Wavelet-based Compression on the FPGA

Dawn Williams

IceCube/AMANDA Collaboration Meeting

Berkeley

Mar. 20, 2005

Motivation

- IceTop has higher fraction of complex, multi-SPE waveforms than in-ice
- Current “road-grader” algorithm keeps samples above a threshold, sets other samples to zero, run length encoding
- Road grader average compression ratio on prototype IceTop waveforms is 3:1
- Want a robust, integer-arithmetic-based compression algorithm for the FPGA

Part 1: Discrete Wavelet Transform

- Analagous to Fourier transform
- <u>finite</u> basis function
- translate/dilate basis
- Haar wavelet: basis is step function
- continuous-->discrete
- Wavelets have many applications, lots of literature

“Lifting” Scheme

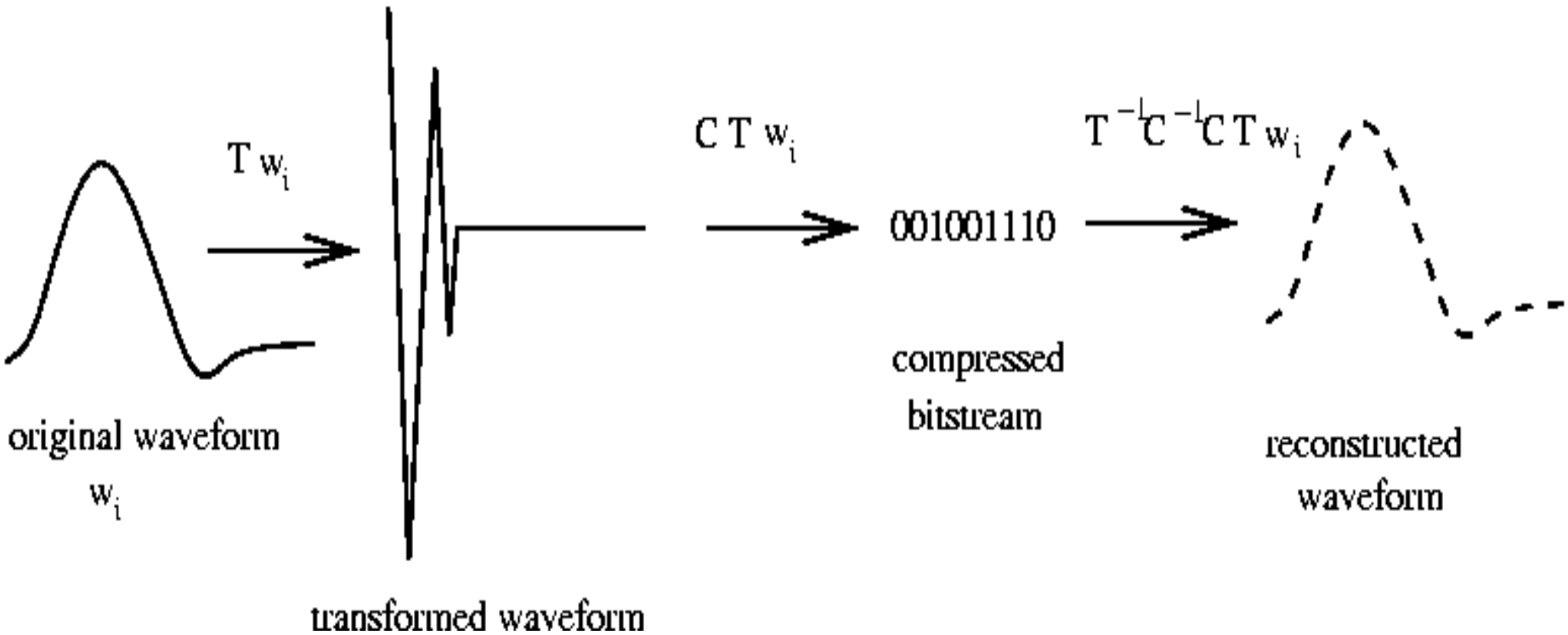
(Daubechies & Sweldens)

- Split signal into even & odd samples, use odd to predict even
  - Step 1: $d = x_o - P(x_e)$ “detail” coefficients
    P is prediction operator
  - Step 2: $s = x_e + U(d)$ “smoothed” coefficients
    U is update operator
  - Repeat with “s” coefficients
- Lifting scheme is invertible
- For Haar wavelet, U = -1, P= -1/2

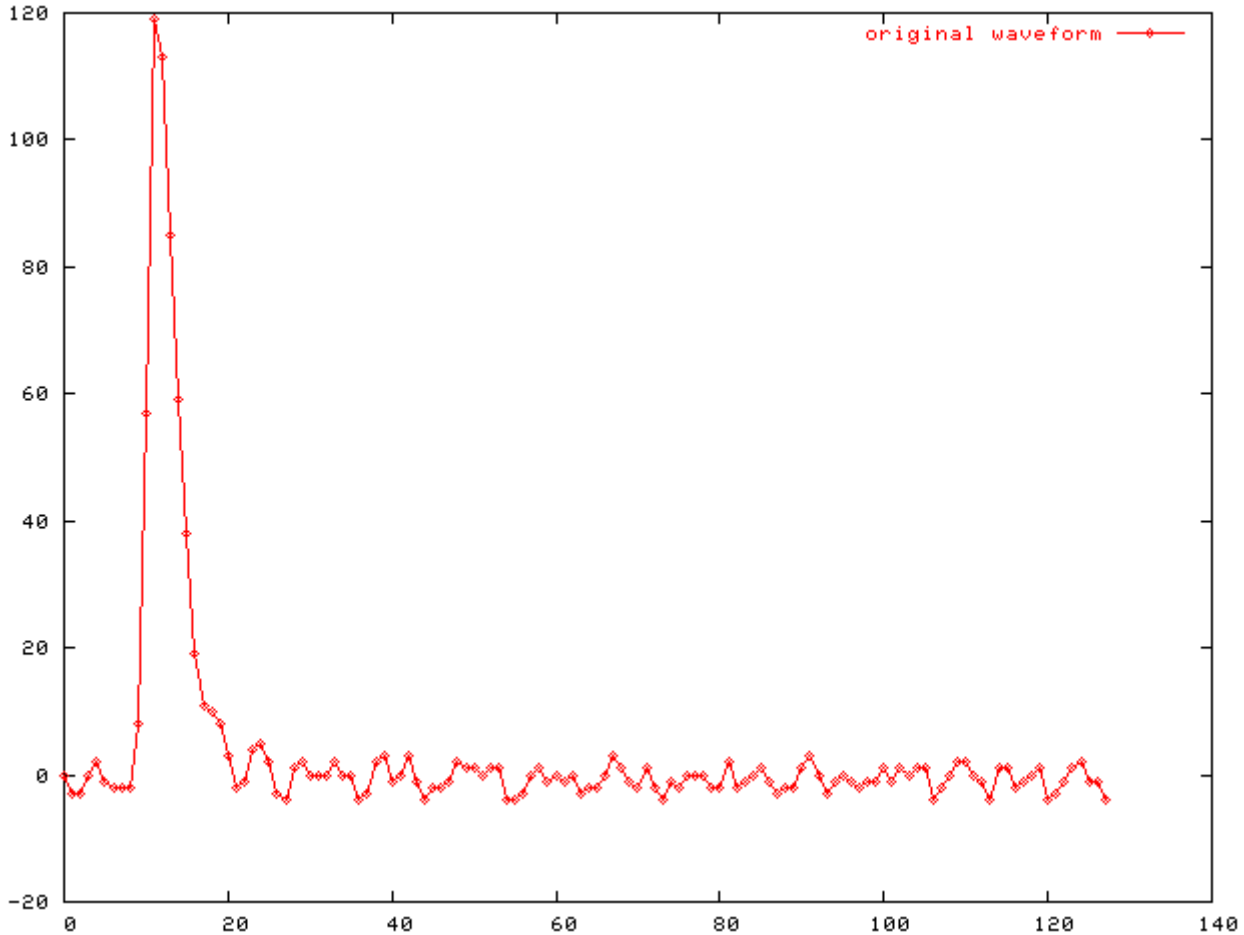
Set Partitioning In Hierarchical Trees
(new since Uppsala)

- Wavelet transform itself does not compress
- SPIHT algorithm designed to work with wavelet-transformed data, but works with raw data at a lower compression ratio
- Basic idea: divide data into sets, encode only sets/members of sets above a certain threshold
  - Threshold is only parameter to be adjusted
- Outputs a stream of bits (further lossless compression ?)

