ıstering-scheduling process is repeated. To preserve ographical cohesiveness, one might consider different seed lection criteria for the next cluster. A simple rule that have used is to bisect the sector just considered, and, the half-sector corresponding to the sweep direction, let ıe unscheduled customer with the smallest angle formed by ıe ray from the depot through that customer and the isector be the seed for a new cluster (see diagram below). he intuition behind partitioning the unscheduled customers n the sector in two subsets is that, assuming a ounterclockwise sweep, the customers in the right ıalf-sector will be relatively far away from the new luster. Hence, hopefully a better schedule could be obtained by scheduling them at a later stage. The process is then repeated until all customers have been scheduled.

A different initial seed can be obtained by rotating all customers. The algorithm can then be restarted with this new reference customer and the best solution obtained over all possible rotations.

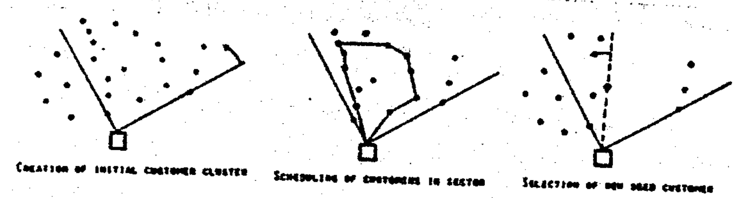

Diagram 3.6.

3.2.5 A giant-tour heuristic for (VRSPTW).

This heuristic is based on the route-first,cluster-second methodology in vehicle routing (see Bodin and Golden(1981)). This approach has been taken by Newton and Thomas(1974), and by Bodin and Berman(1979) in the context of school bus routing, by Stern and Dror(1979) for routing electric meter readers, by Levi et al.(1980) for the fleet size and mix vehicle routing problem, and by Cook and Russell(1978) for a case study in (VRSPTW). The time windows in this case study where manually accomodated for, which was possible due to the relatively small percentage of time constraints accounts. In the context of (VRSPTW) we call this a sequence first, schedule second method. The two phases of the algorithm consist of the construction of a tour over all the customers and the creation of time and pacity feasible subtours by partitioning the original iant tour" into contiguous segments.

Various approaches have been used for the creation of ıe initial (TSP) tour over all the customers. We use a ɔmposite heuristic (tour construction, 2-opt, applied in equence to the previous tour). Results obtained by Golden t al.(1980) found such procedures relatively fast omputationally and giving excellent results.

The gist of the procedure for creating feasible ehicle schedules from this original tour is in the modeling ɔf the problem of splitting a (TSP) tour into feasible subtours of contiguous customers, as a shortest path problem on a transformed network.

Let the (TSP) tour be $(0,N(1),\ldots,N(n),0)$. Then the distance between customers $N(m)$ and $N(p)$ in the new network, denoted by $D(m,p)$ is the total cost of having a vehicle service customers $N(m+1),\ldots,N(p)$, for $m < p$, in that order. Such distances are computed for pairs of customers who are time and capacity feasible, i.e.

$$D(m,p) = \begin{cases} c_{0,N(m+1)} + \sum_{i=m+1}^{p-1} c_{N(i),N(i+1)} + c_{N(p),0} \\ +\infty \ , \text{ if } \sum_{i=m+1}^{p} a_i > b_k, \text{ or } \exists\ i,\ m+1 \le i \le p, \text{ st } t_i > d_i \end{cases}$$

The problem is then to find the shortest path from the central depot to the last customer on the tour in this acyclic network. The number of trucks used will equal the

number of arcs in the solution to this problem.

Efficient data storage and manipulation can significantly reduce the computation time of network algorithms. We use a sorted forward star representation for the network(see Klingman et al.(1977)). Our choice of a shortest path algorithm is that of Pape(1974). A computational study by Klingman et al.(1977) has revealed that this label-correcting approach seems to perform better than other types of algorithms, including "label-setting" procedures.

3.2.6 Improvement heuristics.

The heuristics described in the previous sections could be used as initialization heuristics for k-optimal approaches to (VRSPTW). One could consider k-opt interchanges among and/or within vehicle schedules. An interchange would be performed only if it produces at most the same number of vehicles as before, and it leads to a reduction in overall schedule time and distance. The time and capacity feasibility checks could be implemented through the use of the transformed network described for the giant-tour heuristic.

We have implemented a 3-optimal approach for the

(VRSPTW) and found it computationally prohibitive. Since in problems with sufficiently tight capacity or time window constraints a lot of interchanges will be examined and found infeasible, one has to somehow limit the number of interchanges examined. This is a question for further research. One idea could be to examine only the customers in the vicinity of each customer, and/or the ones requiring service at about the same time as the customer under consideration.

