The Impact of Metaheuristics on Solving the Vehicle Routing Problem: Algorithms, Problem Sets, and Computational Results

by

Bruce L. Golden
University of Maryland

Edward A. Wasil
American University

James P. Kelly
University of Colorado

I-Ming Chao
Chinese Military Academy (Taiwan)

Colloque du C.R.T. sul la gestion de flottes et la logistique
Université de Montréal
February 27, 1997

Focus of Paper

- Briefly review the history of the VRP (pre-1990)
- Talk about recent approaches (metaheuristics)
- Discuss our compilation of recent computational results from the literature
- Make some observations regarding this literature
- Solve some larger VRPs
- Discuss some future directions in vehicle routing research

Introduction to the Standard VRP

- Single depot
- Known customer locations and demands
- Homogeneous fleet of vehicles
- Capacity constraints
- Route-length constraints

- Each vehicle departs from and returns to the same depot
- Each customer is visited once
- Customer demand must be fully satisfied
- Minimize the total distance traveled by the fleet

Issues not Addressed in this Paper

- VRP variants

  Multiple depots
  Time windows
  Backhauls
  Period routing
  Mixed Fleet
  .
  .
  .

- Number of vehicles in the solution

A Brief History of Vehicle Routing

1950s	The VRP is formulated as an IP
	Small problems (10 to 20 customers) are "solved"
1960s	Early route-building heuristics (e.g., Clarke & Wright) are proposed
	2-opt and 3-opt are applied to the VRP (Christofides & Eilon)
	Problems with 30 to 100 customers are "solved"
1970s	A number of two-phase heuristics are proposed (e.g., Gillett & Miller)
	Computational efficiency becomes an issue (e.g., Golden, Magnanti & Nguyen)
	Larger problems (100 to 1,000 customers) are "solved"
	Some problems with 25 to 30 customers can be solved using optimal methods
1980s	Mathematical programming-based procedures are proposed (e.g., Fisher & Jaikumar)
	Interactive (man-machine) heuristics are developed (e.g., Cullen, Jarvis & Ratliff)
	Some problems with approximately 50 customers can be solved using optimal methods
1990s	Metaheuristics are applied to the VRP
	Some problems with 50 to 100 customers can be solved using optimal methods (Fisher)

Metaheuristics

- ❑ Simulated annealing (1983)
- ❑ Deterministic annealing (1990)
    - Threshold accepting
    - Record-to-record travel
    - Great deluge algorithm
- ❑ Genetic algorithms (mid-1980s)
- ❑ Neural networks (1985)
- ❑ Tabu search (1986)

Test Problems from the Literature

- There are 14 benchmark problems
- All problems are described in a 1979 article by Christofides, Mingozzi, and Toth
- They range in size from 50 to 199 nodes
- Seven of the problems have route-length constraints, as well as capacity constraints
- Ten have randomly distributed customers
- In four problems, customers appear in clusters
- In the last ten years, the focus has been on quality of solution

How Large are the VRPs that MS/OR Needs to Solve?

- In a 1986 *EJOR* article, we solve real-world problems with 368 and 650 deliveries per day
- In a 1987 *OR* article, we discuss the application of vehicle routing software to a Mid-Atlantic Coca-Cola problem with more than 1200 customers serviced daily
- The average number of customers visited daily in soft drink and beer distribution is approximately 600
- In sanitation (node) routing, the number of sites visited daily is often between 200 and 1000
- In general, companies with routing problems that involve fewer than 200 stops per day have difficulty justifying the purchase of routing software
- Most real-world applications of vehicle routing software involve at least 200 stops per day
- Bottom line: Heuristics remain the only practical option

Why is the Cost of Running Vehicle Routing Software so High?

- Data
  street database
  address match customers

- Customization and consulting
  special constraints
  interface with billing system

- Internal project management

- Total cost might range from 3 to 6 times the cost of software

Some Recent Approaches to the VRP

- ❑ Neural Networks
  - Ghaziri
- ❑ Genetic Algorithms
  - Van Breedam
- ❑ Simulated Annealing
  - Robusté et al.
  - Alfa
  - Osman
  - Van Breedam
- ❑ Tabu Search
  - Taillard
  - Semet and Taillard
  - Gendreau et al.
  - Rochat and Taillard
  - Rego and Roucairol
  - Xu and Kelly
  - Van Breedam

Observations from the Recent Literature

- Overall Performance
  - NN approaches are not competitive
  - SA has been found to outperform GA
  - TS has been found to outperform SA
  - TS seems to work best when infeasible intermediate solutions are allowed via penalty functions

- Computational Experimentation
  - Recent papers use real distances between customers
  - In general, solution quality has been emphasized at the expense of running time
  - TS procedures perform well, but involve a large number of parameters
  - Parameter values are adjusted in an ad hoc way
  - Some TS procedures have been parallelized