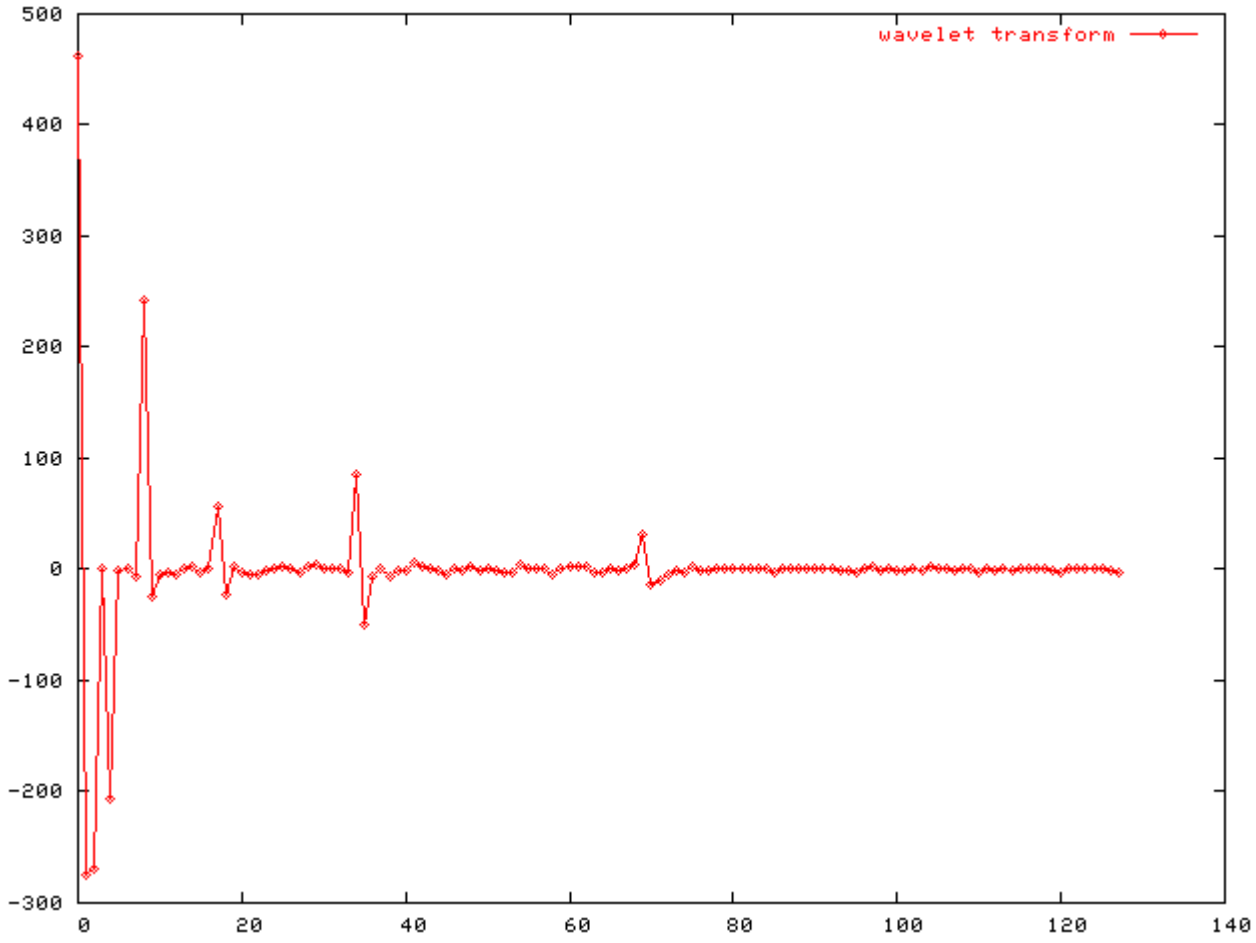
Wavelet-SPIHT Compression Process



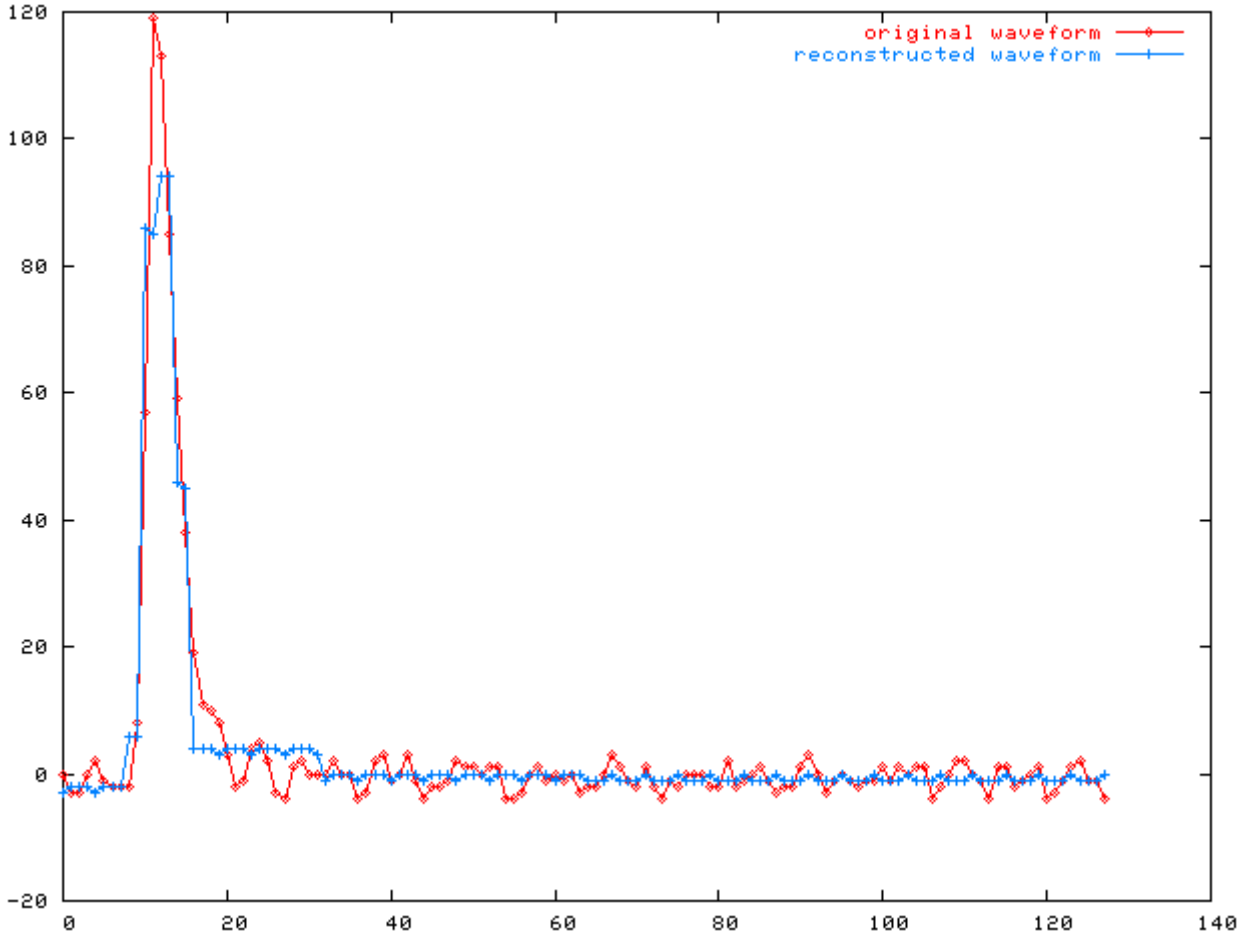
Example: SPE waveform (dark noise from DOM at PSU)



