrix of shortest travel times corresponding to these bus stops and extracting from this matrix the appropriate travel times between each pair of bus stops for that school.

As with other school bus routing and scheduling systems, bus stop locations are selected by the user of the system using whatever criterion that is set up for the district. The ministop concept allows the planner to easily change the bus stop locations based on the output of the routing and scheduling components with a minimum amount of effort. The use of the ministops or something similar to it is a necessity if an interactive school bus routing and scheduling system is to be developed.

The number of students to be picked up at each ministop for a particular school is known from the student data file. However, for a school, not all ministops having students assigned to them are actual bus stops for the school. The problem, therefore, is to assign all of the students for the school to an actual bus stop for that school. The criterion we used is that each student is to be serviced at the bus stop which is the closest to the ministop he is assigned to. Knowing the shortest travel time between all pairs of ministops, each ministop for a school having a student assigned to it can be assigned to its closest bus stop for the school by:

1) Reading in the row of the matrix of shortest travel time corresponding to the ministop, and
2) Picking out the bus stop for the school which is the closest to the ministop.

(In this section we are assuming that the distance between two ministops is proportional to the travel time between these two stops.)

We generated the matrix of shortest travel times one row at a time using the DIJKSTRA[6] shortest chain algorithm. Although this approach reduced the storage requirements considerably, it increased the required computer time to an unacceptable level. To overcome this problem, we decided to compute the shortest travel time from each ministop to its 200 nearest neighboring ministops except for the schools where we computed the shortest travel time to all nodes in G. We found that the 200th nearest neighbor to a ministop was always at least ¾ of a mile from the original ministop in both of the densely populated suburban school districts that we tested. The question then arose—could all the required computations in the routing and scheduling components be carried out with the matrix of shortest travel times not completely filled in.

For this approximation to be acceptable, the following two conditions had to be satisfied:

1) Every bus stop for every school would have another bus stop for the school included among its 200 nearest neighbors. If this were so, the bus stop network for each school could hopefully be connected and we would use either FLOYD'S[7] procedure, DANTZIG'S[5] algorithm, or the Dijkstra algorithm to fill in the *bus stop travel time matrix* for each school. No computational problems with either Floyd's or Dantzig's procedures would arise in this case since most schools had at most 75 bus stops or nodes in their network.
2) Every ministop for a school having at least one student assigned to that ministop would have a bus stop for that school included as one of the 200 nearest neighbors. If this were true, every student could be automatically assigned to the actual bus stop closest to the student's ministop.

We have tested this approximation quite extensively and have yet to find one case where either of the above conditions was violated.

2. THE ROUTING COMPONENT

THE ROUTING component forms a set of bus routes for each school. Each route is feasible with respect to the maximum allowable travel time for the students and maximum capacity for the buses. Input to the routing component for each school is the following:

1) A list of bus stops including the node representing the school,
2) The number of students assigned to each bus stop, and
3) The travel time between each pair of bus stops.

Most procedures for routing school buses are adaptations of either the "route first-cluster second" procedure for routing or the "cluster first-route second" technique (BELTRAMI AND BODIN[2]). The "route first" approach finds the shortest cycle through all the nodes in the network—the traveling salesman problem—and then breaks this chain into a set of feasible routes for the buses. The "cluster first" procedure forms a cluster of bus stops which are feasible with respect to the travel time and bus capacity constraints and then finds the best route (i.e., minimum travel time) for the stops in each cluster by reducing as much as possible the travel time to cover all the bus stops within each cluster. Each of these two procedures is heuristic in nature and gives an approximate solution to the problem. Since Newton and Thomas[9] indicated from their analyses that the "route first-cluster second" approach worked better for the types of networks encountered in school bus routing and scheduling in a fairly densely suburban area, our approach is a variant of the "route first-cluster second" procedure.

If the system described in this paper were to apply the system to a school district in a rural area, then we would use a variant of the "cluster first-route second" procedure. The rural area problem can be complicated by the fact that a bus route can pick up students for more than one school, drop off students at one school and continue to pick up students for a second school, etc. This mixed loading problem has been studied by the Santa Clara (California) County Center for Urban Analysis (personal communication with David Wytock of their staff), but at this writing no procedure to handle this problem has been incorporated within our system.

