CORDIC Accuracy
General

20160125

Fixed Point Simulation

- try to use Octave first

Statistical Analysis

- learn to use ggplot (R)
- Renaming class (C++ implementation)

GpData → RptData

accuracy

- Note.1.General
- Note.2.Stat
- Note.3.Octave → fixed point + etc...
- Note.4.Scaling (K) → scale free CORDIC

idea

- Note.5.Search
- Note.6.LookAhead
- Note.7.Backtrack
- Note.8.Serialize
- Note.9.Precision

CORDIC Accuracy & Precision [edit]

CORDIC.AccPrec (pdf)

C++ Codes for Accuracy and Precision Measurement [edit]

- CORDIC Source (pdf)
  - Makefile (pdf)
  - Core class (pdf)
  - Angles class (pdf, pdf)
  - GPData class (pdf)
  - Figures class (pdf)
  - Interfacing GHDL CORDIC simulation with C (pdf)
  - Calling C++ cordic function from C (pdf)
- batch run bash file for Angles_tb (pdf)
  - fig_basic (pdf, note)
  - fig_tscale (pdf)
  - fig_uscale (pdf)

Testbench Codes and Results [edit]

- cordic testbenches (pdf)
  - cordic testbench 01 (percent error)
  - cordic testbench 02 (path error)
  - cordic testbench 03 (varying tree levels)
    - fig_basic (pdf)
    - fig_tscale (pdf)
    - fig_uscale (pdf)
  - cordic testbench 04 (coarse-fine)
  - batch run bash file for testbench 01 (pdf)
  - batch run bash file for testbench 02 (pdf)
- Angles_wx using wxWidgets & wxGlade (pdf)

CORDIC Accuracy Notes [edit]

1. General (pdf)
2. Statistical Analysis (pdf)
3. Octave Fixed Point Simulation (pdf)
4. Scaling K (pdf)

Idea Sketch [edit]

CORDIC as a Search Algorithm [edit]

- CORDIC as a Search Idea.3.A (pdf)

  5. Search (pdf)

Quad Angle Tree Based CORDIC [edit]

- CORDIC Quad Tree Angles (pdf)

  6. Lookahead (pdf)

Minimizing Latency [edit]

- Latency Minimizing Idea.2.A (pdf)

  7. Backtrack (pdf)

Maximizing Throughput [edit]

- Throughput Maximizing Idea.1.A (pdf)

  8. Serialize (pdf)

Bit-Serial & Bit-Parallel Trade-offs [edit]

- Generalized Multi-Byte CORDIC Idea.4.A (pdf)

  9. Precision (pdf)

② Statistical Analysis

Dense Angle Area

Block & Offset Views

Subtrees and Indices

Residue Angle Distribution

Overlapping & Non-overlapping Region

Jitter Angle

Test Patterns

tscale simulation -- the angle points are the optimal angles to be found

Is there any tendancy in the found angles
-- always larger or smaller or random
-- find out the meaning of the slightly overdamped system

uscale simulation - search the angle point list angles in all node list
and find the one which minimizes the mean squared errors