3.3 Summary

In this chapter, we have introduced the (VRSPTW) formally. It was shown that our model has many attractive features that allow it to accomodate important problem complexities. Given the intrinsic difficulty of (VRSPTW), we have focused on approximate methods for its solution. A variety of heuristic approaches are described. Extensive computational experience with these methods will be reported in the next chapter.

CHAPTER IV

COMPUTATIONAL STUDY

4.1 Introduction

In this chapter we conduct an extensive computational study of the methods described previously. In order to evaluate the computational capabilities of the heuristics presented, given that no benchmark problem set is available in the literature for vehicle routing and scheduling problems with time windows, we have developed such a set of problems. This problem set includes routing and scheduling environments which differ in terms of the type of data being used to generate the problems, the percentage of time windows, their tightness and positioning, and the scheduling horizon.

The first part of the chapter describes the generation of the various problem sets. We then report our computational experience with the heuristics developed and analyze the effect of time window related factors on the behavior of these heuristics. Finally, we present a parametric analysis of the best heuristics.

.2 Development of the Problem Sets.

We have generated six sets of problems. The design of ;hese test problems highlights several factors that can affect the behavior of routing and scheduling heuristics. These factors include: geographical data, the number of customers serviced by a vehicle, and time window characteristics such as percentage of time constrainted customers, and tightness and positioning of the time windows.

As far as the geographical and demand data are concerned, they are randomly generated (denote the corresponding problem sets by R1 and R2), clustered (denote the corresponding problem sets by C1 and C2), and semi-clustered (denote the corresponding problem sets by RC1 and RC2). By a semi-clustered problem we mean a problem containing a mix of randomly generated data and clusters. Problem sets R1, C1, RC1 have a short scheduling horizon. The length of route-time constraint acts as a capacity constraint, which, together with the vehicle capacity constraints, allow only a few customers to be serviced by the same vehicle. In contrast, the sets R2, C2, RC2 have a long scheduling horizon; this, coupled with large vehicle capacities, permit many customers to be on the same vehicle.

Given a certain geographical and demand data, we have created the (VRSPTW) test problems by generating time windows of various widths for different percentages of customers. In terms of time windows' density, i.e. the percentage of customers with time windows, we have created problems with 25 %, 50 %, 75 %, and 100 % time windows.

We now present a method we have designed for the random generation of time window constraints. This method was used for the development of the problem sets R1, R2, RC1, RC2. First, we select the percentage of customers to receive time windows; then, the actual customers are randomly selected. Let these customers be $i_1, \ldots, i_{n_1}$, $n_1 = fn$, $0 < f \leq 1$. The time windows have a randomly generated center and width. We choose the center of the time window for customer i_j, $j=1,\ldots,n_1$, as being a randomly generated number in the interval $(t_{0,i_j}, h-t_{i_j,0}-s_{i_j})$, where h is the length of the scheduling horizon. To create the time window's width for i_j, we generate half the width as a normally distributed random number.

A somewhat different method was used for the clustered problems, C1 and C2. Here, we first run a 3-opt routine on each cluster to create routes and then produce schedules by selecting an orientation for each cluster. The time window constraints are generated by choosing the center as the arrival time at each customer; the width and density are rived as before. The rationale for using this method of onstruction is that it leads to the creation of structured roblems. Such problems serve a double purpose: Not only an one examine the relative performance of heuristics on lustered problems, but one can also test the robustness and ntelligence of the heuristics being examined since their ehavior can be compared in the absolute against a very good solution (believed to be optimal).

All the test problems are 100 customer problems. This problem size is not limiting for the methods presented, as much larger problems could be solved. Rather, we selected this size since it strikes a nice compromise between problem realism and the computer resources made available for this comprehensive computational study. Travel times between customers are taken to equal the corresponding distances. Furthermore a homogeneous fleet is assumed. We now present a detailed description of each set of test problems.

Problem Set R1.

The customer coordinates and demand data for this problem set are those of problem 8 from the standard set of routing test problems given in Christofides et al.(1979). Without time window constraints, a fleet of 10 vehicles, each of capacity 200 units, is necessary to satisfy the

demand requirements. Each customer requires 10 units of service time. A maximum allowable route time of 230 units is assumed. We have selected problem 8 among all the problems in the test set since it allows enough customers per truck to permit the nontrivial introduction of the time window constraints. Moreover, as mentioned above, its size and the length of route constraint make this problem realistic. This problem is depicted in diagram 4.1. We have generated 12 test problems in this set. Summary characteristics of these problems are given in table 4.1.

