Sorting Networks

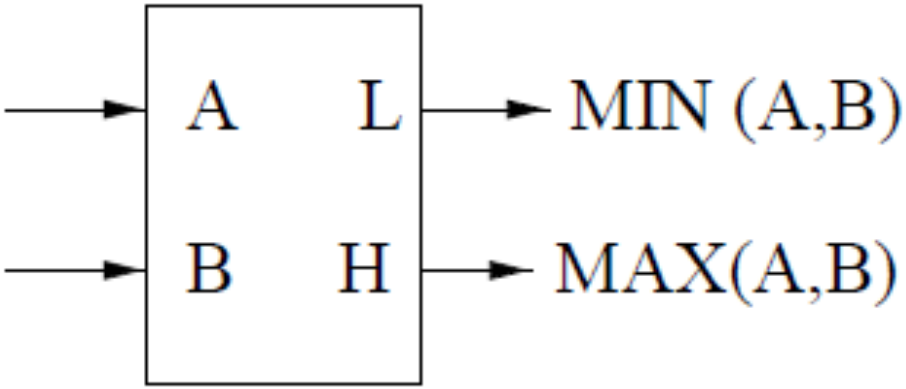
Vincent Derkinderen & Mathias Dekempeneer
3e bachelor Informatica KULeuven
Begeleider: Tom Schrijvers

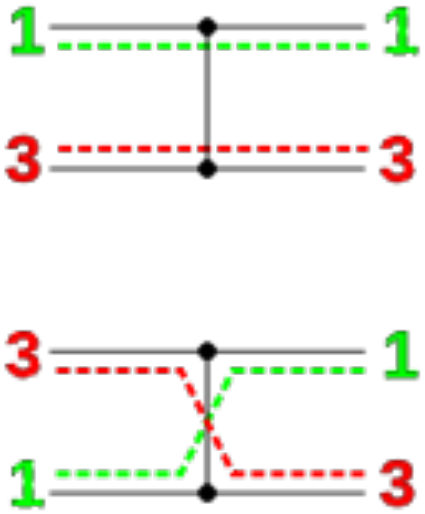
Overzicht

- Inleiding: sorting network
- Toepassingen
- Probleemstelling
- Aanpak hoofdpaper
- Conclusie

Wat is een comparator?

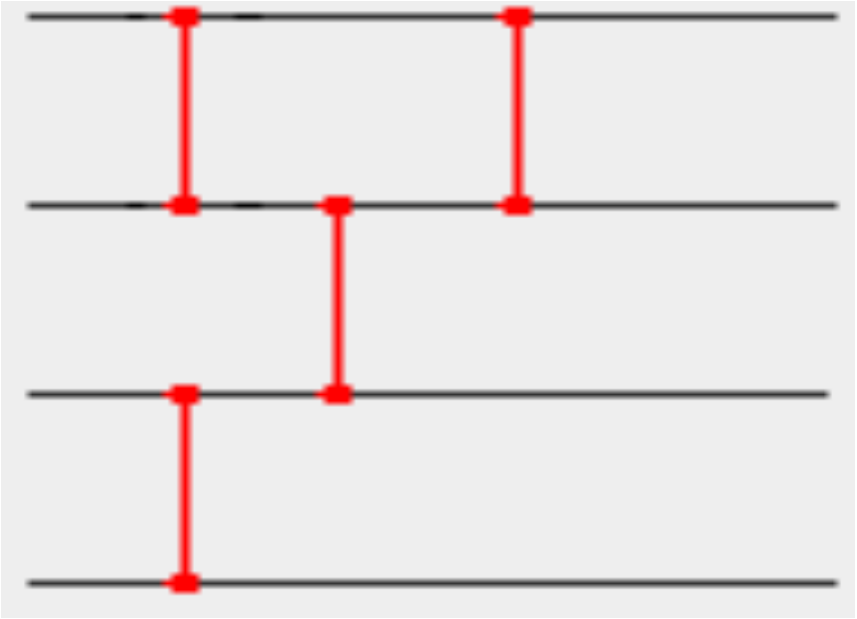
- Comparators (a b) met a < b
  - A = w(a) < w(b) = B
  - w(a) > w(b): w(a) ↔ w(b)





