Available online at www.sciencedirect.com
ScienceDirect
Discrete Optimization 5 (2008) 168–173
DISCRETE OPTIMIZATION
www.elsevier.com/locate/disopt

George Dantzig's contributions to integer programming

Martin Grötschel[a], George L. Nemhauser[b,*]

[a] *Konrad-Zuse-Zentrum für Informationstechnik, Berlin (ZIB) and TU Berlin, Germany*
[b] *School of Industrial and Systems Engineering, Georgia Institute of Technology, USA*

Received 21 March 2006; accepted 15 August 2007
Available online 22 October 2007

Abstract

This paper reviews George Dantzig's contributions to integer programming, especially his seminal work with Fulkerson and Johnson on the traveling salesman problem.

Keywords: History of integer programming; Traveling salesman problem

1. Introduction

George Dantzig wrote only a few papers on integer programming, including two on integer programming modeling [4,5]; specifically, how a variety of nonlinear and nonconvex optimization problems could be formulated as mixed-integer programs with 0–1 variables. For example, he presented the use of 0–1 variables to model fixed charges and variable upper bound constraints, semi-continuous variables, and nonconvex piecewise linear functions.

In [5] he also proposed a very simple cutting plane for separating a fractional basic optimal solution from the convex hull of integer solutions in a pure integer program with nonnegative variables. The cut simply says that at least one of the nonbasic variables must be a positive integer, i.e., the sum of the nonbasic variables is at least one. While this is not a very strong cut, since it does not yield a finitely convergent algorithm [10], a slightly tightened version of it does yield a finite cutting plane algorithm [2].

However, Dantzig's impact on integer programming is huge. His work in the 1950s with D. Ray Fulkerson and Selmer Johnson [6–8] on the traveling salesman problem was the precursor of the branch-and-cut algorithms that form the basis of modern mixed-integer computational systems that are widely used in practice to solve optimization models in supply chains, telecommunications, manufacturing, transportation, and many other areas.

The NP-hard traveling salesman problem (TSP) has provided a remarkable source of ideas for solving hard combinatorial optimization problems including cutting planes, branch-and-bound, and Lagrangian duality. Dantzig, Fulkerson, and Johnson (DFJ from now on) pioneered the idea of employing linear programming relaxation and valid inequalities to solve integer programs by solving (including a proof of optimality) a 49-city TSP. Their paper also has ideas about implicit enumeration. Moreover, the DFJ paper constitutes one of the first serious computational studies of a hard combinatorial optimization problem. It is absolutely astonishing that the three authors were able to find an optimal solution of such a large TSP instance and to prove its optimality by manual computation.

* Corresponding author.
E-mail addresses: groetschel@zib.de (M. Grötschel), george.nemhauser@isye.gatech.edu (G.L. Nemhauser).

1572-5286/$ - see front matter
doi:10.1016/j.disopt.2007.08.003

Although DFJ's seminal contribution of more than 50 years ago is acknowledged in books and survey papers on integer programming and combinatorial optimization, it has not been presented with any detail in recent literature except in a very recent book [1]. Therefore it seems appropriate in this issue devoted to the contributions of George Dantzig to review the work of DFJ, and to honor Ray Fulkerson (1924–1976) and Selmer Johnson as well. DFJ were all at the Rand corporation through the 1950s as part of what may have been the most remarkable group of mathematicians working on optimization ever assembled.

2. The TSP and linear programming

Given a set of n cities and the $n(n-1)/2$ distances $d(ij)$ between all unordered pairs i, j of cities, the (symmetric) TSP is to find a shortest tour for a salesman starting from his home city, then visiting all of the other cities, and finally returning to the home city. In graph theory terms, we are given a complete undirected graph $G = (V, E)$, where the node set V corresponds to the set of cities, the edge set E corresponds to all pairs of cities, and where the edge $e = ij$ has length $d(e)$ which is a number representing the "distance" (measured in minutes, miles, or whatever is appropriate for the particular instance) between the nodes i and j. The problem is to find a cycle C that contains all n nodes (i.e., a Hamiltonian cycle) and whose total distance is minimum. It is well known that the TSP is NP-hard although, of course, DFJ were unlikely to be thinking about complexity then.