Observations from the Recent Literature (continued)

❑ Computational Reporting

- ◆ Two ways to report results
  - ✍ Some authors only report best results
  - ✍ Other authors report single pass results, as well as best results
- ◆ Four ways to report running times
  - ✍ No running time
  - ✍ Time to obtain the best solution
  - ✍ Time to obtain a solution within 1%, 5% of the best-known solution
  - ✍ Total computation time
- ◆ Some authors include the routes of their best solutions

Summary of Solutions to VRPs from CMT: Tabu Search Procedures

Pr	*n*	*G91*	*T92*	*T93*	*OTSF93*	*OTSB93*	*GSP94*	*GB94*	*RT95*	*RRS95*	*RRP95*	*XK96*
1	50	**524.61**	**524.61**	**524.61**	**524.61**	**524.61**	**524.61**	**524.61**		**524.61**	**524.61**	**524.61**
2	75	836.37	835.32	**835.26**	844	844	835.77	835.32		837.50	835.32	**835.26**
3	100	**826.14**	828.98	**826.14**	838	835	829.45	**826.14**		827.53	827.53	**826.14**
4	150	1034.90	1029.64	**1028.42**	1044.35	1052	1036.16	1031.07		1054.29	1044.35	1029.56
5	199	1329.29	1300.89	1298.79	1334.16	1354	1322.65	1311.35	**1291.45**	1338.49	1334.55	1298.58
6	50	**555.43**	**555.43**	**555.43**	**555.44**	**555.44**	**555.43**	**555.43**		**555.43**	**555.43**	
7	75	913.23	**909.67**	**909.68**	911	913	913.23	**909.68**		**909.68**	**909.68**	
8	100	**865.94**	**865.94**	**865.94**	878	866.75	**865.94**	**865.94**		868.29	866.75	
9	150	1189.79	1164.24	**1162.55**	1184	1188	1177.76	1162.89		1178.84	1164.12	
10	199	1421.88	1403.21	1397.94	1441	1422	1418.51	1404.75	**1395.85**	1420.84	1420.84	
11	120	1043.94	1073.05	**1042.11**	1043	**1042.11**	1073.47	**1042.11**		1043.54	**1042.11**	**1042.11**
12	100	822.85	**819.56**	**819.56**	819.59	819.59	**819.56**	**819.56**		**819.56**	**819.56**	**819.56**
13	120	1551.63	1550.15	**1541.14**	1545.98	1547	1573.81	1545.93		1550.17	1550.17	
14	100	**866.37**	**866.37**	**866.37**	**866.35**	**866.35**	**866.37**	**866.37**		866.53	**866.37**	

Bold indicates best-known solution. Problems 1 and 12 are optimal solutions.

Tabu Search Procedures

G91 Gendreau et al. June 1991 working paper
T92 Taillard's March 1992 working paper
T93 Taillard's 1993 Networks article
OTSF93 is Osman's tabu search algorithm, first-best-admissible strategy, 1993 Annals of OR
OTSB93 is Osman's tabu search algorithm , best-admissible strategy, 1993 Annals of OR
GSP94 Gendreau et al. 1994 Management Science article – single pass of Taburoute algorithm
GB94 Gendreau et al. 1994 Management Science article – best solution in multiple passes of Taburoute
RT95 Rochat and Taillard's 1995 Journal of Heuristics article
RRS95 Rego and Roucairol's sequential algorithm in Meta-Heuristics: Theory & Applications
RRP95 Rego and Roucairol's parallel algorithm in Meta-Heuristics: Theory & Applications
XK96 Xu and Kelly's 1996 Transportation Science article

Summary of Tabu Search Solutions to VRPs from CMT: Percent Above Best-known Solution

Pr	*n*	*G91*	*T92*	*T93*	*OTSF93*	*OTSB93*	*GSP94*	*GB94*	*RT95*	*RRS95*	*RRP95*	*XK96*
1	50	0	0	0	0	0	0	0		0	0	0
2	75	0.133	0.007	0	1.046	1.046	0.061	0.007		0.268	0.007	0
3	100	0	0.344	0	1.436	1.072	0.401	0		0.168	0.168	0
4	150	0.63	0.119	0	1.549	2.293	0.753	0.258		2.516	1.549	0.111
5	199	2.93	0.731	0.568	3.307	4.843	2.416	1.541	0	3.642	3.337	0.552
6	50	0	0	0	0	0	0	0		0	0	
7	75	0.39	0	0	0.145	0.365	0.39	0		0	0	
8	100	0	0	0	1.393	0.094	0	0		0.271	0.094	
9	150	2.343	0.145	0	1.845	2.189	1.308	0.029		1.401	0.135	
10	199	1.865	0.527	0.149	3.235	1.873	1.623	0.638	0	1.79	1.79	
11	120	0.176	2.969	0	0.085	0	3.009	0		0.137	0	0
12	100	0.401	0	0	0.004	0.004	0	0		0	0	0
13	120	0.681	0.585	0	0.314	0.38	2.119	0.311		0.586	0.586	
14	100	0	0	0	0	0	0	0		0.018	0	
Avg		0.682	0.388	0.051	1.026	1.011	0.863	0.199		0.771	0.548	0.095

Bold indicates best-known solution. Problems 1 and 12 are optimal solutions.

