

16

The Shortest Path Problem

The shortest path problem asks for the computation of the path from a source to a destination node that minimizes the sum of the positive weights¹ of its constituent links. The related shortest path tree (SPT) is the union of the shortest paths from a source node to a set of m other nodes in the graph with N nodes. If $m = N - 1$, the SPT connects all nodes and is termed a spanning tree. The SPT belongs to the fundamentals of graph theory and has many applications. Moreover, powerful shortest path algorithms like that of Dijkstra exist. Section 15.7 studied the hopcount, the number of hops (links) in the shortest path, in sparse graphs with unit link weights. In this chapter, the influence of the link weight structure on the properties of the SPT will be analyzed. Starting from one of the simplest possible graph models, the complete graph with i.i.d. exponential link weight, the characteristics of the shortest path will be derived and compared to Internet measurements.

The link weights seriously impact the path properties in QoS routing (Kuipers and Van Mieghem, 2003). In addition, from a traffic engineering perspective, an ISP may want to tune the weight of each link such that the resulting shortest paths between a particular set of in- and egresses follow the desirable routes in its network. Thus, apart from the topology of the graph, the link weight structure clearly plays an important role. Often, as in the Internet or other large infrastructures, both the topology and the link weight structure are not accurately known. This uncertainty about the precise structure leads us to consider both the underlying graph and each of the link weights as random variables.

¹ A zero link weight is regarded as the coincidence of two nodes (which we exclude), while an infinite link weight means the absence of a link.

16.1 The shortest path and the link weight structure

Since the shortest path is mainly sensitive to the smaller, positive link weights, the probability distribution of the link weights around zero will dominantly influence the properties of the resulting shortest path. A *regular* link weight distribution $F_w(x) = \Pr[w \leq x]$ has a Taylor series expansion around $x = 0$,

$$F_w(x) = f_w(0)x + O(x^2)$$

since $F_w(0) = 0$ and $F_w'(0) = f_w(0)$ exists. A regular link weight distribution is thus linear around zero. The factor $f_w(0)$ only scales all link weights, but does not influence the shortest path. The simplest distribution of the link weight w with a distinct different behavior for small values is the polynomial distribution

$$F_w(x) = x^\alpha 1_{x \in [0,1]} + 1_{x \in [1,\infty)}, \quad \alpha > 0, \quad (16.1)$$

The corresponding density is $f_w(x) = \alpha x^{\alpha-1} 1_{x \in [0,1]}$. The exponent

$$\alpha = \lim_{x \downarrow 0} \frac{\log F_w(x)}{\log x}$$

is called the *extreme value index* of the probability distribution of w and $\alpha = 1$ for regular distributions. By varying the exponent α over all non-negative real values, any extreme value index can be attained and a large class of corresponding SPTs, in short α -trees, can be generated.

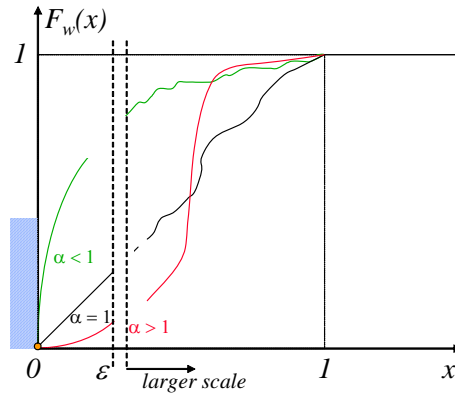


Fig. 16.1. A schematic drawing of the distribution of the link weights for the three different α -regimes. The shortest path problem is mainly sensitive to the small region around zero. The scaling invariant property of the shortest path allows us to divide all link weights by the largest possible such that $F_w(1) = 1$ for all link weight distributions.

Figure 16.1 illustrates schematically the probability distribution of the link weights around zero $(0, \epsilon]$, where $\epsilon > 0$ is an arbitrarily small, positive real number. The larger link weights in the network will hardly appear in a shortest path provided the network possesses enough links. These larger link weights are drawn in Fig. 16.1 from the double dotted line to the right. The nice advantage that only small link weights dominantly influence the property of the resulting shortest path tree implies that the remainder of the link weight distribution (denoted by the arrow with larger scale in Fig. 16.1) only plays a second order role. To some extent, it also explains the success of the simple SPT model based on the complete graph K_N with i.i.d. exponential link weights, which we derive in Section 16.2. A link weight structure effectively thins the complete graph K_N – any other graph is a subgraph of K_N – to the extent that a specific shortest path tree can be constructed.

Finally, we assume the independence of link weights, which we deem a reasonable assumption in large networks, such as the Internet with its many independent autonomous systems (ASs). Apart from the Section 16.7, we will mainly consider the case for $\alpha = 1$, which allows an exact analysis.