DFJ studied an instance consisting of the road distances between 49 cities, the then 48 state capitals in the U.S. and Washington DC. The data DFJ used came from a distance table that was prepared by the Rand Corporation. Table 1 shows a copy of the table of distances between the cities, hand written by Ray Fulkerson (we are indebted to Bob Bland for making the original available). If d'_{ij} denotes the original distance between the cities i and j in miles then the entry d_{ij} of the DFJ table was obtained using $d_{ij} := \left[\frac{1}{17}(d'_{ij} - 11)\right]$. where the brackets [.] denote rounding to the next integer. This looks somewhat strange. The authors remark that they wanted to obtain distances smaller than 256 to permit compact storage in binary representation. However, it turned out that they made no use of it. DFJ's formulation of the TSP contains the variables $x(e) = 1$ or 0 to indicate whether edge e is in the tour or not and the obvious constraints that each node has degree 2 in a cycle. They realized, of course, that this was not enough because the resulting solution might contain subtours, i.e., cycles on subsets $S \subset V$. However, DFJ knew that subtours could be removed using the *subtour elimination constraints*, which they stated in the following two forms:

$$\sum_{e \in E(S)} x(e) \le |S| - 1, \qquad \sum_{e \in \delta(S)} x(e) \ge 2,$$

where $E(S)$ denotes the set of edges in G with both ends in the node set S, and $\delta(S)$ denotes the set of edges with one end in S. DFJ observed that the two versions of the subtour elimination constraints are equivalent because they can be transformed into each other utilizing the degree constraints.

Still, this would not be enough for solving TSPs by linear programming for any but the smallest values of n. There are two reasons for this observation:

1. The number of subtour elimination constraints grows exponentially with n, and therefore, all of them could not be considered explicitly.
2. Even if a linear programming formulation of the form

$$\min \sum_{e \in E} d(e)x(e) \tag{1}$$

$$x(e) \ge 0 \quad e \in E \tag{2}$$

$$x(e) \le 1 \quad e \in E \tag{3}$$

$$\sum_{e \in \delta(v)} x(e) = 2, \quad v \in V \tag{4}$$

$$\sum_{e \in \delta(S)} x(e) \ge 2 \quad S \subset V, \qquad 2 \le |S| \le |V| - 2 \tag{5}$$

could be solved, the solution would not necessarily correspond to a tour since the LP solution might be fractional.