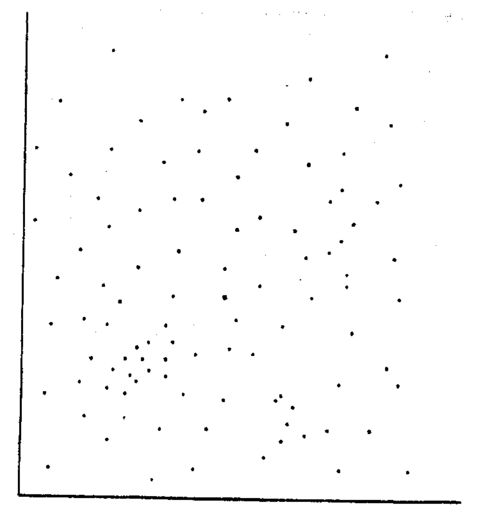

Diagram 4.1

Table 4.1 Problem Set R1.

Problem	% TW	Mean	St.D.	TWM	TWSD	ETM	ETSD	LTM	LTSD
1	100%	5.	0.	10.	0.	96.48	45.56	123.52	45.56
2	75%	5.	0.	10.	0.	100.04	45.50	119.96	45.50
3	50%	5.	0.	10.	0.	101.26	48.38	118.74	48.38
4	25%	5.	0.	10.	0.	104.60	47.89	115.40	47.89
5	100%	15.	0.	30.	0.	86.73	43.80	113.27	43.80
6	75%	15.	0.	30.	0.	90.25	43.31	109.75	43.31
7	50%	15.	0.	30.	0.	91.18	45.80	108.82	45.80
8	25%	15.	0.	30.	0.	93.96	45.33	106.04	45.33
9	100%	30.	7.50	58.89	8.93	73.01	38.79	98.10	37.85
10	50%	60.	30.	86.50	39.27	60.18	33.90	83.32	36.02
	50%	30.	15.						
11	25%	15.	3.75	93.10	54.73	61.35	40.49	75.55	42.19
	25%	30.	7.50						
	25%	60.	15.						
	25%	90.	22.50						
12	100%	60.	15.	117.64	17.45	47.20	22.67	65.16	22.93

Column 2 represents the percentage of time windows in the problem or the problem 's density. The values in the columns headed Mean and St.D. are the means and standard deviations of the normal distributions used to create the time windows' widths. Columns 5 and 6 show the actual mean and standard deviation of the time windows obtained. The average time window, when compared to the length of the scheduling horizon, provides a measure of the tightness of the time windows in the problem. The values in the columns 7 and 8 are the sample means and standard deviations of the intervals $(0,e_{i_j})$, $j=1,\dots,n_1$, while the columns headed LTM and LTSD give the sample means and standard deviations of the intervals (d_{i_j},h), $j=1,\dots,n_1$. They describe the positioning of the time windows within the scheduling horizon. Overall, the above measures indicate the relative difficulty of a given (VRSPTW).

Problem Set C1.

For this problem set, the geographical and demand data are those of problem 12 (Christofides et al.(1979)); (see diagram 4.2). Vehicle capacity is assumed to be 200 units. Each customer requires 90 units of service time. A restriction on the maximum allowable time per route of 1236 units is imposed. The best cluster by cluster solution we found for this problem class (believed to be optimal) requires 10 vehicles for a total schedule time of 9829 units, a distance of 829 units, and no waiting time. Table 4.2 gives the summary characteristics of this problem set.

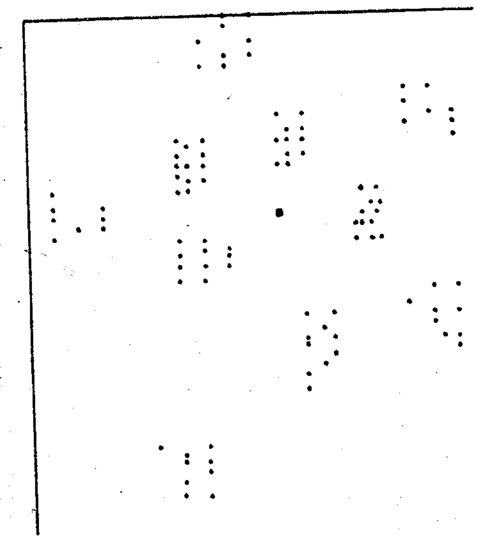

Diagram 4.2 Problem Set C1.

