A Framework for High Level Synthesis using TED

(Taylor Expansion Diagram)

[redacted]

Prepared by
Daniel Gomez-Prado

Aug 10 - 2011

Outline

- Introduction
  - Overview of Taylor Expansion Diagrams (TED)
  - Optimizing DFGs for HW Implementations
  - Functional Retiming
  - Computational Accuracy
- Overview of the data structures used
- Other Projects

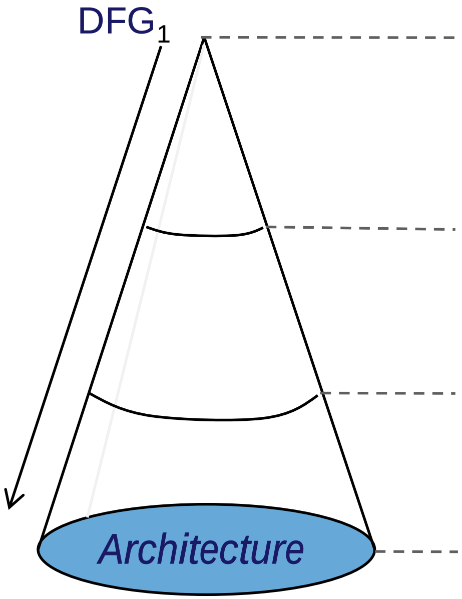
Current CAD Flow

Algorithm

High Level Description of the design (VHDL, c++)

Parsing and elaboration
Compiler like optimizations

Structural DFG



Data Flow Graph

Tree balancing
Common sub-expression elimination

High Level Synthesis

Scheduling, Allocation, Binding

Logic & Physical Synthesis

Library mapping, Routing

All possible Architectures reachable from DFG_1

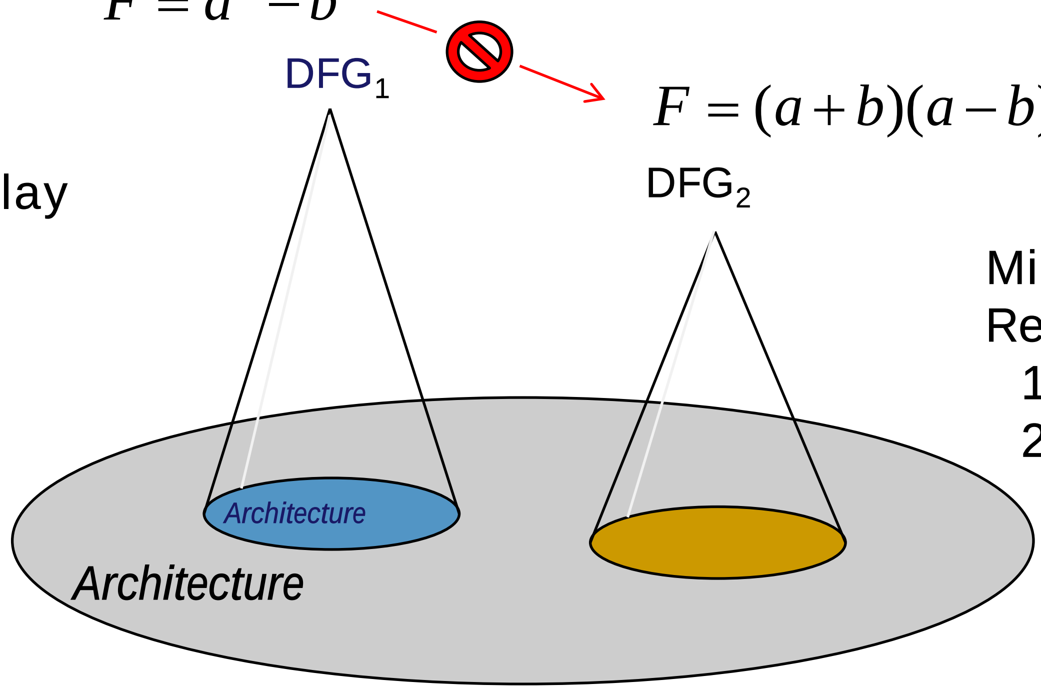
But some DFGs produce better Architectures

$$F = a^2 - b^2$$

No automation

$$F = (a+b)(a-b)$$

Minimum delay
Requires:
- 2 MPYs
- 1 ADD



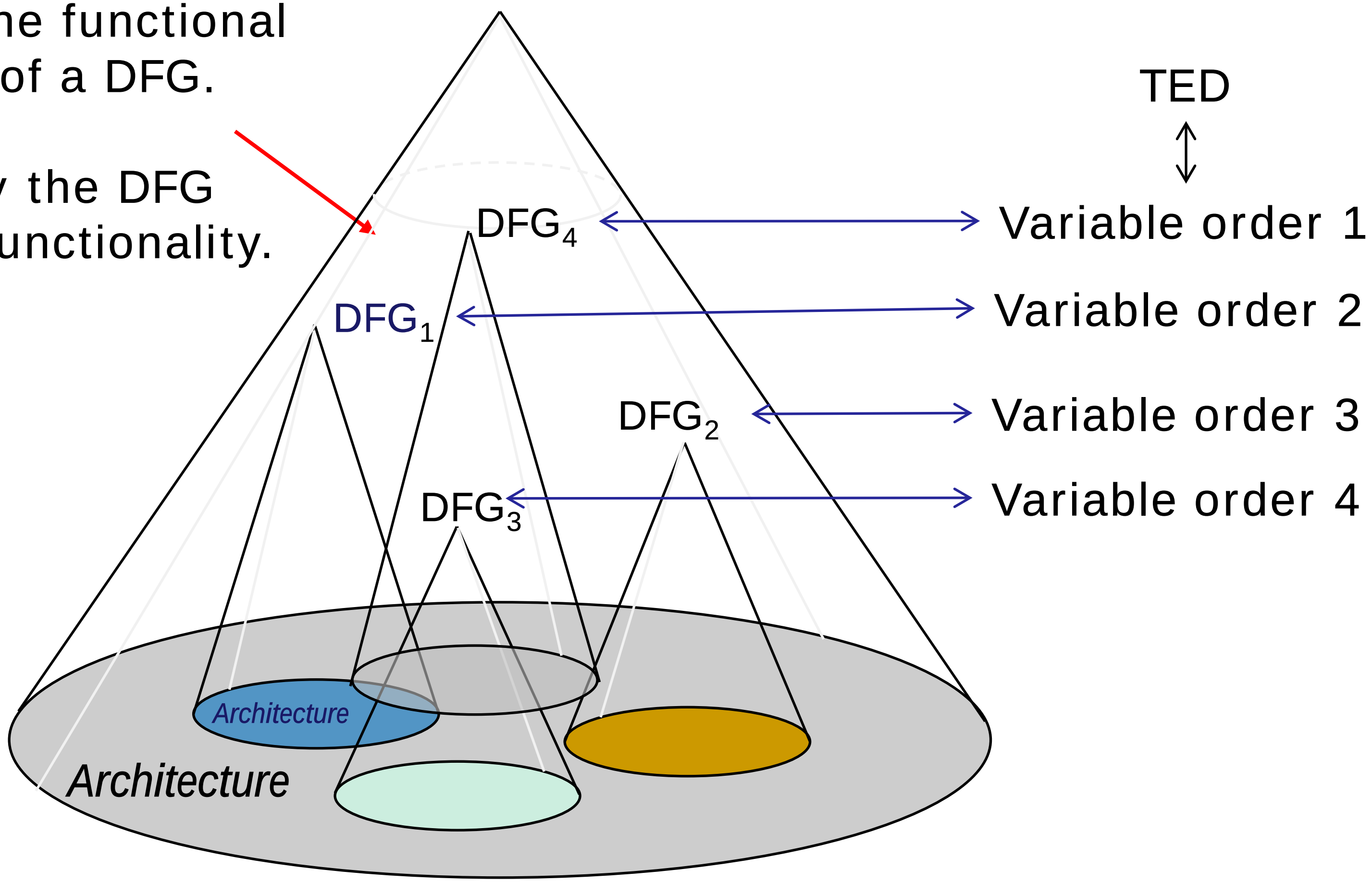
Minimum delay
Requires:
- 1 MPY
- 2 ADDs

How do we find the best DFG?
How do we generate another functionally equivalent DFG?

Taylor Expansion Diagrams

TED captures the functional representation of a DFG.

TED can modify the DFG preserving its functionality.



