Optimizing Transportation by Polyhedra

A. Schrijver

1. Introduction

Historically, there is a strong interaction between combinatorial optimization and polyhedral methods. The development of the basic polyhedral tool *linear programming* in fact was motivated to a large extent by application to transportation problems.

Linear programming studies minimizing (or maximizing) a linear function $c^T x$ over a given polyhedron P. A *polyhedron* is the solution set of a system $Ax \leq b$ of linear inequalities. Generally, an optimum solution is attained at a vertex of P.

The basic method, the *simplex method*, was already described by Fourier: make a trip along the vertices and edges of the polyhedron, throughout decreasing $c^T x$, until an optimum vertex is reached. What makes the simplex method interesting is that it applies if the polyhedron is given by a set of defining inequalities. Listing all vertices is not necessary; it would also be impracticable for most problems because of the huge number of vertices (while there is a relatively small number of defining inequalities).

The interaction with combinatorial applications originates from the founding fathers of linear programming, L.V. Kantorovich, F.L. Hitchcock, Tj.C. Koopmans, and G.B. Dantzig, who independently designed polyhedral tools during the years 1939-1949. (Dantzig introduced the terms linear programming and simplex method.)

2. Origins of transportation and linear programming problems

In 1939, Kantorovich introduced the idea of linear programming. He gave a wealth of practical applications, which he motivated by the Soviet plan economy.

One of the applications mentioned by him is a problem now known as the *multicommodity flow problem*:

Let there be several points A, B, C, D, E, which are connected to one another by a railroad network (see figure 1). It is possible to make the shipments from B to D by the shortest route BED, but it is also possible to use other routes as well: namely BCD, BAD. Let there also be given a schedule of freight shipments; that is, it is necessary to ship from A to B a certain number of carloads, from D to C a certain number, and so on. The problem consists of the following. There is given a maximum capacity for each route under the given conditions (it can of course change under new methods of operation in transportation). It is necessary to distribute the freight flows among the different routes in such a way as to complete the necessary shipments with a minimum expenditure of fuel, under the condition of minimizing the empty runs of freight cars and taking account of the maximum capacities of the routes.

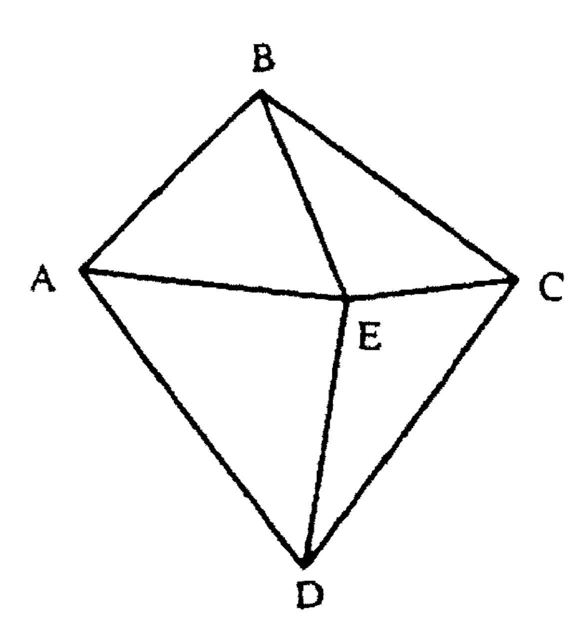


Figure 1.

Independently, at about the same time, Hitchcock at the Massachussetts Institute of Technology introduced and studied the *transportation problem*: given a nonnegative $m \times n$ matrix $C = (c_{i,j})$ and vectors $s \in \mathbb{R}^m_+$ and $d \in \mathbb{R}^n_+$, find a nonnegative $m \times n$ matrix $X = (x_{i,j})$ satisfying

$$\sum_{j=1}^{n} x_{i,j} = s_i \text{ for } i = 1, \ldots, m, \qquad (2.1)$$
$$\sum_{i=1}^{m} x_{i,j} = d_j \text{ for } j = 1, \ldots, n,$$

and minimizing $\sum_{i,j} c_{i,j} x_{i,j}$.

The interpretation is that there are m factories and n customers. Factory i can produce a quantity of s_i of a certain product, while customer j needs a quantity of d_j of this product. The cost of transporting one unit of the product from factory i to customer j is equal to $c_{i,j}$. Then $x_{i,j}$ gives the amount transported from factory i to customer j, at minimum total cost.

Thus the transportation problem is a problem of minimizing a linear function over the polyhedron determined by (2.1) (and by the nonnegativity constraints)—the *transportation polyhedron*. It is a polyhedron in nm-space, and the optimum matrix X can be found by making a trip along the vertices of this polyhedron. Hitchcock showed a simple procedure for doing this.

Independently of Kantorovich and Hitchcock, also Koopmans studied transportation problems. During the Second World War, Koopmans was as a statistician on the staff of the Combined Shipping Adjustment Board, a British-American agency dealing with merchant shipping problems during the Second World War. Influenced by his teacher J. Tinbergen, he was interested in ship freights and capacities. At the Board he studied the assignment of ships to convoys so as to accomplish prescribed deliveries, minimizing empty voyages. Koopmans found his results in 1943, but due to wartime restrictions he published them only after the war. (See figure 2.)