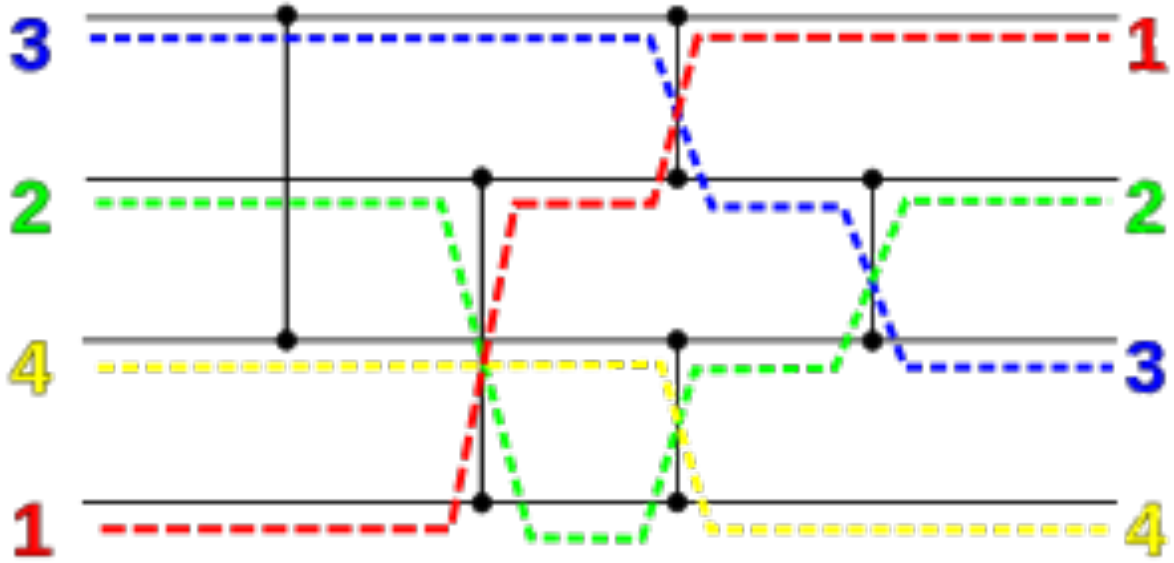
Wat is een comparator network?

- Kanalen
  - Input
  - Output
- Comparators
- Layer
  - Parallelle comparators



(1 2)(3 4)(2 3)(1 2)

Werking comparator network

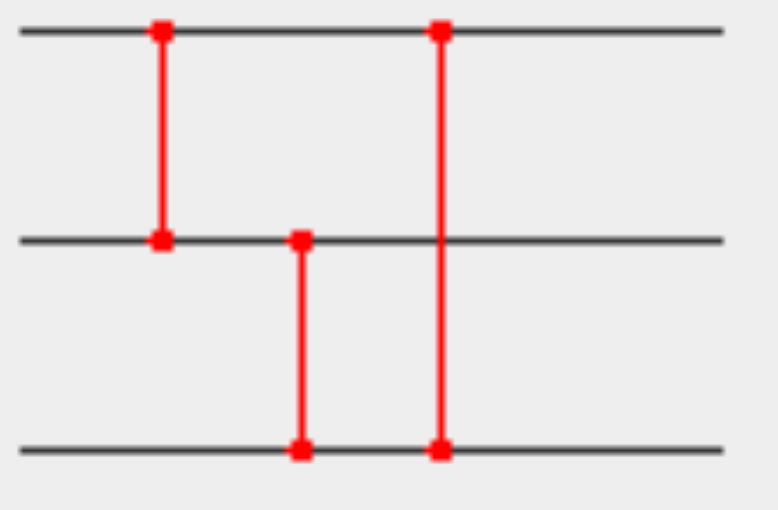


Wat is een sorting network?