TED ↕ Variable order 1

DFG_4 ↔ Variable order 1

DFG_1 ↔ Variable order 2

DFG_2 ↔ Variable order 3

DFG_3 ↔ Variable order 4

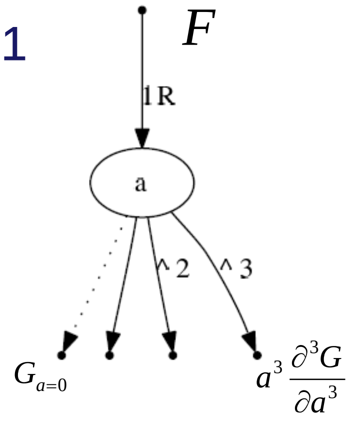
TED Graph Notation

Follows Taylor decomposition:

$$F_{(a,\cdots)} = G^{1R}_{(a,\cdots)}$$

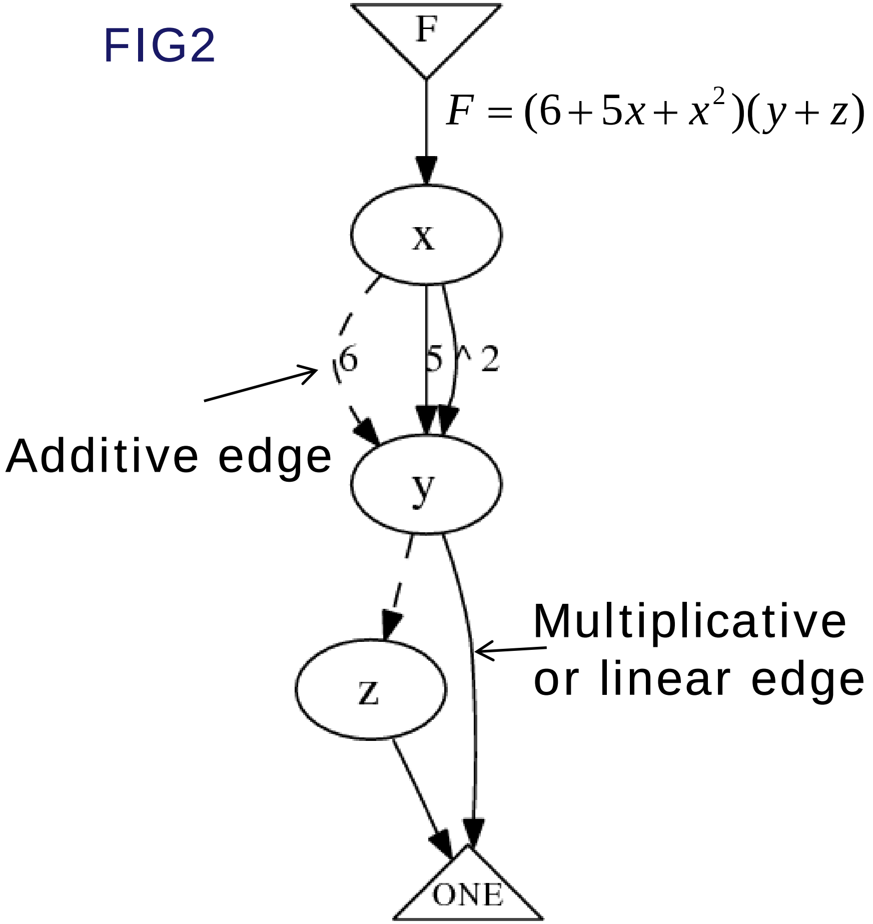
$$G = G_{(a=0)} + a\frac{\partial G}{\partial a}_{|a=0} + a^2\frac{\partial^2 G}{\partial a^2}_{|a=0} + a^3\frac{\partial^3 G}{\partial a^3}_{|a=0}$$