Table 1
Road distances between cities in adjusted units

City	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
1																																										
2	8																																									
3	39	45																																								
4	37	47	9																																							
5	50	49	21	15																																						
6	61	62	21	20	17																																					
7	58	60	16	17	18	6																																				
8	59	60	15	20	26	17	10																																			
9	62	66	20	25	31	22	15	5																																		
10	81	81	40	44	50	41	35	24	20																																	
11	103	107	62	67	72	63	57	46	41	23																																
12	108	117	66	71	77	68	61	51	46	26	11																															
13	145	149	104	108	114	106	99	88	84	63	49	40																														
14	181	185	140	144	150	142	135	124	120	99	85	76	35																													
15	187	191	146	150	156	142	137	130	125	105	90	81	41	10																												
16	161	170	120	124	130	115	110	104	105	90	72	64	34	31	27																											
17	142	146	101	104	111	97	91	85	86	75	51	59	29	53	48	21																										
18	174	178	133	138	143	129	123	117	118	107	83	84	54	46	35	26	31																									
19	185	186	142	143	140	130	126	124	128	118	93	101	72	69	58	58	43	26																								
20	164	165	120	123	124	106	106	105	110	104	86	97	71	93	82	62	42	45	22																							
21	137	139	94	96	94	80	78	77	84	77	56	64	65	90	87	58	36	68	50	30																						
22	117	122	77	80	83	68	62	60	61	50	34	42	49	82	77	60	30	62	70	49	21																					
23	114	118	73	78	84	69	63	57	59	48	28	36	43	77	72	45	27	59	69	55	27	5																				
24	85	89	44	48	53	41	34	28	29	22	23	35	69	105	102	74	56	88	99	81	54	32	29																			
25	77	80	36	40	46	34	27	19	21	14	29	40	77	114	111	84	64	96	107	87	60	40	37	8																		
26	87	89	44	46	46	30	28	29	32	27	36	47	78	116	112	84	66	98	95	75	47	36	39	12	11																	
27	91	93	48	50	48	34	32	33	36	30	34	45	77	115	110	83	63	97	91	72	44	32	36	9	15	3																
28	105	106	62	63	64	47	46	49	54	48	46	59	85	119	115	88	66	98	79	59	31	36	42	28	33	21	20															
29	111	113	69	71	66	51	53	56	61	57	59	71	96	130	126	98	75	98	85	62	38	47	53	39	42	29	30	12														
30	91	92	50	51	46	30	34	38	43	49	60	71	103	141	136	109	90	115	99	81	53	61	62	36	34	24	28	20	20													
31	83	85	42	43	38	22	26	32	36	51	63	75	106	142	140	112	93	126	108	88	60	64	66	39	36	27	31	28	28	8												
32	89	91	55	55	50	34	39	44	49	63	76	87	120	159	156	129	100	123	109	86	62	71	78	52	49	39	44	35	24	15	12											
33	95	97	64	63	56	42	49	56	60	75	86	97	126	160	155	128	104	128	113	90	67	76	82	62	59	49	53	40	29	25	23	11										
34	74	81	44	43	35	23	30	39	44	62	78	89	121	159	155	127	108	136	124	101	75	79	81	54	50	42	46	43	39	23	14	14	21									
35	67	69	42	41	31	25	32	41	46	64	83	90	130	164	160	148	114	146	134	111	85	84	86	59	52	47	51	53	49	32	24	24	30	9								
36	74	76	61	60	42	44	51	60	66	83	102	110	147	185	179	155	133	159	146	122	98	105	107	79	71	66	70	70	60	48	40	36	33	25	18							
37	57	59	46	41	25	30	36	47	52	71	93	98	136	172	172	148	126	158	147	124	120	97	99	71	65	59	63	67	62	46	38	37	43	23	13	17						
38	45	46	41	34	20	34	38	48	53	73	96	99	137	176	178	151	131	163	159	135	108	102	103	73	67	64	69	75	72	54	46	49	54	34	24	29	12					
39	35	37	35	26	18	34	36	46	51	70	93	97	134	171	176	151	129	161	163	139	118	102	101	71	65	65	70	84	78	58	50	56	62	41	32	38	21	9				
40	29	33	30	21	18	35	33	40	45	65	87	91	117	166	171	144	125	157	156	139	113	95	97	67	60	62	67	79	82	62	53	59	66	45	38	45	27	15	6			
41	3	11	41	37	47	57	55	58	63	83	105	109	147	186	188	164	144	176	182	161	134	119	116	86	78	84	83	101	108	88	80	86	92	71	64	71	54	41	32	25		
42	5	12	55	41	53	64	61	61	66	84	111	113	150	186	192	166	147	180	188	167	140	124	119	90	87	90	94	107	114	77	86	92	98	80	74	77	60	48	38	32	6	