Example: wavelet-transformed waveform

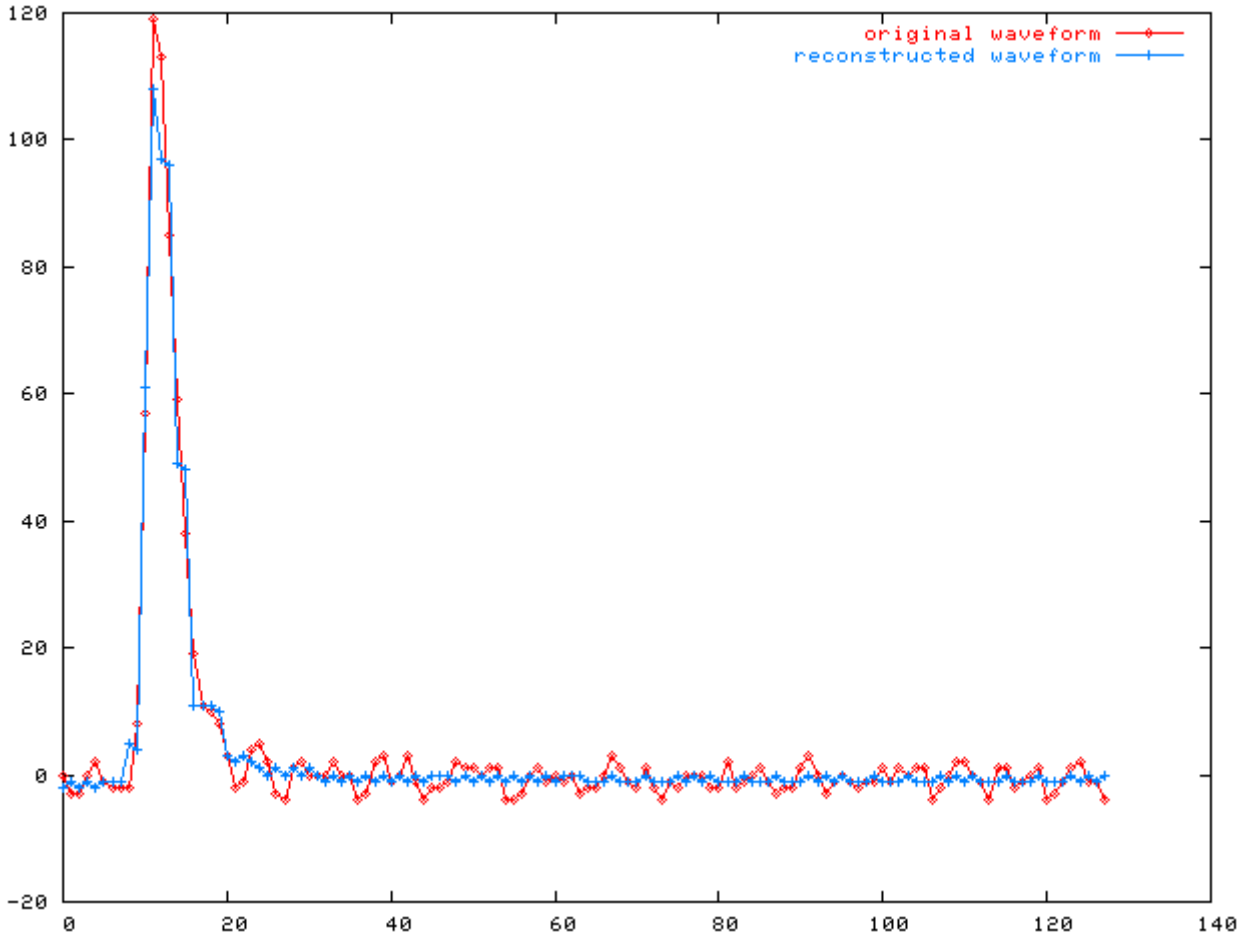


Example: reconstructed waveforms



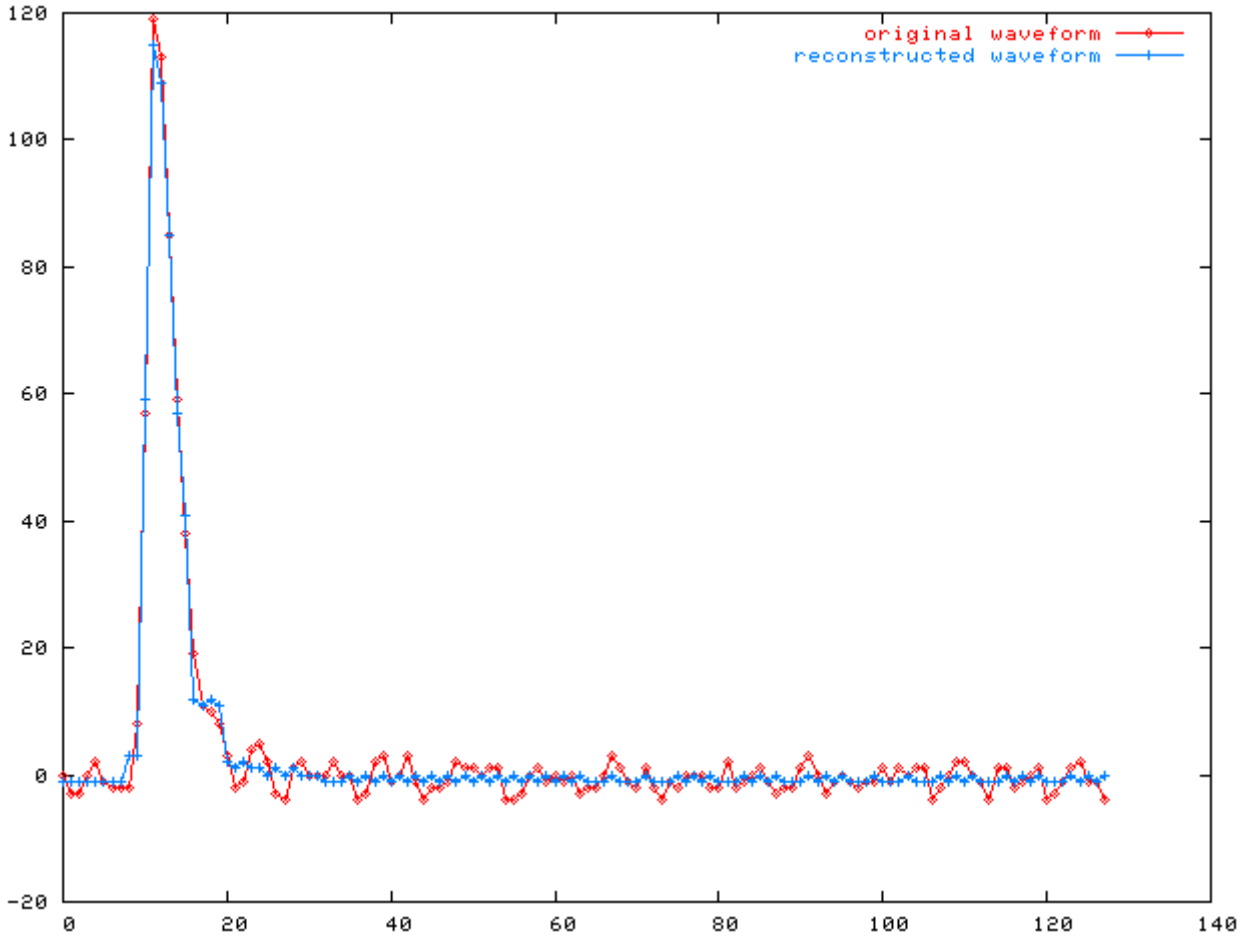
Compression ratio

14:1

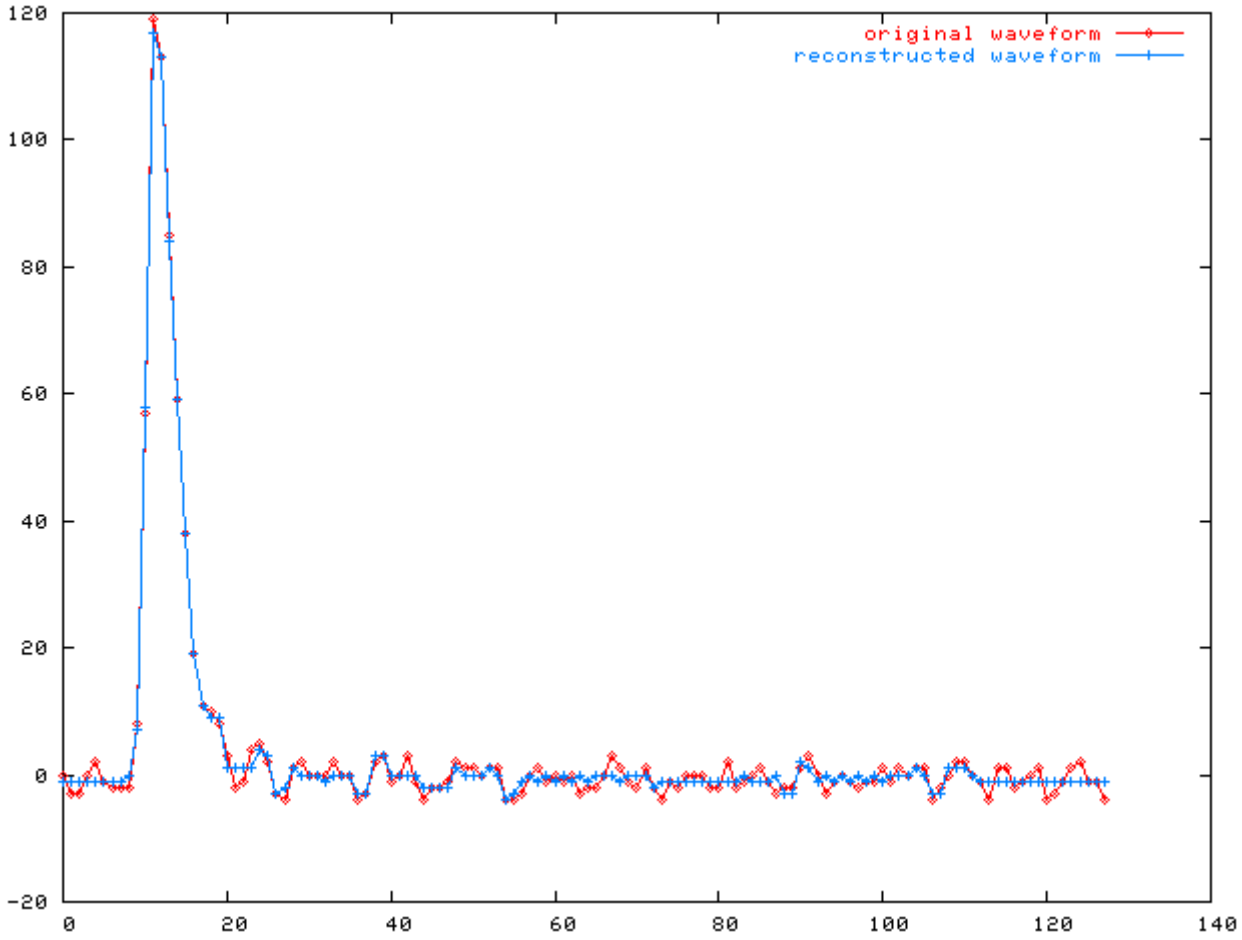


11:1

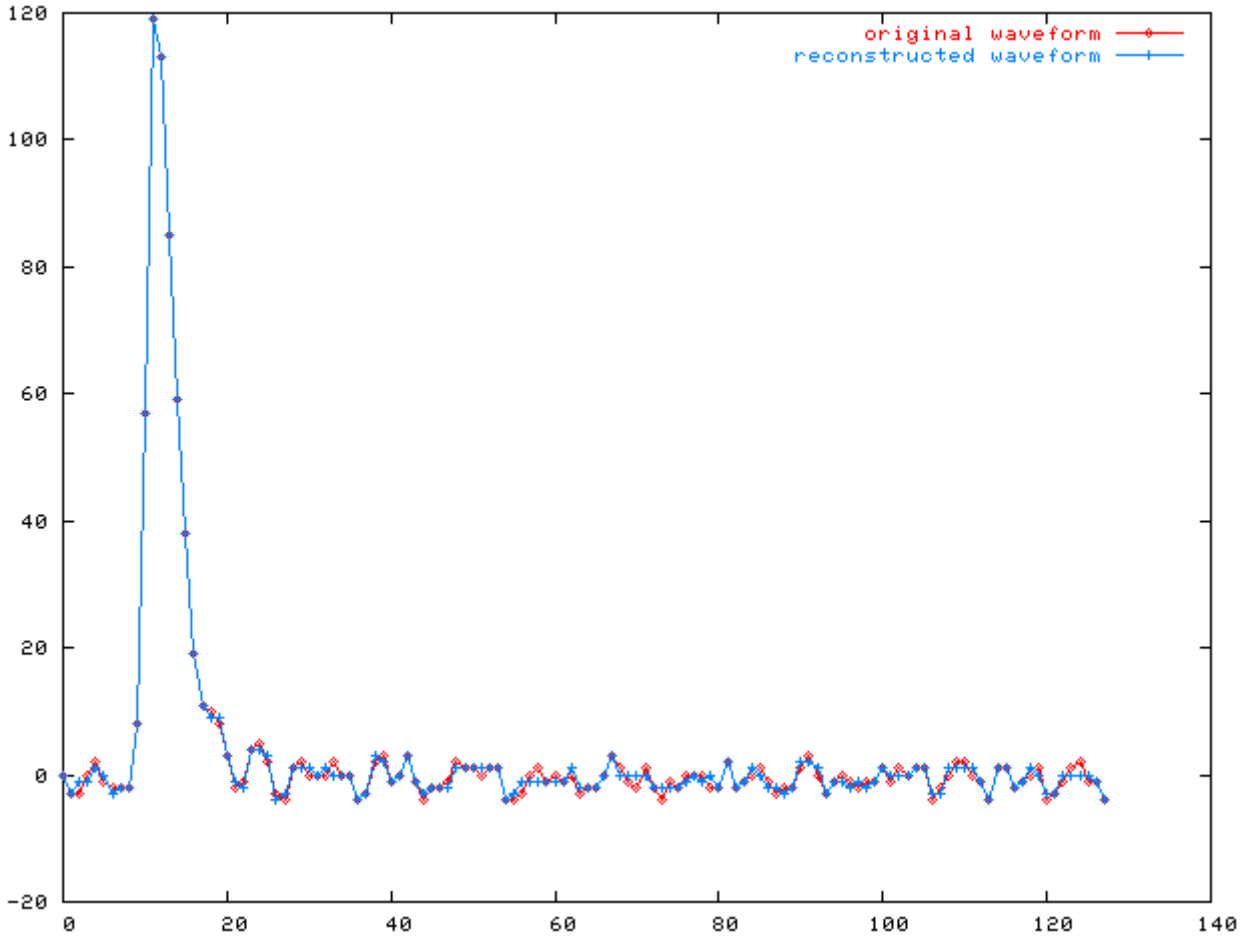
same as road-grader


original waveform
reconstructed waveform
120
100
80
60
40
20
0
-20
0
20
40
60
80
100
120
140

6:1

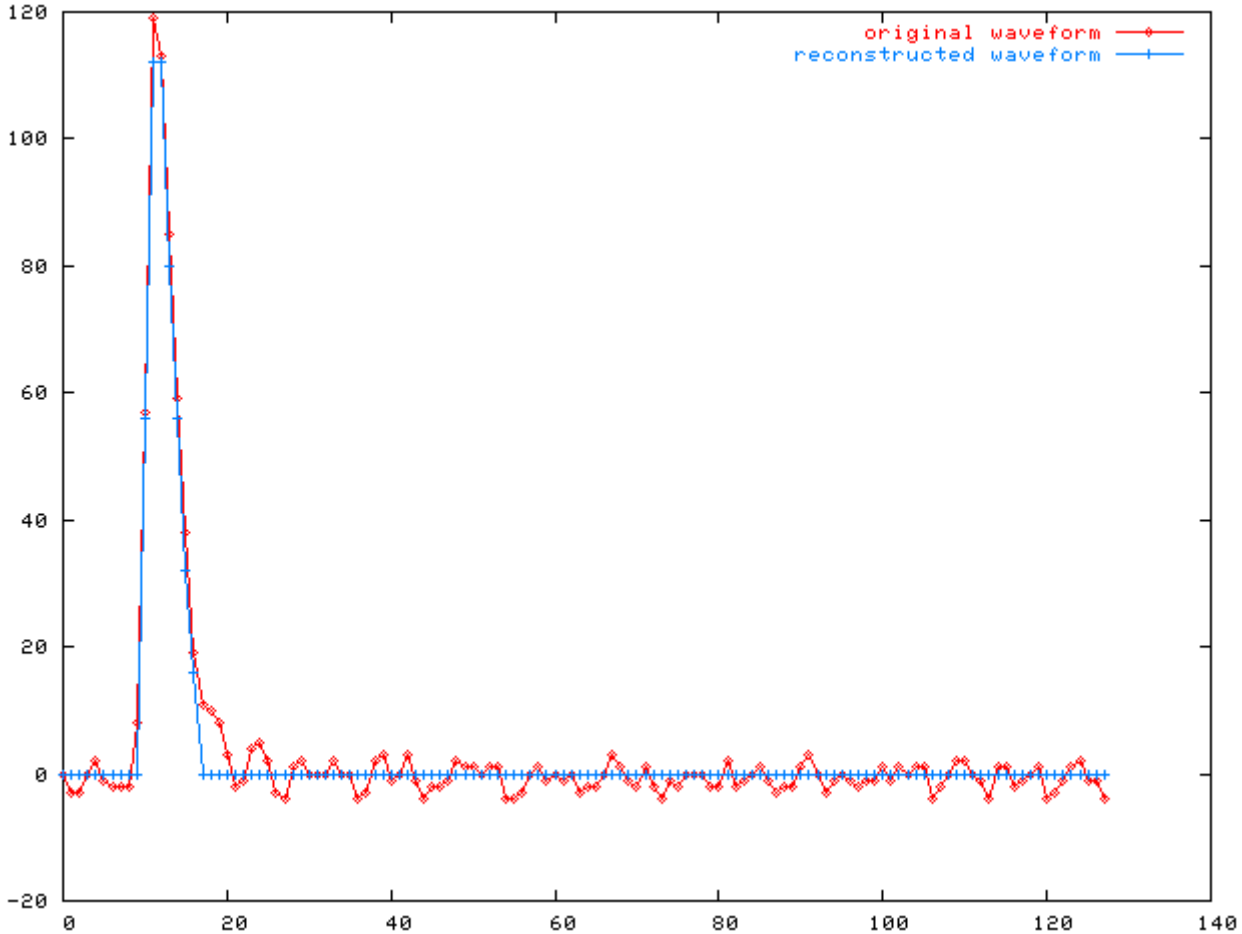

original waveform
reconstructed waveform
120
100
80
60
40
20
0
-20
0
20
40
60
80
100
120
140