Tabu Search Procedures

G91 Gendreau et al. June 1991 working paper
T92 Taillard's March 1992 working paper
T93 Taillard's 1993 Networks article
OTSF93 is Osman's tabu search algorithm, first-best-admissible strategy, 1993 Annals of OR
OTSB93 is Osman's tabu search algorithm , best-admissible strategy, 1993 Annals of OR
GSP94 Gendreau et al. 1994 Management Science article -- single pass of Taburoute algorithm
GB94 Gendreau et al. 1994 Management Science article -- best solution in multiple passes of Taburoute
RT95 Rochat and Taillard's 1995 Journal of Heuristics article
RRS95 Rego and Roucairol's sequential algorithm in Meta-Heuristics: Theory & Applications
RRP95 Rego and Roucairol's parallel algorithm in Meta-Heuristics: Theory & Applications
XK96 Xu and Kelly's 1996 Transportation Science article

Procedure	*Average*	*High*
T93	0.051	0.568
XK96	0.095	0.552
GB94	0.199	1.541
T92	0.388	2.969
RRP95	0.548	3.337
G91	0.682	2.343
RRS95	0.771	3.642
GSP94	0.863	3.009
OTSB93	1.011	4.843
OTSF93	1.026	3.307

Summary of Solutions to VRPs from CMT: Simulated Annealing and Neural Networks

Pr	*n*	*OSA93*	*HGH95*	*HGS95*
1	50	528	545.1	539.2
2	75	838.62		
3	100	829.18	911.2	893.4
4	150	1058	1133.2	1084.7
5	199	1376	1421.5	1401.4
6	50	**555.44**	567.3	561.3
7	75	**909.68**		
8	100	866.75	902.4	894.3
9	150	1164.12	1304.5	1264
10	199	1417.85	1604.3	1580.4
11	120	1176	1113.4	1102.4
12	100	826	826.6	825.1
13	120	1545.98	1645.8	1608.5
14	100	890	881.5	879.8

Bold indicates best-known solution. Problems 1 and 12 are optimal solutions.

Simulated Annealing Procedures
OSA93 Osman's simulated annealing procedure, 1993 Annals of OR

Neural Network Procedures
HGH95 Ghaziri's hierarchical deformable net algorithm in Meta-heuristics: Theory & Applications
HGS95 Ghaziri's supervised hierarchical deformable net algorithm in Meta-heuristics: Theory & Applications

Summary of Solutions to VRPs from CMT: Simulated Annealing and Neural Networks

Pr	*n*	*OSA93*	*HGH95*	*HGS95*
1	50	0.646	3.906	2.781
2	75	0.402		
3	100	0.368	10.296	8.141
4	150	2.876	10.188	5.472
5	199	6.547	10.07	8.514
6	50	0	2.135	1.055
7	75	0		
8	100	0.094	4.211	3.275
9	150	0.135	12.21	8.727
10	199	1.576	14.933	13.221
11	120	12.848	6.841	5.785
12	100	0.786	0.859	0.676
13	120	0.314	6.791	4.371
14	100	2.727	1.746	1.55
Avg		2.094	7.016	5.297

Procedure	*Average*	*High*
OSA93	2.094	12.848
HGS95	5.297	13.221
HGH95	7.016	14.933

Bold indicates best-known solution. Problems 1 and 12 are optimal solutions.

Simulated Annealing Procedures

OSA93 Osman's simulated annealing procedure, 1993 Annals of OR

Neural Network Procedures

HGH95 Ghaziri's hierarchical deformable net algorithm in Meta-heuristics: Theory & Applications

HGS95 Ghaziri's supervised hierarchical deformable net algorithm in Meta-heuristics: Theory & Applications

CMT Problems -- Running Times

"We do not give CPU times ... because it is very difficult to give them as our algorithm does not find the best-known solution of every problem at each run."