16.2 The shortest path tree in K_N with exponential link weights

16.2.1 The Markov discovery process

Let us consider the shortest path problem in the complete graph K_N , where each node in the graph is connected to each other node. The problem of finding the shortest path between two nodes A and B in K_N with exponentially distributed link weights with mean 1 can be rephrased in terms of a Markov discovery process. The discovery process evolves as a function of time and stops at a random time T when node B is found. The process is shown in Fig. 16.2.

The evolution of the discovery process can be described by a continuous-time Markov chain $X(t)$, where $X(t)$ denotes the number of discovered nodes at time t , because the characteristics of a Markov chain (Theorem 10.2.3) are based on the exponential distribution and the memoryless property. Of particular interest here is the property (see Section 3.4.1) that the minimum of n independent exponential variables each with parameter α_i is again an exponential variable with parameter $\sum_{i=1}^n \alpha_i$.

The discovery process starts at time $t = T_0$ with the source node A and for the initial distribution of the Markov chain, we have $\Pr[X(T_0) = 1] = 1$. The state space of the continuous Markov chain is the set S_N consisting of all positive integers (nodes) n with $n \leq N$. For the complete graph K_N , the

transition rates are given by

$$\lambda_n = n(N - n), \quad n \in S_N \quad (16.2)$$

Indeed, initially there is only the source node A with label² 0, hence $n = 1$. From this first node A precisely $N - 1$ new nodes can be reached in the complete graph K_N . Alternatively one can say that $N - 1$ nodes are competing with each other each with exponentially distributed strength to be discovered and the winner amongst them, say C with label 1, is the one reached in shortest time which corresponds to an exponential variable with rate $N - 1$.

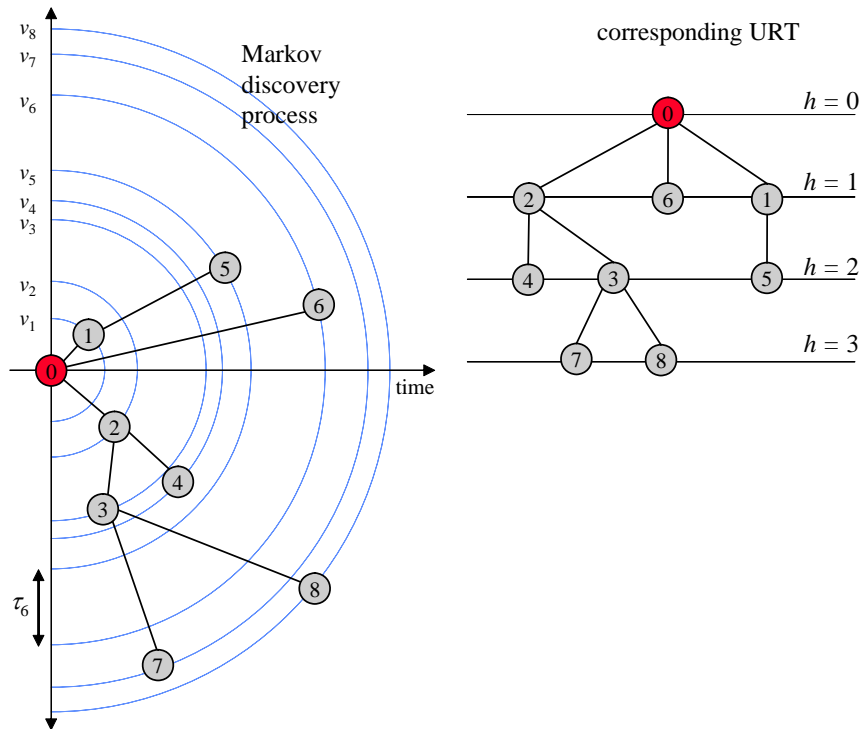


Fig. 16.2. On the left, the Markov discovery process as function of time in a graph with $N = 9$ nodes. The circles centered at the discovering node A with label 0 present equi-time lines and v_k is the discovering time of the k -th node, while $\tau_k = v_k - v_{k-1}$ is the k -th interattachment time. The set of discovered nodes redrawn per level are shown on the right, where a level gives the number of hops h from the source node A . The tree is a uniform recursive tree (URT).

² When continuous measures such as time and weight of a path are computed, the source node is most conveniently labeled by zero, whereas in counting processes, such as the number of hops of a path, the source node is labeled by one.