3:1


original waveform
reconstructed waveform
120
100
80
60
40
20
0
-20
0
20
40
60
80
100
120
140

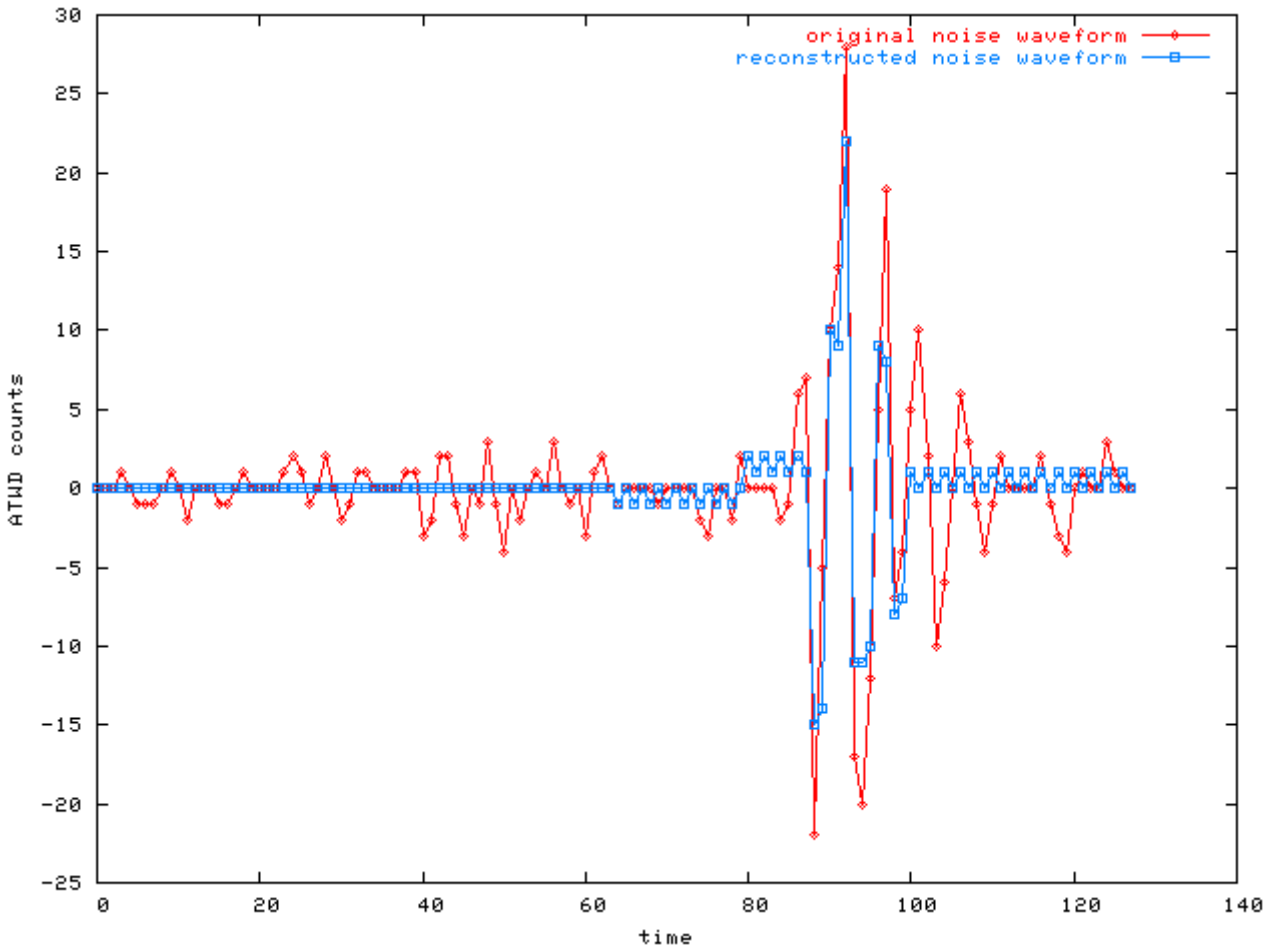
2:1

Reconstructed waveform, SPIHT only



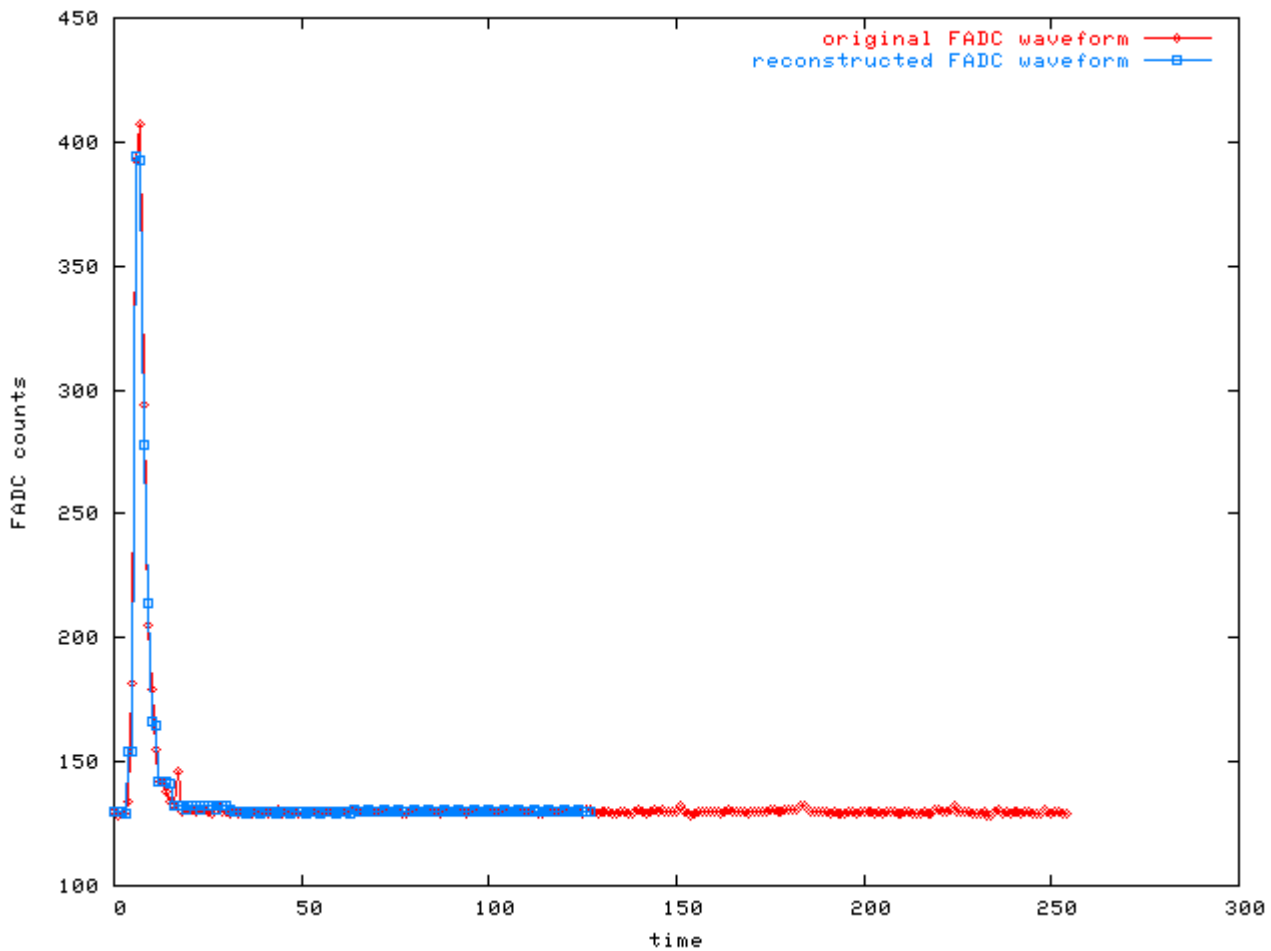
12:1

Noise waveform