FIG1



It is a canonical data structure

FIG2

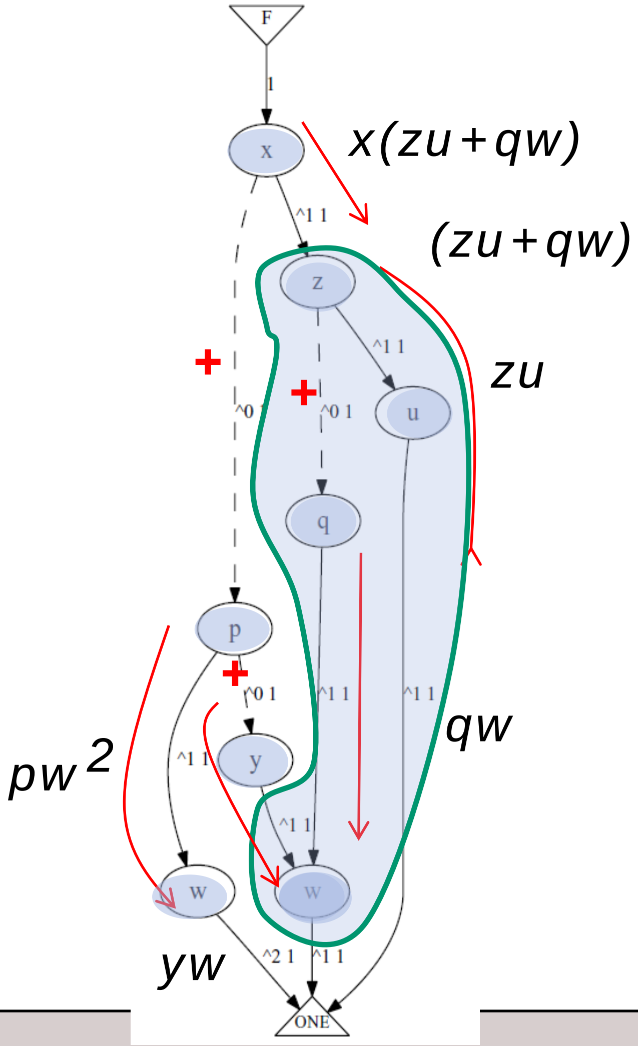


$F = (6+5x+x^2)(y+z)$

Additive edge

Multiplicative or linear edge

TED Construction

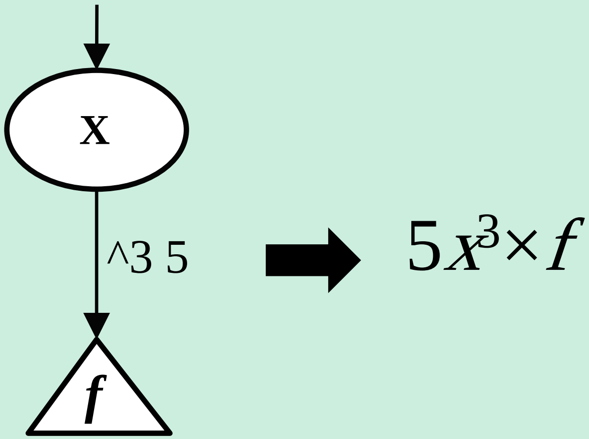


TED & TDS Overview

$$\boxed{x(\boxed{zu}+\boxed{qw})}+\boxed{pw^2}+\boxed{yw}$$

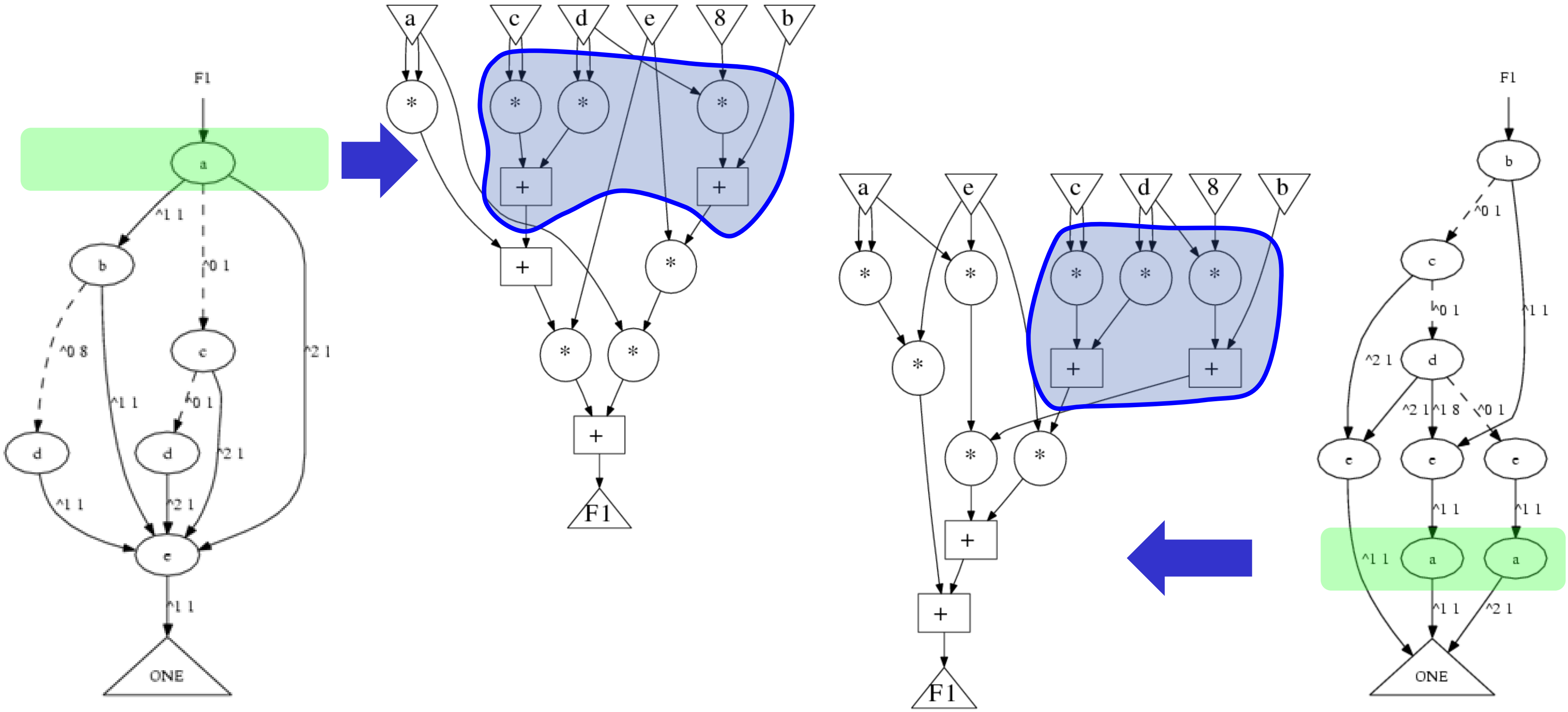
Canonical for the given order: *x,z,u,q,p,y,w*

Notation:



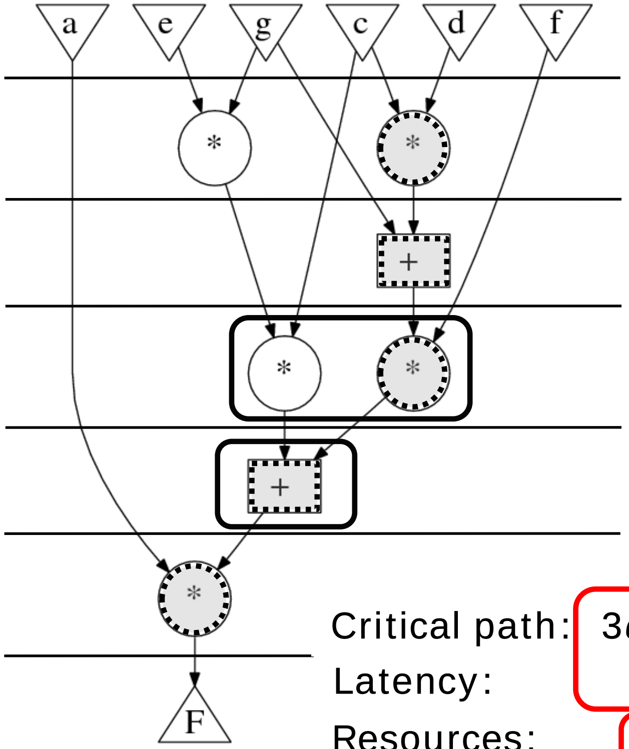
TED Variable Ordering

$$F1 = e(a^2 + c^2 + d^2 + a(b + 8d))$$



TED optimizes HW not operation count

$$F = a\cdot(f\cdot(g+d\cdot c)+c\cdot e\cdot g) \Big\} \text{ 5MUL, 2ADD}$$



CSE optimizes operations

FIG1

Critical path:	$3d_{MPY}+2d_{ADD}$
Latency:	5 steps
Resources:	2MPY 1ADD

$$F = a\cdot g\cdot(f+c\cdot e)+a\cdot f\cdot d\cdot c \Big\} \text{ 6MUL, 2ADD}$$

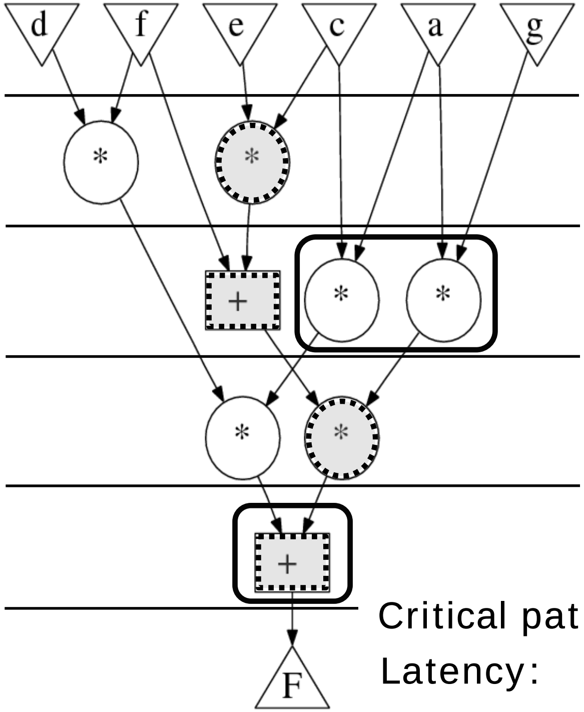


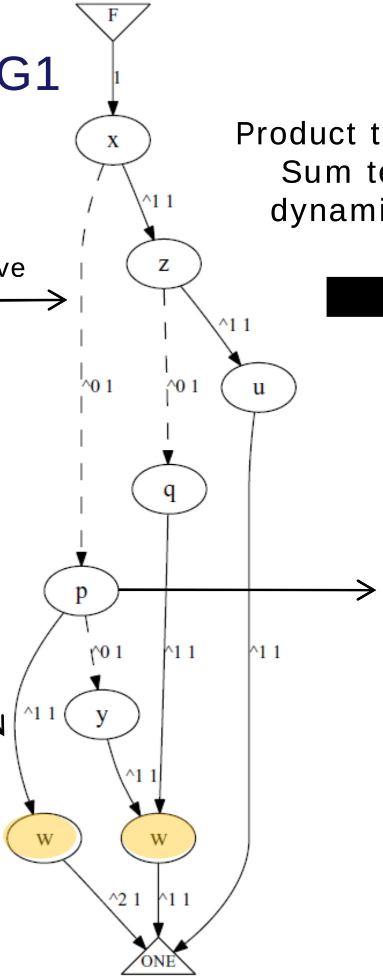
FIG2

TED optimizes HW objectives

Critical path:	$2d_{MPY}+2d_{ADD}$
Latency:	4 steps
Resources:	2MPY 1ADD

Optimizing DFGs for HW (TED Decomposition)