Running Times (minutes)

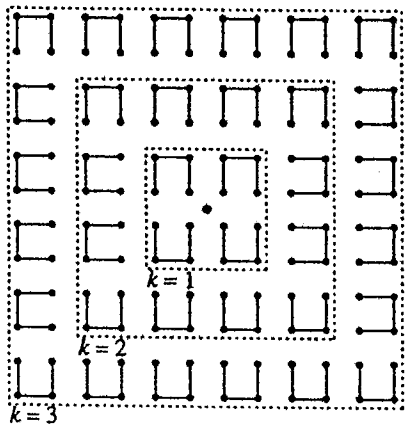
Pr	*n*	*OSA93*	*OTSF93*	*OTSB93*	*GSP94*	*XK96*
1	50	2.79	1.90	1.12	6	29.92
2	75	107.24	2.98	1.18	53.8	48.80
3	100	155.56	25.72	11.25	18.4	71.93
4	150	83.54	59.33	51.25	58.8	149.90
5	199	38.64	54.10	32.88	90.9	272.52
6	50	56.84	2.88	2.34	13.5	
7	75	10.44	17.62	3.38	54.6	
8	100	15.95	49.97	20.00	25.6	
9	150	1405.02	79.26	40.73	71	
10	199	95.13	76.02	55.17	99.8	
11	120	5.26	24.09	23.31	22.2	91.23
12	100	10.53	14.87	6.79	16	56.61
13	120	127.04	47.23	22.38	59.2	
14	100	5.09	19.59	92.98	65.7	
Avg		151.36	33.97	26.05	46.82	102.99

XK96 Xu and Kelly, total time on a DEC ALPHA Workstation (DEC OSF/1 v3.0)
OSA93, OTSF93, OTSB93 Osman SA and TS procedure, CPU to end, Vax 8600 computer
GSP94 Gendreau et al., total computation time, Silicon Graphics workstation, 36 MHz, 5.7Mflops

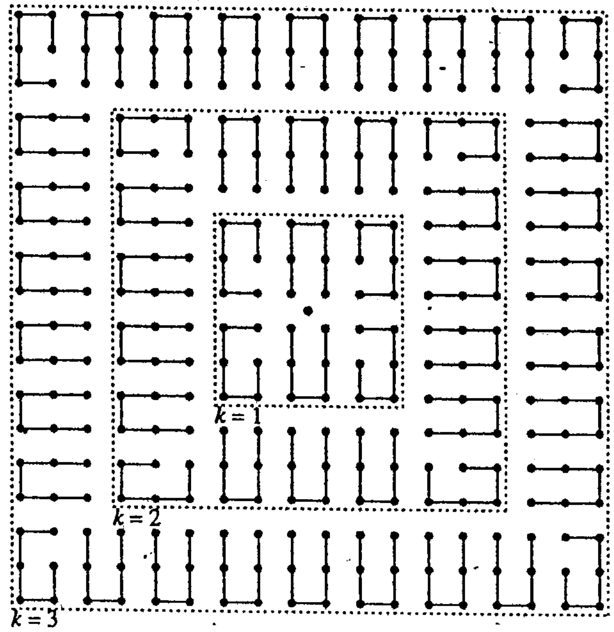
Some Larger Vehicle Routing Problems

- Authors construct families of problems
- These problems are highly symmetric
- One can estimate good solutions by inspection

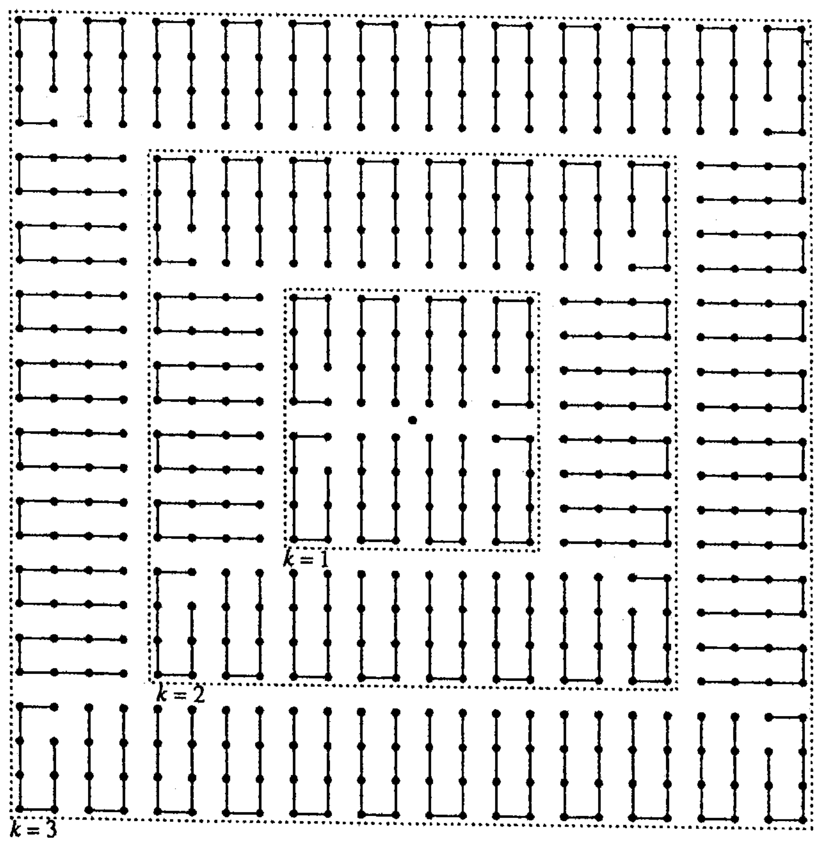
Grid Problems of Taillard (144, 324, and 576 cities) with Pseudo-optimum Solutions