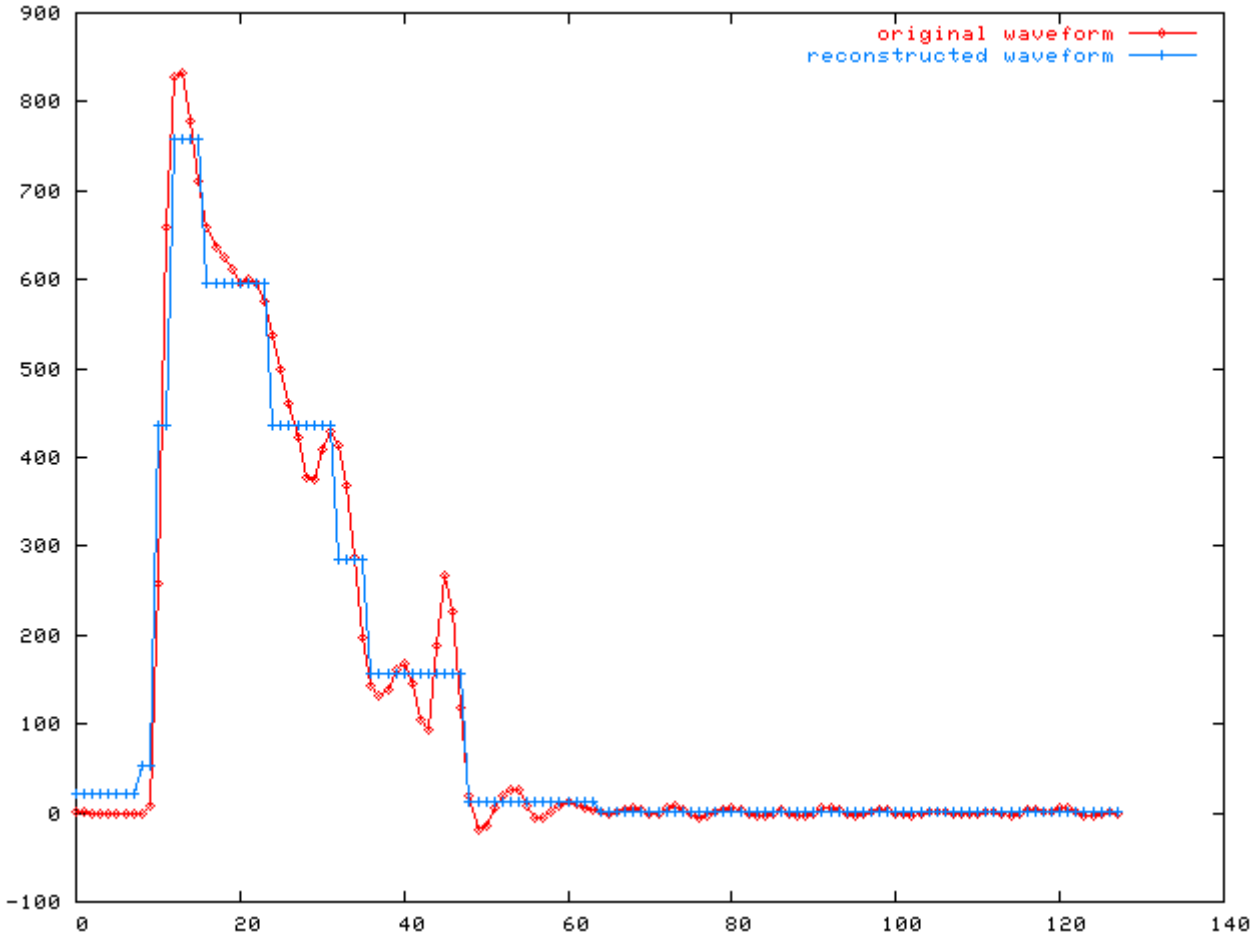
No peaks introduced into noise

FADC compression

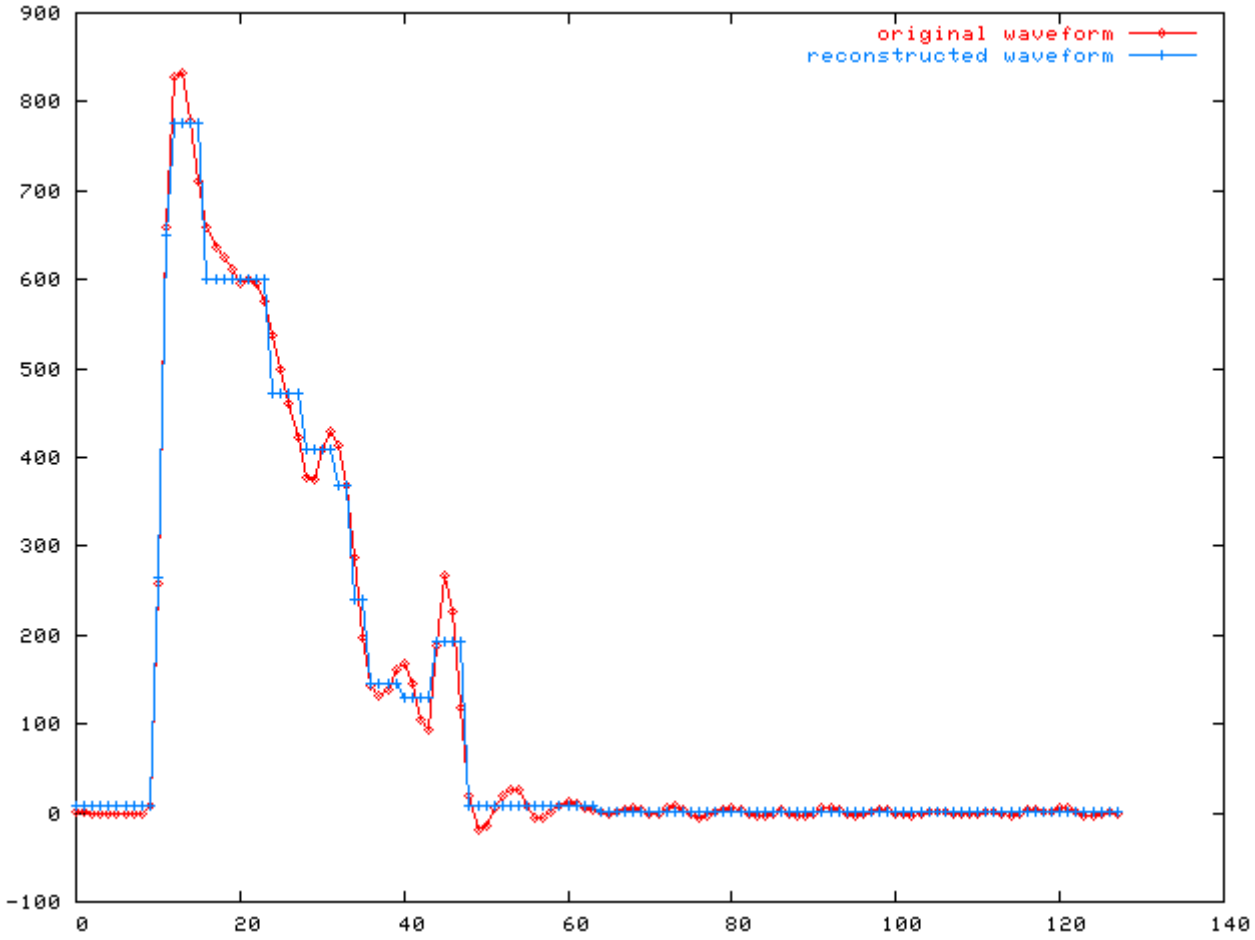


18:1 compression ratio

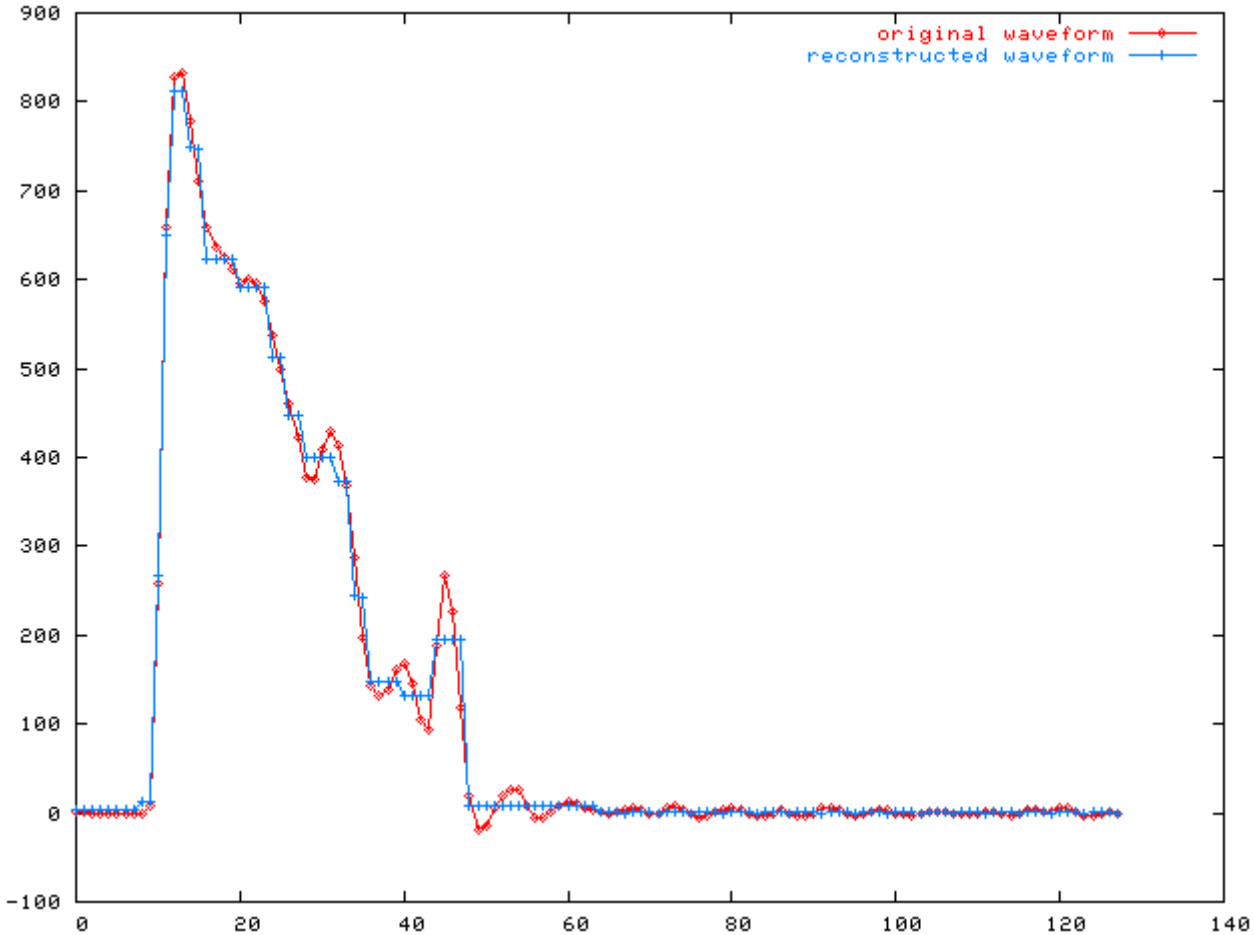
MPE waveform (IceTop prototype)