Despite these difficulties, DFJ demonstrated that a linear programming approach to the TSP was viable, and in the process gave the first steps towards a theory that we now call *polyhedral combinatorics* that provides one of the main ingredients of successful modern integer programming software. Given a tour T of an n-city TSP, $n \geq 3$, let us define the vector $\chi^T \in \mathfrak{R}^E$ by setting $\chi^T(e) = 1$, if $e \in T$, and $\chi^T(e) = 0$, if $e \notin T$. Call χ^T the *incidence vector* of tour T. The convex hull of the incidence vectors of all tours, i.e.,

$$Q_T^n := \text{conv}\{\chi^T \in \mathfrak{R}^E \mid T \text{ a } n\text{-city tour}\}$$

is called the (symmetric) *traveling salesman polytope*. The study of Q_T^n (and its asymmetric companion) began in the mid-fifties and is still thriving today; see [1,17].

In a Rand preprint [6] to the published paper [7], DFJ indicate that although Heller [14] had shown that the constraints (2)–(5) are sufficient for describing Q_T^5 completely, the polytope that they define has fractional extreme point solutions for $n \geq 6$. Just to formulate the TSP as an integer program requires an exponential number of inequalities, and it was clear from the work of Heller [14] and Kuhn [16] that a huge number of linear inequalities are needed to characterize Q_T^n for even modest values of n. The tremendous sizes of the LPs that may have to be solved might lead one to give up on the linear programming approach to the TSP.

But rather than giving up, at this point DFJ made some key observations that have had a big impact on the development of modern integer programming.

- To solve an LP with a huge number of constraints, you don't need to begin with all of them. It suffices to start with a relatively small subset as long as you have a way of telling whether the solution to the relaxed problem satisfies all of the omitted constraints, and if not, of finding one that is violated by the current solution. Of course, for the TSP, if the current LP solution is a tour, by definition it satisfies all of the unknown inequalities that define Q_T^n and therefore is an optimal solution to the TSP. Hence the stopping rule for this approach to solving the TSP is obvious. Terminate with an optimal tour if and only if the LP solution represents a tour. Otherwise tighten the LP by adding another constraint that cuts off the current solution.

This observation is probably the earliest appearance of what we now call *separation* or *cutting plane recognition*. Given a polyhedron P and a point y in $\mathfrak{R}^n$, decide whether $y \in P$, and if not, find a hyperplane separating y from P. About twenty-five years later [12] it was discovered that "separation" and "optimization" are equivalent with respect to polynomial time solvability. More precisely, one can solve linear programs over a class of polyhedra (such as the traveling salesman polytopes Q_T^n) in polynomial time if and only if the separation problem for this class of polyhedra can be solved in polynomial time. Specifically, because the TSP is known to be NP-hard, the separation problem for Q_T^n is also NP-hard. In other words, given a point $y \in \mathfrak{R}^E$, checking whether $y \in Q_T^n$ and if not finding an inequality that is satisfied by all incidence vectors of tours but not by y is NP-hard. We can assume that DFJ didn't understand all of the formalities of separation, but they used their ingenuity to take advantage of some properties of the TSP.

- For the TSP all the incidence vectors of tours are extreme points of a relaxation that contains the nonnegativity constraints (2) and the degree constraints (4), and they give a polytope all of whose extreme points correspond to tours, subtours, or isolated edges of value 2. So if we begin with only constraints (2) and (4), it is trivial to recognize whether the optimal LP solution contains subtours or isolated edges, and it is also simple to find an inequality (3) or (5) that separates the solution from Q_T^n. Moreover, since there are only a small number of upper bound constraints (3), we could add all of them to begin with. DFJ didn't do that, but remember that all of their computations were done by hand. Once we begin to add subtour elimination constraints or upper bound constraints, the polytope is no longer integral. Since any fractional extreme point solution cannot be in Q_T^n, whenever an optimal LP solution is fractional or is integral and contains subtours, we know that we have to continue adding constraints. But how do we find the right ones?