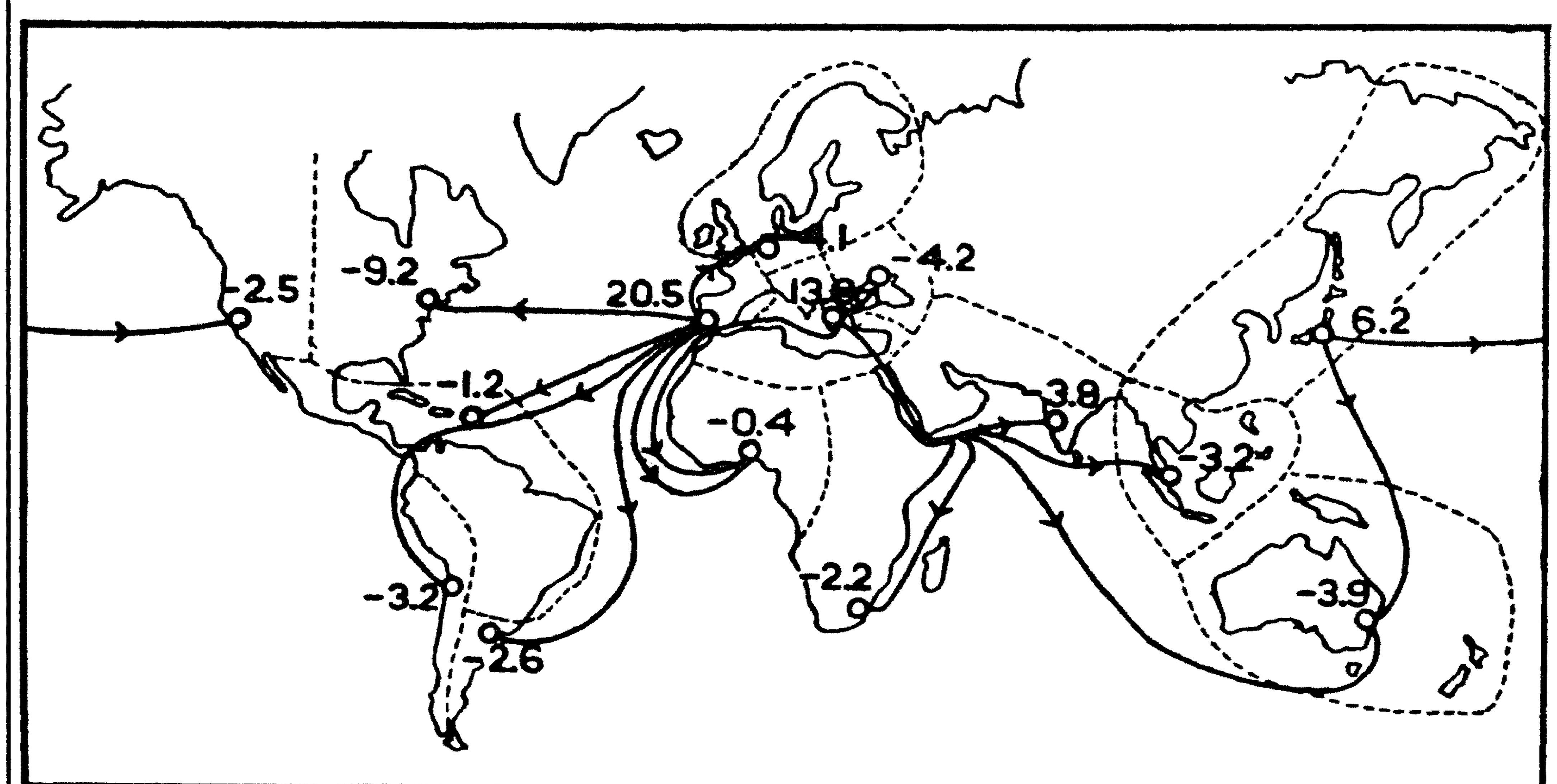
In the second half of the 1940's, the work of Dantzig gave the breakthrough of linear programming, especially due to his description of the simplex method in a compact tableau-form with an easy 'pivoting' rule. Dantzig also observed that if the method is applied to the Hitchcock-Koopmans transportation problem and if the supplies and demands are integer-valued, then there exists an optimum solution X that is integer-valued.

This makes it possible to apply linear programming methods to several other combinatorial optimization problems, in particular to problems where articles are indivisible, like the *optimum assignment problem*—the problem of assigning men (or machines) to jobs so as to minimize costs: given an $n \times n$ matrix $C = (c_{i,j})$, find a permutation π of $\{1, \ldots, n\}$ minimizing $\sum_{i=1}^{n} c_{i,\pi(i)}$. In a different terminology, it asks for a minimum-weight perfect matching in a bipartite graph.

The assignment problem is a special case of the transportation problem, as it is equivalent to minimizing $\sum_{i,j} c_{i,j} x_{i,j}$ over all nonnegative matrices $X = (x_{i,j})$ satisfying

$$\sum_{j=1}^{n} x_{i,j} = 1 \text{ for all } i, \text{ and } \sum_{i=1}^{n} x_{i,j} = 1 \text{ for all } j. \tag{2.2}$$

Nonnegative matrices satisfying (2.2) are called *doubly stochastic*, and the theorem behind is the *Birkhoff-von Neumann theorem*: each doubly stochastic matrix is a convex combination of permutation matrices. That is, each vertex of the polyhedron determined by (2.2) has integer coordinates only.