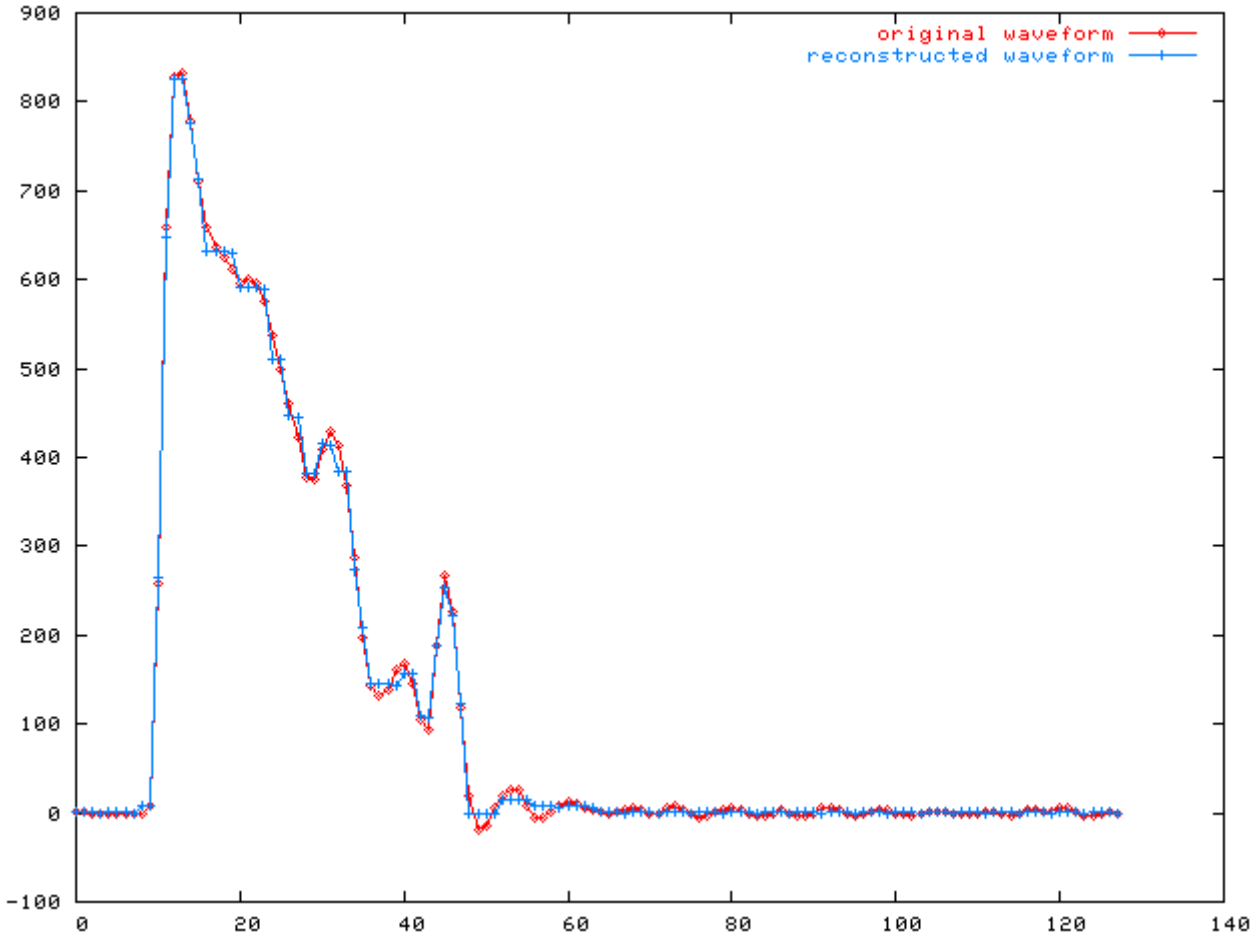
13:1


original waveform
reconstructed waveform
900
800
700
600
500
400
300
200
100
0
-100
0
20
40
60
80
100
120
140

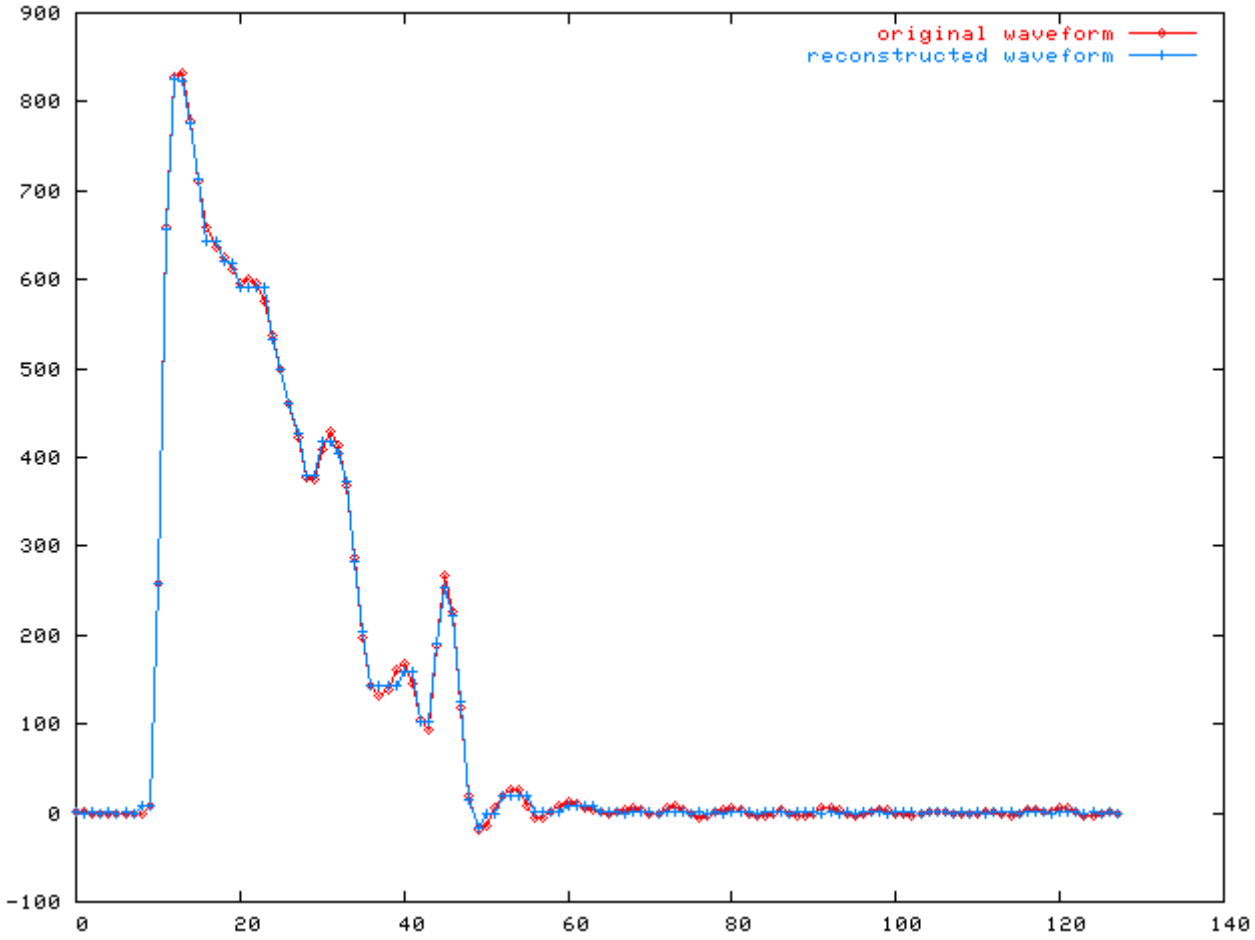
8:1


original waveform
reconstructed waveform
900
800
700
600
500
400
300
200
100
0
-100
0
20
40
60
80
100
120
140

6:1


original waveform
reconstructed waveform
900
800
700
600
500
400
300
200
100
0
-100
0
20
40
60
80
100
120
140