$q = 2$



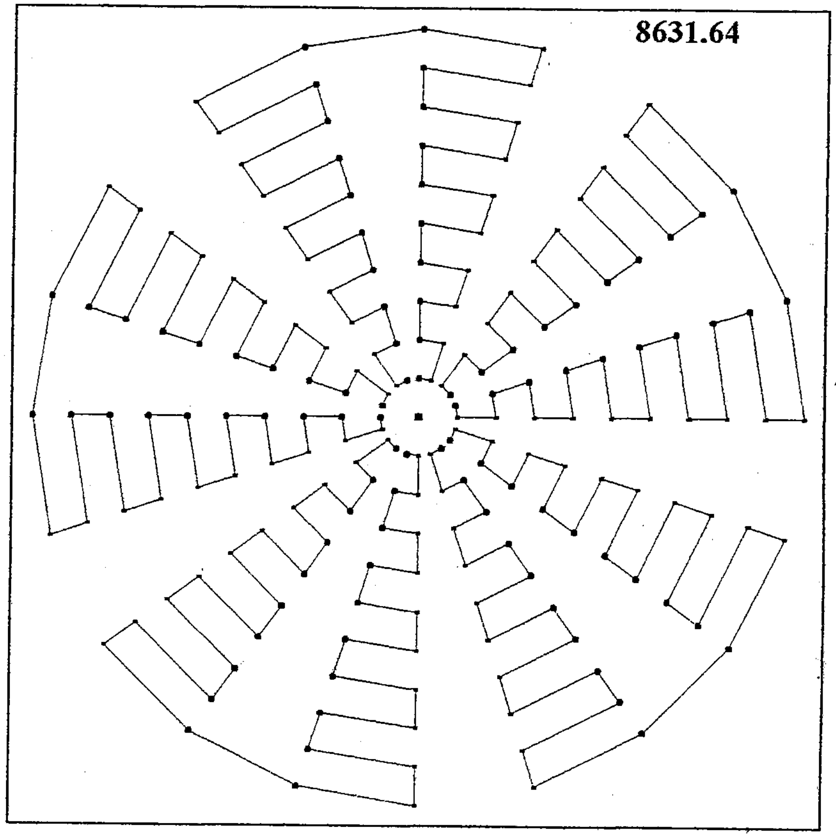
$q = 3$



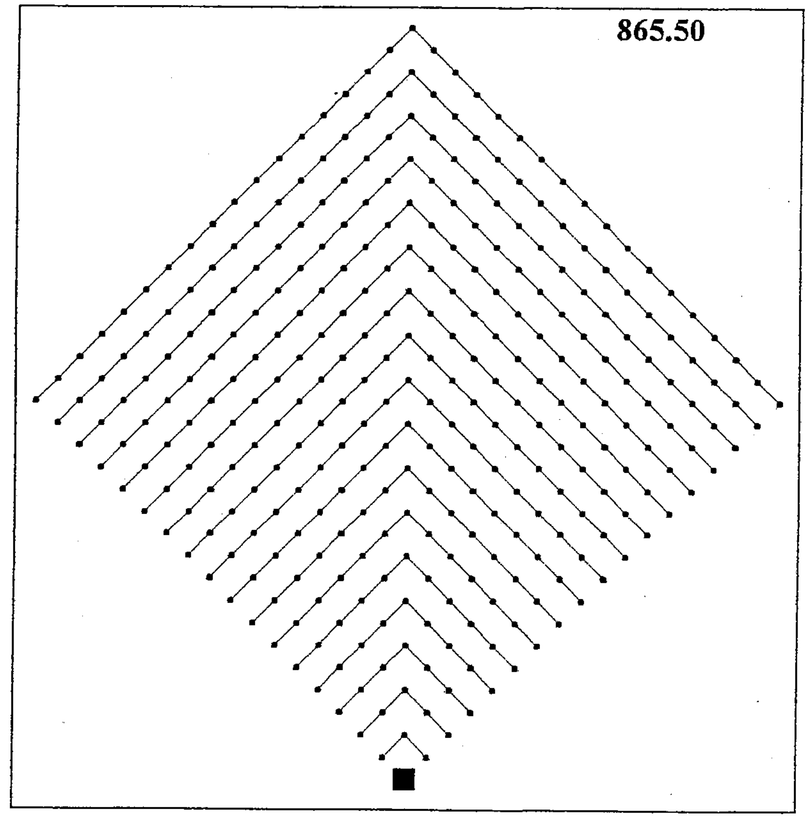
$q = 4$

Examples of Large-scale Vehicle Routing Problems

Estimate Solution to LSVRP No. 5

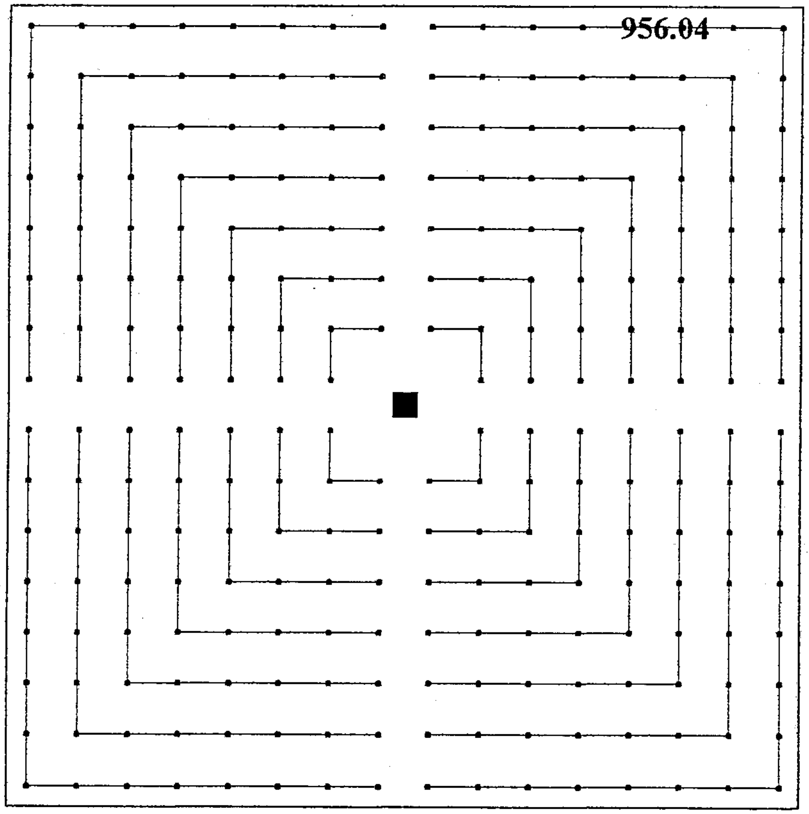


Estimate Solution to LSVRP No. 10

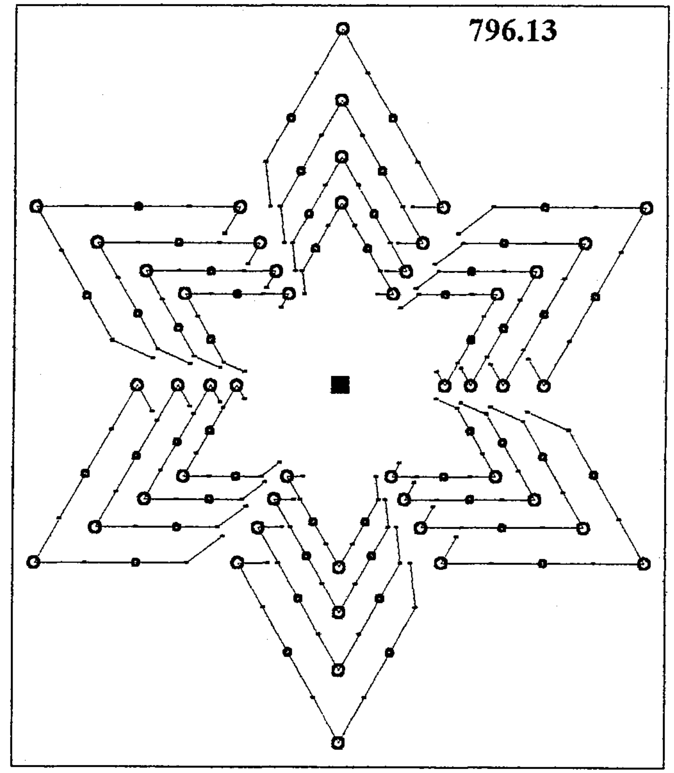


Examples of Large-scale Vehicle Routing Problems (continued)

Estimate Solution to LSVRP No. 13



Estimate Solution to LSVRP No. 17



Projected Running Times for Large-scale VRPs

Pr	n	Actual Total Time (minutes) GSP94*	XK96**
1	50	6.0	29.92
2	75	53.8	48.80
3	100	18.4	71.93
4	150	58.8	149.90
5	199	90.9	272.52
11	120	22.2	91.23
12	100	16.0	56.61

* Silicon Graphics Workstation, 36 MHz, 5.7 Mflops
** DEC ALPHA Workstation (DEC OSF/1 v 3.0)
roughly five times as fast as an SG Workstation