Before exploring DFJ's use of valid inequalities further, we present some of their other innovations that have become important in computational integer programming. DFJ used what is now called *warm start*. That is, since the incidence vector of a tour is an extreme point of the initial LP relaxation, it is possible to begin the simplex algorithm with a basic solution corresponding to a good tour. For the given US instance, DFJ simply guessed what they thought might be an optimal tour and then, setting the constraints $x(e) \leq 1$ to equality for all edges in the tour to form a basis, obtained a basic solution corresponding to that tour.

For a TSP on a complete graph with Euclidean distances, many long edges can be excluded from an optimal tour in a straightforward way. For example in the 49-city instance, one can easily argue by bounds that it would not be optimal to go directly from an east coast state capital to a west coast state capital, and therefore, such edges can be eliminated from the instance. However, much more fixing of this type can be done using linear programming in a more advanced manner. DFJ introduced the idea of what is now called *reduced cost fixing*. Suppose we have solved an LP relaxation and an edge is currently nonbasic at value zero with reduced cost $r(e)$. Let $z(LP)$ be the value of the LP solution and $z(T)$ be the value of the best known tour. Then if

$$z(LP) + r(e) > z(T), \tag{6}$$

edge e is not in any optimal tour. Similarly, for a nonbasic edge at value 1, if (6) holds, then edge e is in every optimal tour. DFJ observed that reduced cost fixing is a powerful tool for reducing the size of a TSP and when the problem became small enough in the number of remaining edges, they could use "combinatorial arguments" to establish an optimal solution. They were not very specific on how this was done, but it wouldn't be surprising if their combinatorial arguments were a type of tree search enumeration suggestive of implicit enumeration or *branch-and-bound*. Finally, DFJ began the solution of the 49-city instance by reducing it to a 42-city instance by observing that the shortest path between Washington and Boston passed through seven other state capitals, and therefore, these seven cities could be eliminated and replaced by a single edge. (That is why Table 1 shows only 42 cities.) Here they were using a form of what we now call *preprocessing*.

DFJ do not give all of the iterative details on their solution to the 42-city capitals instance. They luckily guessed the optimal solution at the outset. This tour provided their initial basis for the LP relaxation. To solve the LP relaxation to obtain the provably optimal tour as a basic feasible solution, they needed nonnegativity, the 42 degree constraints, 16 upper bound constraints, 7 subtour elimination constraints and 2 other valid inequalities.

We mentioned that solving the separation problem for Q_T^n is hard. However, that does not exclude that, for some subclasses of the class of all facets of Q_T^n, polynomial time separation routines exist. Finding such algorithms is still an active research area, and the progress in this respect is, to a large extent, responsible for the enormous success of the cutting plane approach to the TSP; see [1]. The fact that one can solve the separation problem for subtour elimination constraints by viewing it as a min-cut problem [11] was first observed in [15,18]. DFJ did not know that, of course, and finding violated subtour elimination constraints for fractional solutions by hand is not as straightforward as it may look nowadays. Finally, the remaining two constraints, whose validity was proved using neat combinatorial arguments given to DFJ by I. Glicksberg, a colleague at Rand, are essentially what is known today as *comb inequalities* [3,13]. See [1] for a detailed discussion of these two inequalities.

3. Conclusions

Although DFJ were not the first to develop a connection between linear programming and combinatorial optimization, see, e.g., the work of Heller and Kuhn cited earlier, they were the first to demonstrate that linear programming could be used to attack large-scale combinatorial optimization problems by actually solving such an instance. Let us recall from the discussion above the concepts (in modern terminology) that were employed by DFJ in their 1954 study:

- preprocessing,
- warm start,
- variable fixing,
- reduced cost exploitation,
- cutting plane recognition,
- elements of branch-and-bound.

The authors were certainly not aware of the full power of their contribution. They close their paper with the following remark:

"It is clear that we have left unanswered practically any question one might pose of a theoretical nature concerning the traveling-salesman problem; however, we hope that the feasibility of attacking problems involving a moderate number of points has been successfully demonstrated, and that perhaps some of the ideas can be used in problems of similar nature."

which – compared to the marketing jargon one often reads today, even in the scientific literature – appears to be a very modest self-assessment of their own work. Nevertheless, the DFJ paper caught the interest of the public press. *Newsweek Magazine* published an article on this "ingenious application of linear programming" on July 26, 1954.