For each school in the District, we used the 3-opt branch exchange procedure of LIN[8] to find the traveling salesman tour through all the bus stops for a school. In this tour the bus stop representing the school is left out. The 3-opt procedure gives an approximate but usually good solution to the problem, is reasonably fast computationally, and is easy to implement. We felt that an exact solution to the traveling salesman problem was not necessary in this case since this tour was to be partitioned into routes for the buses.

The second step in this approach is to break the tour up into a set of routes which are feasible with respect to the capacity of the bus and travel time of the students. A route for a bus as it comes out of the partitioning looks like the following (see Figure 1):

$$\text{School } i, \text{ bus stop } j_1, \text{ bus stop } j_2, \cdots, \text{ bus stop } j_m, \text{ school } i$$

In practice, when picking up students, a bus comes from a school (or some other stop) in the previous period, covers bus stops $j_1, j_2, \cdots, j_m$ in some order and then goes to school i (Figure 2). When dropping off students, a bus comes from stop k (which could be a school), goes to

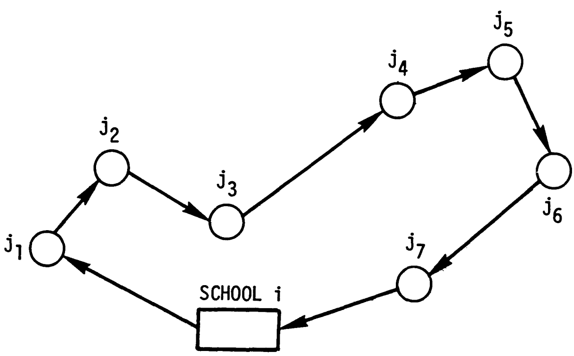

Fig. 1 (with $m = 7$).

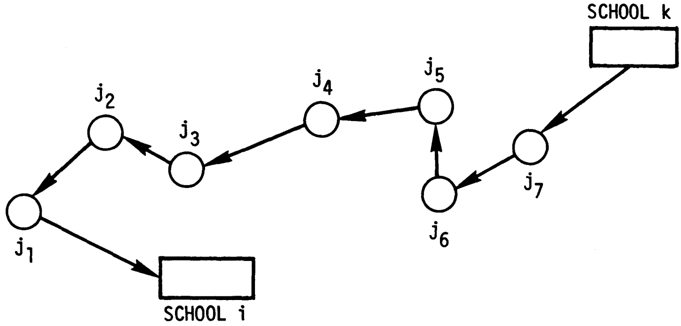

Fig. 2

school i and then covers bus stops $j_1, j_2, \cdots, j_m$ in that order (if the time to go from school i to stop j_1 is less than the time to go from school i to j_m). Otherwise the route is stop k, school i, stop j_m, stop $j_{m-1}, \cdots, j_1$. However, it is not known at this point which bus, if any, utilized in time period $t - 1$ will service this route in time period t, since the bus schedules are formed in the scheduling component.

As such, the routing component only constructs a partial route:

$$\text{School } i, \text{ bus stop } j_1, \text{ bus stop } j_2, \cdots, \text{ bus stop } j_m.$$

In this partial route, the time to go from school i to bus stop j_1 is assumed to be less than the time to go from school i to bus stop j_m. If not, the order of the stops would be reversed to get the shortest travel time on the partial route.

In partitioning the traveling salesman tour in this fashion, the upper bound on travel time allowed for a partial route would be smaller than the time assumed needed to complete a route since the time to go from the last stop in the previous time period to the first stop on this route has to be added in. Since our districts were fairly densely populated, this adjustment did not cause any serious problems because the capacity of the bus was normally exhausted before the vehicle ran out of time on its route.

To derive effective routes, the following two heuristics were utilized in the partitioning portion of the routing component.