Optimal Routes of Empty Shipping Corresponding to World Dry-Cargo Flows in 1925.

Figure 2. A map, presented in the 1947 paper by Tj.C. Koopmans, one of the earliest studies on optimization in transportation.

So linear programming automatically gives an integer optimum solution X, and such an integer matrix is a permutation matrix.

3. THE TRAVELING SALESMAN PROBLEM

Having such efficient methods for the assignment problems, one is tempted to try similar methods to similar problems, and one of the most challenging turned out to be the *traveling salesman problem*: given an $n \times n$ matrix C, find a cyclic permutation π of $\{1, \ldots, n\}$ minimizing $\sum_{i=1}^{n} c_{i,\pi(i)}$. (A permutation is *cyclic* if it has exactly one orbit.)

Solving the traveling salesman problem is not only mathematically intriguing, but also of high practical importance due to its numerous occurrences in practice, in several forms (like in vehicle routing and production planning). M.M. Flood was one of the first studying the traveling salesman problem for practical purposes. He considered the problem in 1937 in relation to the routing of school buses. According to Flood, the idea of using polyhedral methods to solve the traveling salesman problem was brought to his attention by Koopmans in 1948. Flood next popularized this approach at the RAND Corporation, the intellectual centre of operations research, where several pioneers of linear programming and transportation were employed, and where polyhedral tools were successfully utilized, by Dantzig, L.R. Ford Jr., and D.R. Fulkerson, to solve flow problems coming from routing trains.

If we wish to solve the traveling salesman problem with polyhedral methods, the question arises what are the inequalities describing the *traveling salesman polytope*, that is, the convex hull of the cyclic permutation matrices.

We can take the equalities (2.2) as a basis, but they are obviously not enough, as each noncyclic permutation matrix satisfies (2.2). The noncyclic permutation matrices can be excluded by adding the following *subtour elimination constraints*:

$$\sum_{i\in I, j\notin I} x_{i,j} \geq 1 \text{ for each } I \subseteq \{1,\ldots,n\} \text{ with} \qquad (3.1)$$

$$\emptyset \neq I \neq \{1,\ldots,n\}.$$

It would be very nice if adding these constraints gives a complete description of the traveling salesman polytope; that is, if the polyhedron determined by (2.2) and (3.1) has integer vertices only. It would enable us to use the simplex method to solve the traveling salesman problem; moreover, it would imply (as we shall see below) that the traveling salesman problem is solvable in polynomial time. This is possible because, although the number of constraints in (3.1) grows exponentially with n, these yet can be checked in polynomial time: given a doubly stochastic matrix X, we can test in polynomial time if it satisfies (3.1) (by a reduction to minimum-capacity cut computations).

However, while the inequalities (3.1) are enough to cut off the noncyclic permutation matrices from the polytope of doubly stochastic matrices, they yet do not yield all facets of the traveling salesman polytope (if $n \geq 5$). There exist doubly stochastic matrices, of any order $n \geq 5$, that satisfy (3.1) but are not a convex combination of cyclic permutation matrices.

This disappointing fact has stimulated a stream of research. In a seminal paper of Dantzig, Fulkerson, and S.M. Johnson (1954) (according to A.J. Hoffman and Ph. Wolfe 'one of the principal events in the history of combinatorial optimization'), several new methods for solving the traveling salesman problem were introduced that are basic in combinatorial optimization up to today.

One of their basic observations is that, although we do not know a full description of the traveling salesman polytope, we obtain a lower bound for the minimum tour length if we optimize over the constraints (2.2) and (3.1).

