D.S. JOHNSON, J.K. LENSTRA, A.H.G. RINNOOY KAN
THE COMPLEXITY OF THE NETWORK DESIGN PROBLEM
Preprint
Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O).
THE COMPLEXITY OF THE NETWORK DESIGN PROBLEM

D.S. JOHNSON
Bell Laboratories, Murray Hill, New Jersey, USA

J.K. LENSTRA
Mathematisch Centrum, Amsterdam, The Netherlands

A.H.G. RINNOOY KAN
Graduate School of Management, Delft, The Netherlands

ABSTRACT

The network design problem is the problem of finding a subgraph of a weighted undirected graph, subject to a budget constraint on the sum of its edge weights, that minimizes the sum of the shortest path lengths between all vertex pairs. In this note we establish NP-completeness for the network design problem, even for the simple case where all edge weights are equal and the budget restricts the choice to spanning trees. This result justifies the development of enumerative optimization methods and of approximation algorithms, such as those described in a recent paper by R. Dionne and M. Florian.

KEY WORDS & PHRASES: network design problem, sum of all shortest path lengths, NP-completeness, knapsack, exact 3-cover

NOTE: This report is not for review; it will be submitted for publication in a journal.
The network design problem is the problem of finding a subgraph of a weighted undirected graph, subject to a budget constraint on the sum of its edge weights, that minimizes the sum of the shortest path lengths between all vertex pairs. In this note we establish NP-completeness \([6;7]\) for the network design problem. Briefly, this result implies that a polynomial-bounded method for its solution could be used to construct similar algorithms for a large number of combinatorial problems which are notorious for their computational intractability, such as the travelling salesman problem and the multicommodity network flow problem. Since none of these problems is known to be solvable in polynomial time, NP-completeness of the network design problem justifies the development of enumerative optimization methods and of approximation algorithms, such as those described in \([2]\).

For our purposes, we formulate the problem in the following way.

**NETWORK DESIGN PROBLEM (NDP):** Given an undirected graph \(G = (V,E)\), a weight function \(L: E \rightarrow \mathbb{N}\), a budget \(B\) and a criterion threshold \(C (3, C \in \mathbb{N})\), does there exist a subgraph \(G' = (V,E')\) of \(G\) with weight \(\sum_{[i,j] \in E'} L([i,j]) \leq B\) and criterion value \(F(G') \leq C\), where \(F(G')\) denotes the sum of the lengths of the shortest paths in \(G'\) between all vertex pairs?

By way of introduction to a quite involved NP-completeness proof for a simplified version of NDP, we shall first present a simple proof establishing NP-completeness for the general NDP.

**THEOREM 1.** NDP is NP-complete.

**Proof.** Consider the following problem.

**KNAPSACK:** Given positive integers \(t, a_1, \ldots, a_t, b\), does there exist a subset \(S \subseteq T = \{1, \ldots, t\}\) such that \(\sum_{i \in S} a_i = b\)?

We will show that KNAPSACK is reducible to NDP, i.e., that for any instance of KNAPSACK an instance of NDP can be constructed in polynomial-bounded time such that solving the instance of NDP solves the instance of KNAPSACK as well.
KNAPSACK: $t = 4, a_1 = 2, a_2 = 3, a_3 = 5, a_4 = 6, b = 7$.

NDP: $G$ & $L$:

![Diagram](attachment:image.png)

$B = 39, C = 249.$

**Figure 1** Equivalent instances of KNAPSACK and NDP.

The theorem then follows from the NP-completeness of KNAPSACK [6] and the solvability of NDP by polynomial-depth backtrack search.

Given any instance of KNAPSACK, we write $A = \sum_{i \in T} a_i$ and define an instance of NDP as follows:

- $V = \{0\} \cup \{i, i': i \in T\}$,
- $E = \{(0, i), (0, i'), (i, i'): i \in T\}$,
- $L((0, i)) = L((0, i')) = L((i, i')) = a_i$ ($i \in T$),
- $B = 2A + b$,
- $C = 4tA - b$.

Figure 1 illustrates this reduction. We claim that KNAPSACK has a solution if and only if $G = (V, E)$ contains a subgraph with weight at most $B$ and criterion value at most $C$.

It is easily seen that any feasible NDP solution can be assumed to contain a star graph $G^* = (V, \{(0, i), (0, i'): i \in T\})$; $G^*$ has weight $2A = B - b$ and criterion value $4tA = C + b$. Adding an edge $(i, i')$ to $G^*$ increases the weight.
by \( a_i \) and decreases the criterion value by \( a_i \), since \((i,i')\) will appear only in the shortest path between \(i\) and \(i'\). The equivalence now follows in a straightforward way.