Gesorteerd

+ Output is gesorteerd

Ongesorteerd



+ Niet alle output gesorteerd (110 → 101)

Nul – één principe

- Testen van comparator network
  - Nul – één principe $\Rightarrow$ testen alle combinaties {0,1}

Nul – één principe

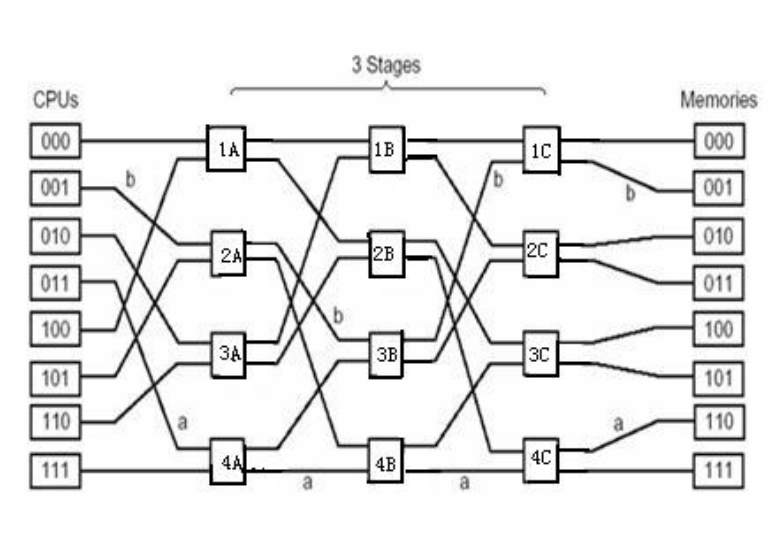
Sorteert alle combinaties {0,1}

$\Downarrow$

sorteert alle combinaties van totaal geordende set

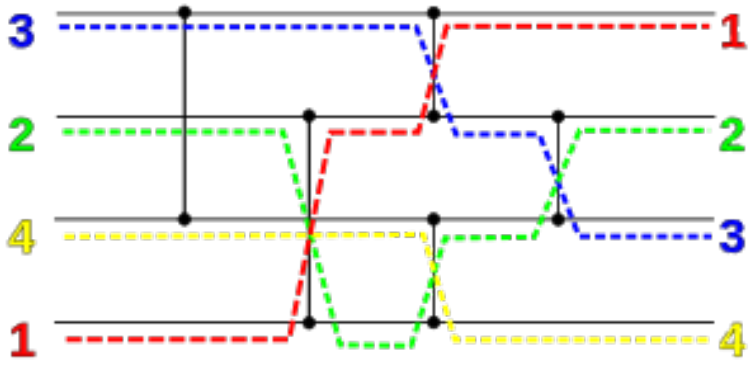
Toepassingen

- Switching network with buffering
  - Input = bestemmingsgetal
- Multiaccess memory
- Multiprocessor
- Om te sorteren



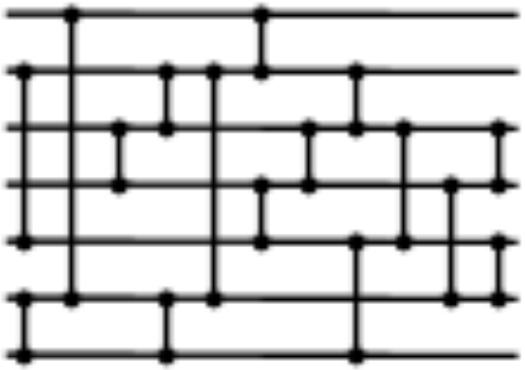
Toepassingen

- Switching network with buffering
- Multiaccess memory
- Multiprocessor
- Om te sorteren
  - Sorteren van data