FIG1



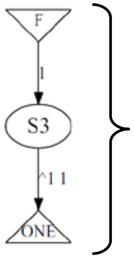
additive edge →

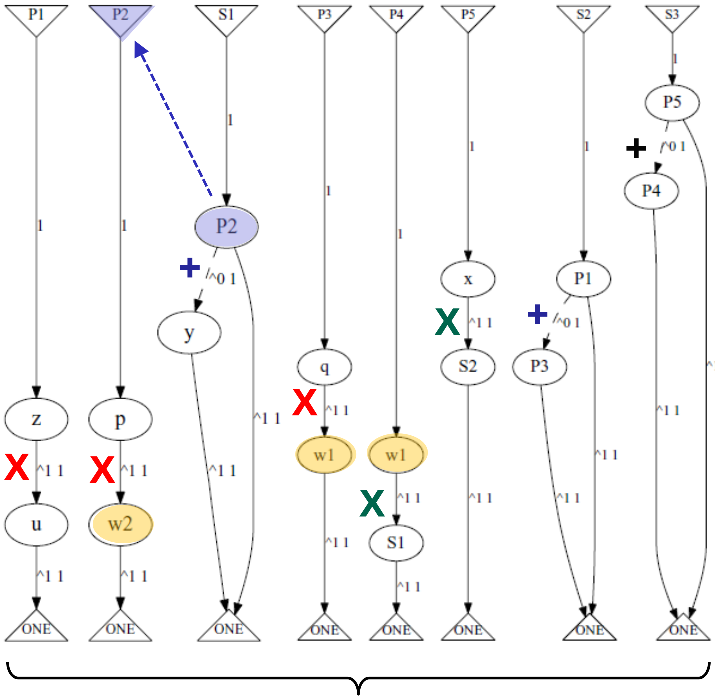
linear edge

$$G = pw^2 + yw$$
$$= (pw + y)w$$

Product term extraction
Sum term extraction
dynamic factorization
reordering

FIG2



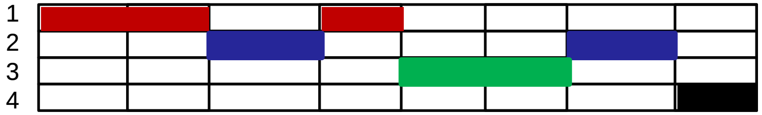


DFG

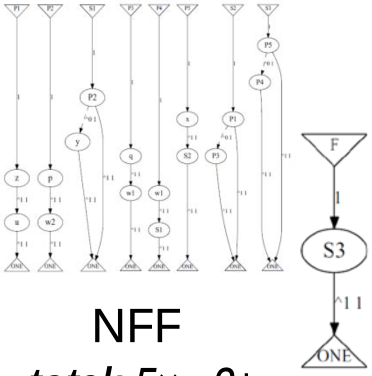
ASAP schedule

Resources: 3MPY
2ADD

Latency: 4

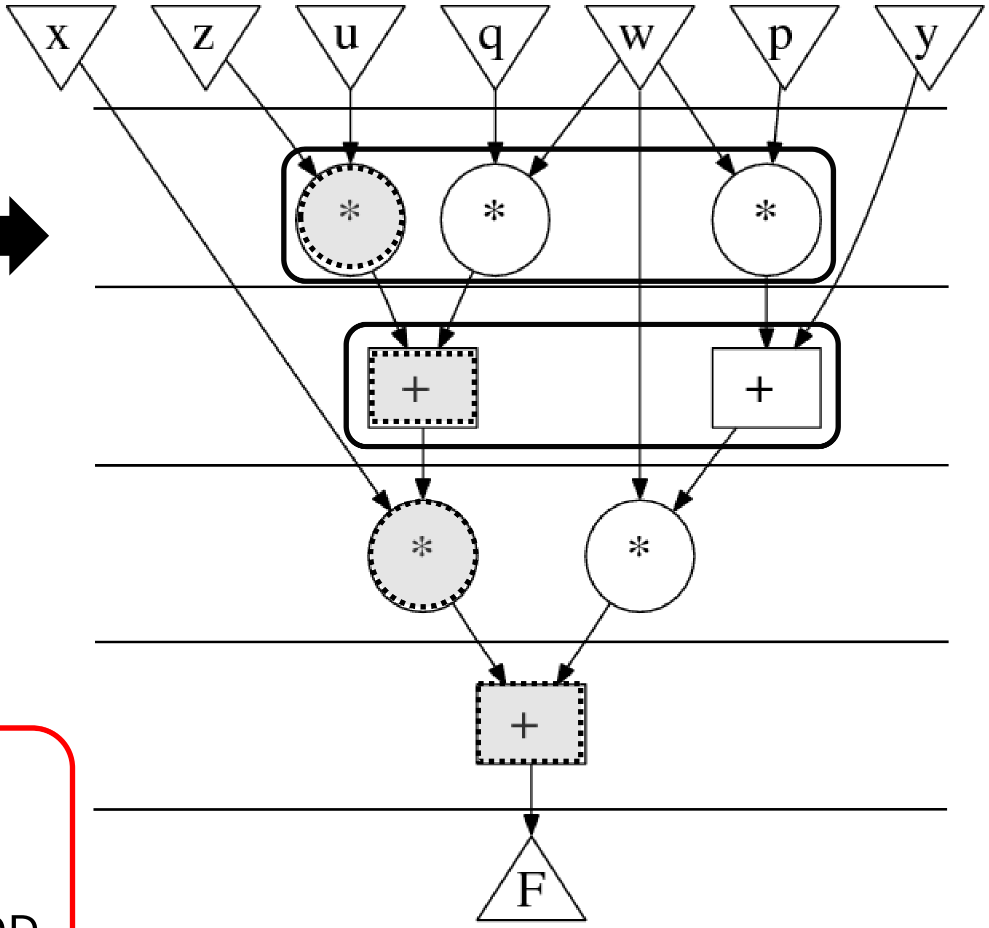


Data Flow Graph



NFF

total: *5× , 3+*



Reordering cost

Path delay $2d_{MPY}+2d_{ADD}$

Latency 4 steps

Resource 3MPY & 2ADD

Functional Retiming in TED

The retime operator $()^R$ is lineal because it is the convolution * with the time delayed delta Dirac δ.

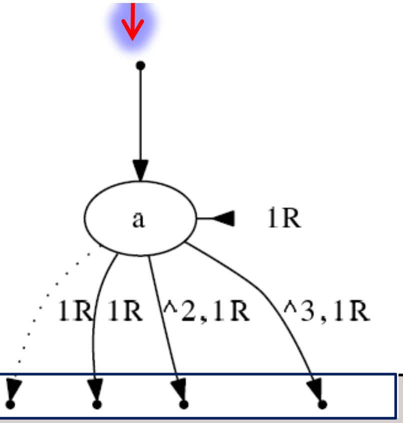
Because $()^R$ is a lineal operator, it can be applied to the TED graph.

Taylor decomposition:

$$G = G_{(a=0)} + a\frac{\partial G}{\partial a}_{|a=0} + a^2\frac{\partial^2 G}{\partial a^2}_{|a=0} + a^3\frac{\partial^3 G}{\partial a^3}_{|a=0}$$

Lineal operator $()^R$:

$$F = G^{1R}_{(a=0)} + a^{1R}\left(\frac{\partial G}{\partial a}\right)^{1R}_{|a=0} + (a^{1R})^2\left(\frac{\partial^2 G}{\partial a^2}\right)^{1R}_{|a=0} + (a^{1R})^3\left(\frac{\partial^3 G}{\partial a^3}\right)^{1R}_{|a=0}$$



