# Netchange

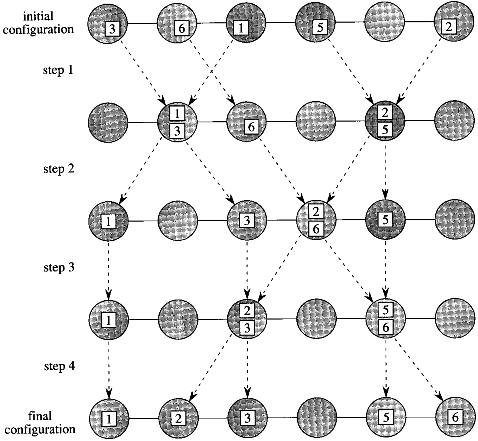
```
var Neigh_u : set of nodes ;      (* The neighbors of u *)
    D_u     : array of 0.. N ;    (* D_u[v] estimates d(u, v) *)
    Nb_u    : array of nodes ;    (* Nb_u[v] is preferred neighbor for v *)
    ndis_u  : array of 0.. N ;    (* ndis_u[w, v] estimates d(w, v) *)

Initialization:
    begin forall w ∈ Neigh_u, v ∈ V do ndis_u[w, v] := N ;
          forall v ∈ V do
                begin D_u[v] := N ; Nb_u[v] := udef end ;
          D_u[u] := 0 ; Nb_u[u] := local ;
          forall w ∈ Neigh_u do send ⟨ mydist, u, 0 ⟩ to w
    end

Procedure Recompute (v):
    begin if v = u
             then begin D_u[v] := 0 ; Nb_u[v] := local end
             else begin (* Estimate distance to v *)
                        d := 1 + min{ndis_u[w, v] : w ∈ Neigh_u} ;
                        if d < N then
                           begin D_u[v] := d ;
                                 Nb_u[v] := w with 1 + ndis_u[w, v] = d
                           end
                        else begin D_u[v] := N ; Nb_u[v] := udef end
                  end ;
          if D_u[v] has changed then
             forall x ∈ Neigh_u do send ⟨ mydist, v, D_u[v] ⟩ to x
    end

Processing a ⟨ mydist, v, d ⟩ message from neighbor w:
    { A ⟨ mydist, v, d ⟩ is at the head of Q_wu }
    begin receive ⟨ mydist, v, d ⟩ from w ;
          ndis_u[w, v] := d ; Recompute (v)
    end

Upon failure of channel uw:
    begin receive ⟨ fail, w ⟩ ; Neigh_u := Neigh_u \ {w} ;
          forall v ∈ V do Recompute (v)
    end

Upon repair of channel uw:
    begin receive ⟨ repair, w ⟩ ; Neigh_u := Neigh_u ∪ {w} ;
          forall v ∈ V do
                begin ndis_u[w, v] := N ;
                      send ⟨ mydist, v, D_u[v] ⟩ to w
                end
    end
```

## korektnosť

lexikograficky klesá hodnota $[t_0, t_1, \ldots, t_N]$
kde $t_i$ je počet správ $\langle$**mydist**, $i\rangle +$ počet dvojíc $u, v$ kde $D_u[v] = i$

# packet routing

- synchrónny režim
- vrcholy majú pakety (uložené v bufferoch)
- v jednom kroku po jednej linke ide max. jeden paket
- algoritmus = odchádzajúce linky + priorita bufferov
- celkový čas



# packet routing na mriežke $\sqrt{N} \times \sqrt{N}$

## vstup

Každý vrchol má 1 paket, do každého smeruje 1 paket (permutation routing)

## algoritmus

Najprv riadok, potom stĺpec. Prednosť má ten s najdlhšou cestou.

## analýza: stačí $2\sqrt{N} - 2$ krokov

- po $\sqrt{N} - 1$ krokoch je každý v správnom stĺpci (nebrzdia sa)
- routovanie v stĺpci ide v $\sqrt{N} - 1$ krokoch
  - pre každé $i$ platí: po $N - 1$ krokoch sú koncové pakety na koncových miestach
  - dôvod: zdržujú sa iba navzájom

veľkosť buffra v najhoršom prípade: $2/3\sqrt{N} - 3$

# veľkosť buffra: priemerný prípad I

## setting

Každý vrchol má jeden paket s **náhodným cieľom**

max. veľkosť buffra $\approx$ počet zahnutí vo vrchole

psť, že aspoň $r$ zahne $\leq \binom{\sqrt{N}}{r}\left(\frac{1}{\sqrt{N}}\right)^r < \left(\frac{e}{r}\right)^r$

pre $r = \frac{e \log N}{\log \log N}$ je psť $o(N^{-2})$

# veľkosť buffra: priemerný prípad II

wide-channel: nepredbiehajú sa

## lema

psť, že vo wch prejde aspoň $\alpha\Delta/2$ paketov cez hranu $e$ počas $t+1, t+2, \ldots, t+\Delta$ je najviac $e^{(\alpha-1-\alpha\ln\alpha)\Delta/2}$

očakávaný počet paketov na hrane $(i,j) \mapsto (i+1,j)$ je

$$\frac{2i(\sqrt{N}-i)\Delta}{N} \leq \frac{\Delta}{2}$$

chceme ukázať, že s veľkou psťou ich neprejde príliš viac

# Černovov odhad

**lema**

Majme $n$ nezávislých Bernoulihho náh. prem. $X_1, \dots, X_n$, pričom $Pr[X_k = 1] \leq P_k$. Potom

$$Pr[X \geq \beta P] \leq e^{(1-\frac{1}{\beta}-\ln\beta)\beta P}$$

kde $X = \sum X_i$, $P = \sum P_i$

$$E[e^{\lambda X_k}] \leq 1 + P_k(e^{\lambda} - 1) \leq e^{P_k(e^{\lambda}-1)}$$

$$E[e^{\lambda X}] \leq e^{P(e^{\lambda}-1)}$$

$$P[e^{\lambda X} \geq e^{\lambda\beta P}] \leq \frac{E[e^{\lambda X}]}{e^{\lambda\beta P}} \leq e^{P(e^{\lambda}-1)-\lambda\beta P}$$

