Independence
additional examples

Suppose a biased coin comes up heads with probability p, *independent* of other flips

P(n heads in n flips) $= p^n$

P(n tails in n flips) $= (1-p)^n$

P(exactly k heads in n flips) $= \binom{n}{k} p^k (1-p)^{n-k}$

Aside: note that the probability of *some* number of heads = $\sum_k \binom{n}{k} p^k (1-p)^{n-k} = (p + (1-p))^n = 1$

as it should, by the binomial theorem.

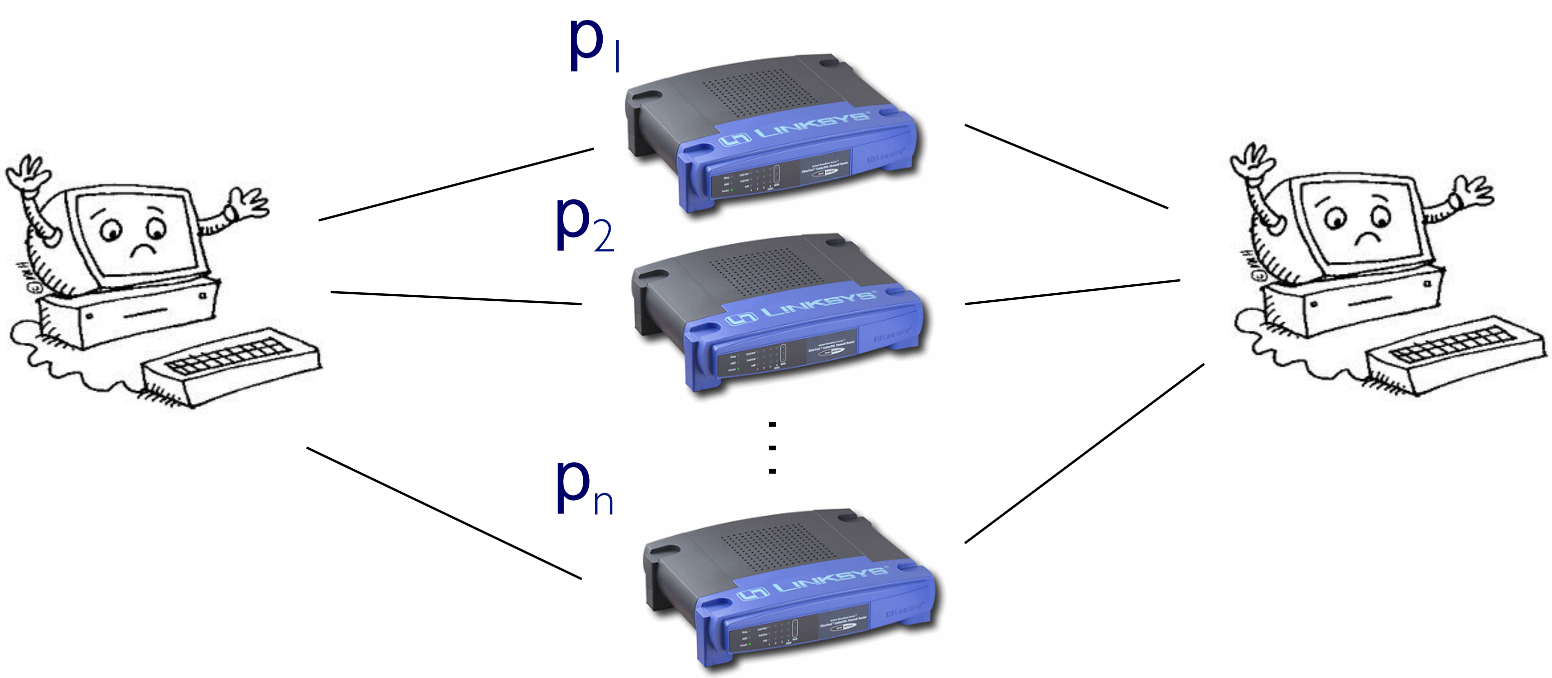
Suppose a biased coin comes up heads with probability p, *independent* of other flips