Example of Functional Retiming

$$F = d^{1R}(b+a)^{1R} + (ca+bc)^{1R}$$

Classic data path retiming

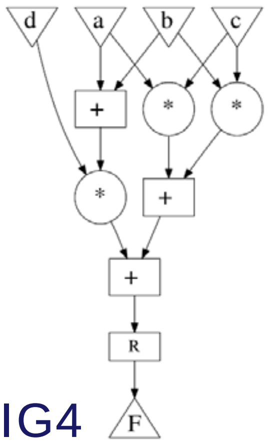


FIG4

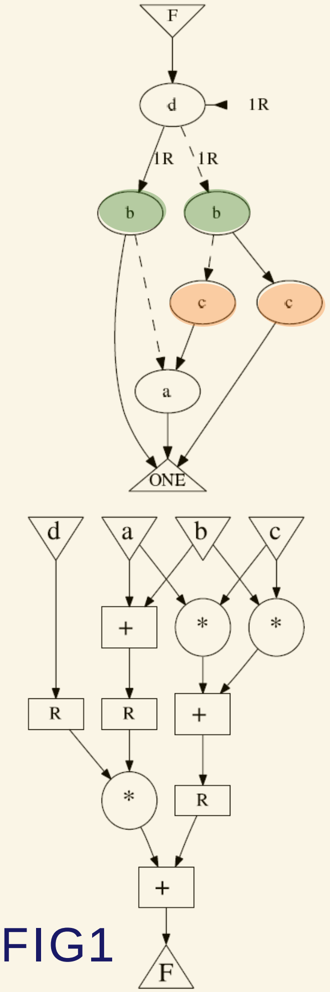


FIG1

ordering

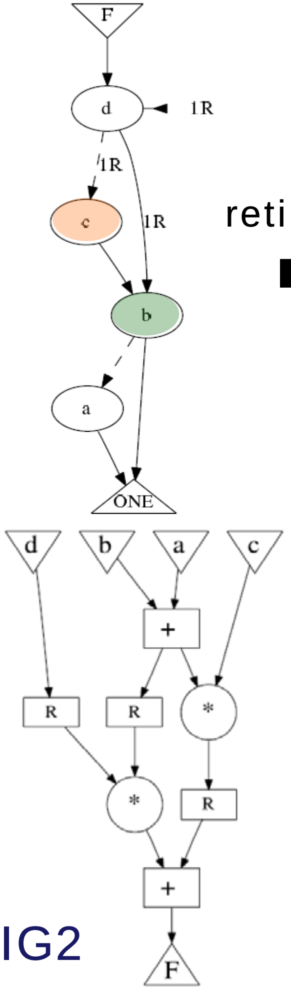


FIG2

retiming

$$F = \left(d(b+a) + ca + bc\right)^{1R}$$

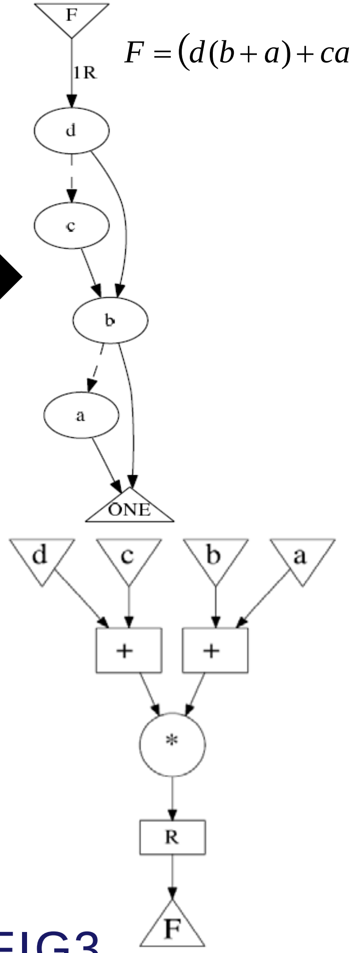
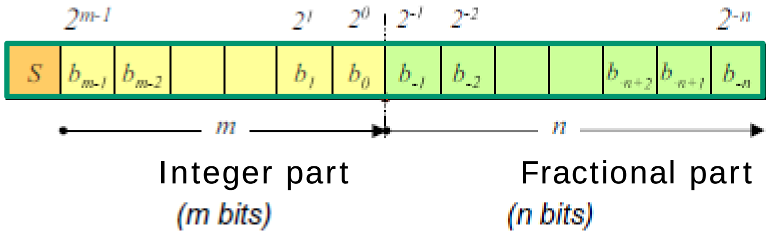


