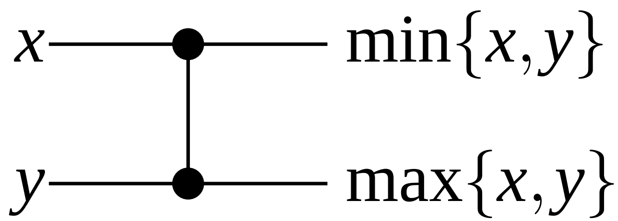
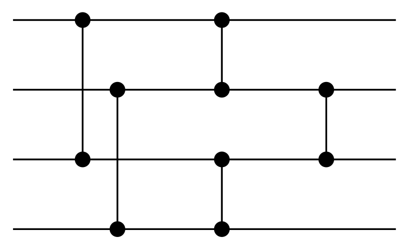
Fast Periodic Correction Networks

Grzegorz Stachowiak
University of Wrocław

Comparator Sorting Networks

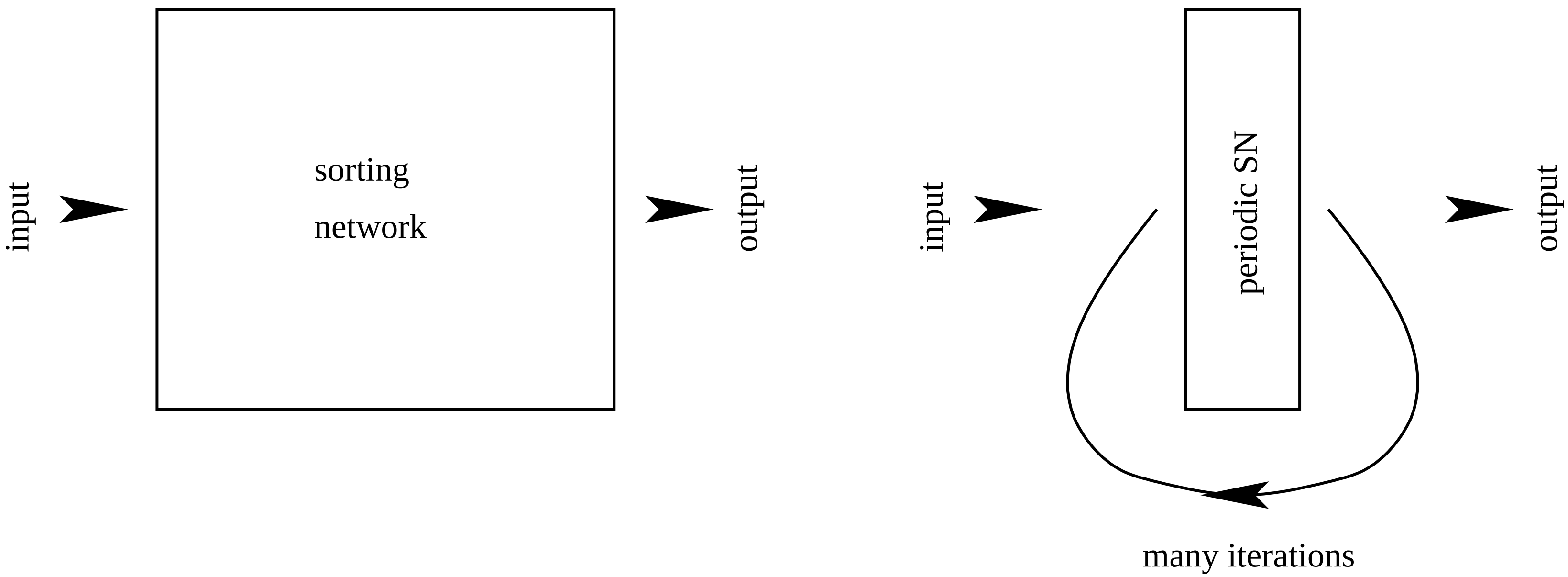
- Comparator: 

- In a *sorting network* comparators are arranged into a series of layers.