After having reached C from A at hitting time v_1 , two nodes $n = 2$ are found and the discovery process restarts from both A and C . Although at time v_1 we were already progressed a certain distance towards each of the $N - 2$ other, not yet discovered, nodes, the memoryless property of the exponential distribution tells us that the remaining distance to these $N - 2$ nodes is again exponentially distributed with the same parameter 1. Hence, this allows us to restart the process from A and C by erasing the previously partial distance to any other not yet discovered node as if we ignore that it were ever travelled. From the discovery time v_1 of the first node on, the discovery process has double strength to reach precisely $N - 2$ new nodes. Hence, the next winner, say D labeled by 2, is reached at v_2 in the minimum time out of $2(N - 2)$ traveling times. This node D has equal probability to be attached to A or C because of symmetry. When D is attached to A (the argument below holds similarly for attachment to C), symmetry appears to be broken, because D and C have only one link used, whereas A has already two links used. However, since we are interested in the shortest path problem and since the direct link from A to D is shorter than the path $A \rightarrow C \rightarrow D$, we exclude the latter in the discovery process, hereby establishing again the full symmetry in the Markov chain. This exclusion also means that the Markov chain maintains single paths from A to each newly discovered node and this path is also the shortest path. Hence, there are no cycles possible. Furthermore, similar to Dijkstra's shortest path algorithm, each newly reached node is withdrawn from the next competition round, which guarantees that the Markov chain eventually terminates. Besides terminating by extinction of all available nodes, after each transition when a new node is discovered, the Markov chain stops with probability equal to $\frac{1}{N-n}$, since each of the n already discovered nodes has precisely 1 possibility out of the remaining $N - n$ to reach B and only one of them is the discoverer. The stopping time T is defined as the infimum for $t \geq 0$ at which the destination node B is discovered. In summary, the described Markov discovery process, a pure birth process with birth rate $\lambda_n = n(N - n)$, models exactly the shortest path for all values of N .

16.2.2 The uniform recursive tree

A uniform recursive tree (URT) of size N is a random tree rooted at A . At each stage a new node is attached uniformly to one of the existing nodes until the total number of nodes is equal to N . The hopcount h_N (equivalent to the depth or distance) is the smallest number of links between the root A and a destination chosen uniformly from all nodes $\{1, 2, \dots, N\}$.

Denote by $\{X_N^{(k)}\}$ the k -th level set of a tree T , which is the set of nodes in the tree T at hopcount k from the root A in a graph with N nodes, and by $X_N^{(k)}$ the number of elements in the set $\{X_N^{(k)}\}$. Then, we have $X_N^{(0)} = 1$ because the zeroth level can only contain the root node A itself. For all $k > 0$, it holds that $0 \leq X_N^{(k)} \leq N - 1$ and that

$$\sum_{k=0}^{N-1} X_N^{(k)} = N \quad (16.3)$$

Another consequence of the definition is that, if $X_N^{(n)} = 0$ for some level $n < N - 1$, then all $X_N^{(j)} = 0$ for levels $j > n$. In such a case, the longest possible shortest path in the tree has a hopcount of n . The level set

$$L_N = \{1, X_N^{(1)}, X_N^{(2)}, \dots, X_N^{(N-1)}\}$$

of a tree T is defined as the set containing the number of nodes $X_N^{(k)}$ at each level k . An example of a URT organized per level k is drawn on the right in Fig. 16.2 and in Fig. 16.3. A basic theorem for URTs proved in van der Hofstad *et al.* (2002b), is the following:

Theorem 16.2.1 *Let $\{Y_N^{(k)}\}_{k,N \geq 0}$ and $\{Z_N^{(k)}\}_{k,N \geq 0}$ be two independent copies of the vector of level sets of two sequences of independent URTs. Then*

$$\{X_N^{(k)}\}_{k \geq 0} \stackrel{d}{=} \{Y_{N_1}^{(k-1)} + Z_{N-N_1}^{(k)}\}_{k \geq 0}, \quad (16.4)$$

where on the right-hand side the random variable N_1 is uniformly distributed over the set $\{1, 2, \dots, N - 1\}$.

Theorem 16.2.1 also implies that a subtree rooted at a direct child of the root is a URT. For example, in Fig. 16.3, the tree rooted at node 5 is a URT of size 13 as well as the original tree without the tree rooted at node 5. By applying Theorem 16.2.1 to the URT subtree, any subtree rooted at a member of a URT is also a URT.

An arbitrary URT U consisting of N nodes and with the root labeled by 1 can be represented as

$$U = (n_2 \longleftarrow 2)(n_3 \longleftarrow 3) \dots (n_N \longleftarrow N) \quad (16.5)$$

where $(n_j \longleftarrow j)$ means that the j -th node is attached to node $n_j \in [1, j - 1]$ and $n_2 = 1$. Hence, n_j is the predecessor of j and the predecessor relation is indicated by the arrow " \longleftarrow ". Moreover, n_j is a discrete uniform random variable on $[1, j - 1]$ and all n_2, n_3, \dots, n_N are independent.