GSP94

Estimated Total Time = $0.001784\ n^{2.025}$ $R^2 = 0.95$ (with second point removed)

XK96

Estimated Total Time = $0.049688\ n^{1.591}$ $R^2 = 0.95$

n	Projected Total Time (mins) GSP94	XK96†
250	128.0	324.6
300	185.2	433.9
350	253.0	554.5
400	331.6	685.7
450	420.9	827.0
500	521.0	977.9

† Actual times were 480 to 2880 mins on large-scale VRPs

Large Scale VRPs

LSVRPs

Pr	n	Capacity	*Max. Route Length*	*Estimated Solution*	*C & W*	*XK*	*Running Time (mins)**	*RTR*	*Running Time (mins)***
1	240	550	650	5859.62	5974.23	**5646.46**	802.87	5834.60	3.68
2	320	700	900	**8566.04**	9257.23	8570.28	898.53	9002.26	22.66
3	400	900	1200	**11649.06**	12282.90	11880.37	1749.27	11879.95	40.04
4	480	1000	1600	15108.68	16054.22	15250.78	2432.42	**14639.32**	122.61
5	200	900	1800	8631.64	7230.47	7361.29	591.40	**6702.73**	11.24
6	280	900	1500	9843.01	9372.33	9088.66	913.70	**9016.93**	18.79
7	360	900	1300	**11047.69**	11901.54	11411.85	1062.73	11213.31	22.55
8	440	900	1200	**12250.06**	12982.65	12825.00	1586.20	12514.20	111.37
9	255	1000	∞	678.82	663.57	589.10		**587.09**	23.01
10	323	1000	∞	865.50	838.92	**746.56**		749.15	31.49
11	399	1000	∞	1074.80	1052.12	**932.68**		934.33	69.19
12	483	1000	∞	1306.73	1270.99	1140.72		**1137.18**	101.09
13	252	1000	∞	956.04	952.74	881.07		**881.04**	6.01
14	320	1000	∞	1220.27	1221.69	1118.09		**1103.69**	21.83
15	396	1000	∞	1516.48	1512.66	1377.79		**1364.23**	32.62
16	480	1000	∞	1844.67	1774.68	**1656.66**		1657.93	47.55
17	240	200	∞	796.13	771.70	**666.84**		720.44	5.69
18	300	200	∞	1133.25	1069.29	**973.60**		1029.21	8.15
19	360	200	∞	1554.64	1467.16	**1338.78**		1403.05	12.42
20	420	200	∞	2081.38	1963.38	**1836.71**		1875.17	31.05

Bold indicates best-known solution.
* Run on a DEC ALPHA workstation
** Run on a 100 MHz Pentium-based PC

Solution Procedures
C&W Clarke and Wright algorithm
RTR Record-to-record algorithm of Chao
XK Tabu search algorithm of Xu and Kelly

Parameters

"Clearly, the algorithm employs dynamic penalty functions that vary with different scenarios based on the feasibility and solution quality. Therefore many parameters are required."

Tabu Search: Xu and Kelly (XK96)

Nine Penalty Parameters

$\rho_1\ \rho_2\ \rho_3\ r_1\ r_2\ \rho_4\ \bar{\rho}_4\ \dot{\rho}_4\ \rho_4'$

Nineteen Time-related and Control Parameters

$n_1\ n_2\ n_3\ n_4\ n_5\ n_6\ n_7\ n_8\ n_9\ n_{10}\ s_1\ L_1\ H_1$

$\tau_1\ \tau_2\ \tau_3\ \varepsilon\ \lambda_1\ \lambda_2$

Other Parameters

Maximum number of iterations

Iteration to begin restart/recovery strategy

Number of stored elite solutions

Number of network moves

Tabu Search: Gendreau et al. (GSP94)

Eleven Parameters

$W\ q\ p_1\ p_2\ g\ h\ \lambda\ \theta_{min}\ \theta_{max}\ \alpha\ \beta$

Other Parameters

n_{max}

Simulated Annealing: Van Breedam (VBSASR95)

Six Parameters

$T_{begin}\ R_c\ N_{tot}\ N_{temp}$ Acceptance Ratio T_{end}

Other Parameters

DL SL

Fine-tuning an Algorithm

Tabu Search: Xu, Chiu, and Glover (1996)

❑ Developed a tabu search algorithm for solving the Steiner Tree-Star problem

❑ Fine-tuned five factors ordered by importance

1 Tabu tenures 2 Base probability for move selection 3 Fine-tuned probability for move selection
4 Recovery strategy 5 Frequency of activating swap moves

❑ Start with first factor: Tabu tenures

18 Candidate Settings ----> Execute on 19 problems ----> Use nonparametric tests to eliminate inferior settings
(18 runs) (Friedman test, Wilcoxon signed-rank test)

----> Two runs survive for future consideration

❑ Go to second factor: Base probability for move selection

6 Candidate Settings ----> Execute on 19 problems ----> Use nonparametric tests to eliminate inferior settings
(6 runs for each survivor) (Friedman test, Wilcoxon signed-rank test)

----> One run survives for future consideration

•
•
•

❑ Go to fifth factor: Frequency of activating swap moves

❑ Authors examined only 44 runs out of 9,270 possible runs and concluded

" ... *Run* 33, the fine-tuned algorithm, significantly improves upon the base algorithm ... it yields improved solutions in fourteen cases ..."