- Zero-one principle: It is enough to check if a comparator network sorts all 0-1 inputs.
- Batcher sorting network has depth $\sim 1/2 \log^2 N$ and AKS sorting network has depth $O(\log N)$.

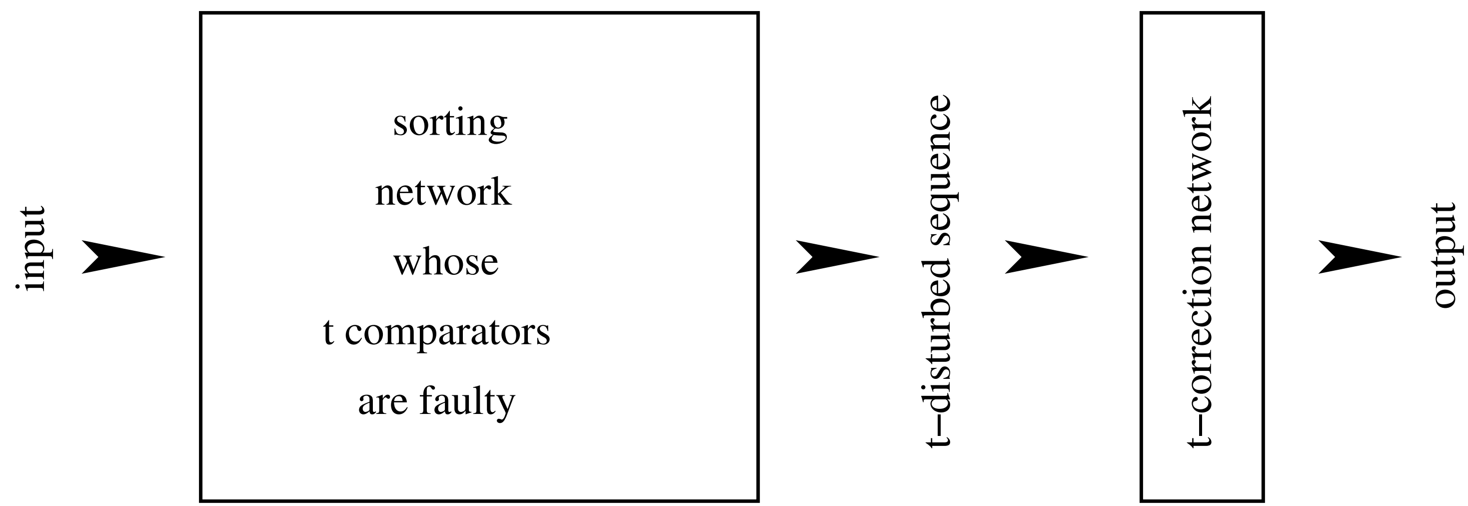
Nonperiodic & periodic sorting



- Dowd,Perl,Saks,Rudolph – depth: $\log n$ and $\log n$ iterations
- Loryś et al. – depth: 3–5 and $O(\log^3 n)$ iterations.

Correction networks

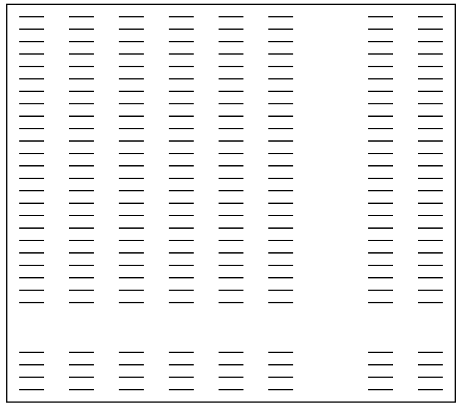
- *t-disturbed* input.
  00000100000001000000011111101111111110111111
- *t-correction* network sorts all t-disturbed inputs.
- Best result: depth $1.44 \log N$ for constant t.
- main application:



