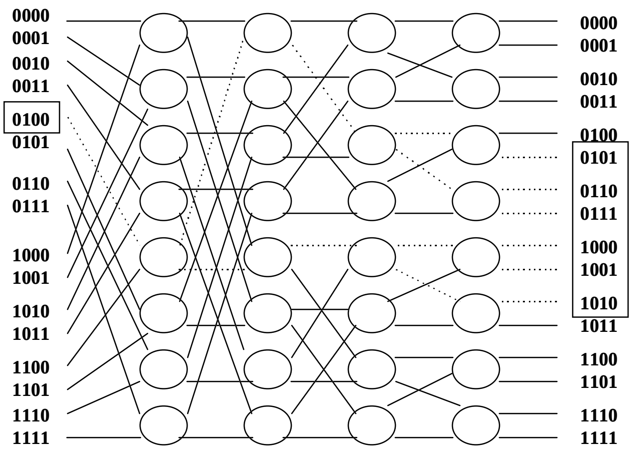
Kytkentäfunktioiden monimutkaisuuden alaraja, Copy-funktio, Itsereitittävyys

Kopioiden muodostus binääriverkolla

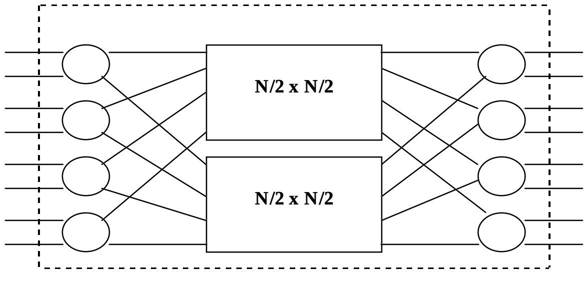
- ✔ **Kopiointi toteutetaan verkkoa edeltävällä levityspuulla, jonka jälkeen on ‘järjestelijä‘.**
- ✔ **Kopioinnissa tarvitaan tieto tulojohdosta ja kopioiden lukumäärästä.**
- ✔ **Kopioinnissa verrataan kopiovälin binääriarvoja ja suoritetaan kopiointi- / reitityspäätös arvojen perusteella.**

Banyan-pohjainen levityspuu



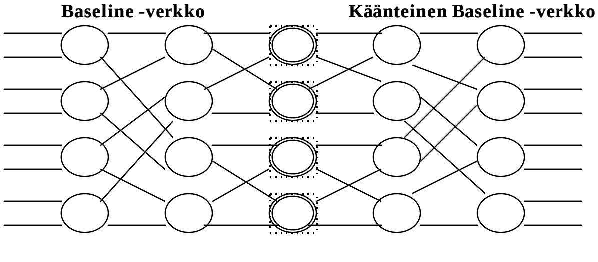
Esim N=8

1)



2)

BENES -verkko



Benes -verkko

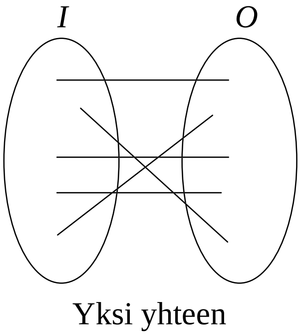
- ✔ **Benes -verkko on rekursiivisesti 2x2 -kytkimistä muodostettu *uudelleen järjesteltävä* verkko.**
- ✔ **1-puoliverkko tunnetaan nimellä baseline -verkko**
- ✔ **2-puoliverkko tunnetaan nimellä käänteinen baseline -verkko**
- ✔ **Benes -verkossa on ($2\log_2 N$-1) porrasta.**
- ✔ **Kytkentä pisteiden lukumäärä on**

$$4(N/2)(2\log_2 N-1) = 4N\log_2 N-2N \sim 4N\log_2 N$$

Kytkentäkonfiguraatiot

C **-** Kytkentöjen kuvaus

Pt-to-pt kytkentäfunktio



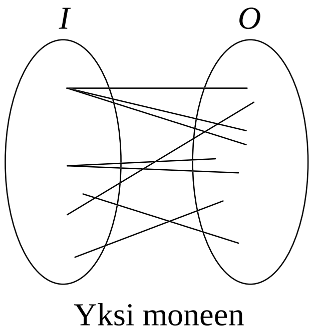
Yksi yhteen

$C = \{(i,o) | \ i \in I, o \in O\}$

$(i,o) \in C$ ja $(i,o') \in C \Rightarrow o = o'$

$(i,o) \in C$ ja $(i',o) \in C \Rightarrow i = i'$

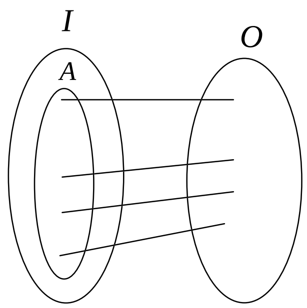
Multicastfunktio



Yksi moneen

Tavoite: Selvittää kytkentöjen kokonaismääräkentässä ja kentän vähimmäismonimutkaisuus.