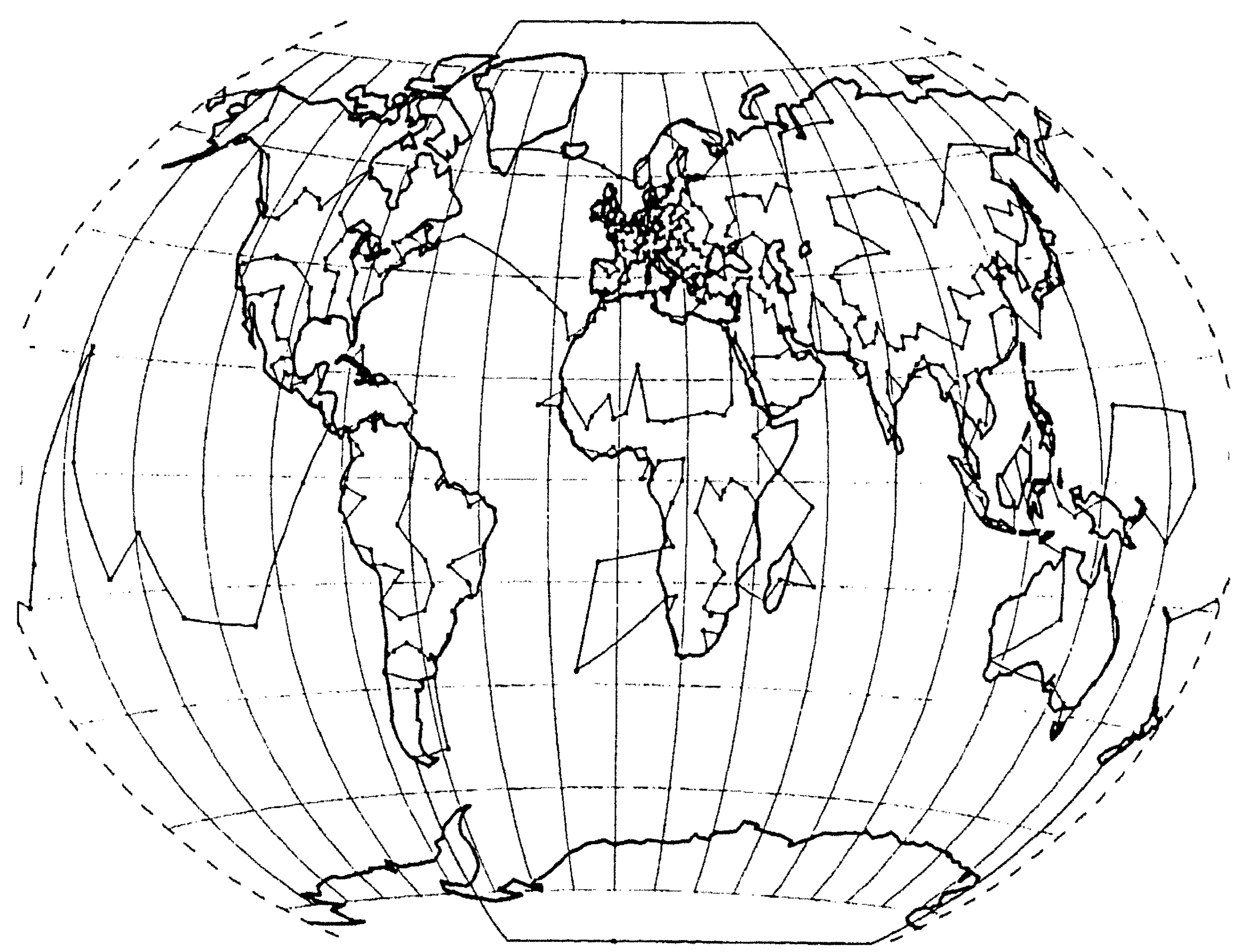


Figure 3. Optimal world-tour (666 cities).

This lower bound can be calculated with the simplex method, taking the (exponentially many) constraints (3.1) as *cutting planes* that can be added during the process when necessary. In this way, Dantzig, Fulkerson, and Johnson were able to find the shortest tour along cities in the 48 U.S. states and Washington, D.C. (See figure 3.)

This general approach has turned out to be extremely fruitful also in attacking other combinatorial optimization problems. In particular, around 1965, J. Edmonds at the National Bureau of Standards showed the applicability of the method, as an exact method, to many important classes of problems, most prominently matching problems.

4. Complexity

Edmonds also advertized *polynomial-time solvability* as a touchstone of the complexity of a problem. An algorithm is called *polynomial-time* if the number of steps is bounded by a polynomial in the size of the input. Generally, such an algorithm is fast in practice.

Polynomial-time algorithms for the assignment and transportation problems (the 'Hungarian method' and extensions) were designed in the 1950's and 1960's, but no such algorithm was found for the traveling salesman problem.

Note that checking all possible traveling salesman tours isn't a polynomial-time method, since there are exponentially many ($(n-1)!$) cyclic permutations. (Observe also that the number of feasible (not necessarily optimum) solutions is not a measure for the complexity of a problem: the number of cyclic permutations is smaller than the number of all permutations ($n!$), but yet selecting an optimum permutation (the assignment problem) turns out to be easier than selecting an optimum cyclic permutation (the traveling salesman problem).)

The general feeling that the traveling salesman problem is much more difficult than the transportation and assignment problem, got mathematical foundation at the start of the 1970's, by the work of S.A. Cook and R.M. Karp on the complexity classification of problems. The introduction of the complexity classes P and NP gave a key to distinguish problems on their complexity. The class P consists of all problems that can be solved in polynomial time; the transportation and the assignment problem belong to this class.

The class NP is potentially much wider than P. It includes all optimization problems with the property that it has an optimum solution the feasibility of which can be checked in polynomial time. About any combinatorial optimization problem belongs to NP, for instance the traveling salesman problem. What Karp showed was that the traveling salesman problem, and many other important combinatorial optimization problems, are the hardest in the class NP; in technical terms, they are NP-*complete*. It means that each problem in NP can be reduced, in polynomial time, to the traveling salesman problem. So if NP≠P then the traveling salesman problem is not solvable in polynomial time.

The definition of NP is very little restrictive, and there is no reason to believe that NP=P. But as yet, no mathematical proof has been found that these classes are really different. Beside being an intriguing mathematical problem, knowing whether NP≠P holds is also of practical importance. If NP=P can be proved, it might imply a revolutionary new algorithm, or, alternatively, it might mean that the concept of polynomial-time algorithm is completely meaningless. If NP≠P can be shown, the proof might give a clue why certain problems are harder than other, and might direct us to attack the kernel of the problems.

