Bang Ye Wu and Kun-Mao Chao

Spanning Trees and Optimization Problems (Excerpt)

CRC PRESS
Boca Raton London New York Washington, D.C.

Chapter 1

Counting Spanning Trees

A spanning tree for a graph G is a subgraph of G that is a tree and contains all the vertices of G.

How many trees are there spanning all the vertices in Figure 1.1?

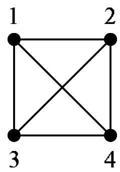

FIGURE 1.1: A four-vertex complete graph K_4.

Figure 1.2 gives all 16 spanning trees of the four-vertex complete graph in Figure 1.1.

DEFINITION 1.1 *A Prüfer sequence of length $n-2$, for $n \geq 2$, is any sequence of integers between 1 and n, with repetitions allowed.*

LEMMA 1.1
There are n^{n-2} Prüfer sequences of length $n-2$.

Example 1.1
The set of Prüfer sequences of length 2 is $\{(1,1), (1,2), (1,3), (1,4), (2,1), (2,2), (2,3), (2,4), (3,1), (3,2), (3,3), (3,4), (4,1), (4,2), (4,3), (4,4)\}$. In total, there are $4^{4-2} = 16$ Prüfer sequences of length 2. □

Algorithm: Prüfer Encoding
Input: A labeled tree with vertices labeled by $1, 2, 3, \ldots, n$.
Output: A Prüfer sequence.
Repeat $n-2$ times

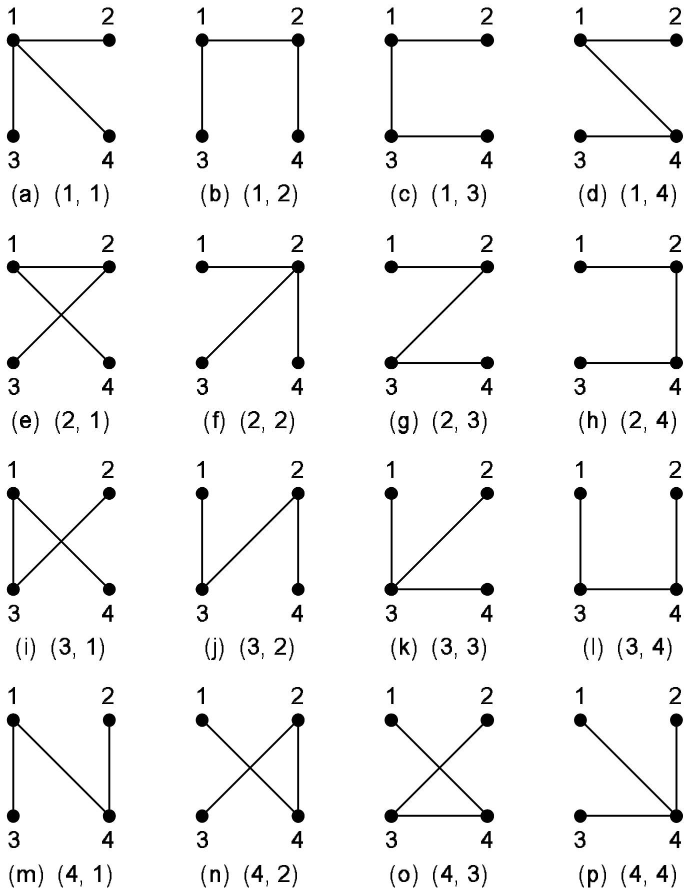

FIGURE 1.2: All 16 spanning trees of K_4.

$v \leftarrow$ the leaf with the lowest label
Put the label of v's unique neighbor in the output sequence.
Remove v from the tree.

Now consider a more complicated tree in Figure 1.3. What is its corresponding Prüfer sequence?

Figure 1.4 illustrates the encoding process step by step.

Algorithm: Prüfer Decoding
Input: A Prüfer sequence $P = (p_1, p_2, \ldots, p_{n-2})$.
Output: A labeled tree with vertices labeled by $1, 2, 3, \ldots, n$.
$P \leftarrow$ the input Prüfer sequence
$n \leftarrow |P| + 2$

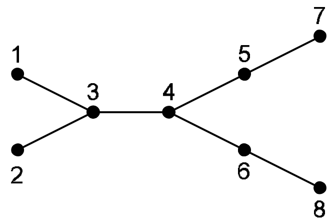

FIGURE 1.3: An eight-vertex spanning tree.