FIG3

Fixed Point Arithmetic

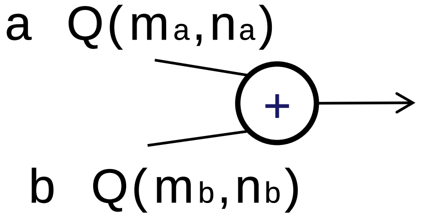


$$x = -2^m S + \sum_{i=-n}^{m-1} b_i 2^i$$

$$q = 2^{-(n)}$$

$$\mathcal{D} = [-2^m; 2^m - q]$$

Addition


a $Q(m_a,n_a)$

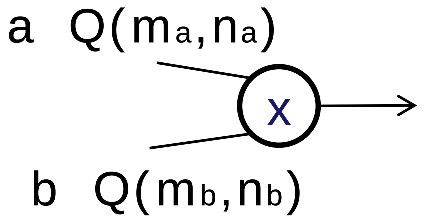
$+$

c

b $Q(m_b,n_b)$

$$n_c = max(n_a, n_b)$$
$$m_c = max(m_a, m_b) + 1$$

Multiplication


a $Q(m_a,n_a)$

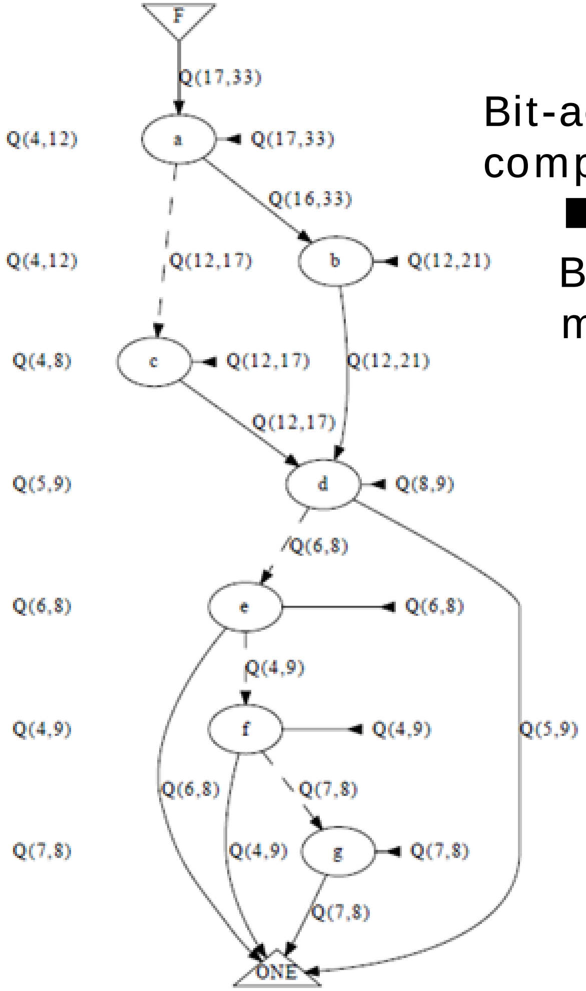
x

c

b $Q(m_b,n_b)$

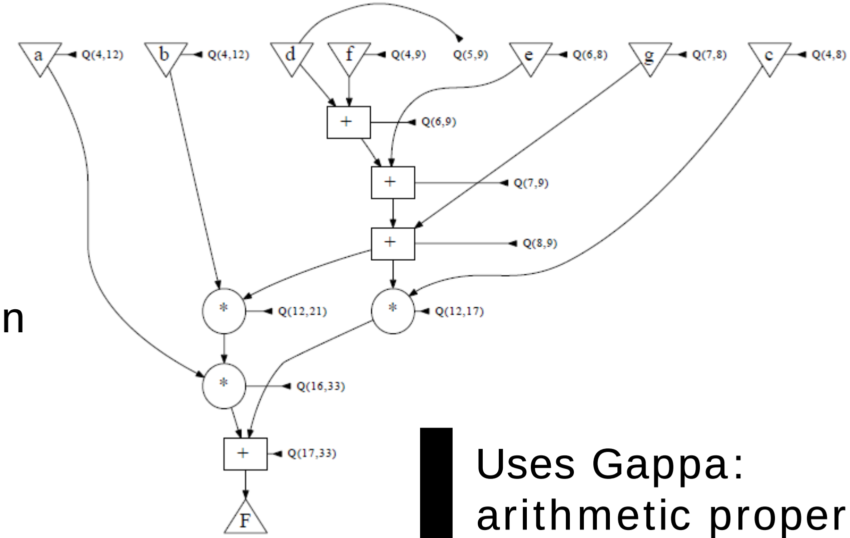
$$n_c = n_a + n_b$$
$$m_c = m_a + m_b$$

TED With Fixed Point Arithmetic



Bit-accurate computation

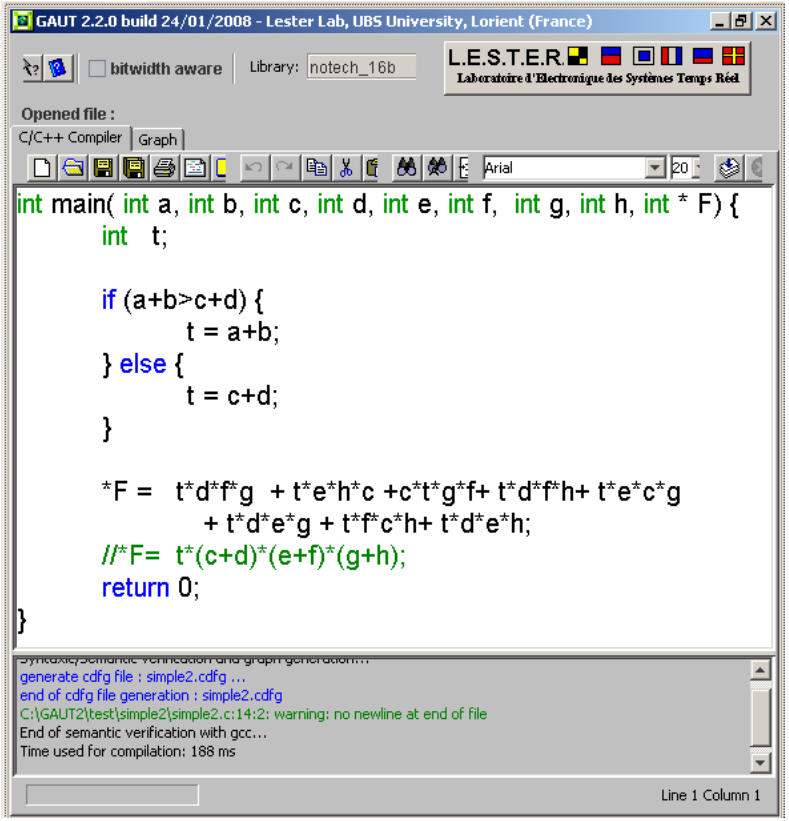
Bit-width minimization



Uses Gappa: arithmetic property tool

DFG optimization under accuracy constraints
HW accuracy certification

Integration with GAUT



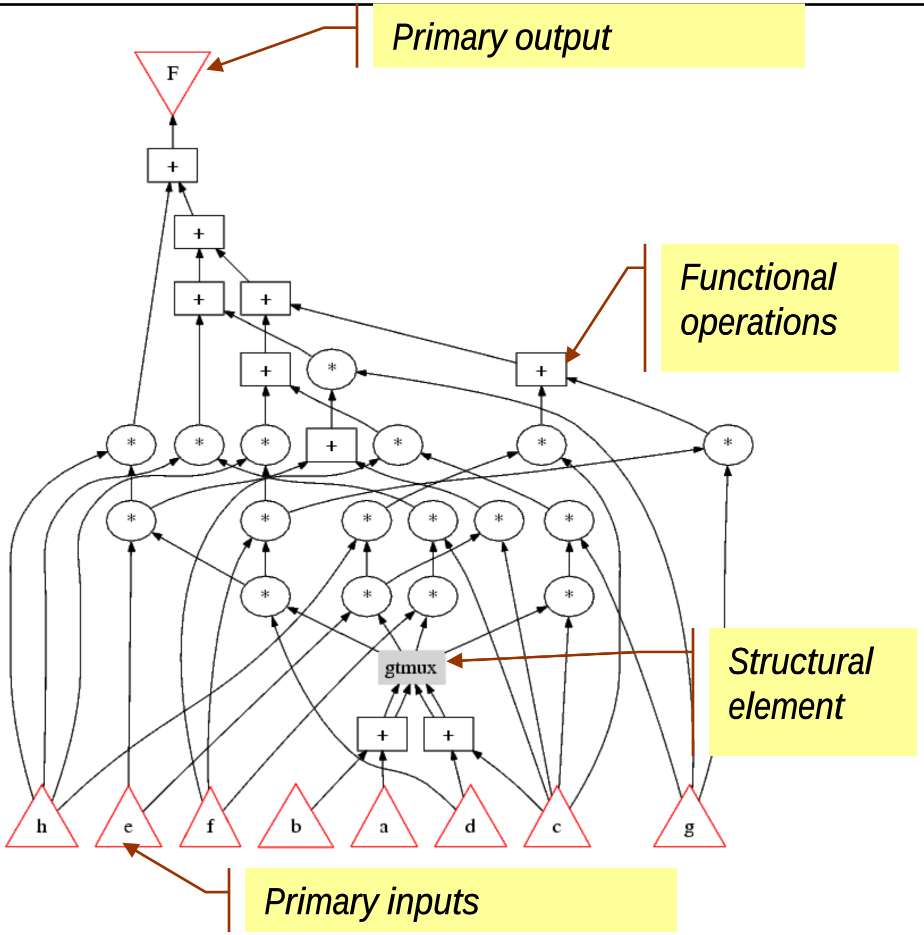
GAUT 2.2.0 build 24/01/2008 - Lester Lab, UBS University, Lorient (France)
bitwidth aware
Library:
notech_16b
L.E.S.T.E.R.
Laboratoire d'Electronique des Systèmes Temps Réel
Opened file :
C/C++ Compiler
Graph
Arial
20

```
int main( int a, int b, int c, int d, int e, int f,  int g, int h, int * F) {
        int   t;

        if (a+b>c+d) {
                t = a+b;
        } else {
                t = c+d;
        }

        *F =   t*d*f*g  + t*e*h*c +c*t*g*f+ t*d*f*h+ t*e*c*g
                  + t*d*e*g + t*f*c*h+ t*d*e*h;
        //*F=  t*(c+d)*(e+f)*(g+h);
        return 0;
}
```

```
Syntaxic/Semantic verification and graph generation...
generate cdfg file : simple2.cdfg ...
end of cdfg file generation : simple2.cdfg
C:\GAUT2\test\simple2\simple2.c:14:2: warning: no newline at end of file
End of semantic verification with gcc...
Time used for compilation: 188 ms
```