Thus the traveling salesman problem is pivotal — if it is polynomial-time solvable, then about any combinatorial optimization problem is polynomial-time solvable. What impact does it have on the value of the polyhedral method? Maybe there is an alternative method that gives us a polynomial-time algorithm.

Only in 1979 it was shown that the class of linear programming problems belongs to P. This was shown by L.G. Khachiyan in 1979, by adapting the *ellipsoid method* for nonlinear programming. This implies a polynomial-time algorithm for any combinatorial optimization problem if we know all inequalities describing the corresponding polyhedron. But in what sense do we need to know them?

In 1981, M. Grötschel, L. Lovász, and A. Schrijver showed that one does not need an explicit list of inequalities describing the polyhedron. It suffices to be able to solve the *separation problem* for the corresponding polyhedron P in polynomial time: given a vector x, does x belong to P, and if $x \notin P$, find a hyperplane separating x and P.

The separation problem trivially is polynomial-time solvable if we have an explicit list of all inequalities determining P completely written out before us—in that case we can check them one by one; but it is not necessary to have such a list. In fact, what is shown is that the polynomial-time solvability of a combinatorial optimization problem is *equivalent* to the polynomial-time solvability of the separation problem. Thus an appropriate description of the corresponding polytope is necessary and sufficient for the polynomial-time solvability of the optimization problem.

For the traveling salesman problem it means that it is polynomial-time solvable if and only if there is a polynomial-time algorithm checking if a given doubly stochastic matrix belongs to the traveling salesman polytope. Thus the polyhedral approach is in a sense necessary and sufficient.

5. Computational and other work

In any case, the polyhedral method can give very good bounds, especially if we include it in a *branch-and-bound* algorithm. A branch-and-bound algorithm consists of a branched case checking (by setting entries in the variable matrix X to 0 or 1), guided by lower bounds calculated for each case considered. At each iteration, the case with the smallest lower bound is split into two new cases, by setting a new entry to 0 and to 1 respectively. Having good lower bounds that can be calculated fast is essential for the computational behaviour of a branch-and-bound method.

To obtain better lower bounds, more facet-defining inequalities can be added to (3.1), so as to describe the traveling salesman polytope more and more accurate. To shortcut the branching process, heuristics finding and improving tours can be incorporated, so that 'hopeless' branches can be eliminated.

Such methods form the basis for the continuous progress in solving large-scale traveling salesman problems. Around 1980, Grötschel found an op-