Test Patterns

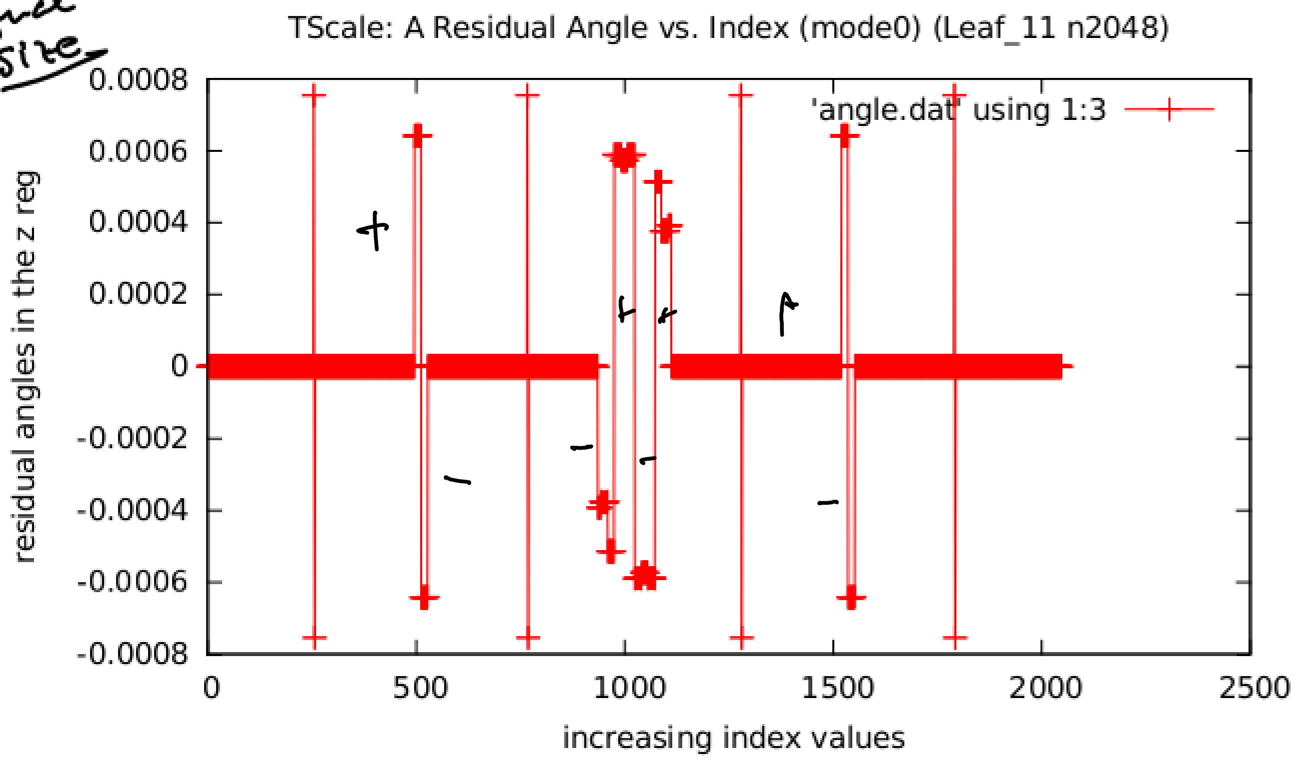
- Random data
  - uniform distribution
  - gaussian distribution centered on dense area
- uniform scale data more resolution
  - increasing order
  - decreasing order
  - strided access
- tree scale data
  - all nodes
  - leaf node only
  - shifted combinations of angle points

- tree scale data
  - to know how cordic works well for a known set of angles

- increasing order
- decreasing order
- strided access

☆ (think about "incremental compile"
is there any way to exploit current iterations for the next angle?)

parallel, pipeline .....



with break

without break

③ Octave : Fixed point simulation

Binary Angle Routine

Level

Angles

Sorting

Angle Spacing & Resolution

representative angle spacing value?

④ Scale Free

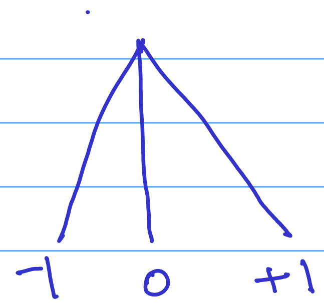
Scaling Constant K

K After initial phase

Ternary Tree

Ternary Tree

after enough iterations,
constant scaling
problem vanishes



⑤ # Search

Greedy algorithm.

Optimal Solution

DFS/BFS

Heuristics

Cost functions

Optimal Solution

for a given set of angle points

how well corzpic algorithm finds

optimum value.