Doel

- Optimal size
  - Minimaal # comparators
  - Size 11 kanalen = ?
- Optimal depth
  - Minimaal # layers



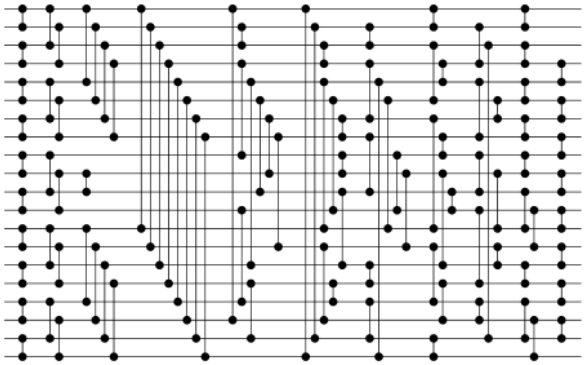


Fig. 4. A sorting network for 20 channels of depth 11.

Doel

- Optimal size
  - Minimaal # comparators
- Doel: optimal size
  11 kanalen?
  - Best case:
    55^{33} mogelijke netwerken
  - Worst case:
    55^{35} mogelijke netwerken

verschillende comparators:
11*10/2 = 55

33,34,35 comparators

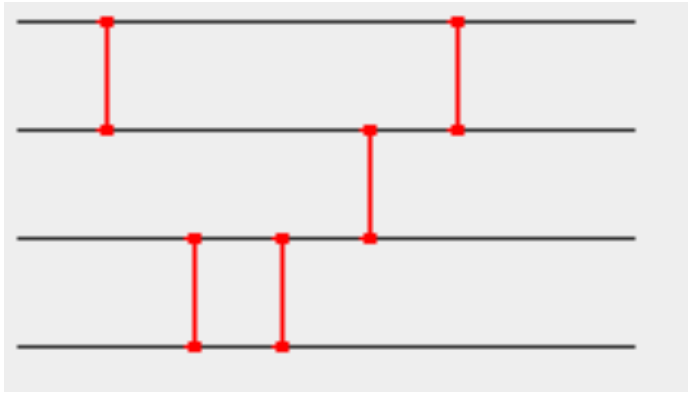
Doel

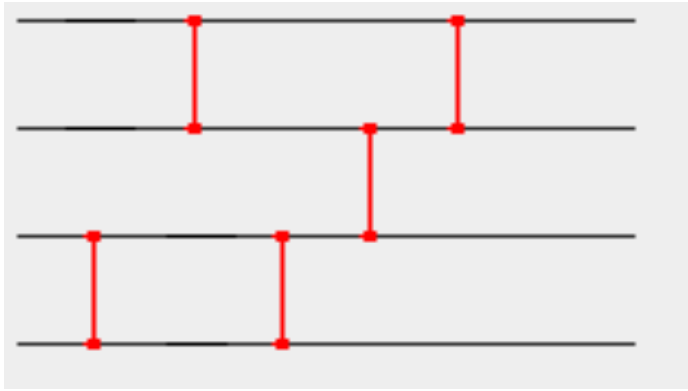
- Optimal size
- Lower bound:
  $S(n+1) \geq S(n) + \lceil \log_2 n \rceil$

n	1	2	3	4	5	6	7	8	9	10	11	12	13
Depth	0	1	3	3	5	5	6	6	7	7	8	8	9
Upper bound	0	1	3	5	9	12	16	19	25	29	35	39	45
Lower bound	0	1	3	5	9	12	16	19	25	29	33	37	41

Doel

- Optimalisatie test
- Optimalisatie domein (generate + prune)
  - Bijvoorbeeld: Overbodige comparators verwijderen