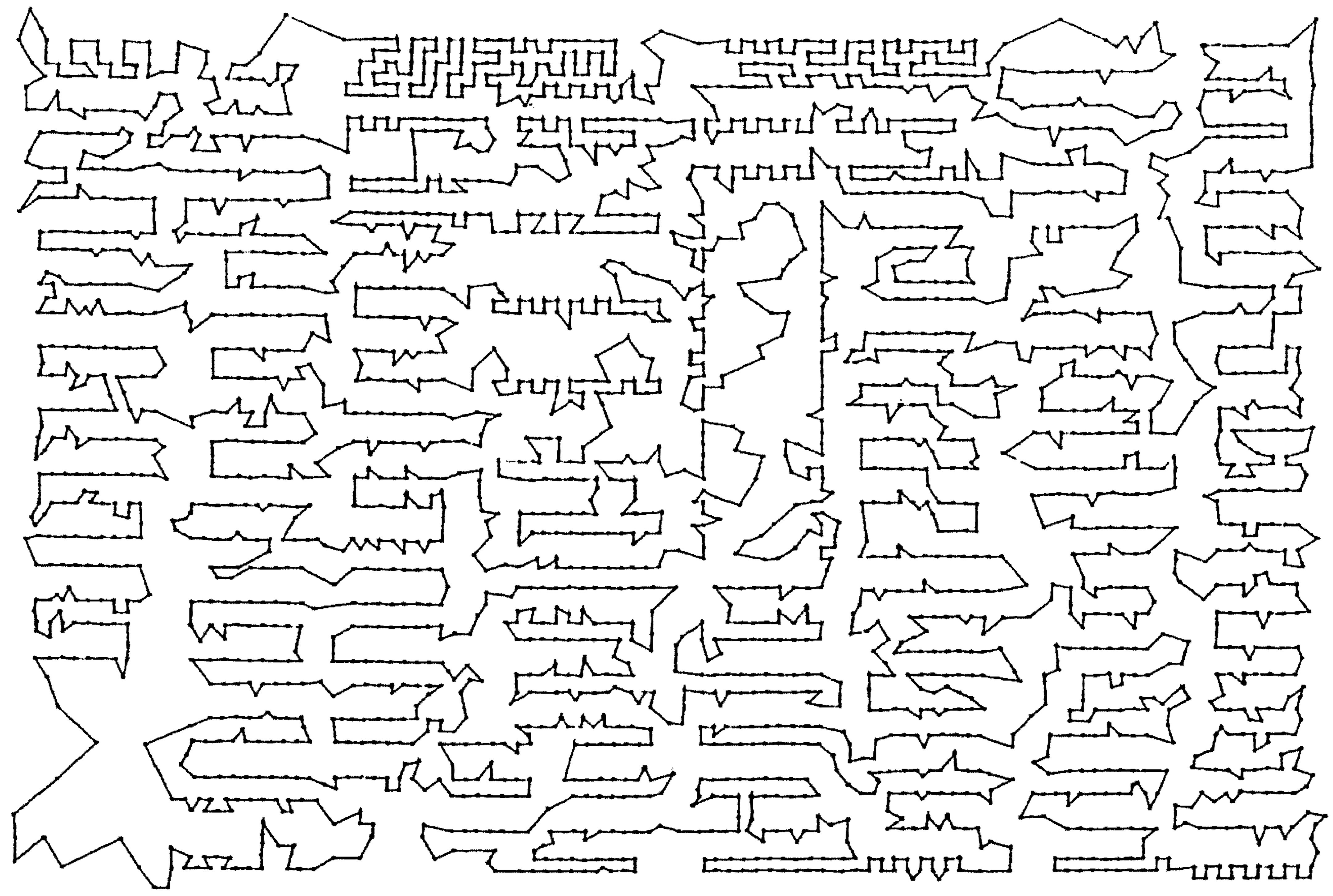


Figure 4. Record optimum tour along 3038 holes in a printed circuit board (1993).

timum tour along 120 cities in the Federal Republic of Germany, and H. Crowder and M. Padberg found one along 318 holes in a certain printed circuit board. Subsequent improvements during the last 15 years have led to the solution of a 7397 city problem by D. Applegate, R.E. Bixby, V. Chvátal, and W.J. Cook in 1994. (See also figure 4.)

At CWI research has been done on the complexity analysis of methods for the traveling salesman and related problems and on practical applications. For Van Gend & Loos, a Dutch transport company, the vehicle-routing system CAR (Computer-Aided Routing) was developed and installed. Important ingredient is a method to solve a traveling salesman problem with time-windows; that is, each 'city' can be visited only during a certain time period. (See figure 6.)

Moreover, research at CWI on polyhedral methods in combinatorial optimization has been pointed to identifying problem classes that are polynomial-time solvable with the polyhedral method. These problems include disjoint paths and trees, in particular in relation to routing wires on a VLSI-circuit.

At a more elementary level, CWI developed polyhedral methods for the problem of the most economical circulation of railway stock, a problem pre-

ride number		2123	2127	2131	2135	2139	2143	2147	2151	2155	2159	2163	2167	2171	2175	2179	2183	2187	2191
Amsterdam	V		6.48	7.55	8.56	9.56	10.56	11.56	12.56	13.56	14.56	15.56	16.56	17.56	18.56	19.56	20.56	21.56	22.56
Rotterdam	A		7.55	8.58	9.58	10.58	11.58	12.58	13.58	14.58	15.58	16.58	17.58	18.58	19.58	20.58	21.58	22.58	23.58
Rotterdam	V	7.00	8.01	9.02	10.03	11.02	12.03	13.02	14.02	15.02	16.00	17.01	18.01	19.02	20.02	21.02	22.02	23.02	
Roosendaal	A	7.40	8.41	9.41	10.43	11.41	12.41	13.41	14.41	15.41	16.43	17.43	18.42	19.41	20.41	21.41	22.41	23.54	
Roosendaal	V	7.43	8.43	9.43	10.45	11.43	12.43	13.43	14.43	15.43	16.45	17.45	18.44	19.43	20.43	21.43			
Vlissingen	A	8.38	9.38	10.38	11.38	12.38	13.38	14.38	15.38	16.38	17.40	18.40	19.39	20.38	21.38	22.38			

ride number		2108	2112	2116	2120	2124	2128	2132	2136	2140	2144	2148	2152	2156	2160	2164	2168	2172	2176
Vlissingen	V			5.30	6.54	7.56	8.56	9.56	10.56	11.56	12.56	13.56	14.56	15.56	16.56	17.56	18.56	19.55	
Roosendaal	A			6.35	7.48	8.50	9.50	10.50	11.50	12.50	13.50	14.50	15.50	16.50	17.50	18.50	19.50	20.49	
Roosendaal	V		5.29	6.43	7.52	8.53	9.53	10.53	11.53	12.53	13.53	14.53	15.53	16.53	17.53	18.53	19.53	20.52	21.53
Rotterdam	A		6.28	7.26	8.32	9.32	10.32	11.32	12.32	13.32	14.32	15.32	16.32	17.33	18.32	19.32	20.32	21.30	22.32
Rotterdam	V	5.31	6.29	7.32	8.35	9.34	10.34	11.34	12.34	13.35	14.35	15.34	16.34	17.35	18.34	19.34	20.35	21.32	22.34
Amsterdam	A	6.39	7.38	8.38	9.40	10.38	11.38	12.38	13.38	14.38	15.38	16.40	17.38	18.38	19.38	20.38	21.38	22.38	23.38