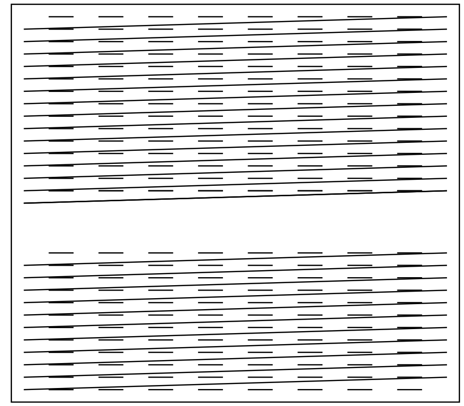
Periodic Correction Networks

- A *periodic t-correction network* is a periodic compatator network which sorts any t-disturbed input.
- Kik, Piotrow constructed periodic t-correction networks of 6-8 layers and requring $O(\log N + t)$ iterations.
- This work: 3 layers and $O(\log N + (\log\log N)^2(\log t)^3)$ iterations.

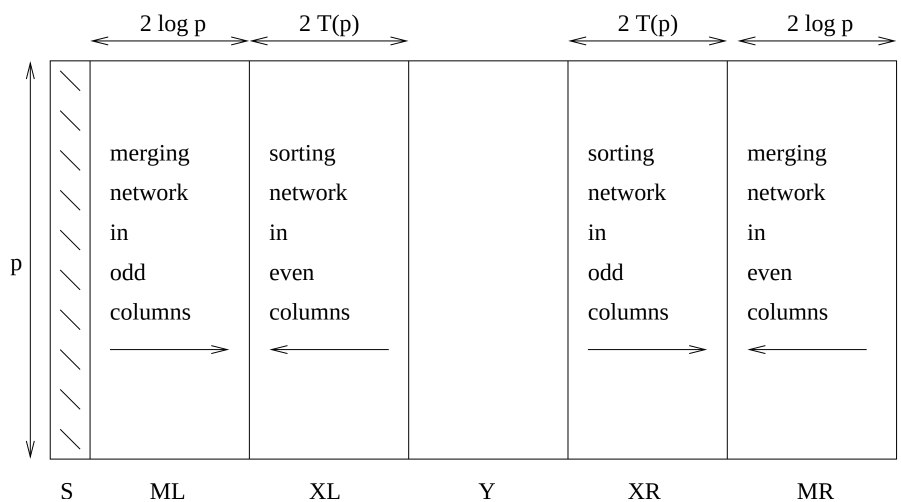
3-layer periodification scheme



layer A

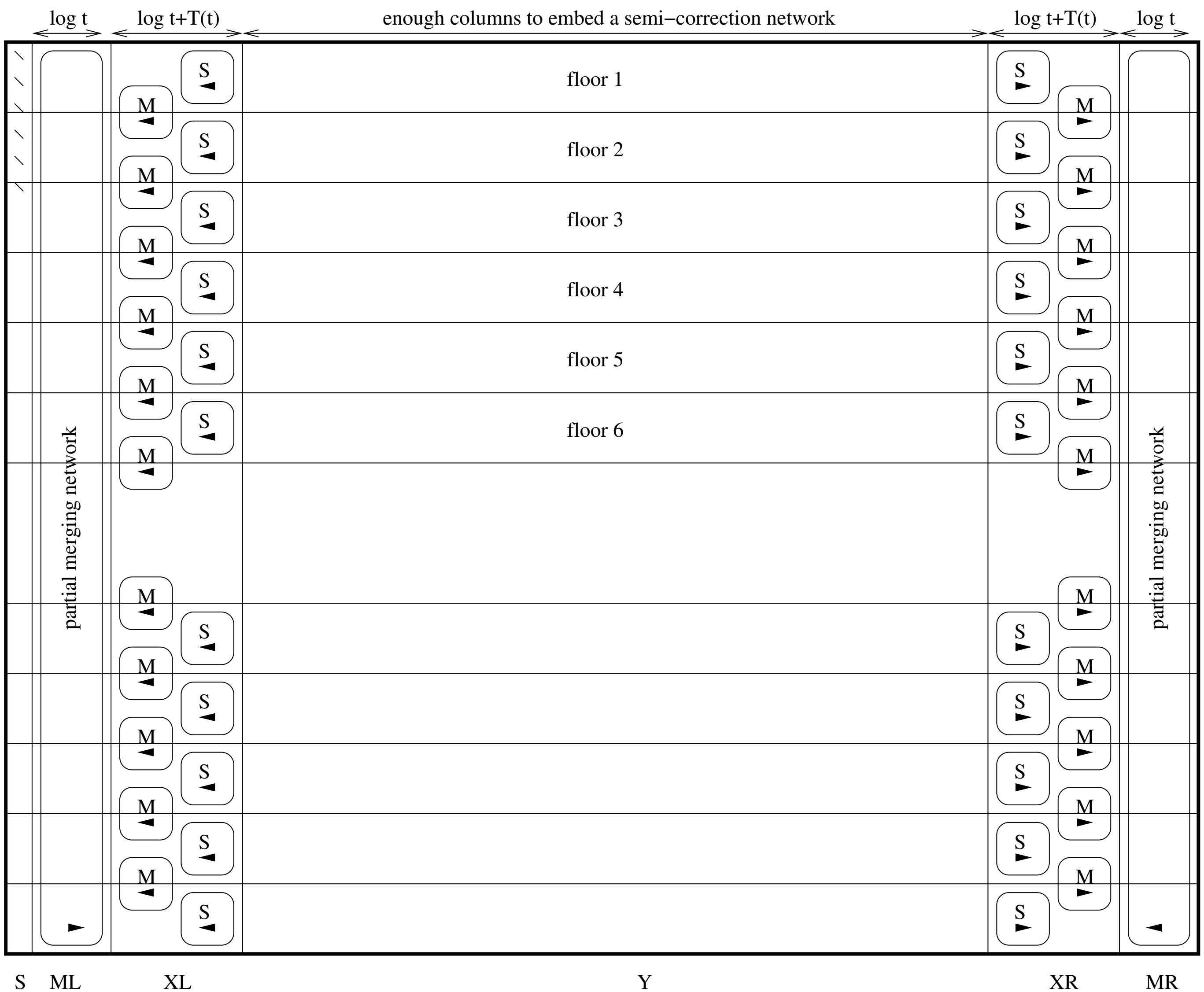


layer B



layer C

Layer C in our network



How it works

- We have $q = O((\log N + (\log \log N)^2 (\log t)^3)/\log t)$ columns.
- We divide rows into floors having $O(t)$ rows each.
- Comparators are arranged so that a right running displaced 1 goes 1 floor down passing X_R, M_R, S, M_L, X_L.
- The same happens to a displaced leftrunning 1 if it remains left running.
- All 1s that are close to the border do not go down 1 floor anymore, but if akk displaced 1s are close to the border then $O(\log N + (\log \log N)^2 (\log t)^3)$ rounds are enough to finish sorting.
- We divide floors into families of floors. Family r consists of all floors with numbers $r + k \log t$.

How it works

- $O(\log N + (\log\log N)^2(\log t)^3)$ rounds are enough for a right running 1 to get to the family 1.
- Passing all areas in family r a right running 1 reduces its distance to the border from $2N^{1-(r-1)/\log t}$ to $2N^{1-r/\log t}$ due to a semi-correction network embeded in this family and then gets to the family $r+1$.
- $O(\log N + (\log\log N)^2(\log t)^3)$ rounds are enough to reduce this distance to $\log t$ floors.
- Next $O(\log N + (\log\log N)^2(\log t)^3)$ rounds are enough to reduce this distance to 2 floors.
- The same happens to a left-running 1 as long as it stays left-running.