Hardware to Solve Sparse Systems of Linear Equations over GF(2)

Willi Geiselmann
Rainer Steinwandt

Institut für Algorithmen und Kognitive Systeme
Universität Karlsruhe
Germany

Sieving Algorithms for Factorization

- Pre-computation
- Sieving step
- Linear algebra step

  Aim: solve a sparse system of linear equations

  The Block Wiedemann algorithm reduces this task to many matrix-vector multiplications:

  $A \cdot v, A \cdot (A \cdot v), A \cdot (A^2 \cdot v), \ldots$

  with $A \in GF(2)^{m \times m}$ sparse, $v \in GF(2)^m$
- Post-computation

Matrix-Vector Multiplication by Sorting

Schimmler's sorting algorithm:

Sorts 2^{2l} numbers in $\approx 8 \cdot 2^l$ steps

on a mesh of $2^l \times 2^l =: M \times M$ processing units

communication occurs between neighbours only.

Proposal by Bernstein (2001):

Calculate $A \cdot v$ on a mesh of 2^{2l} processors by sorting three times,

where $2^{2l} \geq$ number of non-zero entries of A + size of v.

"Bernstein Hardware"

One processing unit requires ≈ 2000 transistors or
0.07 mm $\times$ 0.07 mm (using a 0.13 μm process)

$\log_2(n)$	# proc	M	area	time
512	$4.3 \cdot 10^8$	2^{15}	2.1 m $\times$ 2.1 m	18 h
1024	$4.0 \cdot 10^9$	2^{16}	4.5 m $\times$ 4.5 m	207 h

Splitting this area into pieces requires ≈ 7500 pins per cm borderline

Design by Lenstra, Shamir, Tomlinson, Tromer

- Routing algorithm instead of sorting algorithm
- Many matrix entries stored per processing unit (DRAM)
- Scalable: area $\leftrightarrow$ time
- One processing unit requires $\approx$ 0.25 mm $\times$ 0.25 mm

$\log_2(n)$	# proc	M	area	time
512	$1.6 \cdot 10^5$	400	9.5 cm $\times$ 9.5 cm	17 min
1024	$9.5 \cdot 10^5$	1024	26.3 cm $\times$ 26.3 cm	6.5 h

Splitting into pieces requires $\approx$ 2,000 to 10,000 pins per cm borderline

Block Matrix Multiplication 1

Split A and v with $A_{i,j}$ of size $\approx m/s$ and approximately the same number of non-zero entries:

$$A = \left(\begin{array}{c|c|c|c} A_{1,1} & A_{1,2} & \dots & A_{1,s} \\ \hline A_{2,1} & A_{2,2} & \dots & A_{2,s} \\ \hline \vdots & \vdots & \vdots & \vdots \\ \hline A_{s,1} & A_{s,2} & \dots & A_{s,s} \end{array}\right)$$

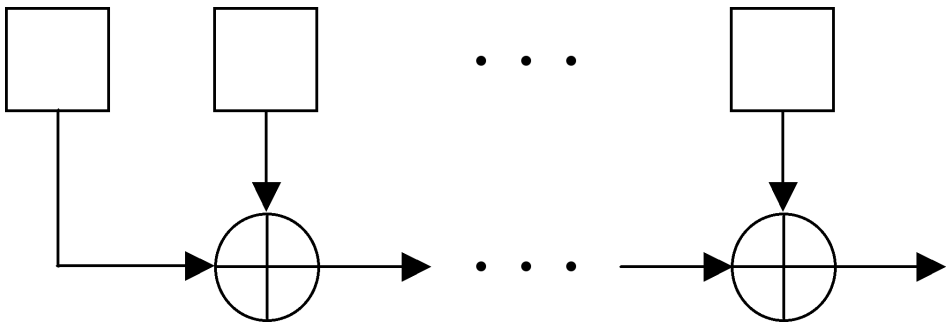
$$v = \begin{pmatrix} v_{1,1} \\ \vdots \\ v_{1,s} \end{pmatrix} = \ldots = \begin{pmatrix} v_{s,1} \\ \vdots \\ v_{s,s} \end{pmatrix}.$$

Then calculate

$$A \cdot v = \begin{pmatrix} \sum_{j=1}^{s} A_{1,j} \cdot v_{1,j} \\ \vdots \\ \sum_{j=1}^{s} A_{s,j} \cdot v_{s,j} \end{pmatrix}.$$

Block Matrix Multiplication 2

- Assign the "multiplication circuit" $\bar{A}_{i,j}$ to the submatrix $A_{i,j}$ (for $1 \leq i, j \leq s$) and load the matrix;
- Distribute/load the appropriate parts of v to $\bar{A}_{i,j}$;
- Multiply $A_{i,j} \cdot v_{i,j}$ on $\bar{A}_{i,j}$;
- Output the subproducts and add them in a pipeline structrue.



Performance with "Bernstein Circuits"

$\log_2(n)$	b	s^2	chip size	LA time
512	(single unit)	1	2.14 m × 2.14 m	17.6 h
512	2048	11^2	144 mm × 144 mm	1.1 h
512	1024	24^2	72 mm × 72 mm	0.6 h
512	1024	55^2	36 mm × 36 mm	0.3 h
1024	(single unit)	1	4.5 m × 4.5 m	210 h
1024	2048	37^2	144 mm × 144 mm	6.8 h
1024	1024	84^2	72 mm × 72 mm	3.4 h

b: number of I/O-pins per unit

s^2: number of units

Performance with LSTT Circuits

$\log_2(n)$	b	K	s^2	chip size	LA time
512	128	65	10^2	11.4 mm × 11.4 mm	73 min
512	512	63	10^2	20 mm × 20 mm	19 min
512	1024	40	16^2	29 mm × 29 mm	6 min
1024	(single unit)	208	1	265 mm × 265 mm	6.1 h
1024	(single unit)	42	1	162 mm × 162 mm	94.7 h
1024	128	30	16^2	11.4 mm × 11.4 mm	29.7 h
1024	512	100	16^2	20 mm × 20 mm	7.0 h
1024	1280	135	16^2	36 mm × 36 mm	2.8 h