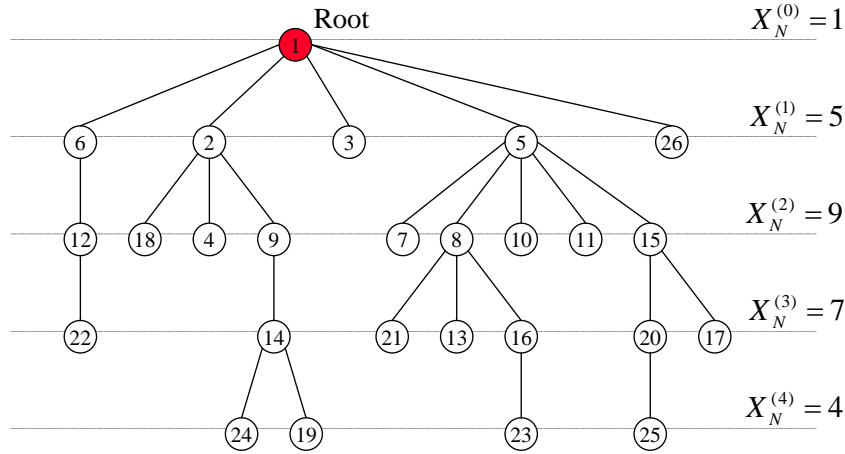


Fig. 16.3. An instance of a uniform recursive tree with $N = 26$ nodes organized per level $0 \leq k \leq 4$. The node number (inside the circle) indicates the order in which the nodes were attached to the tree.

Theorem 16.2.2 *The total number of URTs with N nodes is $(N - 1)!$*

Proof: (a) Let the nodes be labeled in the order of attachment to the URT and assign label 1 for the root. The URT growth law indicates that node 2 can only be attached in one way, node 3 in two ways, namely to node 1 and node 2 with equal probability. The k -th node can be attached in $k - 1$ possible nodes. Each of these possible constructions leads to a URT.

(b) By summing over all allowable configurations in (16.5), we obtain

$$\sum_{n_2=1}^1 \sum_{n_3=1}^2 \dots \sum_{n_N=1}^{N-1} 1 = (N - 1)!$$

and this proves the theorem. \square

In general, Cayley’s Theorem (Appendix B.1 art. 3) states that there are N^{N-2} labeled trees possible. The URT is a subset of the set of all possible labeled trees. Not all labeled trees are URTs, because the nodes that are further away from the root must have larger labels.

The shortest path tree from the source or root A to other nodes in the complete graph is the tree associated with the Markov discovery process, where the number of nodes $X(t)$ at time t is constructed as follows. Just as the discovery process, the associated tree starts at the root A . We now investigate the embedded Markov chain (Section 10.4) of the continuous-time discovery process. After each transition in the continuous-time Markov chain, $X(t) \rightarrow$

$X(t) + 1$, an edge of *unit* length is attached *randomly* to one of the n already discovered nodes in the associated tree because a new edge is equally likely to be attached to any of the n discovering nodes. Hence, the construction of the tree associated with the Markov discovery process and illustrated in Fig. 16.2 on the right demonstrates that the shortest path tree in the complete graph K_N with exponential link weights is an *uniform recursive tree*. This property of the shortest path tree in K_N with exponential link weights is an important motivation to study the URT. More generally, in van der Hofstad *et al.* (2001) we have proved that, for a fixed link density p and sufficiently large N , the shortest path tree in the *class RGU*, the class of random graphs $G_p(N)$ with exponential or uniformly distributed link weights, is a URT. Smythe and Mahmoud (1995) have reviewed a number of results on recursive trees that have appeared in the literature from the late 1960s up to 1995.

16.3 The hopcount h_N in the URT

16.3.1 Theory

The hopcount h_N from the root to an arbitrary chosen node in the URT equals the number of links or hops from the root to that node. We allow the arbitrary node to coincide with the root in which case $h_N = 0$.