Table 4.2 Problem Set C1

Problem	% TW	Mean	St.D.	TWM	TWSD	ETM	ETSD	LTM	LTSD
1	100%	30.	7.50	60.76	10.53	426.80	282.20	748.44	282.05
2	75%	30.	7.50	61.27	9.89	401.11	265.88	773.63	265.04
3	50%	30.	7.50	59.90	9.76	430.74	256.65	745.36	256.06
4	25%	30.	7.50	60.64	9.37	410.32	243.80	765.04	244.13
5	100%	60	15.	121.61	20.98	399.16	277.29	715.23	276.89
6	25% 25% 25% 25%	30. 60. 90. 120.	15. 30. 45. 60.	156.15	91.86	383.74	272.73	696.11	280.06
7	100%	90.	0.	180.	0.	372.98	272.92	683.02	272.92
8	100%	120.	30.	243.28	41.96	347.22	261.43	645.50	261.36
9	100%	180.	0.	360.	0.	298.33	245.98	577.67	245.98

Problem Set RC1.

This problem set was created by using data from the two routing problems mentioned above. Geographically, one distinguishes five clusters (see diagram 4.3). These are the outside clusters of problem 12 containing 43 customers, consolidated with 7 additional customers. The other 50 customers are selected from those of problem 8. A limitation of 240 time units per route is assumed. Unloading time is 10 units per customer. The capacity of each vehicle is 200 units. A fleet of 10 vehicles is necessary to satisfy demand. This problem set is described in table 4.3.

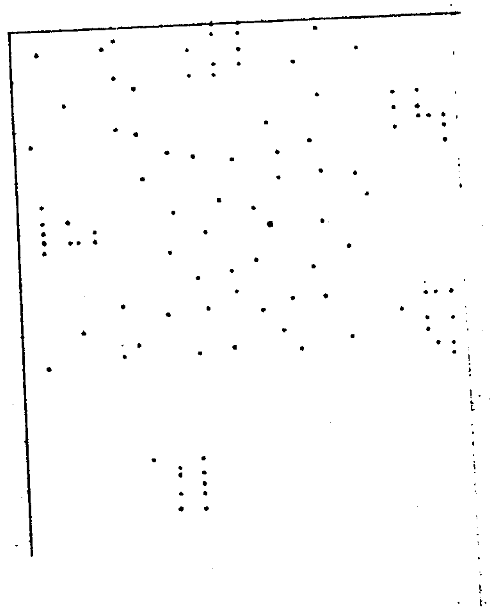

Diagram 4.3

Table 4.3 Problem Set RC1.

Problem	% TW	Mean	St.D.	TWM	TWSD	ETM	ETSD	LTM	LTSD
1	100%	15.	0.	30.	0.	91.82	42.23	118.18	42.23
2	75%	15.	0.	30.	0.	94.84	42.43	115.16	42.43
3	50%	15.	0.	30.	0.	95.76	46.01	114.24	46.01
4	25%	15.	0.	30.	0.	95.92	42.39	114.08	42.39
5	25% 25% 25% 25%	5. 15. 30. 60.	0. 3.75 7.50 0.	54.33	41.81	82.59	44.04	103.08	42.16
6	100%	30.	0.	60.	0.	77.54	37.52	102.46	37.52
7	50% 50%	60. 30.	15. 7.50	88.21	32.82	64.12	31.45	87.67	34.03
8	100%	60.	30.	112.33	30.80	55.09	24.70	72.58	25.02

Problem Set R2.

To permit the servicing of many customers by a
hicle, we have modified the data of problem set R1 by
icreasing the allowable route time to 1000 units and the
apacity of each vehicle to 1000 units. Two vehicles are
ufficient to satisfy the customers' requirements if time
indows are not present. Summary characteristics of these
roblems can be found in table 4.4.

Problem Set C2.

Using the data from problem 12, we have relocated some of the customers to create a structured problem with three large clusters of customers (see diagram 4.4). A fleet of three vehicles, each of capacity 700 units, is assumed. The maximum allowable time per route is 3390, and the customer service times are 90 units. For this set of problems, the best schedule we found (believed to be optimal) requires a total of 9591 units of time and no waiting time. The total distance traveled is 591 units. Table 4.5 summarizes this problem class.

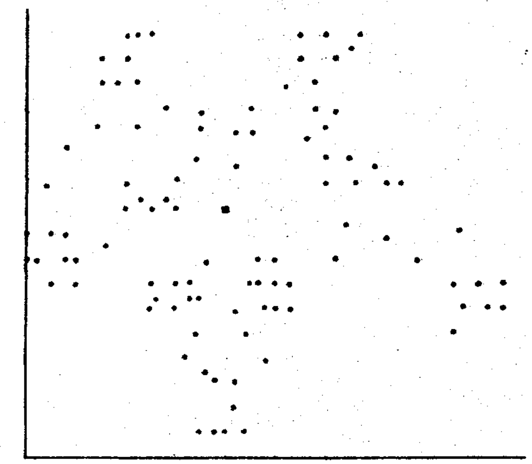

Diagram 4.4

Problem Set RC2.

