to I, now in the tree. The edge IE becomes a candidate. (See Fig. 4.18(e).) Values of *dist* for new fringe vertices must be computed.

Does this method work? The questionable step is the selection of the next fringe vertex and candidate edge. For an arbitrary candidate e, $d(v, tail(e))+W(e)$ is not necessarily equal to $d(v, head(e))$ because shortest paths to $head(e)$ might not pass through $tail(e)$. (In Fig. 4.18, for example, the shortest path to H does not go through G, although GH is a candidate in Figs. 4.18(c), (d), and (e).) We claim that, if e is chosen by minimizing $d(v, tail(e))+W(e)$ over all candidates, then e does give a shortest path. This claim is proved in the following theorem.

Theorem 4.2 Let $G=(V, E, W)$ be a weighted graph or digraph with weights in $\mathbf{Z}^+$. Let V' be a subset of V and let v be a member of V'. If e is chosen to minimize $d(v, tail(e))+W(e)$ over all edges with one vertex in V' and one in $V-V'$, then the path consisting of e adjoined to the end of a shortest path from v to $tail(e)$ is a shortest path from v to $head(e)$.

Proof. Look at Fig. 4.19. Suppose e is chosen as indicated. Let $e = yz$, where y is in V', and let $v, x_1, \ldots, x_r, y$ be a shortest path from v to y. Let $P = v, x_1, \ldots, x_r, y, z$. $W(P)=d(v,y)+W(e)$. Let $v, z_1, \ldots, z_l, \ldots, z$ be any path from v to z; call it P'. We must show that $W(P) \leq W(P')$. Let z_l be the first vertex in P' that is not in V'. (z_l may be z. If $l=1$, interpret z_0 as v. In the algorithm, $z_{l-1}z_l$ would be a candidate edge.) $W(P)=d(v,y)+W(e) \leq d(v,z_{l-1})+W(z_{l-1}z_l)$ (by the choice of e) $\leq W(P')$ since $v, z_1, \ldots, z_l$ is part of the path P'. □

If there is a path from v to w at all, then w will be a leaf in the tree grown from v. There is no way to tell which of the tree edges are in the path to w until the algorithm terminates, so all of the paths that branch out from v are retained by using *parent* as in the minimum spanning tree algorithm.

The shortest-path algorithm uses virtually the same data structure as the minimum spanning tree algorithm; see Fig. 4.16. The only change is that *dist* replaces *fringeWgt*. For each fringe vertex z, *dist*[z] is the weight of the path from v

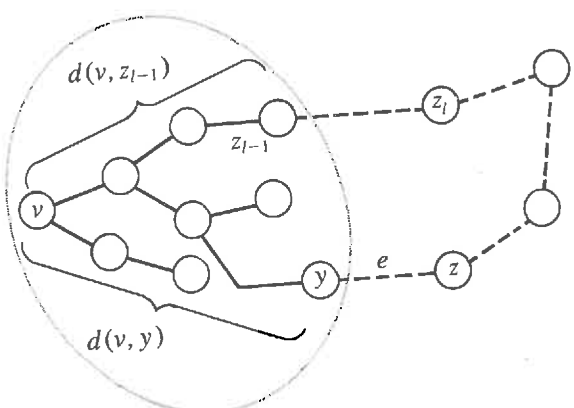

Figure 4.19 For the proof of Theorem 4.2.

(p. 170)
Sara Baase.
Computer Algorithms:
Introduction to
Design and
Analysis, 2nd ed.
Addison-Wesley,
1988.