Keskitinfunktio

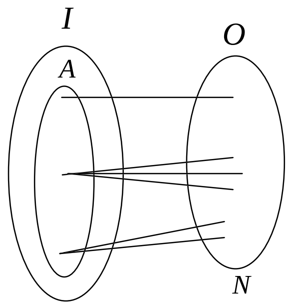


$C = \{(i,o) | \; i \in A \subset I, o \in O\}$

$(i,o) \in C$ ja $(i,o') \in C \Rightarrow o = o'$

$(i,o) \in C$ ja $(i',o) \in C \Rightarrow i = i'$

Copy-funktio



$C = \{(i,n_i) | \; i \in I, \Sigma n_i = N\}$

Lähtöjen n_i järjestyksellä ja identiteetillä ei väliä (output unspecific).

Kentän kombinatorinen monimutkaisuus

ζ(G) - Graafin G toteuttamien legitiimien erilaisten kytkentäkonfiguraatioiden *C* lukumäärän kaksi-kantainen logaritmi.

R **- kentän kytkinten lukumäärä.**

2^R - kentän, jossa on *R* kytkintä, tilojen määrä.

Karkea yläraja:

$\zeta \leq R$

Tarkempia ylärajoja saadaan:

- poistetaan ei legitiimit tilat esim. joissa kaksi kytkintä on liittynyt samaan lähtöön

- joissa kaksi kentän tilaa tuottaa saman *C*.

Voidaan tarkastella eri funktioiden toteuttamien kytkentäkonfiguraatioiden C määrää ζ

$\zeta_{\text{pt-pt-kytkinfunktio}}$

$\zeta_{\text{multicastfunktio}}$

$\zeta_{\text{keskitinfunktio}}$

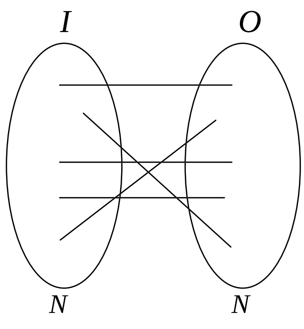
Määritelmän mukaan sanomme, että graafi G on uudelleen järjestettävästi estoton, jos se toteuttaa kaikki kytkentäkonfiguraatiot *C*.

Siis $\zeta_f \leq \zeta(G)$

Siis tarkastelemalla eri funktioiden kytkentä-konfiguraatioiden lukumäärää saamme selville uudelleen järjestettävän kentän monimutkaisuuden alarajan.

Pt-pt-kytkentäfunktion monimutkaisuuden alaraja

Pt-pt-kytkentäfunktio



Yksi yhteen

Kaikki *i* kytketty tasan yhteen *o*.

Haluamme siis toteuttaa *N*! erilaista *C*.

Käytetään Sterlingin likiarvoa:

$$N! \approx \sqrt{2\pi}\, N^{N+1/2} e^{-N}$$

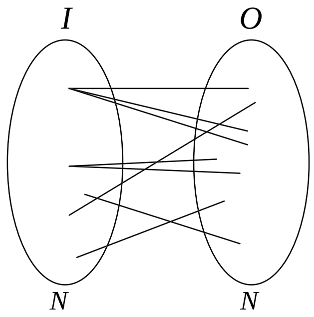
$$= \sqrt{2\pi}\ \exp_2(N \log_2 N - N \log_2 e + 1/2 \log_2 N)$$

$$\zeta_{\text{pt-pt}} = \log_2 N! \approx N \log_2 N - 1.44N + 1/2 \log_2 N$$

HUOM: Benes verkon kytkinten määrä oli $N \log_2 N - N/2$, joka on hyvin lähellä alarajaa $\zeta_{\text{pt-pt}}$.

Multicastfunktion monimutkaisuuden alaraja

Multicastfunktio



$C = \{(o, i) | \ i \in I, o \in O\}$

Jokainen $o \in O$ on kytketty johonkin $i \in I$.

Jokainen *o* voi siis valita minkä tahansa *i*.

Siis me haluamme toteuttaa

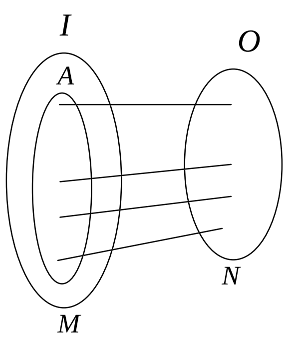
$N^N = \exp_2(N \log_2 N)$ kytkentäkonfiguraatiota *C*.