min { sampling error + quantization error }

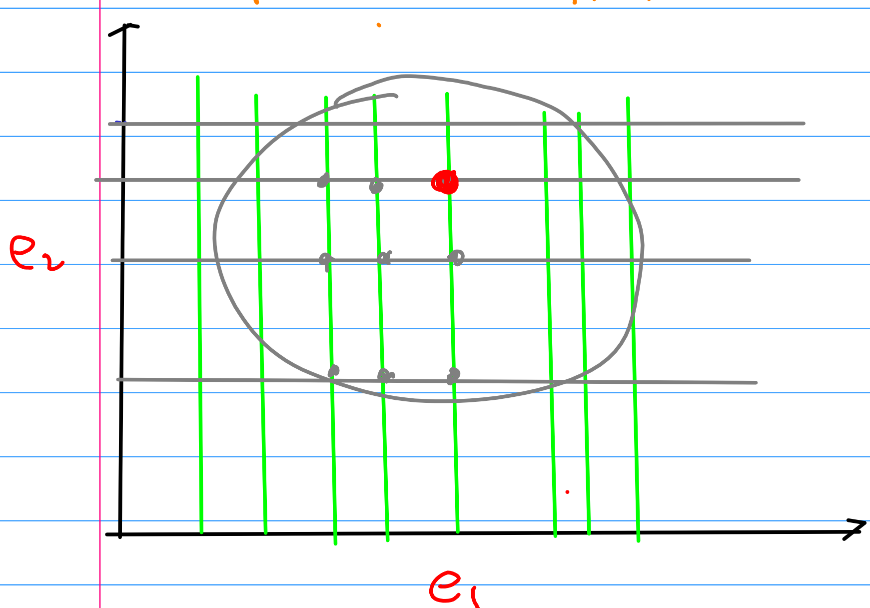
what is the optimum value

↳ search m.s.e

for all angle values

and find the minimum

Optimal solution



a heuristic

① $\min\{e_1\}$

② $\min\{e_2\}$

③ $\min\{e_1 + e_2\}$

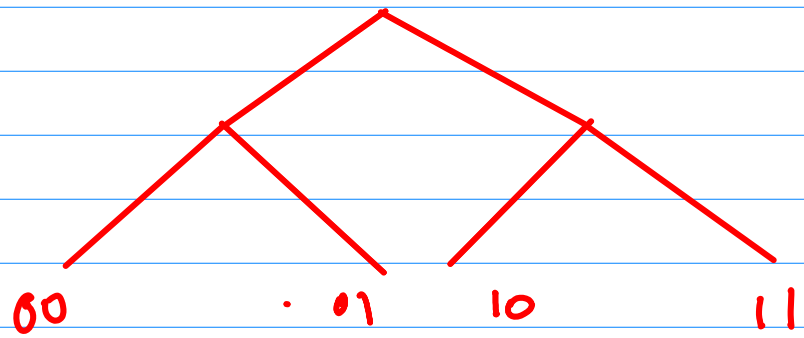
⑥ Look Ahead

Coarse - Fine Approach

2-Step Lookahead (Quad Tree)

Number of Adders

Look ahead 2 Steps



$m_0 = x + y$

$m_1 = x - y = x + \bar{y} + 1$

$m_2 = -x + y = -(x - y) = \bar{x} + y + 1$

$m_3 = -x - y = -(x + y) = \bar{x} + \bar{y} + 2$

$x + y \rightarrow -(x + y) = -x - y$

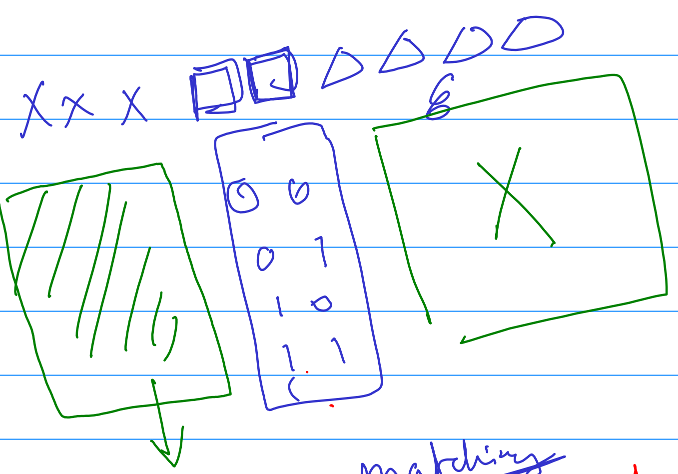
$x - y \rightarrow -(x - y) = -x + y$

$|\theta - m_0|$

$|\theta - m_1|$

$|\theta - m_2|$

$|\theta - m_3|$



Same

~~matching~~

need to check!

7

Backtracking

when to backtrack

how to backtrack?

Comparators

(a) Brute force, traditional backtracking

(b) Heuristics

avoid "dense" angle?

0, 45°, 90°, 135° ......

(8) Serialize

Serialized Addition

Carry Save Adder

Peak Power

⑨ Precision

Data Parallelism

Multi-word precision Multi-task

Objectives: High Precision, min area & power

Resource Sharing

Carry Propagate Network

Space - Time Optimization

Multi CORDIC with initial phase shifting