1) *A look-ahead feature.* If the student load at bus stop j_m, when added to the number of students already assigned to a route, exceeded the capacity of the bus while the student load at bus stop j_{m+1} did not, then bus stop j_{m+1} would be assigned to the first route and the second route would begin with bus stop j_m (assuming the time constraints were not violated). The look-ahead was limited to two bus stops.
2) *A bus stop splitter.* If the sum of the student loads at bus stops j_1,

$j_2, \cdots, j_m$ exceeded the bus capacity, and the student load at j_m was at least 10% of the capacity of the bus, then stop j_m would be partitioned into two bus stops j_m^* and j_m^{**}, where the sum of the student loads at stops $j_1, j_2, \cdots, j_{m-1}, j_m^*$ would not oversaturate the capacity of the bus. The input and routing components for this school were then rerun to get a feasible, well designed set of routes.

The output from the routing component is a set of partial routes for each school. The number of partial routes for each school is used as input to the manual procedure described in the beginning of this paper for determining the starting and ending times for each of the schools. The starting and ending times of the schools and the partial routes for each school are then used as input to the scheduling component.

3. THE SCHEDULING COMPONENT

THE SCHEDULING component organizes the partial routes for each of the schools into daily schedules for the buses. The starting and ending times for each of the schools and the maximum allowable travel time for any of the students at the school has to be specified. (This time is a function of the school the student attends.) The student loads on each of the partial routes have already been satisfied in the routing component; as such, they no longer act as a constraint in the scheduling component.

The objective used in the scheduling component is to minimize the total travel time of all the buses in the district. This objective is closely analogous to the second purpose of a routing and scheduling system described in the introduction to this paper, gives a close to optimal solution for the other two purposes cited in the introduction, and is algorithmically very tractable.

If the starting and ending times of the schools fall into distinct time periods, the problem breaks down into a sequence of assignment problems. These assignment problems connect the buses used in the $(t-1)$st time period with the partial routes found in the routing component which are to be covered in the tth time period.

Since in the earliest time period $(t = 1)$ there are no buses assigned, the first n buses are assigned to cover the n routes to be served in the first time period. For each time period $(t > 1)$, the number of buses needed at each school in time period $t - 1$ and the number of partial routes required during time period t are known. The supply nodes of the assignment problem represent the buses servicing the various routes during time period $t - 1$ and the demand nodes represent the partial routes which have to be covered during time period t. The objective coefficients, c_{ij}, of the assgnment problem represent the total time including student pickup and dropoff time for the ith bus used in time period $t - 1$ to go to the first point on the jth partial route and cover all of its stops. (It is important to note that the ith bus used in the previous period is not necessarily bus number i, but a generic bus number.) If, in the solution of the assignment problem $x_{ij} = 1$, then the jth partial route for the tth period is joined to the daily schedule for the ith generic bus utilized in the $(t - 1)$st period. This procedure is repeated for all time periods.

The following changes are made in the above procedure in implementation:

1) Since the number of buses used in the previous time period will not necessarily equal the number of partial routes found for the present time period, a dummy supply node or a dummy demand node is defined to pick up the excess demand or supply as in the standard transportation problem. In this case, the special structure of the assignment problem is destroyed and a transportation problem (generally under 100 supply and demand nodes) is solved.
2) In the routing component, the bus stops are aggregated to form partial routes for the buses. In ordering the bus stops for this route in the routing component, the order of the stops was based on the minimum time to go from the school through the stops on the route and back to the school again. Since the bus comes from a location on a route in the previous time period, to obtain the minimum travel time to cover the proposed route, the order of the bus stops was changed where necessary using the 2-opt branch exchange procedure of Lin.[8]
3) If the time to complete the last stop on the previous route plus the time to cover the route for the present time period made the (i, j)th pairing infeasible, then c_{ij} was set equal to ∞.
4) If there exists a route j in time period t such that route j is serviced from the dummy supply node discussed in point 1 above, then no bus used in time period $t - 1$ is able to feasibly service partial route j in time period t. In this case, any bus not used in time period $t - 1$ but used in a time period previous to time period $t - 1$ can be assigned to cover the jth partial route. If no such bus is available, a new bus is created whose daily schedule begins with this jth partial route in time period t. We went to great lengths to avoid creating a new bus. This included rerouting the buses in time period t to determine if less efficient routes would still service the students in time period t and not require a new bus and redefining bus stops for the students in time period t. The rerouting simply required a repartitioning of the giant tour. (We saved on file the giant tour through all of the bus stops for each school in time period t.)
5) For a small computer (as with the NCR Century 200 computer), the transportation problem may be too large to fit into the computer.