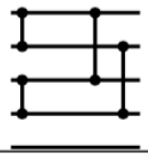
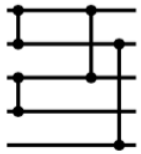
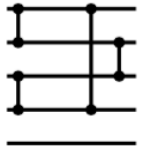


Generate & prune

1. **Generate**
   Voeg een comparator toe achter elk bestaand netwerk van $R^n_k \Rightarrow N^n_{k+1}$
2. **Prune**
   $(C_a \leq_p C_b) \Rightarrow C_b$ niet uitbreiden (als $|a|=|b|$)
   $\neg(C_a \leq_p C_b) \Rightarrow C_b \in R^n_{k+1}$
3. Herhaal het proces tot $|N^n_c| = 1$

Generate & prune: implementatie

- Bewaren van de output
  ⇒ toevoegen comparator = permutatie op outputs
  → grootte van de outputset krimpt

no. of 1s:	0	1	2	3	4	5
C_1 	00000	00001 00010	00011 00110 01010	00111 01011 01110	01111 11110	11111
C_2 	00000	00001 00010	00011 00101 00110 01001	00111 01011 01101	01111 10111	11111
C_3 	00000	00001 00010 00100	00011 00101 00110	00111 01110 10110	01111 10111 11110	11111

Concept subsumes

$$C_a \leq_p C_b \Leftrightarrow P(\text{outputs}(C_a)) \subseteq \text{outputs}(C_b)$$

- C_a: (1 2) en C_b: (2 3)
  outputs(C_a) = {(0 0 0),(0 0 1),(0 1 0),~~(1 0 0)~~,(0 1 1),~~(1 0 1)~~,(1 1 0), (1 1 1)}
  outputs(C_b) = {(0 0 0),(0 0 1),~~(0 1 0)~~,(1 0 0),(0 1 1),(1 0 1),~~(1 1 0),~~ (1 1 1)}

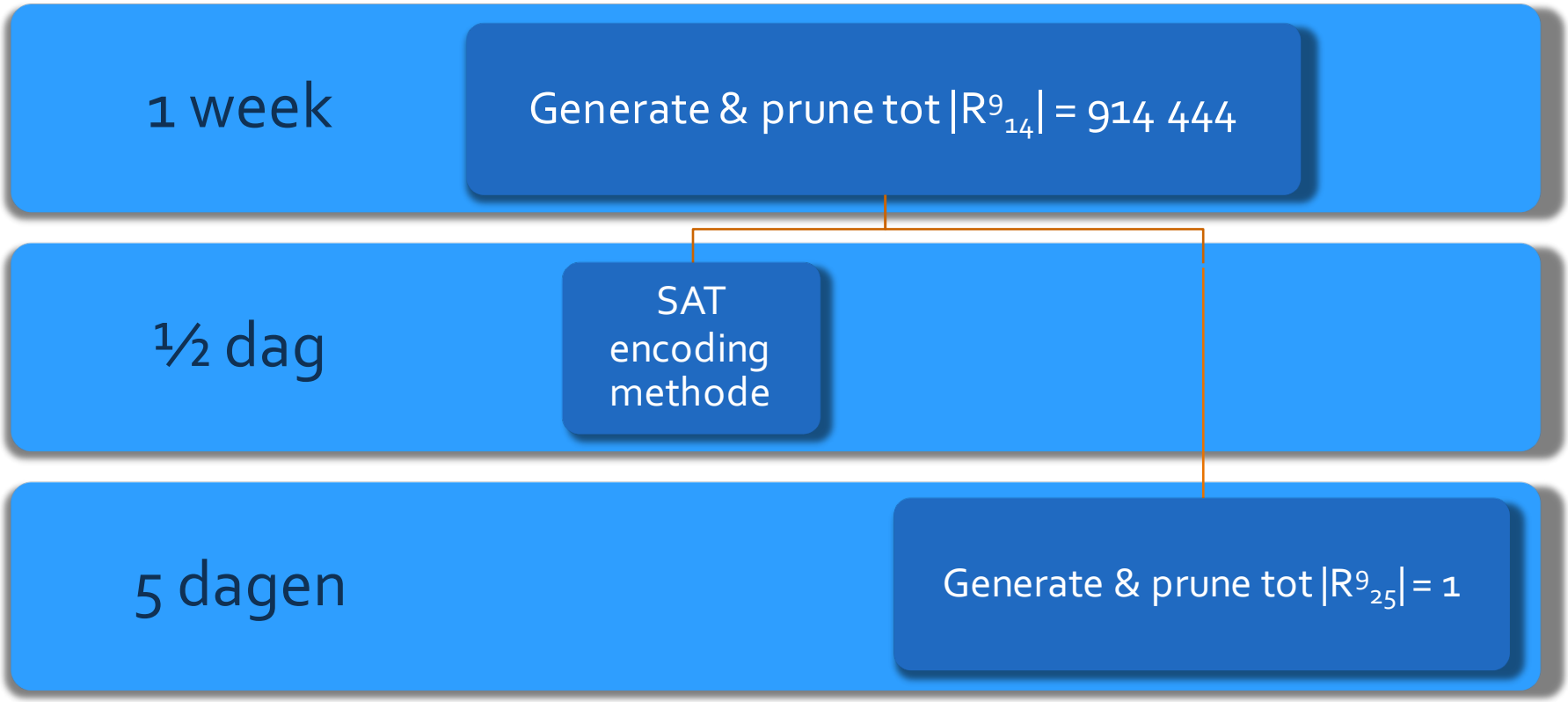
- p = (2 3)(1 2)
  p(outputs(C_a)) = {(0,0,0), (1,0,0), (0,0,1), (1,0,1), (0,1,1), (1,1,1)}

