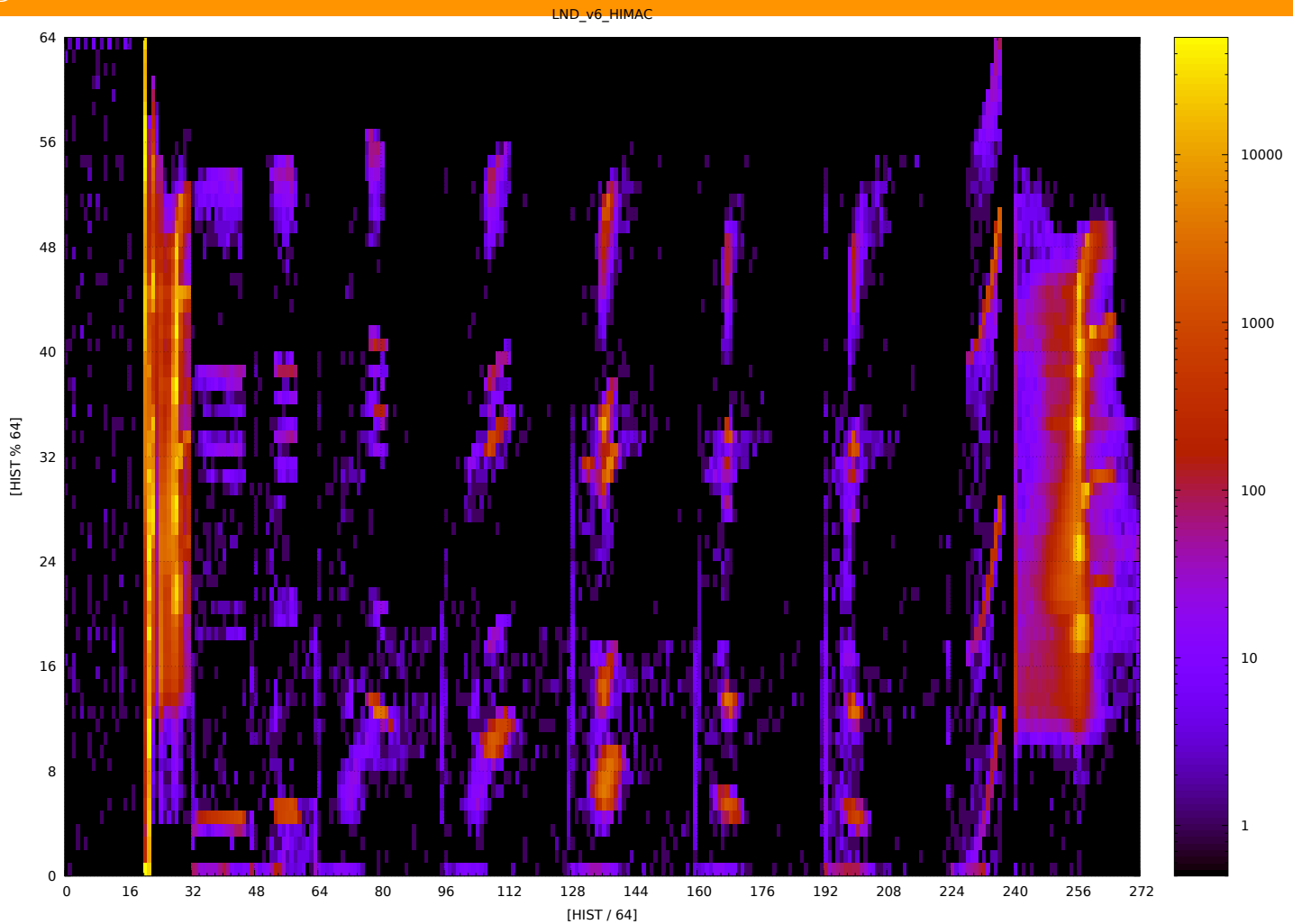
EPD Data Product Compression

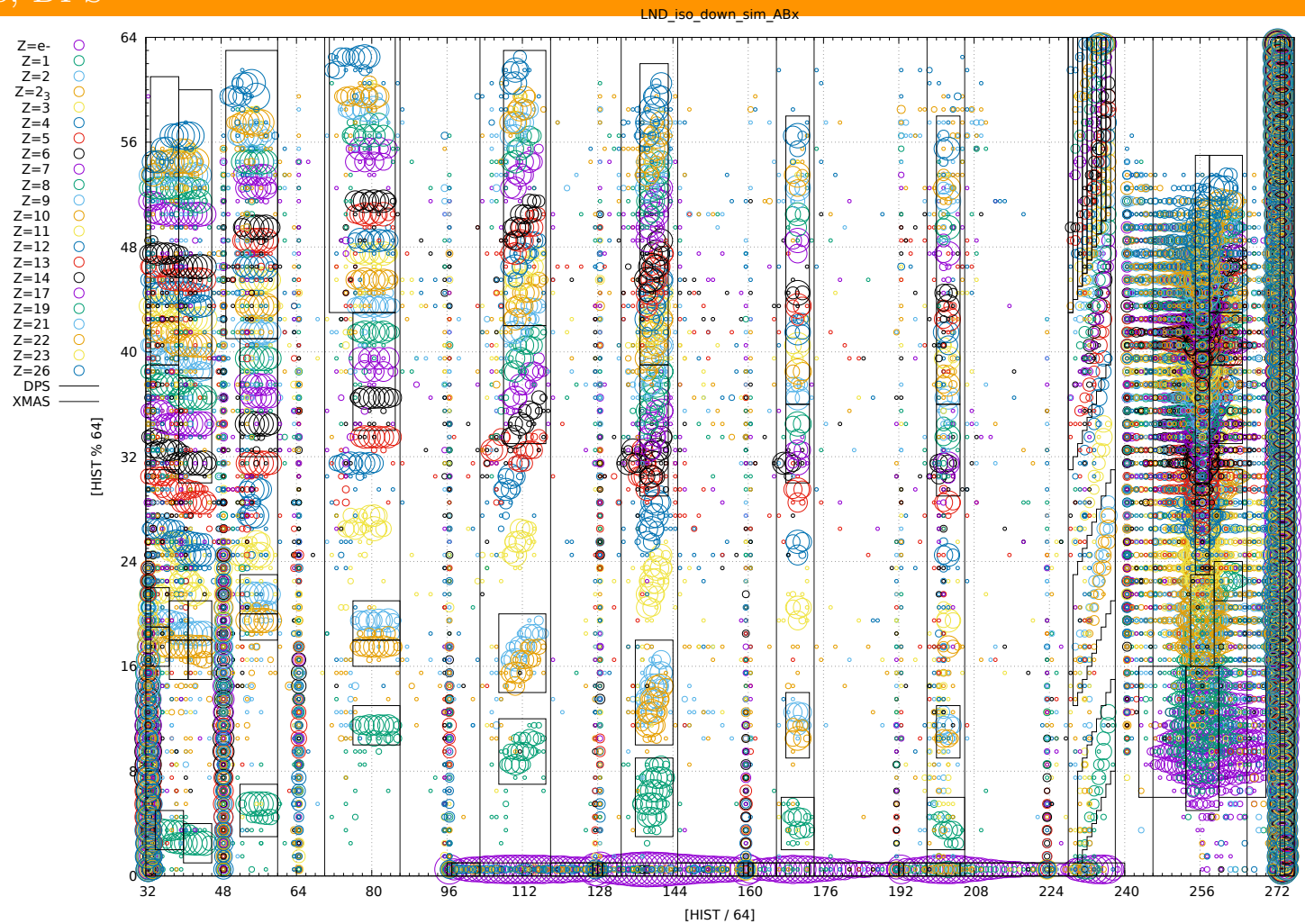
Stephan I. Böttcher

Institut für Experimentelle und Angewandte Physik
Christian Albrechts Universität zu Kiel