In these cases, a smaller transportation problem was created by aggregating the supply nodes of the original transportation problem. In this case, it was assumed that all buses left the last stop on the previous route at the latest possible time (the worst case). Thus, certain pairings were declared infeasible (i.e., certain $c_{ij} = \infty$) in the aggregated transportation problem, when, in fact, if the original disaggregated transportation problem was solved, some of these pairings might have been possible. Because of the manner in which this aggregation was carried out, more buses were required than would have been deemed necessary without this aggregation. In the solution to the transportation problem, $x_{ij} = 1$ means that some bus which serviced school i in time period $t - 1$ would service partial route j in time period t. In examining the solution the earliest finishing bus for school i in time period $t - 1$ was scheduled to service the longest route in duration for which $x_{ij} = 1$, the next earliest finishing bus for the school was scheduled to service the second longest route in duration for which $x_{ij} = 1$, etc.

The following changes to the scheduling component were made to complete the implementation for the Brentwood School District:

1) The supply nodes of the transportation problem were defined to be the schools serviced during time period $t - 1$. The amount of supply at the ith supply node was equal to the number of buses that serviced school i. There was no aggregation of the demand nodes.
2) For certain schools, Brentwood required that the same bus drop off the students at night as delivered the students to the school in the morning. Many districts have this restraint, especially for the elementary and kindergarten students because it is felt that young students would get confused if they were delivered to the school on one bus and dropped off at their home on a second bus. In this case, the buses assigned to service the routes for picking up the students were determined by the scheduling component. Based on this solution, the bus assignment for dropping off the students on these buses was fixed before the scheduling algorithm was executed.
3) Field trips and activity runs for the students and lunch times for the drivers were also scheduled.
4) The upper bound on student travel time was assumed to be 30 minutes or 45 minutes depending upon the school being serviced. As such, the scheduling component was changed so that the first buses that were utilized in time period t were last in service in time period $t - 2$ if the route to be serviced was a 30-minute route or time period $t - 3$ if the route to be serviced was a 45-minute route. No bus utilized in time period $t - 1$ was assumed available for

service in time period t, although it is conceivable that early finishing buses in time period $t - 1$ could be used to service a route in time period t.

4. OUTPUT COMPONENT

THE PREVIOUS components created the following data bases which are used in the output component:

1) The bus stop number each student is assigned to.
2) The bus number each student is assigned for pickup and dropoff.
3) The estimated times of pickup and dropoff for each student.
4) The order of the bus stops for each route.
5) The bus number assigned to each route.
6) The routes and school numbers assigned to each bus.

Given this information, the following six outputs are generated:

1) The daily schedule for each bus broken down by time period, the number of routes each bus is to service during the day and the number of buses needed in each time period.
2) The daily schedule for each bus broken down by route and, within the route, by bus stop. At each bus stop, the estimated time of arrival at each stop and the student load at the stop are denoted.
3) The daily schedule for each bus broken down by route and, within each route, by bus stop. At each stop, the estimated time of arrival, the student load, and the names of the students assigned to this stop are given; this output can be used by the driver to ensure that only the proper riders are picked up.
4) This output is the same format as output 3 except that all routes for a school are aggregated together. This output is utilized by the administrators, principals, and transportation personnel of a district.
5) An identification card is punched out for each student giving the bus stop the student is assigned to, the estimated time of pickup and delivery and the numbers of the buses which will service the student.
6) For each student ordered alphabetically and sorted by school, the bus stop number he is assigned to, the ministop he is assigned to, the numbers for the buses which will pick the student up and drop him off and the estimated time of pickup and dropoff.

From our experience, these outputs encompass most of what a school district requires as far as hard copy from a routing and scheduling system. There may be other outputs that a district wants. More than likely, however, given the ministop concept, these outputs can be readily extracted from the existing data base, or the existing data base can be easily updated to satisfy the district's needs.