Reviewing the development of integer programming in the last fifty years, the DFJ paper of 1954 was a really remarkable contribution that considerably extended, among other things, the "computational IP tool box". It is even more remarkable that this has been done without the help of computers. It seems that DFJ's ideas were too advanced for their contemporaries since, five years later, see [8], they were asked by the editor of *Operations Research* to revisit their 1954 paper and explain its findings again, which they did in a somewhat simplified form on a 10-city example. This was in the days when Ralph Gomory's pioneering work [9] showed how linear programming could be used in a finite algorithm to solve any pure integer program. But in a certain sense, the work of DFJ is closer to the current branch-and-cut systems.

References

[1] D.L. Applegate, R.E. Bixby, V. Chvátal, W.J. Cook, The Traveling Salesman Problem: A Computational Study, Princeton University Press, 2006, 606 pp.

[2] V.J. Bowman, G.L. Nemhauser, A finiteness proof for modified Dantzig cuts in integer programming, Naval Research Logistics Quarterly 17 (1969) 309–313.

[3] V. Chvátal, Edmonds polytopes and weakly Hamiltonian graphs, Mathematical Programming 5 (1973) 29–40.

[4] G.B. Dantzig, Discrete variable extremum problems, Operations Research 5 (1957) 266–277.

[5] G.B. Dantzig, On the significance of solving linear programs with some integer variables, Econometrica 28 (1960) 30–34.

[6] G.B. Dantzig, D.R. Fulkerson, S. Johnson, Solution of a large scale traveling salesman problem, The Rand Corporation, P-510, 12 April 1954.

[7] G.B. Dantzig, D.R. Fulkerson, S. Johnson, Solution of a large scale traveling salesman problem, Operations Research 2 (1954) 393–410.

[8] G.B. Dantzig, D.R. Fulkerson, S. Johnson, On a linear programming combinatorial approach to the traveling salesman problem, Operations Research 7 (1959) 58–66.

[9] R.E. Gomory, Outline of an algorithm for integer solutions to linear programs, Bulletin of the American Mathematical Society 64 (1958) 275–278.

[10] R.E. Gomory, A.J. Hoffman, On the convergence of an integer programming process, Naval Research Logistics Quarterly 10 (1963) 121–123.

[11] R.E. Gomory, T.C. Hu, Multi-terminal network flows, SIAM Journal of Applied Mathematics 9 (1961) 551–556.

[12] M. Grötschel, L. Lovász, A. Schrijver, The ellipsoid method and its consequences in combinatorial optimization, Combinatorica 1 (1981) 167–197.

[13] M. Grötschel, M.W. Padberg, On the symmetric travelling salesman problem I: Inequalities, Mathematical Programming 16 (1979) 281–302.

[14] I. Heller, On the Traveling Salesman Problem, in: Proc. 2nd Symp. on Linear Programming, 2, 1955, pp. 643–665.

[15] S. Hong, A linear programming approach for the travelling salesman problem, Ph.D. Thesis. The Johns Hopkins University, Baltimore, USA, 1972.

[16] H.W. Kuhn, On certain convex polyhedra, Bulletin of the American Mathematical Society 61 (1955) 557–558.

[17] D. Naddef, Polyhedral theory and branch-and-cut algorithms for the symmetric TSP, in: G. Gutin, A.P. Punnen (Eds.), The Traveling Salesman Problem and its Variations, Kluwer, 2002, pp. 29–116 (Chapter 2).

[18] M.W. Padberg, S. Hong, On the symmetric travelling salesman problem: A computational study, Mathematical Programming Study 12 (1980) 78–107.