# veľkosť buffra: priemerný prípad II

## lema

Majme $n$ nezávislých Bernoulihho náh. prem. $X_1, \ldots, X_n$, pričom $Pr[X_k = 1] \leq P_k$. Potom

$$Pr[X \geq \beta P] \leq e^{(1 - \frac{1}{\beta} - \ln \beta)\beta P}$$

kde $X = \sum X_i$, $P = \sum P_i$

## lema

psť, že vo wch prejde aspoň $\alpha\Delta/2$ paketov cez hranu $e$ počas $t+1, t+2, \ldots, t+\Delta$ je najviac $e^{(\alpha - 1 - \alpha \ln \alpha)\Delta/2}$

očakávaný počet paketov na hrane $(i,j) \mapsto (i+1, j)$ je

$$\frac{2i(\sqrt{N} - i)\Delta}{N} \leq \frac{\Delta}{2}$$

chceme ukázať, že s veľkou psťou ich neprejde príliš viac

$n = 2i\Delta$, $P_k = \frac{\sqrt{N} - i}{N}$, $P = \frac{2i(\sqrt{N} - i)\Delta}{N}$, $\beta = \frac{\alpha N}{4i(\sqrt{N} - i)}$

# veľkosť buffra: priemerný prípad II

## lema

ak je paket vo vzd. $d$ od hrany $e$ v čase $T$, a $p$ prejde cez $e$ v čase $T + d + \delta$, potom v každom kroku $[T + d, T + d + \delta]$ prejde paket cez $e$

## dosledok

ak paket prejde cez $e$ v čase $T$ vo wch, a prejde cez $e$ v čase $T + \delta$ v št. , tak v každom kroku $[T, T + \delta]$ prejde paket

## lema

ak počas $[T + 1, T + \Delta]$ prejde cez $e$ $x$ paketov v št., tak pre nejaké $t$ prejde $x + t$ paketov cez $e$ v čase $[T + 1 = t, T + \Delta]$ vo wch.

## lema

psť, že cez $e$ prejde viac ako $\alpha\Delta/2$ paketov počas konkrétneho okna $\Delta$ krokov je najviac $O(e^{(\alpha-1-\alpha \ln \alpha)\Delta/2})$

## dosledok

s psťou $1 - O(\frac{1}{N})$ neprejde po $e$ viac ako $c \log N$ paketov v posebe idúcich krokoch, kde $c = \frac{5 \ln 2}{2 \ln 2 - 1} < 9$.

# dynamické routovanie

## model

V každom kroku sa v každom vrchole s psťou $\lambda$ narodí paket s náhodným cieľom.

## stabilita

Pre $\lambda \geq 4/\sqrt{N}$ je systém nestabilný

## veta

Ak je $\lambda \leq 0.99\frac{4}{\sqrt{N}}$, tak psť zdržania konkrétneho paketu o $\Delta$ krokov je $e^{-O(\Delta)}$.
W.h.p. stačí buffer $O(1 + \frac{\log T}{\log N})$.

# hierarchické routovanie

cieľ: minimalizovať počet rozhodnutí

**veta**

Pre sieť s $N$ vrcholmi stačí $O(\sqrt{N})$ rozhodnutí pri použití 3 farieb.

$s$-klastre:

- každý je súvislý, pokrývajú všetky vrcholy
- každý obsahuje aspoň $s$ vrcholov a má polomer najviac $2s$

kostra spájajúca centrá klastrov: $m$ listov $\Rightarrow m - 2$ vetvení

**veta**

Pre sieť s $N$ a pre $f \leq \log N$ stačí $O(f \cdot N^{1/f})$ rozhodnutí a $2f + 1$ farieb

po $i$ klastrovaniach s parametrom $s$: $m_i$ listov, max. $m_i - 2$ vetvení $\Rightarrow m_{i+1} = m_i(2/s)$
$m_f + fs$ rozhodnutí
$s \approx 2N^{1/f}$

# cvičenia

mriežka $\sqrt{n} \times \sqrt{n}$, prednosť má hocikto; ukážte, že v najhoršom prípade treba viac ako $2\sqrt{n}$ krokov, ale stačí $O(\sqrt{n})$

mriežka $\sqrt{n} \times \sqrt{n}$, v každom vrchole správa do náhodného. Ukážte, že w.h.p. do žiadnoho vrchola nesmeruje viac ako $3 \log n / \log \log n$ správ.

majme cestu z $n$ procesorov, každý chce routovať práve dva pakety (červený a modrý), pričom červené aj zelené tvoria permutáciu. ukážte, že stačí $n$ krokov