Line 1 Column 1

Example behavioral design in C



Primary output

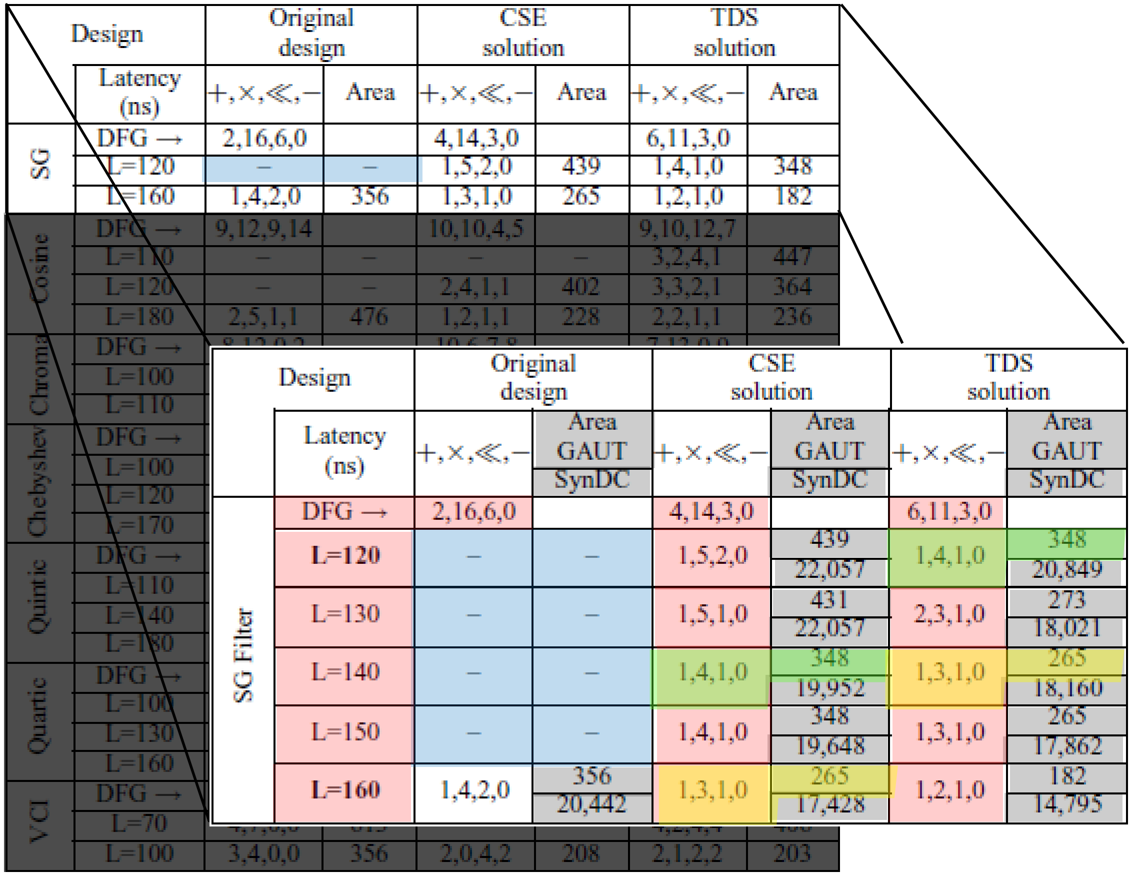
Functional operations

Structural element

Primary inputs

Initial TDS network

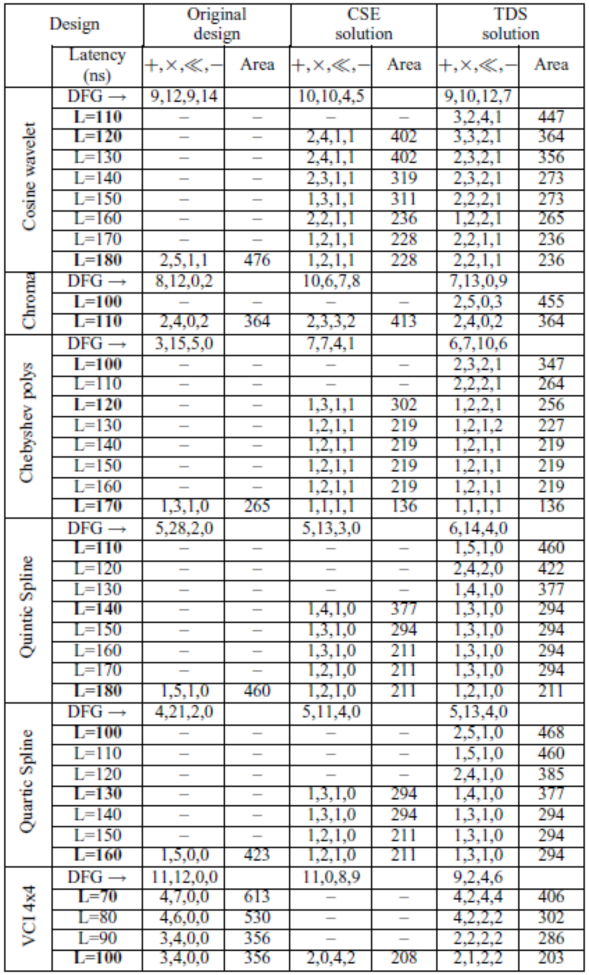
Experiments & Results



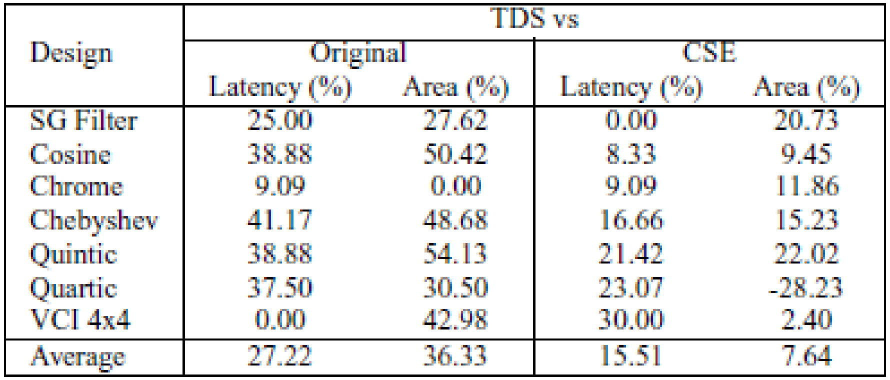
Design		Original design		CSE solution		TDS solution	
	Latency (ns)	+,×,≪,−	Area	+,×,≪,−	Area	+,×,≪,−	Area
SG	DFG →	2,16,6,0		4,14,3,0		6,11,3,0	
	L=120	–	–	1,5,2,0	439	1,4,1,0	348
	L=160	1,4,2,0	356	1,3,1,0	265	1,2,1,0	182
Cosine	DFG →	9,12,9,14		10,10,4,5		9,10,12,7	
	L=110	–	–	–	–	3,2,4,1	447
	L=120	–	–	2,4,1,1	402	3,3,2,1	364
	L=180	2,5,1,1	476	1,2,1,1	228	2,2,1,1	236
Chroma	DFG →						
	L=100						
	L=110						
Chebyshev	DFG →						
	L=100						
	L=120						
	L=170						
Quintic	DFG →						
	L=110						
	L=140						
	L=180						
Quartic	DFG →						
	L=100						
	L=130						
	L=160						
VCI	DFG →						
	L=70						
	L=100	3,4,0,0	356	2,0,4,2	208	2,1,2,2	203

Design		Original design		CSE solution		TDS solution	
	Latency (ns)	+,×,≪,−	Area GAUT SynDC	+,×,≪,−	Area GAUT SynDC	+,×,≪,−	Area GAUT SynDC
SG Filter	DFG →	2,16,6,0		4,14,3,0		6,11,3,0	
	L=120	–	–	1,5,2,0	439 22,057	1,4,1,0	348 20,849
	L=130	–	–	1,5,1,0	431 22,057	2,3,1,0	273 18,021
	L=140	–	–	1,4,1,0	348 19,952	1,3,1,0	265 18,160
	L=150	–	–	1,4,1,0	348 19,648	1,3,1,0	265 17,862
	L=160	1,4,2,0	356 20,442	1,3,1,0	265 17,428	1,2,1,0	182 14,795