P(exactly k heads in n flips) $= \binom{n}{k} p^k (1-p)^{n-k}$

Note when $p=1/2$, this is the same result we would have gotten by considering n flips in the "equally likely outcomes" scenario. But $p \neq 1/2$ makes that inapplicable. Instead, the *independence* assumption allows us to conveniently assign a probability to each of the 2^n outcomes, e.g.:

Pr(HHTHTTT) = $p^2(1-p)p(1-p)^3 = p^{\#H}(1-p)^{\#T}$

Consider the following parallel network



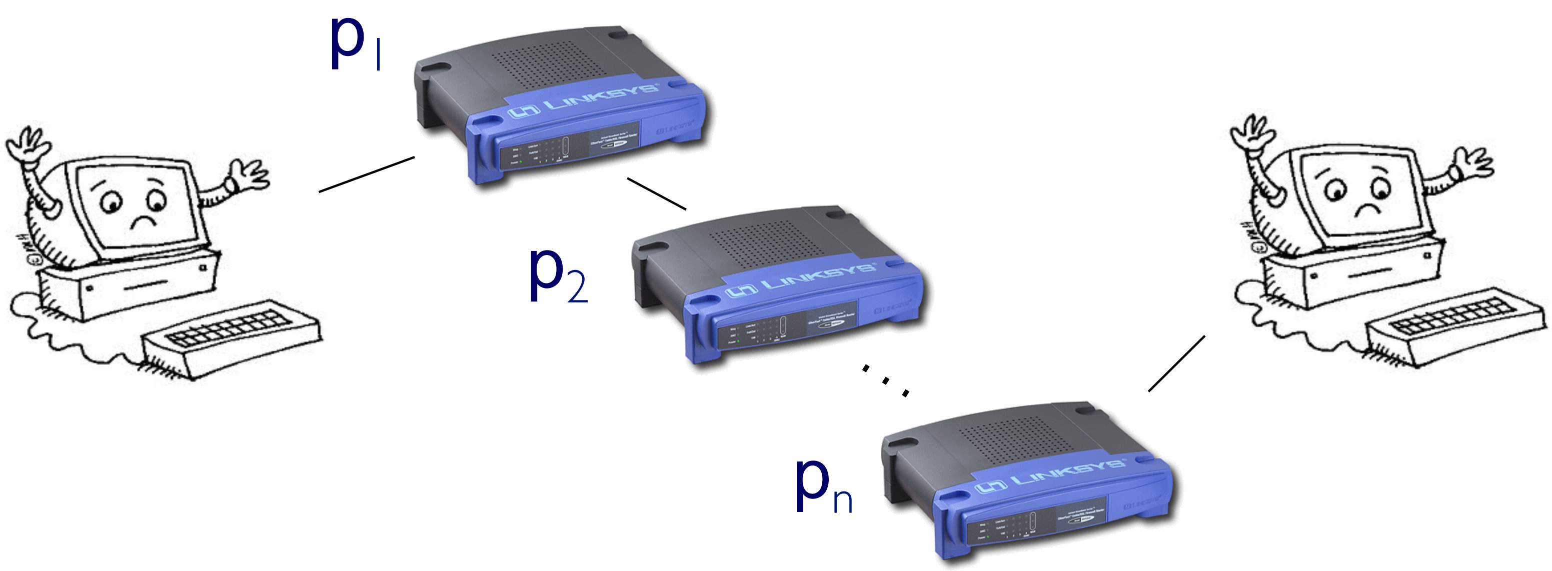
n routers, i^{th} has probability p_i of failing, independently

P(there is functional path) = 1 – P(all routers fail)