Theorem 16.3.1 *The probability generating function of the hopcount in the URT with N nodes is*

$$\varphi_{h_N}(z) = E \left[z^{h_N} \right] = \frac{\Gamma(N + z)}{\Gamma(N + 1)\Gamma(z + 1)} \quad (16.6)$$

Proof: Since the number of nodes at hopcount k from the root (or at level k) is $X_N^{(k)}$, a node uniformly chosen out of N nodes in the URT has probability $\frac{E[X_N^{(k)}]}{N}$ of having hopcount k ,

$$\Pr[h_N = k] = \frac{E[X_N^{(k)}]}{N} \quad (16.7)$$

If the size of the URT grows from n to $n + 1$ nodes, each node at hopcount $k - 1$ from the root can generate a node at hopcount k with probability $1/n$. Hence, for $k \geq 1$,

$$E[X_N^{(k)}] = \sum_{n=k}^{N-1} \frac{E[X_n^{(k-1)}]}{n}$$

With (16.7), a recursion for $\Pr[h_N = k]$ follows for $k \geq 1$ as

$$\Pr[h_N = k] = \frac{1}{N} \sum_{n=k}^{N-1} \Pr[h_n = k - 1]$$

The generating function of h_N equals

$$\begin{aligned} \varphi_{h_N}(z) &= E[z^{h_N}] = \Pr[h_N = 0] + \sum_{k=1}^{N-1} \Pr[h_N = k] z^k \\ &= \frac{1}{N} + \frac{1}{N} \sum_{k=1}^{N-1} \sum_{n=k}^{N-1} \Pr[h_n = k - 1] z^k \\ &= \frac{1}{N} + \frac{1}{N} \sum_{n=1}^{N-1} \sum_{k=1}^n \Pr[h_n = k - 1] z^k = \frac{1}{N} + \frac{z}{N} \sum_{n=1}^{N-1} \varphi_{h_n}(z) \end{aligned}$$

Taking the difference between $(N + 1)\varphi_{h_{N+1}}(z)$ and $N\varphi_{h_N}(z)$ results in the recursion

$$(N + 1)\varphi_{h_{N+1}}(z) = (N + z)\varphi_{h_N}(z)$$

Iterating this recursion starting from $\varphi_{h_1}(z) = E[z^{h_1}] = E[z^0] = 1$ leads to (16.6). \square

Corollary 16.3.2 *The probability density function of the hopcount in the URT with N nodes is*

$$\Pr[h_N = k] = \frac{(-1)^{N-(k+1)} S_N^{(k+1)}}{N!} \tag{16.8}$$

Proof: The probability generating function $\varphi_{h_N}(z)$ in (16.6) is also the generating function of the Stirling numbers $S_N^{(k)}$ of the first kind (Abramowitz and Stegun, 1968, 24.1.3) such that the probability that a uniformly chosen node in the URT has hopcount k equals (16.8). \square

The explicit form of the generating function shows that the average hopcount h_N in a URT of size N equals

$$\begin{aligned} E[h_N] &= \varphi'_{h_N}(1) = \left. \frac{d}{dz} \log \varphi_{h_N}(z) \right|_{z=1} = \sum_{l=2}^N \frac{1}{l} \\ &= \psi(N + 1) + \gamma - 1 \end{aligned} \tag{16.9}$$

where $\psi(z) = \frac{\Gamma'(z)}{\Gamma(z)}$ is the digamma function (Abramowitz and Stegun, 1968, Section 6.3) and the Euler constant is $\gamma = 0.57721\dots$. Similarly,

the variance (2.27) follows from the logarithm of the generating function $L_{h_N}(z) = \log \Gamma(N+z) - \log \Gamma(N+1) - \log \Gamma(z+1)$ as

$$\begin{aligned} \text{Var}[h_N] &= \psi'(N+1) - \psi'(2) + \psi(N+1) + \gamma - 1 \\ &= \psi(N+1) + \gamma - \frac{\pi^2}{6} + \psi'(N+1) \end{aligned}$$

Using the asymptotic formulae for the digamma function leads to

$$E[h_N] = \log N + \gamma - 1 + O\left(\frac{1}{N}\right) \quad (16.10)$$

$$\text{Var}[h_N] = \log N + \gamma - \frac{\pi^2}{6} + O\left(\frac{1}{N}\right) \quad (16.11)$$

For large N , we apply an asymptotic formula of the Gamma function (Abramowitz and Stegun, 1968, Section 6.1.47) to the generating function of the hopcount (16.6),

$$\varphi_{h_N}(z) = \frac{N^{z-1}}{\Gamma(z+1)} \left(1 + O\left(\frac{1}{N}\right)\right)$$

Introducing the Taylor series of $\frac{1}{\Gamma(z)} = \sum_{k=1}^{\infty} c_k z^k$ where the coefficients c_k are listed in Abramowitz and Stegun (1968, Section 6.1.34), we obtain with $N^z = e^{z \log N}$,

$$\begin{aligned} \varphi_{h_N}(z) &= \frac{1}{N} \sum_{k=1}^{\infty} c_k z^{k-1} \sum_{k=0}^{\infty} \frac{\log^k N}{k!} z^k \left(1 + O\left(\frac{1}{N}\right)\right) \\ &= \frac{1 + O\left(\frac{1}{N}\right)}{N} \sum_{k=0}^{\infty} \sum_{m=0}^k c_{m+1} \frac{\log^{k-m} N}{(k-m)!} z^k \end{aligned}$$