$\zeta_{\mathbf{mtcast}} - \zeta_{\mathbf{pt\text{-}pt}} \approx \mathbf{1.44}\ N$

Kenttää, joka toteuttaisi multicastin ja olisi monimutkaisuudeltaan alarajan tuntumassa ei tunneta. Tiedetään, että Benes verkko toteuttaa multicastin, jos kytkinten määrä kaksinkertaistetaan p-to-pt kytkentäkenttään verrattuna.

Keskitinfunktion monimutkaisuuden alaraja

Keskitinfunktio



$C = \{i | \ i \in A, i \text{ lukumäärä} = N\ (< M)\}$

Joukkoja *C* on $\begin{pmatrix} M \\ N \end{pmatrix}$ kappaletta

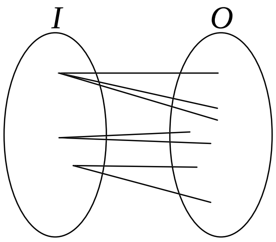
$$\zeta_{\mathbf{keskitin}} = \log_2 \frac{M!}{N!\ (M-N)!}$$

$$\zeta_{\mathbf{keskitin}} = M\,H(c) \qquad c = N/M$$

$$H(c) = -c \log_2 c - (1 - c) \log_2 (1 - c)$$

Eli keskittemen teoreettinen monimutkaisuus on verrannollinen tulojen määrään. Käytännön ratkaisuja ei kuitenkaan tunneta ja tarvitaan tiukasti estottomia keskittimiä => käytetään M logM kenttiä.

Kuvausten lukumäärä Copy-kentässä



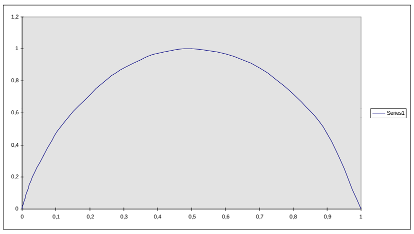
M tuloa $\quad$ N lähtöä

$C = \{(i,n_i) \mid i \in I, \Sigma n_i = N\}$

Erilaisten C lukumäärä on

$$\binom{M-1+N}{M-1}$$

Kuinka monella tavalla N oliota voidaan laittaa M koriin.



$$\zeta \geq (M-1+N)\, H\left(\frac{M-1}{M-1+N}\right)$$

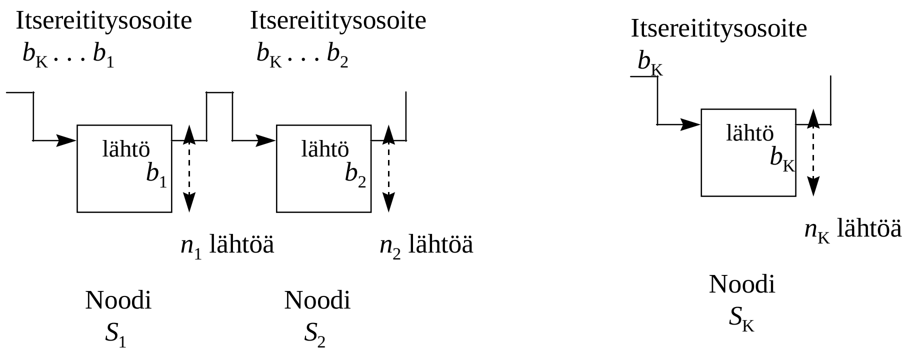
$$H(c) = -c \log_2 c - (1-c) \log_2 (1-c)$$

Multicast kenttien rekursiivinen rakentainen

- Multicast kentän voi rakentaa esim. Copykenttä + pt-to-pt kenttä.
- Tämä johtaa kuitenkin portaiden lukumäärän kasvuun, mikä voi olla epätoivottavaa.
- Moniportaisille kentille on ominaista polun haun/kontrollin monimutkaisuus.
- Tätä ongelmaa yritetään ratkaista mm. itsereitittävyydellä.

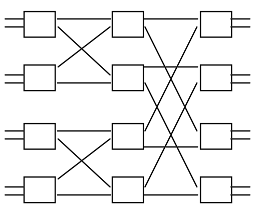
Itsereitittävyys

Itsereitittävyys on pakettikytkentäkentän mahdollinen ominaisuus.
Paketilla on otsikko, jota käytetään polun löytämiseksi kentän läpi.
Polkuja tulolta lähdölle on yksi (tai useampi).