Improvement over CSE and Original Design

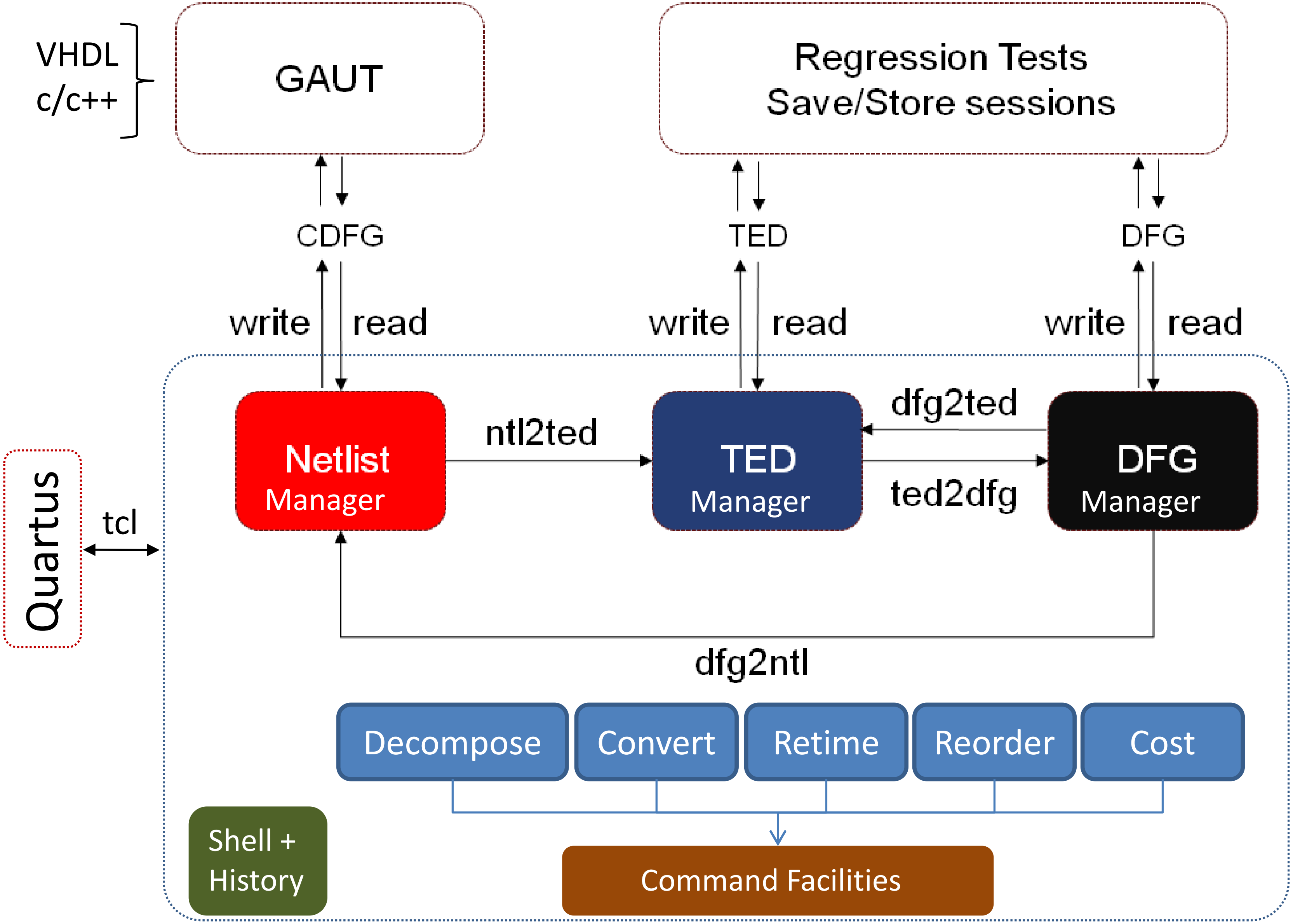


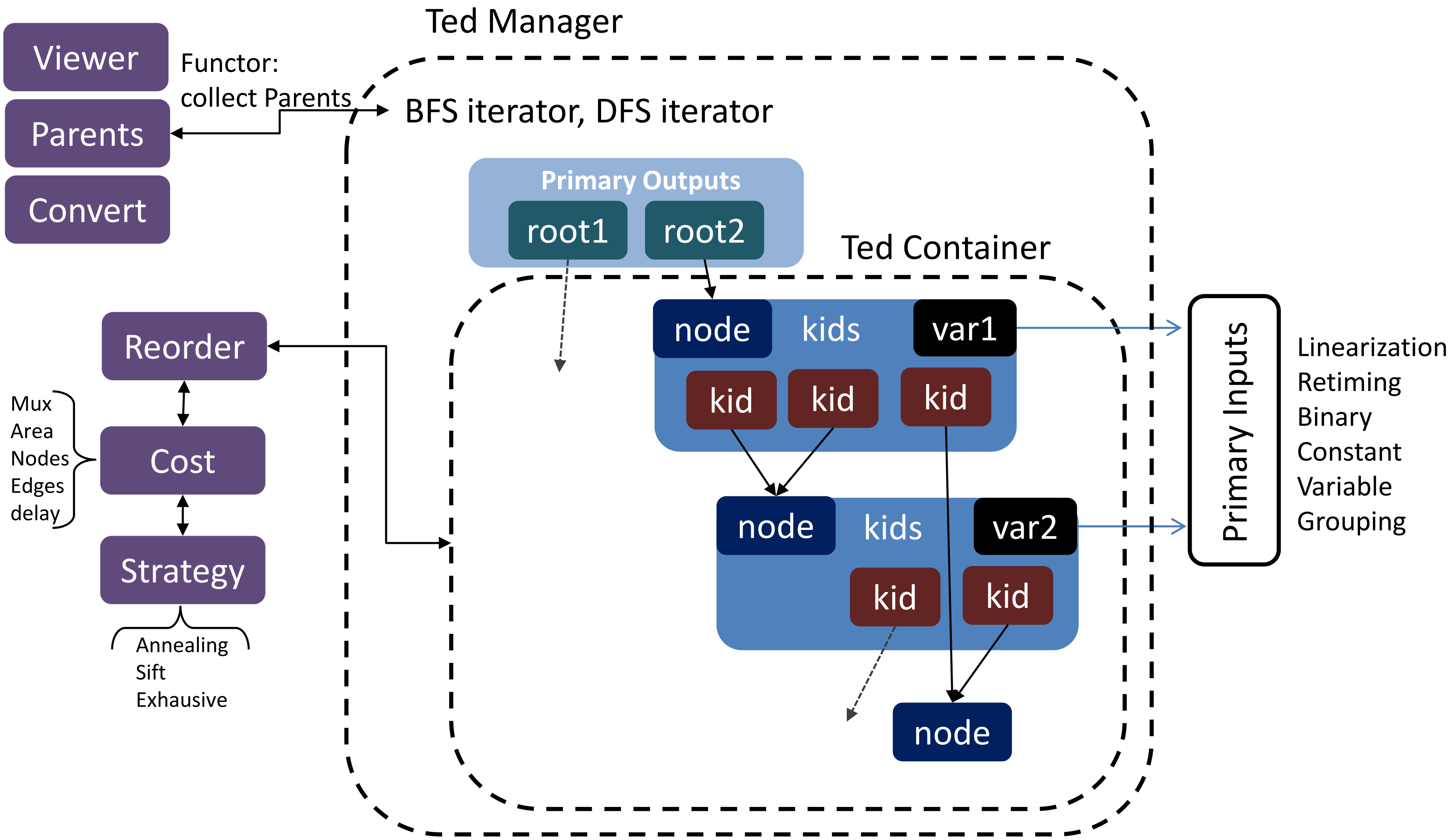
Design		Original design		CSE solution		TDS solution	
	Latency (ns)	+,×,≪,−	Area	+,×,≪,−	Area	+,×,≪,−	Area
Cosine wavelet	DFG →	9,12,9,14		10,10,4,5		9,10,12,7	
	L=110	–	–	–	–	3,2,4,1	447
	L=120	–	–	2,4,1,1	402	3,3,2,1	364
	L=130	–	–	2,4,1,1	402	2,3,2,1	356
	L=140	–	–	2,3,1,1	319	2,3,2,1	273
	L=150	–	–	1,3,1,1	311	2,2,2,1	273
	L=160	–	–	2,2,1,1	236	1,2,2,1	265
	L=170	–	–	1,2,1,1	228	2,2,1,1	236
	L=180	2,5,1,1	476	1,2,1,1	228	2,2,1,1	236
Chroma	DFG →	8,12,0,2		10,6,7,8		7,13,0,9	
	L=100	–	–	–	–	2,5,0,3	455
	L=110	2,4,0,2	364	2,3,3,2	413	2,4,0,2	364
Chebyshev polys	DFG →	3,15,5,0		7,7,4,1		6,7,10,6	
	L=100	–	–	–	–	2,3,2,1	347
	L=110	–	–	–	–	2,2,2,1	264
	L=120	–	–	1,3,1,1	302	1,2,2,1	256
	L=130	–	–	1,2,1,1	219	1,2,1,2	227
	L=140	–	–	1,2,1,1	219	1,2,1,1	219
	L=150	–	–	1,2,1,1	219	1,2,1,1	219
	L=160	–	–	1,2,1,1	219	1,2,1,1	219
	L=170	1,3,1,0	265	1,1,1,1	136	1,1,1,1	136
Quintic Spline	DFG →	5,28,2,0		5,13,3,0		6,14,4,0	
	L=110	–	–	–	–	1,5,1,0	460
	L=120	–	–	–	–	2,4,2,0	422
	L=130	–	–	–	–	1,4,1,0	377
	L=140	–	–	1,4,1,0	377	1,3,1,0	294
	L=150	–	–	1,3,1,0	294	1,3,1,0	294
	L=160	–	–	1,3,1,0	211	1,3,1,0	294
	L=170	–	–	1,2,1,0	211	1,3,1,0	294
	L=180	1,5,1,0	460	1,2,1,0	211	1,2,1,0	211
Quartic Spline	DFG →	4,21,2,0		5,11,4,0		5,13,4,0	
	L=100	–	–	–	–	2,5,1,0	468
	L=110	–	–	–	–	1,5,1,0	460
	L=120	–	–	–	–	2,4,1,0	385
	L=130	–	–	1,3,1,0	294	1,4,1,0	377
	L=140	–	–	1,3,1,0	294	1,3,1,0	294
	L=150	–	–	1,2,1,0	211	1,3,1,0	294
	L=160	1,5,0,0	423	1,2,1,0	211	1,3,1,0	294
VCI 4x4	DFG →	11,12,0,0		11,0,8,9		9,2,4,6	
	L=70	4,7,0,0	613	–	–	4,2,4,4	406
	L=80	4,6,0,0	530	–	–	4,2,2,2	302
	L=90	3,4,0,0	356	–	–	2,2,2,2	286
	L=100	3,4,0,0	356	2,0,4,2	208	2,1,2,2	203



Design	TDS vs			
	Original		CSE	
	Latency (%)	Area (%)	Latency (%)	Area (%)
SG Filter	25.00	27.62	0.00	20.73
Cosine	38.88	50.42	8.33	9.45
Chrome	9.09	0.00	9.09	11.86
Chebyshev	41.17	48.68	16.66	15.23
Quintic	38.88	54.13	21.42	22.02
Quartic	37.50	30.50	23.07	-28.23
VCI 4x4	0.00	42.98	30.00	2.40
Average	27.22	36.33	15.51	7.64

Part II: Data Structure




Ted Manager

Viewer

Parents

Convert

Functor:
collect Parents

BFS iterator, DFS iterator

Primary Outputs

root1

root2

Ted Container

node

kids

var1

kid

kid

kid

node

kids

var2

kid

kid

node

Reorder

Cost

Strategy

Mux
Area
Nodes
Edges
delay

Annealing
Sift
Exhausive

Primary Inputs

Linearization
Retiming
Binary
Constant
Variable
Grouping

Thanks ...