With the definition (2.18) of the probability generating function, we conclude that the asymptotic form of the probability density function (16.8) of the hopcount in the URT is

$$\text{Pr}[h_N = k] = \frac{1 + O\left(\frac{1}{N}\right)}{N} \sum_{m=0}^k c_{m+1} \frac{\log^{k-m} N}{(k-m)!} \quad (16.12)$$

Since the coefficients c_k are rapidly decreasing, approximating the sum in (16.12) by its first term ($m=0$) yields to first order in N ,

$$\text{Pr}[h_N = k] \sim \frac{(\log N)^k}{Nk!} \quad (16.13)$$

which is recognized as a Poisson distribution (3.9) with mean $\log N$. Hence, for large N and to first order, the average and variance of the hopcount in

the URT are approximately $E[h_N] \sim \text{Var}[h_N] \sim \log N$. The accuracy of the Poisson approximation can be estimated by comparison with the average (16.10) and the variance (16.11) found above up to second order in N . For example, if the URT has $N = 10^4$ nodes, the Poisson approximation yields $E[h_N] = \text{Var}[h_N] = 9.21034$, while the average (16.10) is $E[h_N] = 8.78756$ accurate up to 10^{-4} and the variance (16.11) is $\text{Var}[h_N] = 8.14262$. The exact results are $E[h_N] = 8.78761$ and $\text{Var}[h_N] = 8.14277$.

16.3.2 Application of the URT to the hopcount in the Internet

In trace-route measurements explained in Van Mieghem (2004a), we are interested in the hopcount H_N denoted with capital H , which equals h_N in the URT excluding the event $h_N = 0$. In other words, the source and the destination are different nodes in the graph. Since from (16.8) $\Pr[h_N = 0] = \frac{(-1)^{N-1} S_N^{(1)}}{N!} = \frac{1}{N}$ we obtain, for $1 \leq k \leq N-1$,

$$\begin{aligned} \Pr[H_N = k] &= \Pr[h_N = k | h_N \neq 0] = \frac{\Pr[h_N = k, h_N \neq 0]}{\Pr[h_N \neq 0]} \\ &= \frac{N}{N-1} \Pr[h_N = k] \end{aligned}$$

Using (16.8), we find

$$\Pr[H_N = k] = \frac{N}{N-1} \frac{(-1)^{N-(k+1)} S_N^{(k+1)}}{N!} \quad (16.14)$$

with corresponding generating function,

$$\begin{aligned} \varphi_{H_N}(z) &= \sum_{k=1}^{N-1} \Pr[H_N = k] z^k \\ &= \frac{N}{N-1} \sum_{k=0}^{N-1} \Pr[h_N = k] z^k - \frac{N}{N-1} \Pr[h_N = 0] \\ &= \frac{N}{N-1} \left(\varphi_{h_N}(z) - \frac{1}{N} \right) \end{aligned}$$

The average hopcount $E[H_N] = E[h_N | h_N \neq 0]$ is

$$E[H_N] = \frac{N}{N-1} \sum_{l=2}^{N-1} \frac{1}{l} \quad (16.15)$$

Hence, for large N and in practice, we find that

$$\Pr[H_N = k] = \Pr[h_N = k] + O\left(\frac{1}{N}\right)$$

which allows us to use the previously derived expressions (16.12), (16.10) and (16.11).

The histogram of the number of traversed routers in the Internet measured between two arbitrary communicating parties seems reasonably well modeled by the pdf (16.12). Figure 16.4 shows both the histogram of the hopcount deduced from paths in the Internet measured via the trace-route utility and the fit with (16.12). From the fit, we find a rather high number of nodes