5. SOME COMPUTATIONAL EXPERIENCE

In 1972–1973, the Brentwood school district had about 23,000 students attending 2 high schools, 4 junior high schools, 14 elementary schools, and 5 parochial schools. Of these students, about 13,000 required bus transportation each day. Under their manual routing and scheduling for 1972–1973 system, 87 buses are leased for transporting the students—28 buses for 6 hours, 25 for 11 hours, and 34 for 12 hours.

Because this system was implemented on a NCR Century 200 computer with 8K words of storage, the following restrictions were added to the system:

1) In any network, there could be at the most 50 bus stops (including the school). Therefore, if there were more than 50 bus stops, the bus stops were partitioned into smaller networks containing no more than 50 stops each.
2) For all elementary, kindergarten, and parochial schools, it was required that the same bus take the students home that picked them up in the morning.

With these assumptions, the system described in this paper was applied to the Brentwood School District for the school year 1972–1973. The computer program determined that Brentwood needed 75 buses—34 for 12 hours, 13 for 11 hours, and 28 for 6 hours. The savings to the Brentwood School District using the computerized bus scheduling system was about $164,000 for the year over the existing manual system.

In 1974, the East Islip school district had about 9,600 students enrolled of which 4,800 were bused. In 1973, this district had about 4,200 students which were bused. In 1973, 16 buses were required and some bus routes had standees on them. In 1974, 17 buses were utilized and no bus route had any standees preassigned. If the school administration allowed a change in the starting time of one of the elementary schools, then the 4,800 students could have been served by 16 buses, but this change in starting time was infeasible from a practical standpoint. Although no savings in transportation costs were realized with our computerized system, a 15% increase in the number of transported students was serviced with a 6% increase in fleet size. It must be mentioned that the 1973 routes were also found by computer.

The system was originally designed with no interactive features inbedded within it. Each school was to have its own network and each student was assigned to an actual bus stop. It quickly became apparent that such a system did not give the degree of flexibility needed to derive good solutions. Furthermore, the preparation of the networks for each school became very time consuming and quite expensive. As such, the ministop idea evolved. As stated earlier, a data base such as the one described in Section 1 was needed. The other major problems which were encountered