September 18, 2018

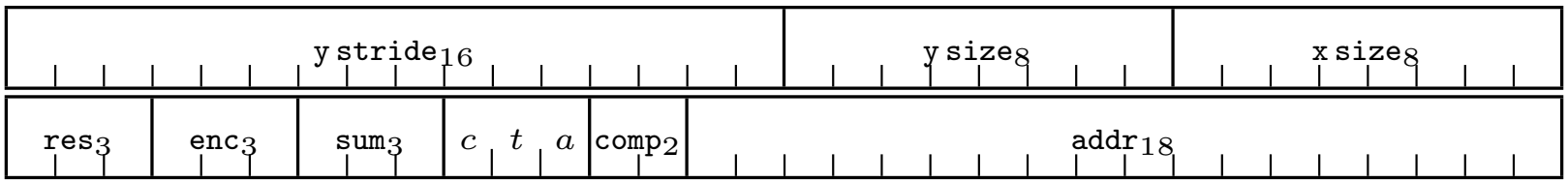


HIMAC ion data X-mas card. L3 histograms, C'E 4 LND.



Monte Carlo, DPS boxes.

- 2048 Boxes, rectangles, parallelograms,
- Eight cadences: 1, 5, 10, 30, 60, 300, 600, 3600s.
- Summing, encoding, and reset cadence.
- Accumulation of boxes.
- Four data formats: 8-bit, 16-bit, 24-bit, variable.



$\texttt{comp}_2{=}0$ 16-bits, unsigned float.

$\texttt{comp}_2{=}1$ 8-bits, $8\log_2(n)$.

$\texttt{comp}_2{=}2$ 24-bits, unsigned integer.

$\texttt{comp}_2{=}3$ Variable length encoding.

Variable length encoding

input	error	encoding	input/output length
0	±0	`0`	0/1
1-15	±0	`1s0xxxx`	4/7
16-31	±1	`1s10xxx-`	5/7
32-63	±2	`1s1100xxx--`	6/9
64-127	±4	`1s1101xxx---`	7/9
128-255	±4	`1s11100xxxx---`	8/11
256-511	±8	`1s11101xxxx----`	9/11
512-1023	±8	`1s111100xxxxx----`	10/13
1024-2047	±16	`1s111101xxxxx-----`	11/13
2048-4095	±16	`1s1111100xxxxxx-----`	12/15
4096-8191	±32	`1s1111101xxxxxx------`	13/15
8192-16383	±32	`1s11111100xxxxxxx------`	14/17
16384-32767	±64	`1s11111101xxxxxxx-------`	15/17
32768-65535	±64	`1s111111100xxxxxxx-------`	16/19
65536-131071	±128	`1s111111101xxxxxxx--------`	17/19
...			

With drop=3

input	error	encoding	input/output length
0-3	±3	`0`	2/1
4-15	±4	`1s0x---`	4/4
16-31	±8	`1s10----`	5/4
32-63	±16	`1s1100-----`	6/6
64-127	±32	`1s1101------`	7/6
128-255	±32	`1s11100x------`	8/8
256-511	±64	`1s11101x-------`	9/8
512-1023	±64	`1s111100xx-------`	10/10
1024-2047	±128	`1s111101xx--------`	11/10
2048-4095	±128	`1s1111100xxx--------`	12/12
4096-8191	±256	`1s1111101xxx---------`	13/12
8192-16383	±256	`1s11111100xxxx---------`	14/14
16384-32767	±512	`1s11111101xxxx----------`	15/14
32768-65535	±512	`1s111111100xxxxx----------`	16/16
65536-131071	±1024	`1s111111101xxxxx-----------`	17/16
...			

$$
\begin{aligned}
& c_0, c_1, c_2, \ldots, c_{59}, \\
n_0 = t_0 &= \mathrm{decode}(\mathrm{encode}_0(c_0)), \\
r_i &= c_i - n_i, \\
l_i &= c_i + r_{i-1} && n_{i-1} \leq 8, \\
l_i &= c_i + r_{i-1} - n_{i-1} && n_{i-1} > 8, \\
t_i &= \mathrm{decode}(\mathrm{encode}_3(l_i)), \\
n_i &= n_{i-1} + t_i, \\
k_0 &= \mathrm{encode}_0(c_0), \\
k_i &= \mathrm{encode}_3(l_i), \\
& k_0, k_1, k_2, \ldots, k_{59}, \mathrm{encode}_0(r_{59}).
\end{aligned}
$$