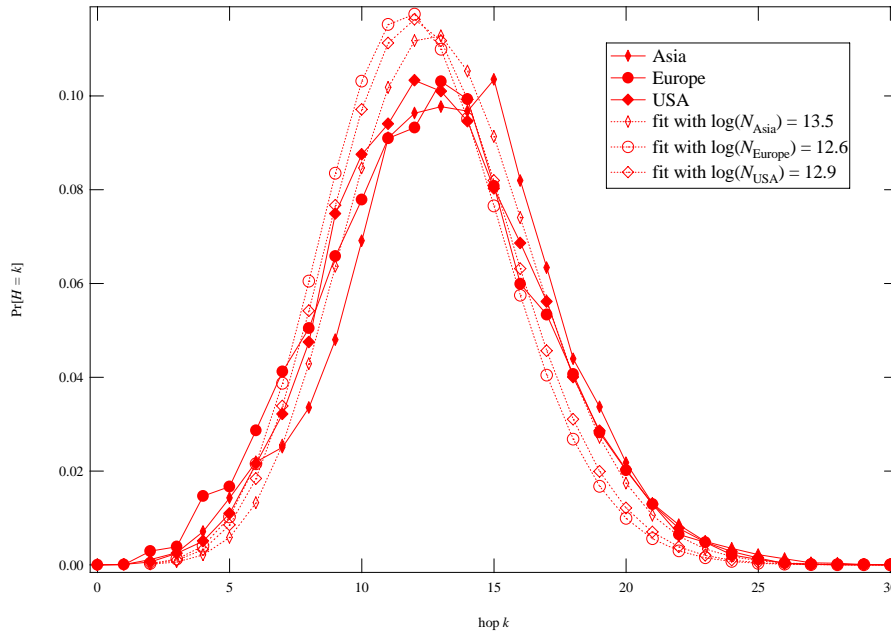


Fig. 16.4. The histograms of the hopcount derived from the trace-route measurement in three continents from CAIDA in 2004 are fitted by the pdf (16.12) of the hopcount in the URT.

$e^{12.6} \approx 3 \cdot 10^5 \leq N \leq e^{13.5} \approx 7 \cdot 10^5$, which points to the approximate nature of modeling the Internet hopcount by that deduced from a URT. The relation between Internet measurements and the properties of the URT is further analyzed in a series of articles (Van Mieghem *et al.*, 2000; van der Hofstad *et al.*, 2001; Van Mieghem *et al.*, 2001b; Janic *et al.*, 2002; van der Hofstad *et al.*, 2002b). At the time of writing, an accurate model of the hopcount in the Internet is not available.

16.4 The weight of the shortest path

The weight – sometimes also called the length – of the shortest path is defined as the sum of the link weights that constitute the shortest path. In Section 16.2.1, the shortest path tree in the complete graph with exponential link weights was shown to be a URT. In this section, we confine ourselves to the same type of graph and require that the source node A (or root) is different from the destination node B .

By Theorem 10.2.3 of a continuous-time Markov chain, the discovery time of the k -th node from node A equals $v_k = \sum_{n=1}^k \tau_n$, where $\tau_1, \tau_2, \dots, \tau_k$ are independent, exponentially distributed random variables with parameter $\lambda_n = n(N-n)$ with $1 \leq n \leq k$. We call τ_j the interattachment time between the discovery or the attachment to the URT of the $j-1$ -th and j -th node in the graph. The Laplace transform of v_k is

$$E[e^{-zv_k}] = \int_0^\infty e^{-zt} \frac{d}{dt} \Pr[v_k \leq t]$$

For a sum of independent exponential random variables, using the probability generating function (3.16), we have

$$E[e^{-zv_k}] = E\left[\exp\left(-z \sum_{n=1}^k \tau_n\right)\right] = \prod_{n=1}^k E[e^{-z\tau_n}] = \prod_{n=1}^k \frac{n(N-n)}{z+n(N-n)} \tag{16.16}$$

The probability generating function³ $\varphi_{W_N}(z) = E[e^{-zW_N}]$ of the weight W_N of the shortest path equals

$$\begin{aligned} \varphi_{W_N}(z) &= \sum_{k=1}^{N-1} E[e^{-zv_k}] \Pr[B \text{ is } k\text{-th attached node in URT}] \\ &= \frac{1}{N-1} \sum_{k=1}^{N-1} \prod_{n=1}^k \frac{n(N-n)}{z+n(N-n)} \end{aligned} \tag{16.17}$$

because any node apart from the root A but including the destination node B has equal probability to be the k -th attached node.

The average weight is

$$E[W_N] = - \left. \frac{d\varphi_{W_N}(z)}{dz} \right|_{z=0} = - \frac{1}{N-1} \sum_{k=1}^{N-1} \left. \frac{d}{dz} \prod_{n=1}^k \frac{n(N-n)}{z+n(N-n)} \right|_{z=0}$$