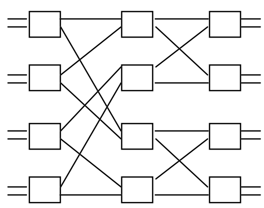


Yhden polun perusverkot

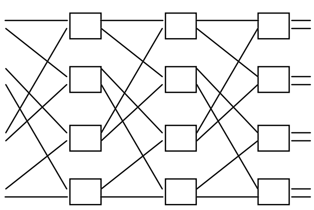
Banyan verkko



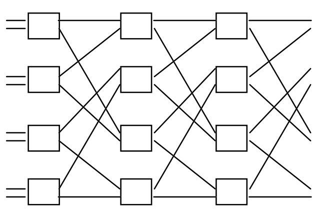
Baseline verkko



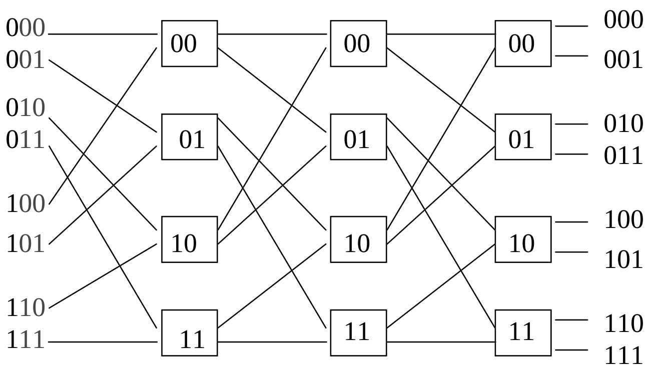
Shuffle exchange (omega) verkko



Flip verkko

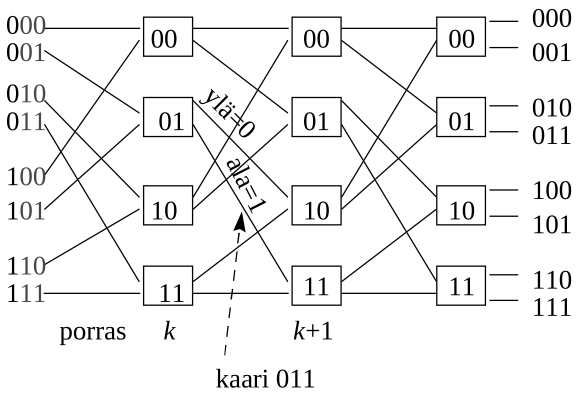


Itse-reitittävä shuffleverkko



$N = 2^n$ tuloa ja 2^n lähtöä. Benesrakenne => Portaita on n kpl ja kytkimiä $N/2 = 2^{n-1}$ kussakin portaassa.
Portaan kytkimet numeroidaan ylhäältä alas $0 \ldots 2^{n-1} - 1$. Portaiden numerointiin tarvitaan $n - 1$ bittiä.
Portaita yhdistävät kaaret numeroidaan ylhäältä alas n bitillä.

Itse-reitittävä shuffleverkon numerointi

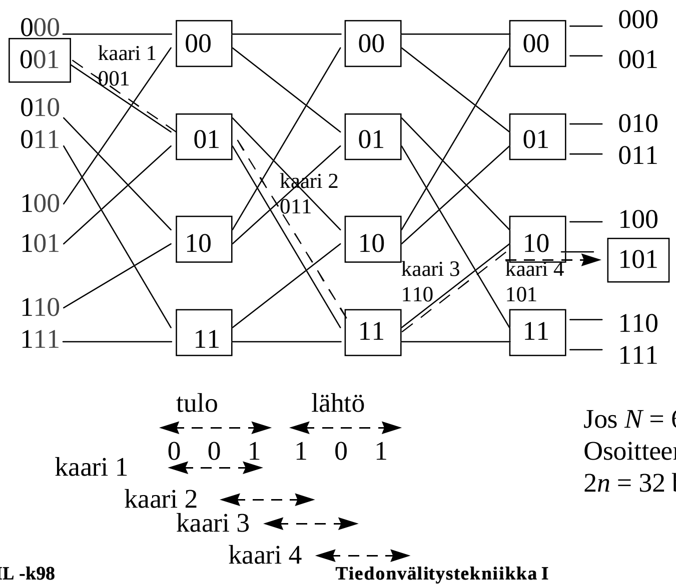


Portaan k solmun nro = kaaren nro - oikean puoleisin bitti

Portaan k +1 solmun nro = kaaren nro - vasemman puoleisin bitti

Lähdön numero toimii suoraan itsereititysosoitteena.

Itse-reititys esimerkki



Jos N = 64 000
Osoitteen pituus on
$2n$ = 32 bittiä.

Ratkaisun rajoitukset

Banyan verkko on estollinen.
Se toteuttaa $\exp_2(1/2N \log_2 N) = (N^N)^{1/2}$ kytkentäkonfiguraatiota.
Tämä on vähemmän kuin vaadittu $N! \approx \exp_2(N \log_2 N)$.

=>
Noodien välisten kaarien määrää voidaan nostaa, tai
monistamalla shufflea Cantor verkon tapaan, tai
laittamalla kaksi shufflea peräkkäin (vrt BENES verkko).

Myös puskurointia väliportaissa voidaan käyttää.