Table 1

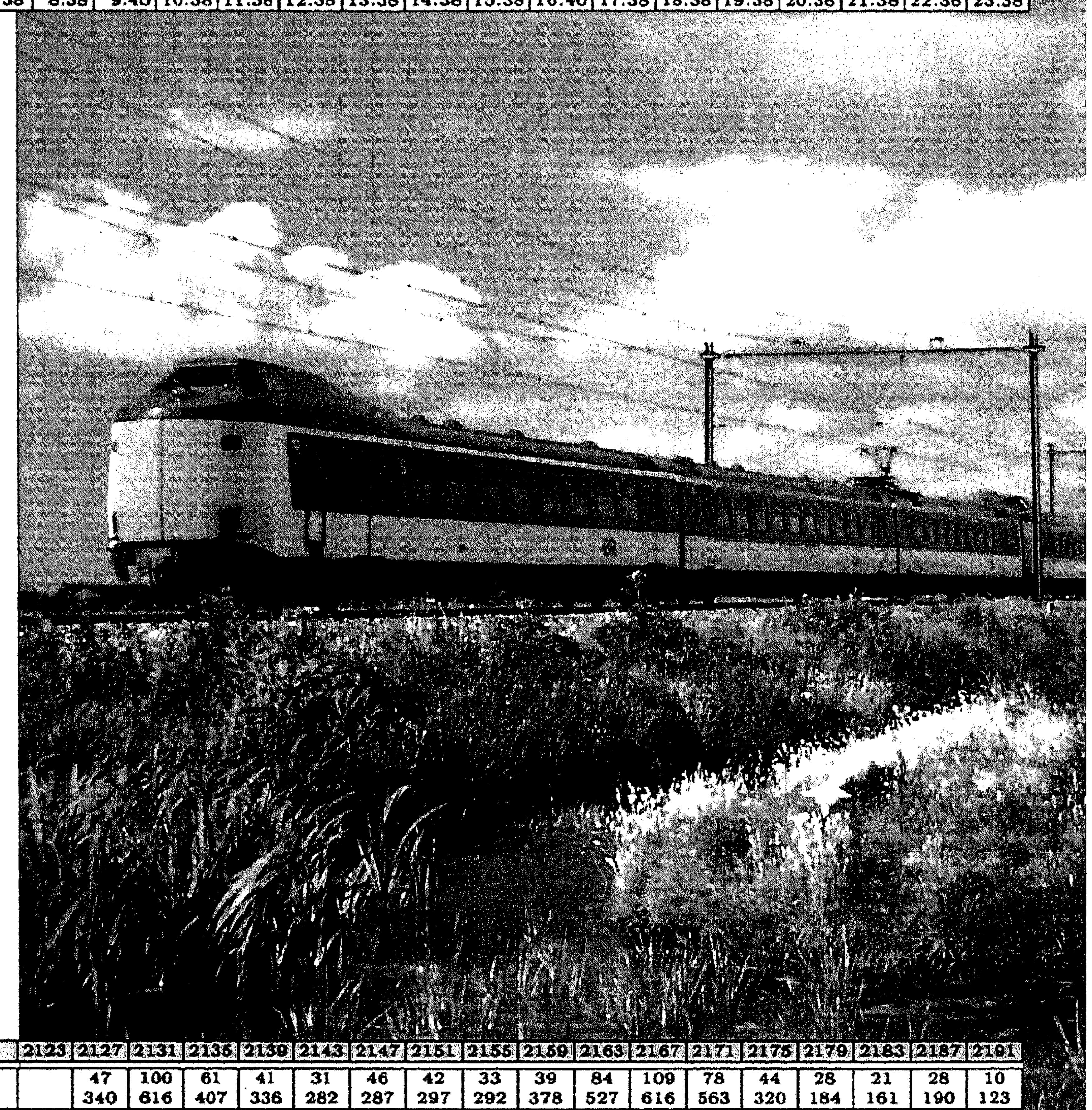


Table 2

train number	2123	2127	2131	2135	2139	2143	2147	2151	2155	2159	2163	2167	2171	2175	2179	2183	2187	2191
Amsterdam-Rotterdam		47 340	100 616	61 407	41 336	31 282	46 287	42 297	33 292	39 378	84 527	109 616	78 563	44 320	28 184	21 161	28 190	10 123
Rotterdam-Roosendaal	4 58	35 272	52 396	41 364	26 240	25 221	27 252	27 267	28 287	52 497	113 749	98 594	51 395	29 254	22 165	13 130	8 77	
Roosendaal-Vlissingen	14 328	19 181	27 270	26 237	24 208	32 188	15 180	21 195	23 290	41 388	76 504	67 381	43 276	20 187	15 136			

train number	2108	2112	2116	2120	2124	2128	2132	2136	2140	2144	2148	2152	2156	2160	2164	2168	2172	2176
Vlissingen-Roosendaal			28 138	100 448	48 449	57 436	24 224	19 177	19 184	17 181	19 165	22 225	39 332	30 309	19 164	15 142	11 121	
Roosendaal-Rotterdam		16 167	88 449	134 628	57 397	71 521	34 281	26 214	22 218	21 174	25 206	35 298	51 422	32 313	20 156	14 155	14 130	7 64
Rotterdam-Amsterdam	7 61	26 230	106 586	105 545	56 427	75 512	47 344	36 303	32 283	34 330	39 338	67 518	74 606	37 327	23 169	18 157	17 154	11 143

Figure 5. Commissioned by Nederlandse Spoorwegen, CWI computed an optimum circulation plan for rolling stock on the Amsterdam-Vlissingen line.