³ If the link weights have mean $\frac{1}{a}$ (instead of 1), then W_N is multiplied by a as explained in Sections 16.2.1 and 3.4.1. The weight of the scaled shortest path $W_{N,a}$ has pgf

$$\varphi_{W_{N,a}}(z) = E[e^{-zaW_N}] = \varphi_{W_N}(az)$$

Using the logarithmic derivative of the product,

$$\begin{aligned} \left. \frac{d}{dz} \prod_{n=1}^k \frac{n(N-n)}{z+n(N-n)} \right|_{z=0} &= \prod_{n=1}^k \frac{n(N-n)}{z+n(N-n)} \frac{d}{dz} \left(\sum_{n=1}^k \log \frac{n(N-n)}{z+n(N-n)} \right) \Big|_{z=0} \\ &= - \sum_{n=1}^k \frac{1}{n(N-n)} \end{aligned}$$

gives

$$\begin{aligned} E[W_N] &= \frac{1}{N-1} \sum_{k=1}^{N-1} \sum_{n=1}^k \frac{1}{n(N-n)} = \frac{1}{N-1} \sum_{n=1}^{N-1} \frac{1}{n(N-n)} \sum_{k=n}^{N-1} 1 \\ &= \frac{1}{N-1} \sum_{n=1}^{N-1} \frac{N-n}{n(N-n)} \end{aligned}$$

The average weight is

$$E[W_N] = \frac{1}{N-1} \sum_{n=1}^{N-1} \frac{1}{n} = \frac{\psi(N) + \gamma}{N-1} \quad (16.18)$$

For large N ,

$$E[W_N] = \frac{\log N + \gamma}{N} + O\left(\frac{1}{N^2}\right)$$

Similarly, the variance is computed (see problem (ii) in Section 16.9) as,

$$\text{Var}[W_N] = \frac{3}{N(N-1)} \sum_{n=1}^{N-1} \frac{1}{n^2} - \frac{\left(\sum_{n=1}^{N-1} \frac{1}{n}\right)^2}{(N-1)^2 N} \quad (16.19)$$

and for large N ,

$$\text{Var}[W_N] = \frac{\pi^2}{2N^2} + O\left(\frac{\log^2 N}{N^3}\right)$$

By inverse Laplace transform of (16.17), the distribution $\Pr[W_N \leq t]$ can be computed. The asymptotic distribution for the weight of the shortest path is (see problem (iii) in Section 16.9)

$$\lim_{N \rightarrow \infty} \Pr[NW_N - \log N \leq x] = e^{-e^{-x}} \quad (16.20)$$

A related but slightly more complex analysis is presented in Section 16.5.1 where we study the flooding time. The interest of such an asymptotic analysis is that it often leads to tractable solutions that are physically more appealing to interpret. Moreover, it turns out that results for finite, not too small N are reasonably approximated by the asymptotic law.

Since W_N equals the sum of the link weights of the shortest path from the root to an arbitrary node and since $H_N = h_N/h_N > 0$ is the number of links in that shortest path (where the arbitrary destination node is different from the root), one may wonder whether there is a relation between them. Although the shortest path has precisely H_N hops, the destination node of that path is not necessarily the H_N -th attached node to the URT grown at the root. The destination node cannot be discovered sooner than the H_N -th attached node, otherwise the hopcount of the shortest path would be shorter than H_N . Hence, the destination node is the k -th discovered node and attached to the URT somewhere in between the $H_N - 1$ -th and the last attached node. Thus, $k \in [H_N, N - 1]$. If $k = H_N$, then all previously discovered nodes belong to the shortest path and the j -th attached node in the URT is linked to the $j - 1$ -th, for all $j \leq k$. If $k > H_N$, precisely $k - H_N$ of the attached nodes do not belong to the shortest path. Hence $W_N = W_k$ provided $k - H_N$ nodes in the URT discovered so far do not belong to the path and precisely H_N do. The latter condition requires the determination of all structurally favorable possibilities which is rather complex.

Curiously, the probability that the shortest path consists of the direct link between source and destination is, with (16.14), (16.18) and $S_N^{(2)} = (-1)^N (N - 1)! \sum_{k=1}^{N-1} \frac{1}{k}$,

$$\Pr[H_N = 1] = \frac{1}{N - 1} \sum_{k=1}^{N-1} \frac{1}{k} = E[W_N]$$

16.5 The flooding time T_N

The most commonly used process that informs each node (router) about changes in the network topology is called *flooding*: the source node initiates the flooding process by sending the packet with topology information to all adjacent neighbors and every router forwards the packet on all interfaces except for the incoming one and duplicate packets are discarded. Flooding is particularly simple and robust since it progresses, in fact, along all possible paths from the emitting node to the receiving node. Hence, a flooded packet reaches a node in the network in the shortest possible time (if overheads in routers are ignored). Therefore, an interesting problem lies in the determination of the flooding time T_N , which is the minimum time needed to inform all nodes in a network with N nodes. Only after a time T_N , all topology databases at each router in the network are again synchronized, i.e. all routers possess the same topology information. The flooding time T_N