The geographical, demand, and service time data for this set of problems are the same as for RC1. We have modified the length of route-time constraint and the vehicle capacities to allow a vehicle to visit many customers on a route. The route-time limitation is 960 units and vehicles are assumed to have a capacity of 1000 units. Without time windows, a fleet of two vehicles is suffient to satisfy demand. Problem characteristics are presented in table 4.6.

Table 4.4 Problem Set R2

Problem	% TW	Mean	St.D.	TWM	TWSD	ETM	ETSD	LTM	LTSD
1	100%	60.	30.	115.96	35.78	391.21	244.71	492.83	238.03
2	75%	60.	30.	115.23	35.56	407.91	244.51	476.87	241.82
3	50%	60.	30.	117.34	34.35	411.70	258.15	470.96	257.39
4	25%	60.	30.	111.80	31.13	431.76	243.53	456.44	251.01
5	100%	120.	0.	240.	0.	330.26	230.27	429.74	230.27
6	75%	120.	0.	240.	0.	346.07	229.53	413.93	229.53
7	50%	120.	0.	240.	0.	349.84	240.08	410.16	240.08
8	25%	120.	0.	240.	0.	362.56	232.45	397.44	232.45
9	50%	240.	120.	349.50	163.84	280.59	202.44	369.91	204.73
	50%	120.	60.						
10	25%	60.	15.	383.27	237.98	282.66	230.98	334.07	226.10
	25%	120.	30.						
	25%	240.	60.						
	25%	360.	90.						
11	100%	240.	60.	471.94	71.67	225.24	175.51	302.82	168.40

Page 74

Table 4.5 Problem Set C2.

Problem	% TW	Mean	St.D.	TWM	TWSD	ETM	ETSD	LTM	LTSD
1	100%	80.	0.	160.	0.	1470.21	921.06	1759.79	921.06
2	75%	80.	0.	160.	0.	1493.44	933.91	1736.56	933.91
3	50%	80.	0.	160.	0.	1472.94	957.90	1757.06	957.90
4	25%	80.	0.	160.	0.	1598.36	974.76	1631.64	974.76
5	100%	160.	0.	320.	0.	1393.28	911.54	1676.72	911.54
6	100%	240.	60.	486.64	83.99	1312.95	898.33	1590.41	901.23
7	25%	120.	30.						
	25%	240.	60.	612.32	302.72	1243.78	862.29	1533.90	939.21
	25%	360.	90.						
	25%	480.	120.						
8	100%	320.	0.	640.	0.	1242.69	879.90	1507.31	879.90

Problem	% TW	Mean	St.D.	TWM	TWSD	ETM	ETSD	LTM	LTSD
1	100%	60.	0.	120.	0.	371.18	229.80	468.82	229.80
2	75%	60.	0.	120.	0.	386.39	232.60	453.61	232.60
3	50%	60.	0.	120.	0.	391.70	247.85	448.30	247.85
4	25%.	60.	0.	120.	0.	405.88	233.59	434.12	233.59
5	25% 25% 25% 25%	30. 60. 120. 240.	0. 15. 30. 0.	223.06	162.66	327.96	237.33	408.98	222.23
6	100%	120.	0.	240.	0.	313.03	214.05	406.97	214.05
7	50% 50%	240. 120.	120. 60.	349.50	163.84	263.34	186.39	347.16	189.32
8	100%	240.	60.	471.93	71.67	208.61	156.10	279.46	149.83

4.3 Computational Results.

The heuristics described in the previous section have been programmed in Fortran, and computational tests were performed on the problem classes described above. We must remark that, in obtaining the computational results,we have not used any k-optimal improvement procedures. In the tables that follow the numbers to the left of each box are the total schedule time, total distance and total waiting time, respectively. The numbers to the right of each box are the number of routes and the CPU time in seconds on the DEC-10, which includes the time to read the data and the computation of intercustomer distances and times. The actual parameter values used to obtain the computational results are provided in the Appendix.

As discussed in chapter 3, we measure solution quality in terms of minimum schedule time and minimum distance , in that order, for the minimum number of trucks that will admit a feasible solution. The relative performances of the methods tested on the problem set R1 are shown in table 4.7.