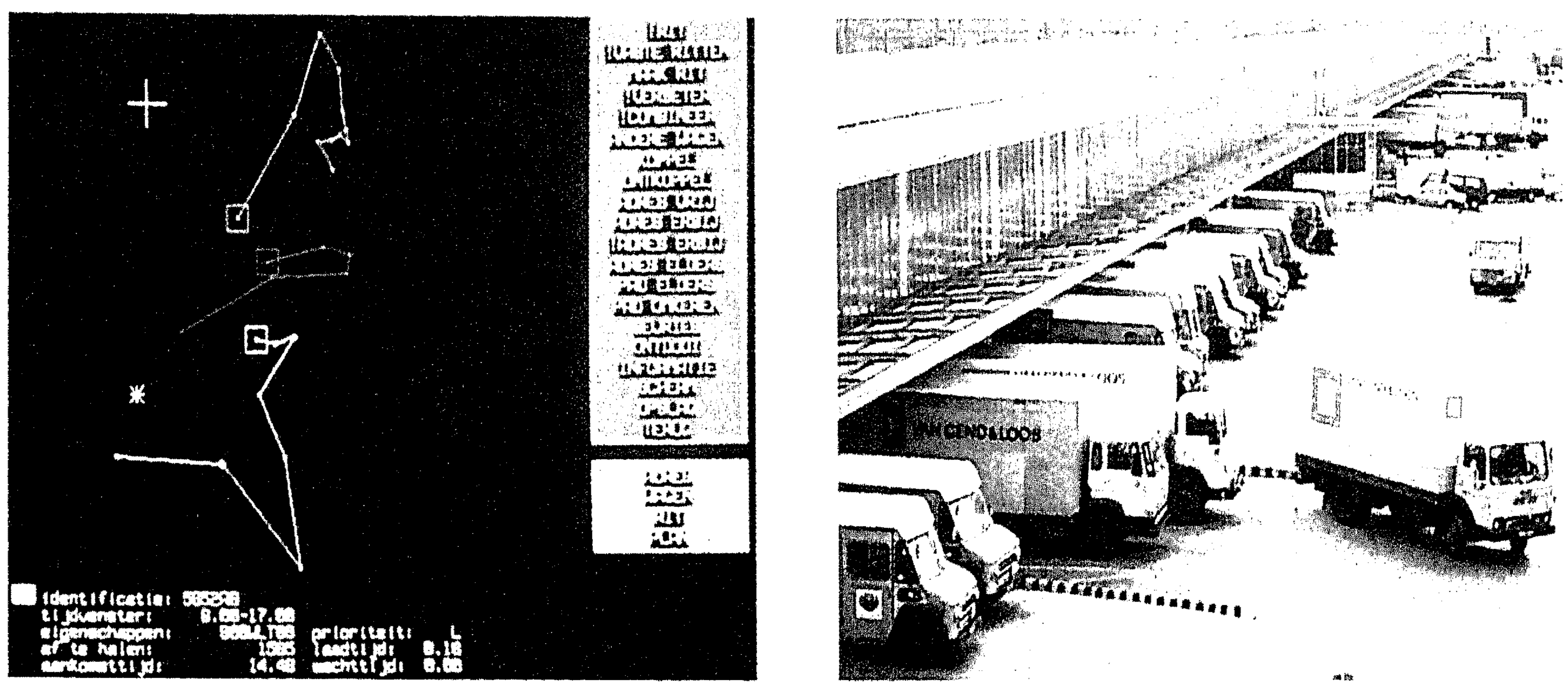


Figure 6. Computer-Aided Routing (CAR) is an interactive software package used as a support tool for physical distribution. CAR was jointly designed and developed by CWI and the Dutch road haulier Van Gend & Loos.

sented by Nederlandse Spoorwegen (see figure 5). This leads us back to a multicommodity flow problem, one of the original motivations of Kantorovich. The problem consists of determining the minimum amount of rolling stock to be purchased by NS in order to guarantee a given number of seats in each of the scheduled train legs. If there would be only one type of stock, one could solve the problem directly with linear programming methods, as it would automatically yield integer solutions. The question of NS was to extend the method to the case where units of several types are available, that can be coupled together. The original circulation problem (with lower bounds) then becomes a multicommodity circulation problem. The solutions are restricted to be integer-valued, since one cannot break train-units.

However, in that case, applying linear programming does not automatically give an optimum solution that is integer-valued. Thus we were bound to embed the polyhedral method in a branch-and-bound framework. To make it work, a number of cutting planes had to be added. This gives an algorithm that finds an optimum circulation plan for the Amsterdam-Vlissingen line (with 99 scheduled legs) within a few seconds.

For background information we refer to the books mentioned below.

REFERENCES

1. M. GRÖTSCHEL, L. LOVÁSZ, A. SCHRIJVER (1988). *Geometric Algorithms and Combinatorial Optimization*, Springer, Berlin.
2. E.L. LAWLER, J.K. LENSTRA, A.H.G. RINNOOY KAN, D.B. SHMOYS

(eds.). (1985). *The Traveling Salesman Problem—A Guided Tour of Combinatorial Optimization*, Wiley, Chichester.

3. J.K. LENSTRA, A.H.G. RINNOOY KAN, A. SCHRIJVER (eds.) (1991). *History of Mathematical Programming—A Collection of Personal Reminiscences*, CWI/North-Holland, Amsterdam.
4. M.W.P. SAVELSBERGH (1992). *Computer Aided Routing*, CWI Tract 75, CWI, Amsterdam.
5. A. SCHRIJVER (1986). *Theory of Linear and Integer Programming*, Wiley, Chichester.