5:1

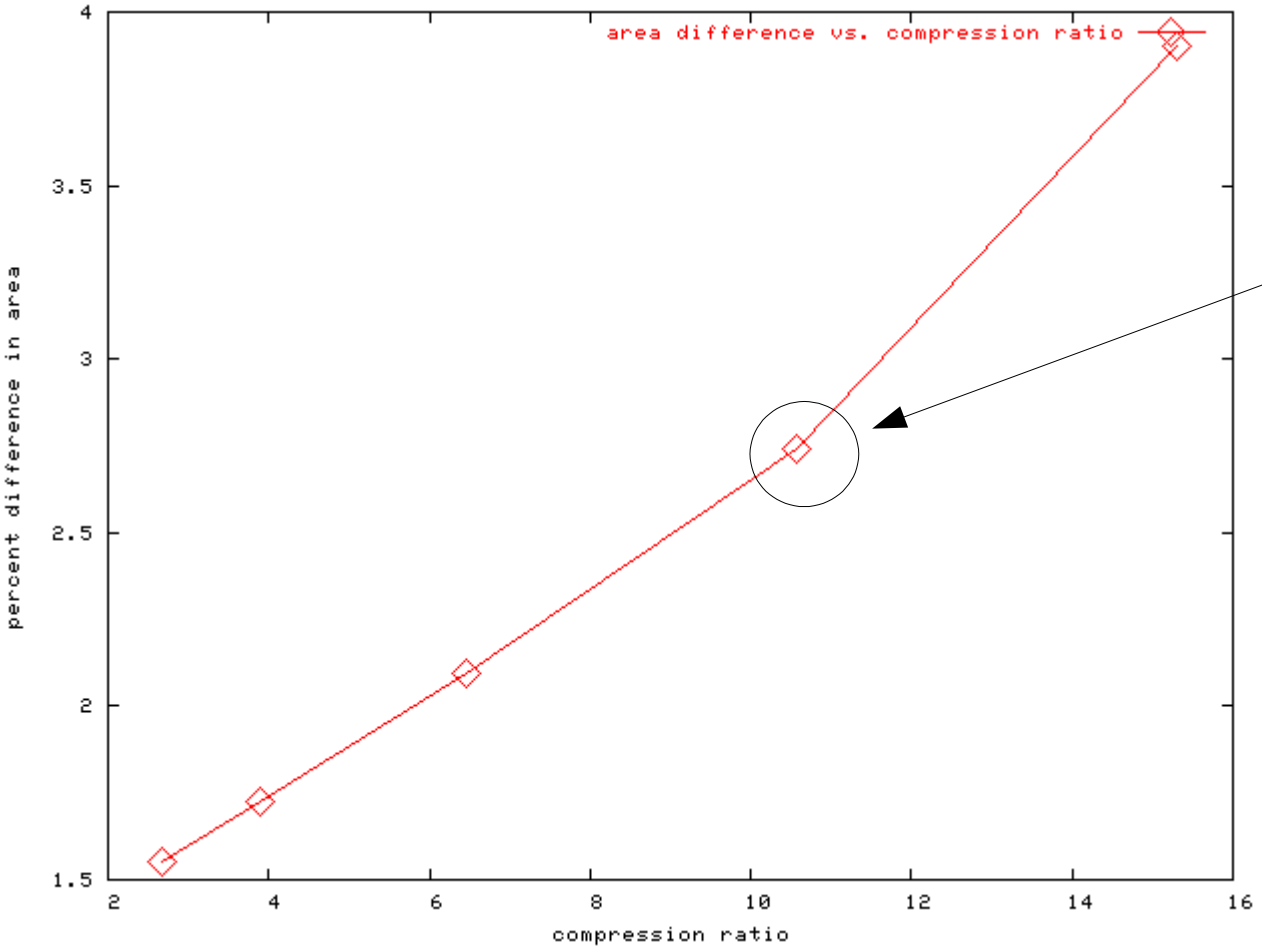


3:1
same as
roadgrader

Goodness of reconstruction criteria

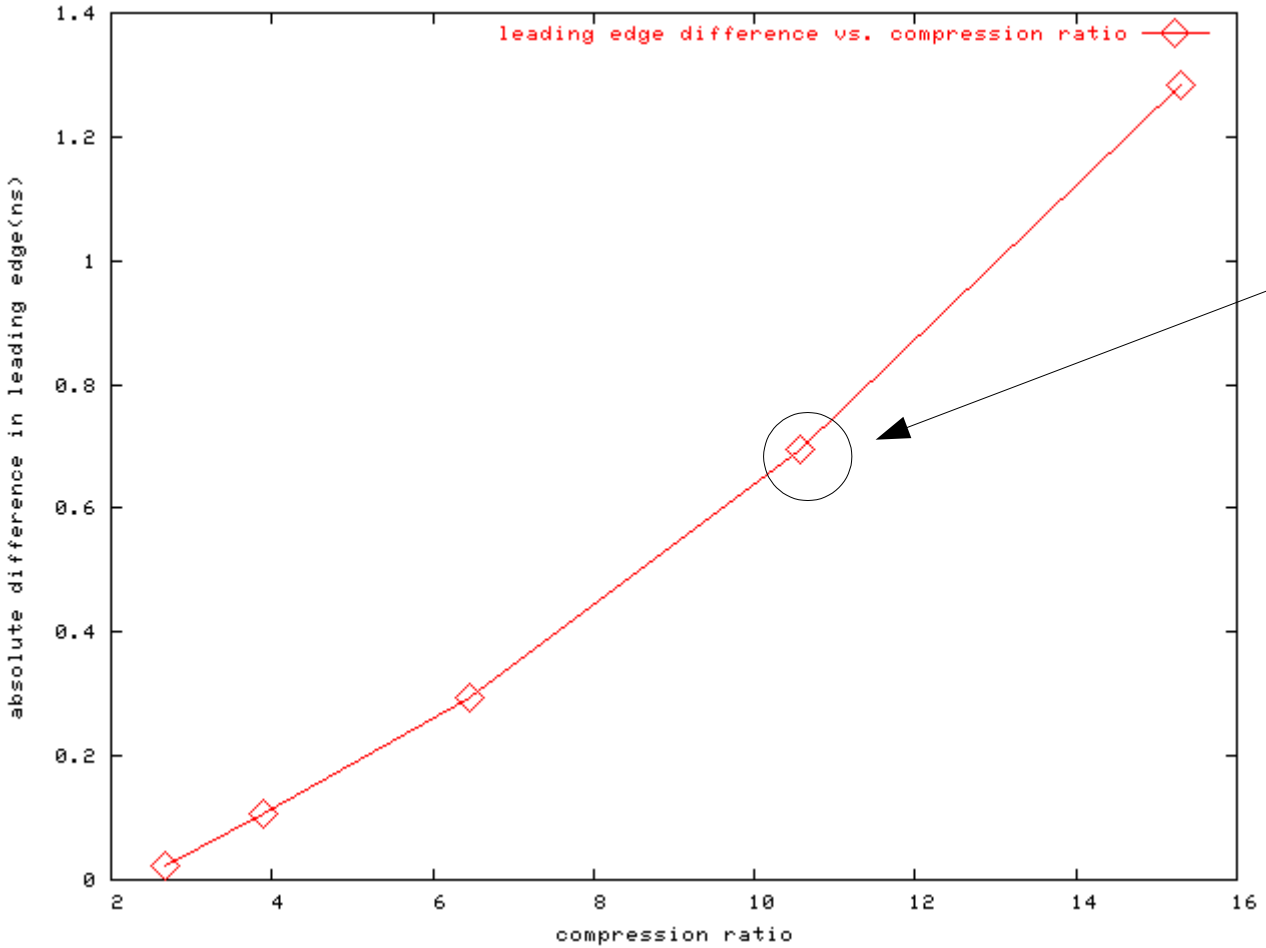
- leading edge and area (primary peak, total area)
- More sophisticated timing resolution on multiple-peaked waveforms (yet to be done)
- Run compression/decompression algorithm on waveforms before they go through high-level reconstruction routines in IceTray, see how results compare to uncompressed waveforms

Area vs. compression ratio (SPE)



10:1 comp. rat.
< 3% difference
in area

Leading Edge vs. compression ratio



10:1 comp. rat.
< 0.8 ns diff. in
leading edge

Feature Extraction in IceTray

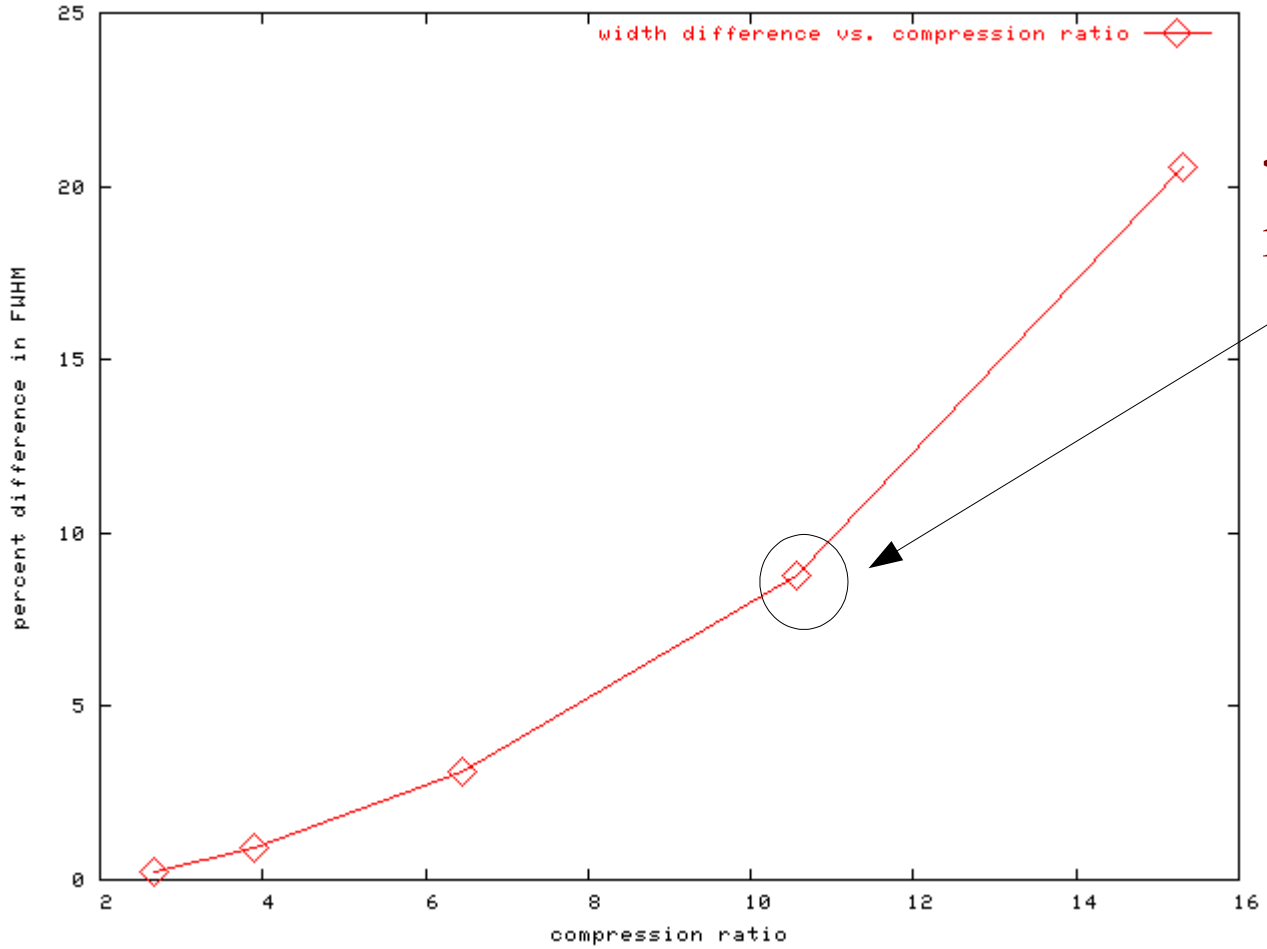
Four parameter fit:

$aux(t) = -(t-p2)/p3$

$y(aux(t)) = p0 + p1*(1 - \exp(-aux*aux))*(\exp(-aux))$

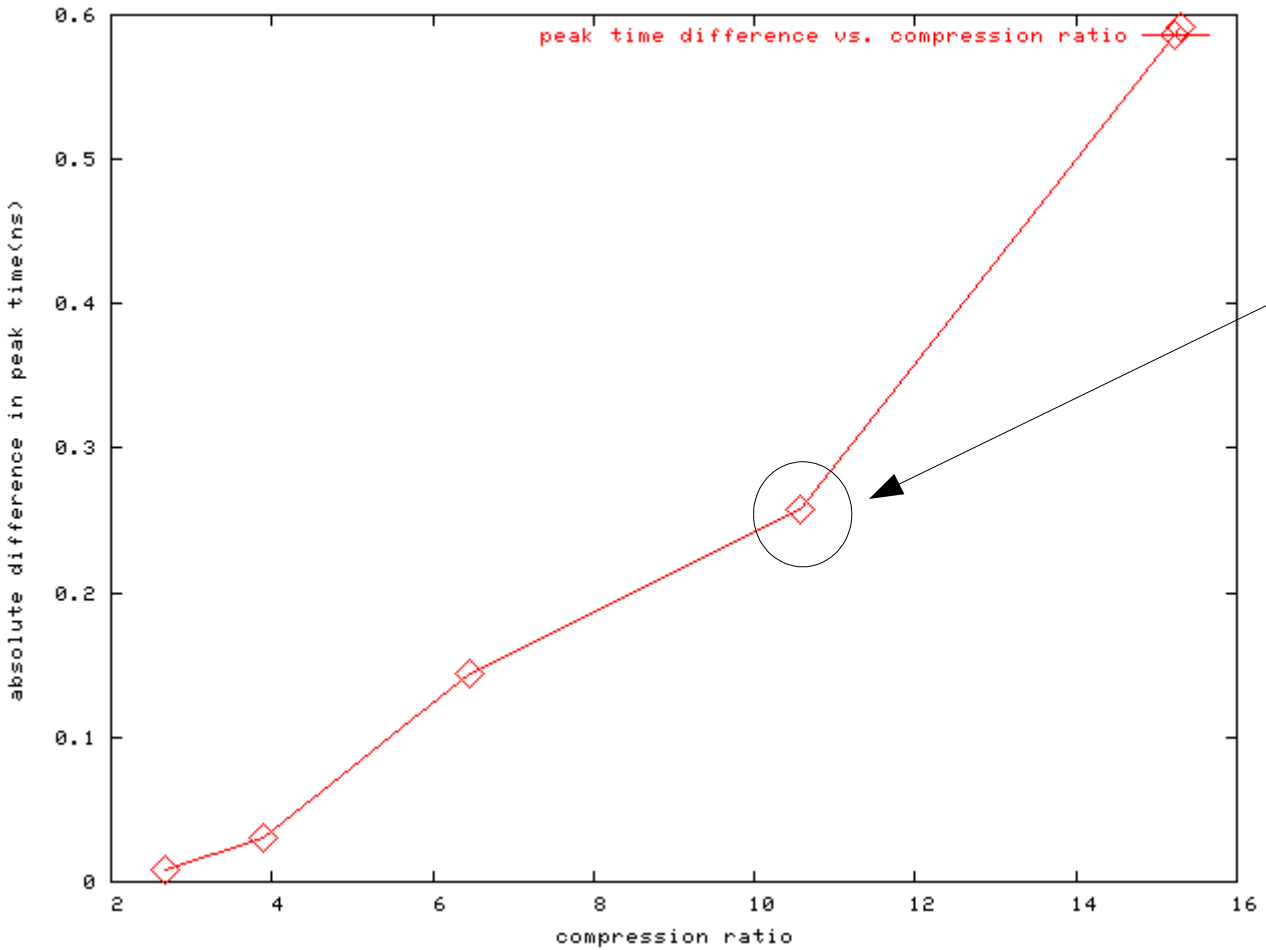
How does the compressed/decompressed waveform affect the fit?