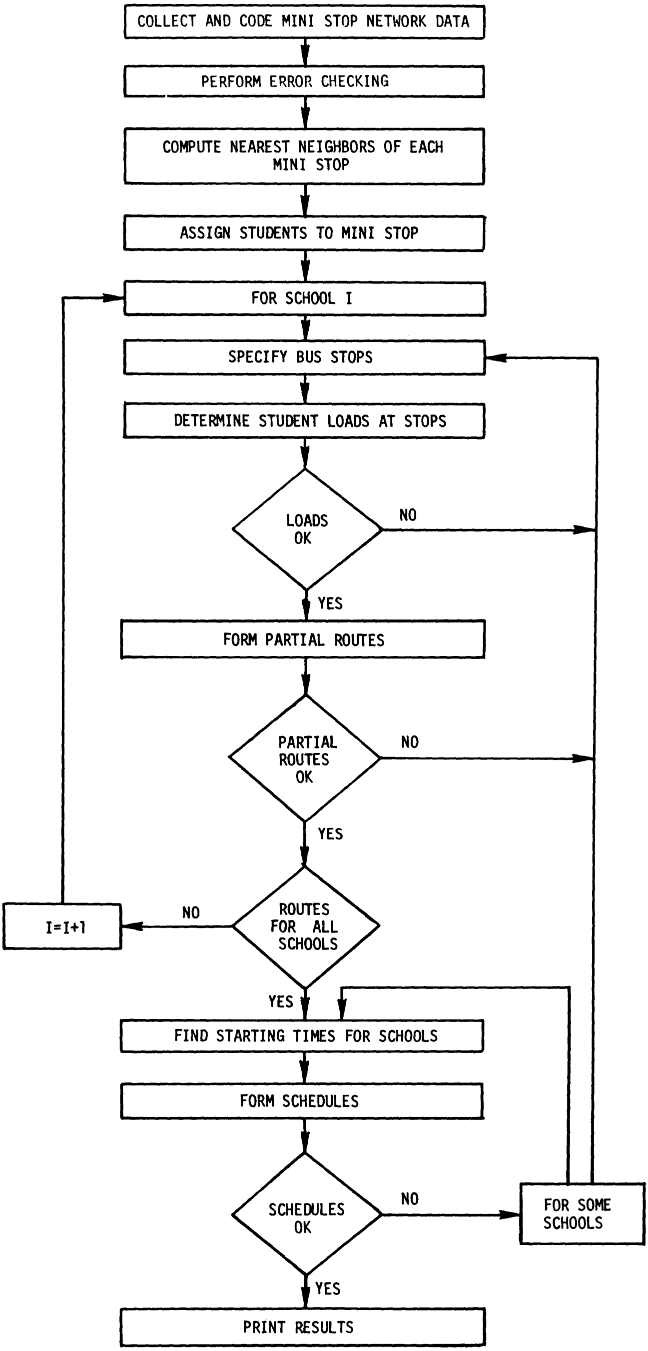

Fig. 3. Flow chart for school bus routing and scheduling procedure.

in developing this system were described at the end of Section 3. There problems primarily occurred in the scheduling component of the systems.

6. UTILIZATION OF THE SCHOOL BUS ROUTING AND SCHEDULING SYSTEM

IN FIGURE 3, a flow chart is presented which reflects our experience in utilizing the four components described in this paper. As illustrated in Figure 3, the manual intervention of the planner throughout the utilization of this system becomes extremely important. The expertize of the planner is queried at each step in the procedure in an attempt to derive good routes and schedules. The authors believe that school bus routing and scheduling can perhaps best be carried out in a man-machine interactive mode and that the system design in this paper can serve as a base for this system. The authors have found that the system presented in this paper is effective in the determination of a reasonable school bus fleet and that the ministop concept represents a significant step forward in the preparation of an efficient and effective data base for this class of problems.

ACKNOWLEDGMENTS

WE WISH to thank Gary Kochman for his programming efforts on this project; and Joe Rotolo, John Krupa, Madelaine Moger, George Klein and other staff members of the Brentwood School District for their assistance, advice and input in the design and implementation of this system.

REFERENCES

1. R. D. ANGEL, W. L. CANDLE, R. NOONAN AND A. WHINSTON, "Computer-Assisted School Bus Scheduling," *Mgmt. Sci.,* **18,** 279–288 (1972).
2. E. BELTRAMI AND L. BODIN, "Networks and Vehicle Routing for Municipal Waste Collection," *J. Networks* **4,** 65–94 (1974).
3. B. BENNETT AND D. GAZIS, "School Bus Routing by Computer," *Trans. Res.* **6,** 317–325 (1974).
4. G. CLARKE AND J. W. WRIGHT, "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points," *Opns. Res.* **12,** 568–581 (1964).
5. G. B. DANTZIG, "All Shortest Routes in a Graph," Technical Report No. 66-3, Operations Research House, Stanford University, 1966.
6. E. W. DIJKSTRA, "A Note on Two Problems in Connection with Graphs," *Numerische Mathematik* **1,** 269–271 (1959).
7. R. W. FLOYD, "Algorithm 97, Shortest Path," *Commun. AMC* **5,** 345 (1962).
8. S. LIN, "Computer Solutions of the Traveling Salesman Problem," *Bell System Tech. J.* **44,** 2245–2269 (1965).
9. R. NEWTON AND W. THOMAS, "Developing a Computer Program for Bus Routing: Final Report," State University of New York Research Foundation, Albany, New York, 1970.

10. C. ORLOFF, "Routing and Scheduling a Fleet of Vehicles to/from Central Facilities—The School Bus Problem," Ph.D. dissertation, Cornell University, 1972.
11. G. TRACZ AND M. NORMAN, "A Computerized System for School Bus Routing," *The Ontario Institute for Studies in Education*, Toronto, Canada, 1970.

(Received June 1976)