However, since KNAPSACK can be solved in \( O(n^2 \log n) \) time [1], Theorem 1 does not exclude the existence of a similar pseudopolynomial algorithm [4] for NDP; the above construction crucially depends on allowing arbitrary positive integers as edge weights and budget. As a stronger result, we shall now prove that NDP is NP-complete even in the simple case where all edge weights are equal and the budget restricts the choice to spanning trees.

**Simple Network Design Problem (SNDP):** NDP with \( L((i,j)) = 1 \) for all \((i,j) \in E\) and \( B = |V|-1 \).

**Theorem 2.** SNDP is NP-complete.

**Proof.** As a starting point we take the following NP-complete problem [3,5].

**Exact 3-Cover:** Given a family \( S = \{ s_1, \ldots, s_r \} \) of 3-element subsets of a set \( T = \{ t_1, \ldots, t_{3t} \} \), does there exist a subfamily \( S' \subset S \) of pairwise disjoint sets such that \( \bigcup_{s \in S'} s = T \)?

We will show that Exact 3-Cover is reducible to SNDP.

Given any instance of Exact 3-Cover, we define an instance of SNDP as follows.

\[
V = R \cup S \cup T, \\
R = \{ r_0, r_1, \ldots, r_r \}, \\
r = C_{SS} + C_{ST} + C_{TT}, \\
E = \{ (i, r_i) : i = 1, \ldots, r \} \cup \{ (r_0, s) : s \in S \} \cup \{ (s, t) : t \in T \}, \\
C = C_{RR} + C_{RS} + C_{RT} + C_{SS} + C_{ST} + C_{TT},
\]

where

\[
C_{RR} = r^2, \quad C_{RS} = 2rs + s, \quad C_{RT} = 9rt + 6t, \quad C_{SS} = s^2 - s, \quad C_{ST} = 9st - 6t, \quad C_{TT} = 18t^2 - 12t.
\]

Figure 2 illustrates this reduction. We will prove that Exact 3-Cover
EXACT 3-COVER: $t = 2, s = 4,$

$$S = \{\{\tau_1, \tau_2, \tau_3\}, \{\tau_2, \tau_3, \tau_5\}, \{\tau_2, \tau_4, \tau_5\}, \{\tau_4, \tau_5, \tau_6\}\}.$$ 

**Figure 2** Equivalent instances of EXACT 3-COVER and SNDP.

has a solution if and only if $G = (V, E)$ contains a spanning tree with criterion value at most $C$. We assume that $G$ is connected, i.e., $\bigcup_{\sigma \in S} \sigma = T$.

Let $G' = (V, E')$ be some spanning tree of $G$ and let $F_{PQ}(G')$ denote the sum of the lengths of all shortest paths in $G'$ between vertex sets $P$ and $Q$ $(P, Q \subseteq V)$. We clearly have $(\rho_i, \rho_0) \in E'$ for all $i = 1, \ldots, r$. If $(\rho_0, \sigma) \notin E'$ for some $\sigma \in S$, then

$$F(G') > F_{RR}(G') + F_{RS}(G') + F_{RT}(G')$$

$$\geq C_{RR} + C_{RS} + 2(r+1) + C_{RT}$$

$$> C,$$

therefore, we may assume that $(\rho_0, \sigma) \in E'$ for all $\sigma \in S$. It follows that in $G'$ each vertex in $T$ is adjacent to exactly one vertex in $S$. Straightforward calculations show that we now have

$$F_{PQ}(G') = C_{PQ} \quad \text{for } P = R, S \text{ and } Q = R, S, T.$$

Denoting the number of vertices in $S$ being adjacent in $G'$ to exactly $h$ vertices in $T$ by $s_h$ $(h = 0, 1, 2, 3)$, we have
\[ F_{TT}(G') = 4\{3t(3t-1)/2 \]
\[ - 2|\{(\tau, \tau') : \tau \neq \tau', \{(\sigma, \tau), (\sigma, \tau')\} \in E' \text{ for some } \sigma \in S\}| \]
\[ = (18t^2 - 6t) - (2s_2 + 6s_3) \]
\[ = C_{TT} + 6(t-s_3) - 2s_2. \]

It is easily seen that \( F_{TT}(G') = C_{TT} \) if and only if \( s_3 = t, s_2 = s_1 = 0, \)
\( s_0 = s-t. \) The first condition is now equivalent to \( F(G') \leq C, \) the second one
to the existence of an EXACT 3-COVER SOLUTION. This completes the proof. \( \square \)

REFERENCES

University Press, Princeton, N.J.

2. P. DINCHE, M. FLOREAN (1976) Exact and approximate algorithms for
optimal network design. Publication 41, Centre de recherche sur les
transports, Université de Montréal.


vation, examples and implications. Bell Laboratories, Murray Hill, N.J.

5. L. HAYAFIL, R.L. RIVEST (1976) Constructing optimal binary decision trees
is NP-complete. Information Processing Lett. 2,15-17.


7. P.M. KARP (1975) On the computational complexity of combinatorial prob-
lems. Networks 2,45-68.