Fit: FWHM vs. compression ratio



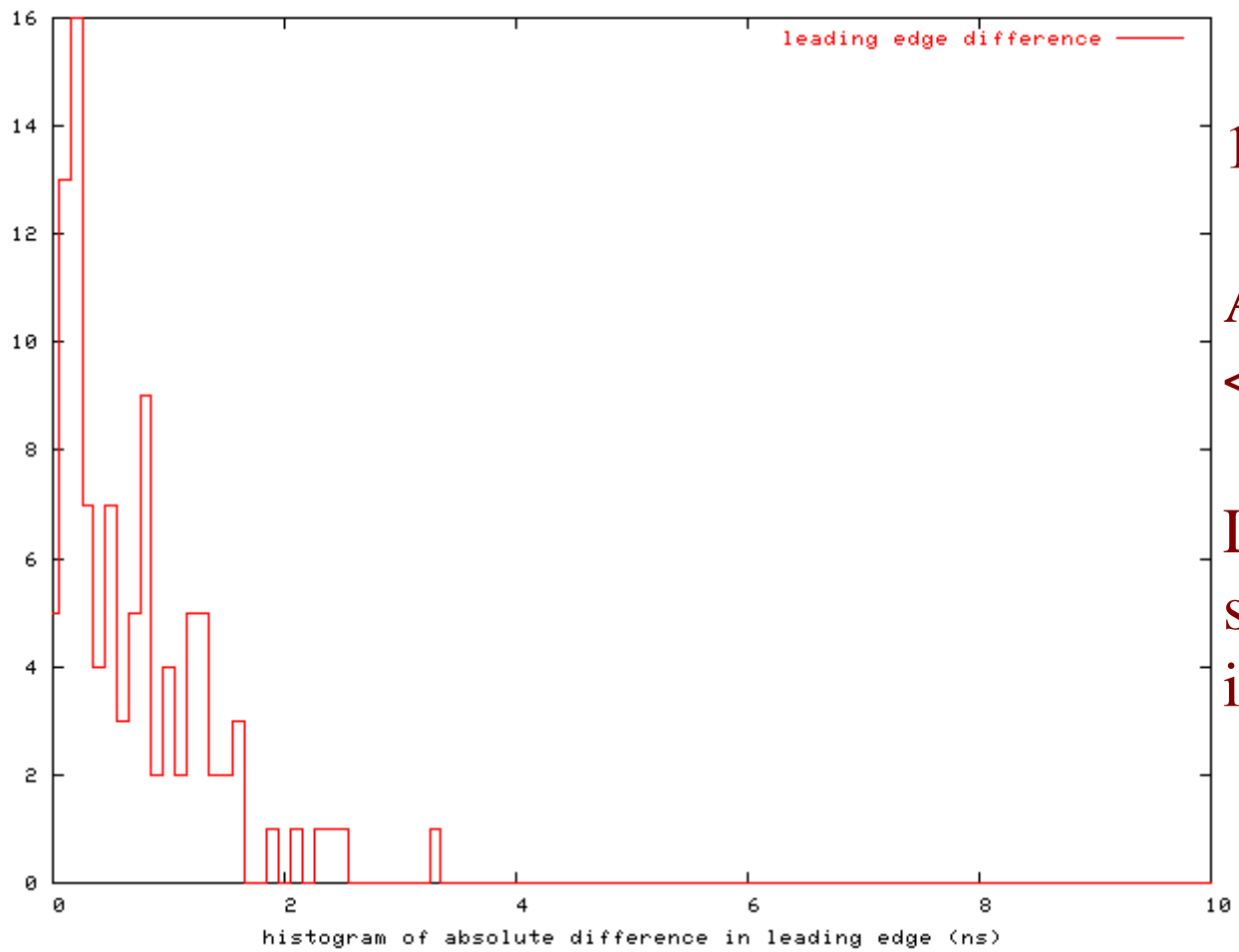
10:1 comp. rat.
< 10% difference
in FWHM

Fit: Peak time vs. compression ratio



10:1 comp. rat.
< 0.3 ns diff. in peak time

Distributions: Leading edge

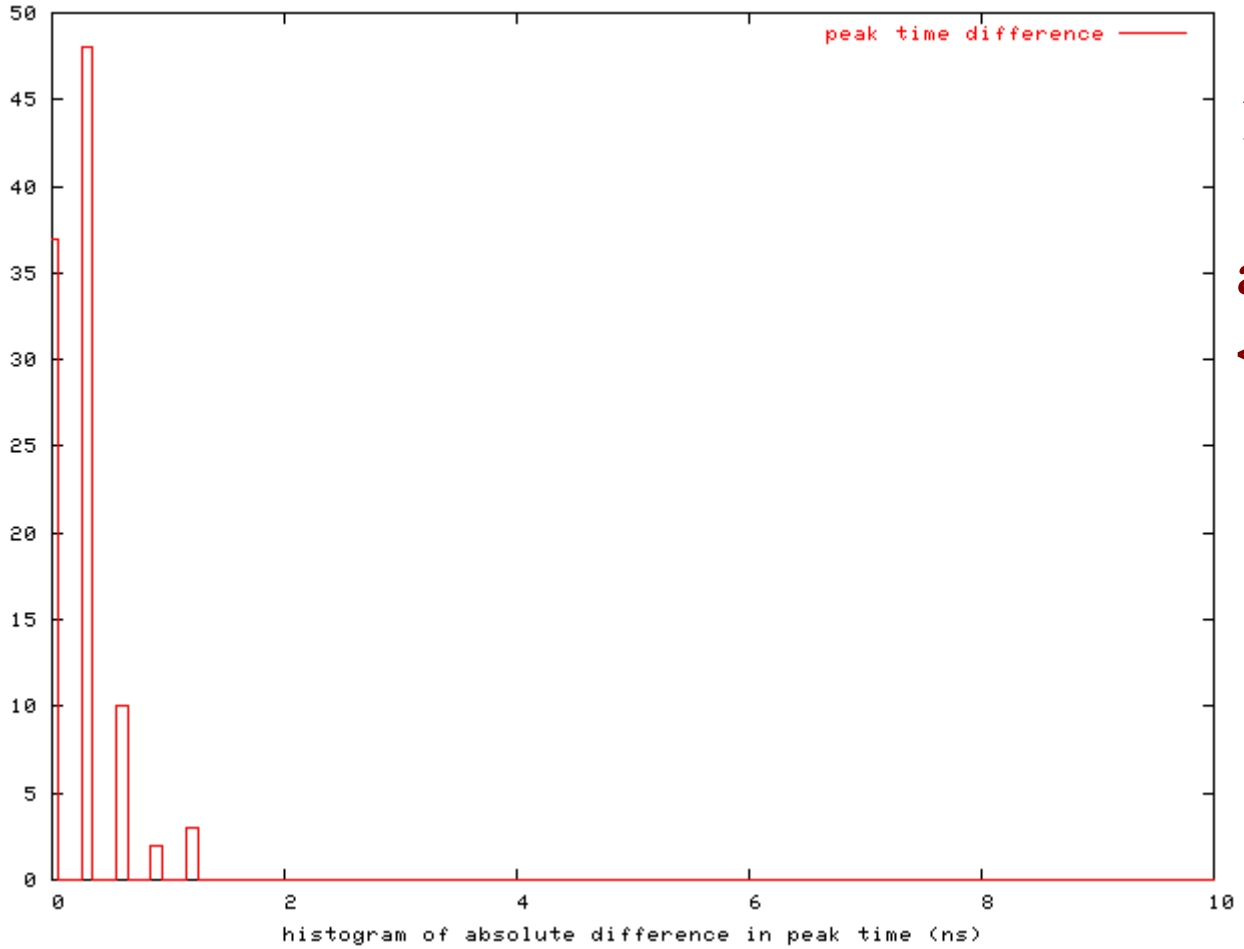


10:1 comp.rat.

Avg. L.E. diff < 0.8 ns

L.E. algorithm still needs improvement

Distributions: Peak Time (fit)



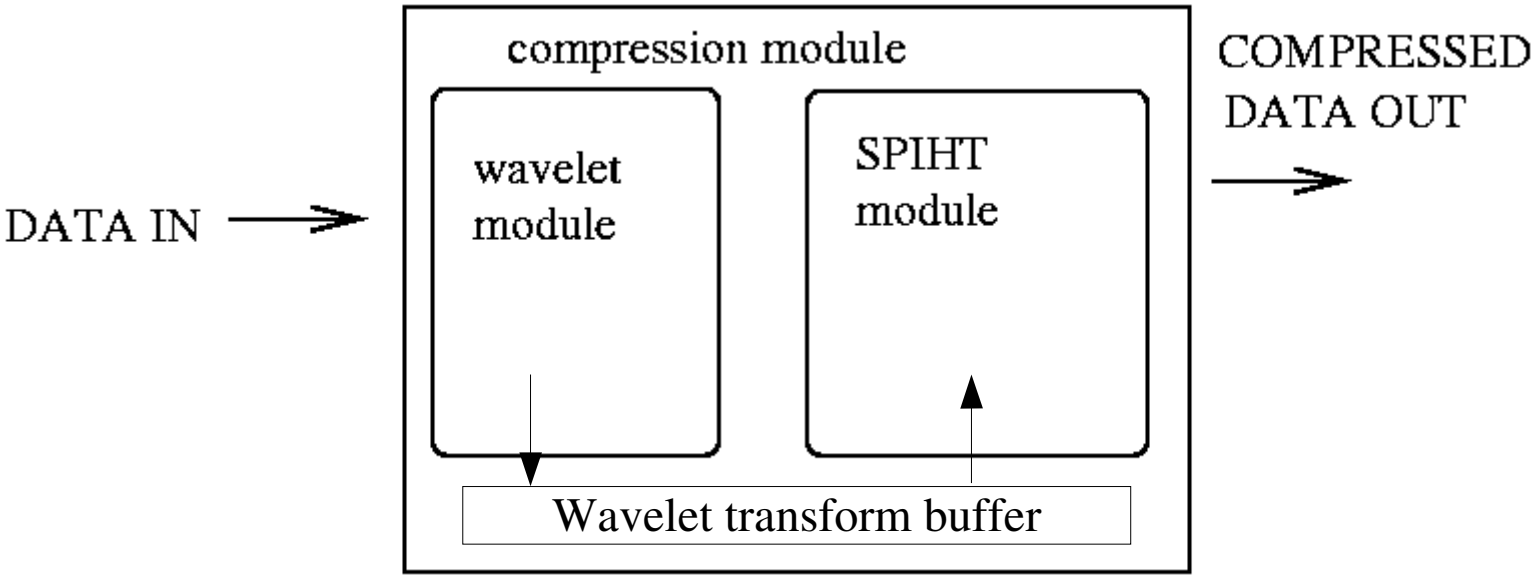
10:1 comp.rat.

avg. P.T. siff <0.3 ns

FPGA Progress

- Writing FPGA version of wavelet/SPIHT module at Penn State
  - Code produces expected bitstream
  - Output in 8-bit words suitable to domapp interface
  - FPGA resource usage: 8% of logic elements, 5% of memory
  - Still need to complete interface with domapp
- Time to produce code is decreasing (good!)

FPGA sketch


compression module
wavelet module
SPIHT module
Wavelet transform buffer
DATA IN
COMPRESSED DATA OUT

Ongoing and Future Work

- Complete interface of wavelet/SPIHT FPGA code with the domapp framework
- Is there a better wavelet than the Haar?
- Explore feature recognition capabilities of wavelets
- J. Hart at PSU implementing IceTray version of wavelet/SPIHT module
- Developing criteria for goodness of reconstruction

Summary

- Wavelets show promising reconstruction vs. compression performance with simplest wavelet
- Wavelet/SPIHT compresses multi-peaked waveforms factor of 2-3 more than road grader
- FPGA work going well, developing synthesizable code that fits in the existing FPGA framework
- Still need to determine how much information loss is acceptable, meet IceTop compression requirements
- So far about equal time spent on reconstruction and FPGA work