Fine-tuning a Neural Network Code

- Constructed a neural network (NN) that models the first-day returns of initial public offerings (IPOs)

    441 technology IPOs from 1989 to 1994

        196 firms in *training set*, 98 in *testing set*, 147 in *validating set*

    16 predictor variables

- Trained the NN model using our computer code (MNC)

    5 parameters required settings

    - number of hidden nodes
    - output node sigmoid slope
    - learning rate
    - momentum
    - network pruning

- Found the "best" settings using a *fractional factorial experiment* and *regression* with the data partitioned into three sets

MNC MODELS

Initial parameter settings

Architecture	16 input nodes, 8 hidden nodes, 1 output node
Hidden layer sigmoid slopes	0.15 node 1, 0.20 node 2, 0.25 node 3, 0.30 node 4, 0.35 node 5, 0.40 node 6, 0.45 node 7, 0.50 node 8 begin; incremented by 0.2 after 100 iterations
Output node sigmoid slope	0.5
Learning rate	0.2
Momentum	0.4
Error threshold	0
Stopping criteria	200 iterations
Procedure	Increase sigmoid slope after 100 iterations.

Parameter changes for fractional factorial experiment

Parameter	+	-
Number of hidden nodes	10	6
Output node sigmoid slope	0.6	0.4
Learning rate	0.3	0.1
Momentum	0.4	0.2
Pruning	On	Off

MNC MODEL: TECHNOLOGY OFFERINGS

Hidden nodes	Sigmoid slope	Learning rate	Momentum	Pruning	MAE
-	-	-	-	+	0.0928
+	-	-	-	-	0.0921
-	+	-	-	-	0.0994
+	+	-	-	+	0.0910
-	-	+	-	-	0.1018
+	-	+	-	+	0.0927
-	+	+	-	+	0.0930
+	+	+	-	-	0.0910
-	-	-	+	-	0.0991
+	-	-	+	+	**0.0904**
-	+	-	+	+	0.0942
+	+	-	+	-	0.0944
-	-	+	+	+	0.0958
+	-	+	+	-	0.0912
-	+	+	+	-	0.0938
+	+	+	+	+	0.0907

R^2	0.677		
Adjusted R^2	0.516		
Observations	16		
	Coefficient	t statistic	p-value
Intercept	0.095	115.9111	0.000
Hidden nodes	-0.003	-3.810	0.034
Sigmoid slope	-0.001	-0.879	0.400
Learning rate	-0.000	-0.360	0.729
Momentum	-0.000	-0.439	0.670
Pruning	-0.003	-2.324	0.043

Number of hidden nodes	MAE
10	0.0904
11	0.0899
12	0.0898
13	0.0898

- Number of hidden nodes and pruning found to be statistically significant.
- Increasing hidden nodes to 12 produces best testing set MAE value

Architecture	16 input nodes
	12 hidden nodes
	1 output node
Output node sigmoid slope	0.4
Learning rate	0.1
Momentum	0.4
Pruning	On

- Validating data set MAE = 0.0903

A Quick Look Back

- There are 14 benchmark problems in the VRP literature
- This has been the case for about 20 years
- Meanwhile, computers have become much faster
- Heuristics have become more complex
- The number of parameters has increased
- Some current heuristics have *more* parameters than the number of benchmark problems

  notion of elegance in heuristic design

  notion of overfitting

Future Directions in Vehicle Routing

- The goal should be to design heuristics that are
  - high-quality -- near-optimal
  - lean/parsimonious -- few parameters
  - computationally efficient
  - robust
  - simple

- Benchmark problems
  - with the new, larger VRPs, there are now 34 benchmark problems
  - the largest of these has nearly 500 customers

- Parameters
  - authors should explicitly list and define parameters
  - without this, reproducibility is impossible
  - parameter values should be determined in a systematic way (e.g., via experimental design)

Future Directions in Vehicle Routing (continued)

❑ Parameters

- a single set of parameter values should be obtained
- alternatively, the heuristic might automatically search over a set of internal (hidden) parameters to find the best solution for each problem

❑ Robustness

- borrowing from the neural network literature, a subset of 20 or so benchmark problems can be used for "training"
- the resulting heuristic can then be applied to the 14 or so remaining benchmark problems
- this address the issue of robustness

❑ Reporting

- total computation time should be reported for each problem
- solutions (routes) should be reported