Generate & prune

Subsumes = dure operatie $\Rightarrow$ extra prune methodes

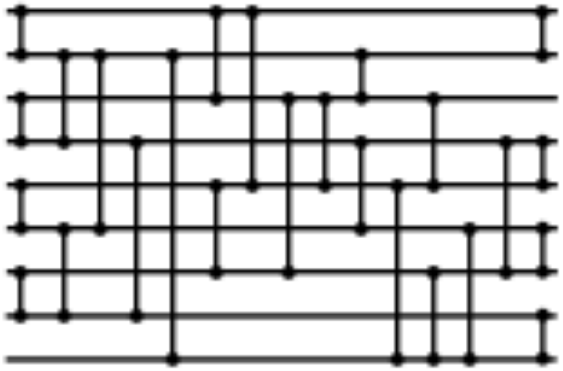
- Contradicties bij subsumes
- Verwijderen overbodige comparators (i j) :
  $\forall$ outputs(C) : $x_i \leq x_j$
- Parallel uitvoeren

Optimal size, 9 channels



Optimal size, 9 channels: resultaten

1. Size(9) = 25



2. Size(10) = 29
   - Lower bound $S(10) \geq S(9) + \lceil \log_2 9 \rceil \geq 25 + 4 \geq 29$
   - Upper bound: 29
   - ➢ Size(10) = 29

Ons onderzoek

- Reproduceren resultaten hoofdpaper
- Optimal sorting network 11 kanalen

Papers

- **Twenty-Five Comparators is Optimal when Sorting Nine Inputs (and Twenty-Nine for Ten)**
  M. Codish, L. Cruz-Filipe, M. Frank, P. Schneider-Kamp
  24 juni 2014.

- **Sorting Networks and Their Applications**
  K.E. Batcher
  2 mei 1968

- **Bounds on the size of test sets for sorting and related networks**
  M. J. Chung, B. Ravikumar
  1990.

- **Sorting Networks: the End Game**
  M. Codish, L. Cruz-Filipe, P. Schneider-Kamp
  24 november 2014.

VRAGEN?