$$= 1 - p_1 p_2 \cdots p_n$$

network failure

Contrast: a *series* network



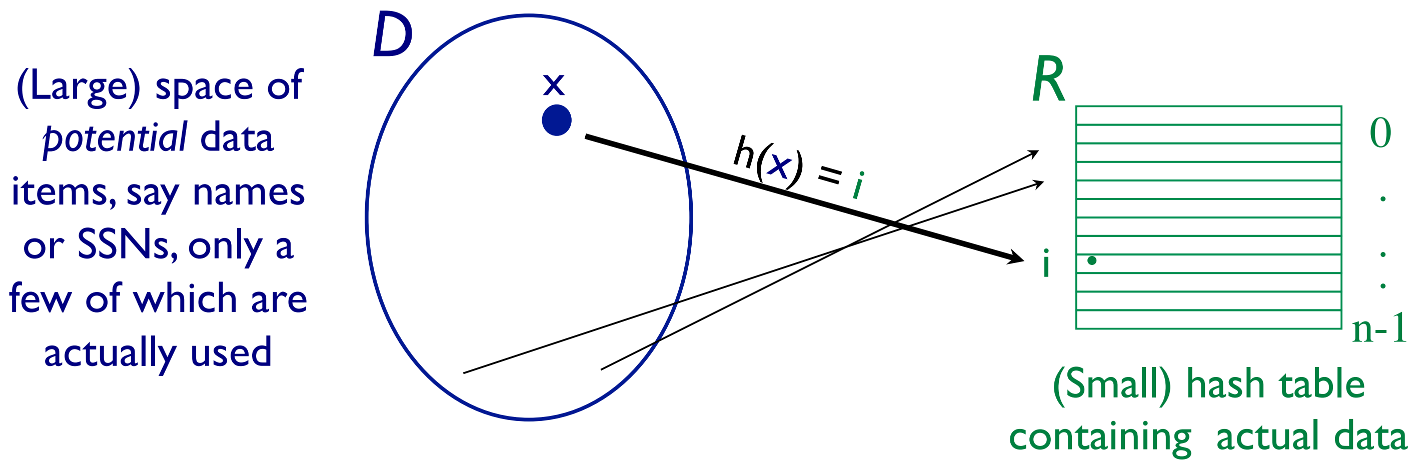
n routers, i^{th} has probability p_i of failing, independently

P(there is functional path) =

P(*no* routers fail)

$$= (1 - p_1)(1 - p_2) \cdots (1 - p_n)$$

A data structure problem: *fast* access to *small* subset of data drawn from a *large* space.



A solution: *hash function* *h*:D→{0,...,n-1} crunches/scrambles names from large space into small one. E.g., if x is integer:

$h(x) = x \bmod n$

Good hash functions *approximately* randomize placement.

m strings hashed (uniformly) into a table with n buckets

Each string hashed is an *independent* trial

E = at least one string hashed to first bucket

What is P(E) ?

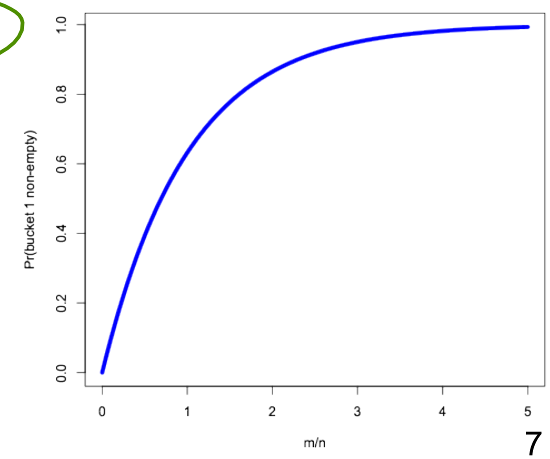
Solution:

F_i = string i *not* hashed into first bucket (i=1,2,…,m)

$P(F_i) = 1 - 1/n = (n-1)/n$ for all i=1,2,…,m

Event $(F_1 F_2 \dots F_m)$ = no strings hashed to first bucket

$$
\begin{aligned}
P(E) &= 1 - P(F_1 F_2 \cdots F_m) \\
&= 1 - P(F_1) P(F_2) \cdots P(F_m) \quad \text{indp} \\
&= 1 - ((n-1)/n)^m \\
&\approx 1 - \exp(-m/n)
\end{aligned}
$$



m strings hashed (non-uniformly) to table w/ n buckets

Each string hashed is an *independent* trial, with probability p_i of getting hashed to bucket i

E = At least 1 of first k buckets gets ≥ 1 string

What is P(E) ?

Solution:

F_i = at least one string hashed into i-th bucket

$$P(E) = P(F_1 \cup \cdots \cup F_k) = 1 - P((F_1 \cup \cdots \cup F_k)^c)$$

$$= 1 - P(F_1^c F_2^c \ldots F_k^c)$$

$$= 1 - P(\text{no strings hashed to buckets 1 to k})$$

$$= 1 - (1 - p_1 - p_2 - \cdots - p_k)^m$$