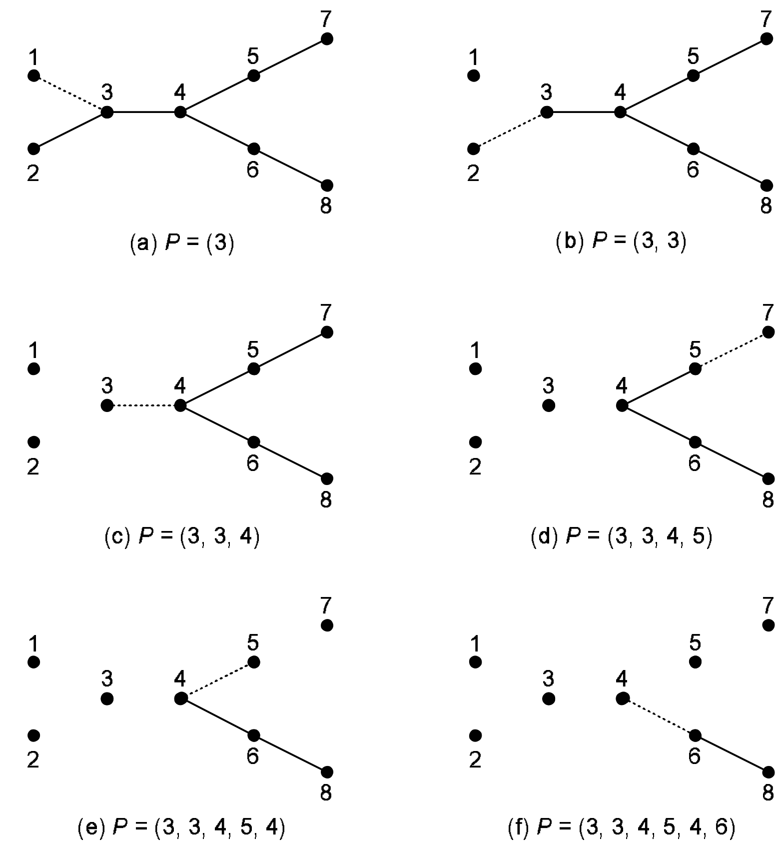

FIGURE 1.4: Generating a Prüfer sequence from a spanning tree.

$V \leftarrow \{1, 2, \ldots, n\}$
Start with n isolated vertices labeled $1, 2, \ldots, n$.
for $i = 1$ **to** $n - 2$ **do**
    $v \leftarrow$ the smallest element of the set V that does not occur in P

Connect vertex v to vertex p_i
Remove v from the set V
Remove the element p_i from the sequence P
/* Now $P = (p_{i+1}, p_{i+2}, \ldots, p_{n-2})$ */
Connect the vertices corresponding to the two numbers in V.

Figure 1.5 illustrates the decoding process step by step.

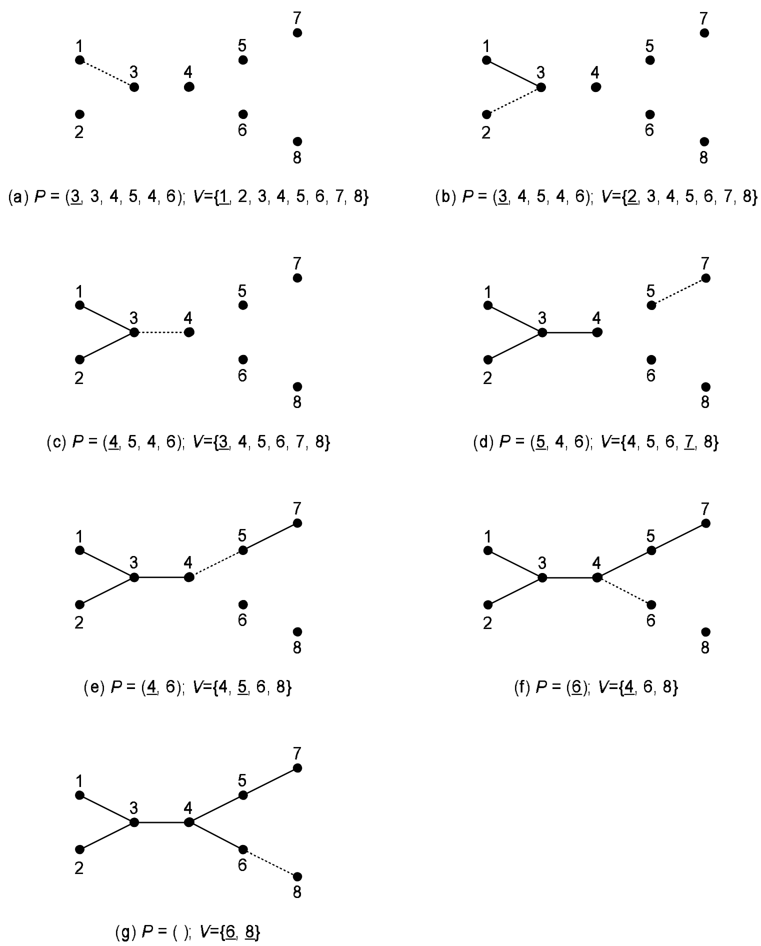

FIGURE 1.5: Recovering a spanning tree from a Prüfer sequence.

THEOREM 1.1
The number of spanning trees in K_n is n^{n-2}.

Let $G - e$ denote the graph obtained by removing edge e from G. Let $G \backslash e$ denote the resulting graph after contracting e in G. In other words, $G \backslash e$ is the graph obtained by deleting e, and merging its ends. Let $\tau(G)$ denote the number of spanning trees of G. The following recursive formula computes the number of spanning trees in a graph.

THEOREM 1.2
$\tau(G) = \tau(G - e) + \tau(G \backslash e)$