A Chink in the Armour of AES ?

Claude Gravel
Université de Montréal

Crypto 2009 - Santa Barbara - Rump Session

NIST's Randomness Testing for Round1 AES Candidates

Security Technology Group
Information Technology Laboratory
NIST

http://www.nist.gov/aes

Preliminary Statistical Analysis

- NIST Statistical Tests
  - Spectral (DFT), Runs, Approximate Entropy
  - Cusum Forward, Cusum Reverse, Long Runs
- Crypt-XB Statistical Tests
  - Frequency, Binary Derivative,
  - Linear Complexity
- DIEHARD Statistical Tests

THE HYPOTHESIS

▶ Given a random key $k \in \{0,1\}^{256}$, the output of AES is uniformly distributed.

A TEST TO TRY

▶ Check if the output bytes are uniformly distributed over the 256 possible values that a byte can take.

▶ Do a goodness-of-fit for the multinomial via a chi-square with 256 cells.

▶ In other words, split an output string into independent blocks of 8 bits, and apply the chi-square with 255 degrees of freedom.

▶ This test seems not to have been done during the AES round 1 contest as we have seen just before BUT (Why ?) it is in the NIST statistical test suite under the name of "Non-Overlapping Template Matching Test" as we can read in :

A STATISTICAL TEST SUITE FOR RANDOM AND PSEUDORANDOM NUMBER GENERATORS FOR CRYPTOGRAPHIC APPLICATIONS

NIST Special Publication 800-22
(with revisions dated May 15, 2001)

Andrew Rukhin, Juan Soto, James Nechvatal,
Miles Smid, Elaine Barker, Stefan Leigh,
Mark Levenson, Mark Vangel, David Banks,
Alan Heckert, James Dray, San Vo

and more specifically :

and more specifically :

Also on page 110 of the document :

The parameter *ALPHA* denotes the significance level that determines the region of acceptance and rejection. NIST recommends that *ALPHA* be in the range [0.001, 0.01].

and more specifically :

Also on page 110 of the document :

> The parameter *ALPHA* denotes the significance level that determines the region of acceptance and rejection. NIST recommends that *ALPHA* be in the range [0.001, 0.01].

What I will show you is that, even with a level of significance of 10^{-9}, the hypothesis should be rejected.

CRITERION FOR APPLYING THE TEST

▶ There must be at least 5 (better 10) elements of the sample per cell for applying the chi-square.

REJECTING THE HYPOTHESIS

▶ Reject if the probability of the chi-square statistics from the sample under the hypothesis, called a p-value, is smaller than the level of significance, say α (badly equidistributed), or greater than $1 - \alpha$ (too well equidistributed).

A TYPICAL SAMPLE

▶ An input message is 128 bits = 16 bytes = 16 blocks of 8 bits. An input message once encrypted gives 16 elements of a sample.

▶ So given a pseudo-random 256 bits key, encrypting in ECB mode the following 2^{11} messages gives a sample of $2^{11+4} = 32768$ elements or bytes where

$$\begin{aligned} \text{Messages} &= \text{span}\{e_j \ ; \ j \in \text{Set of indices}\} \\ \text{Set of indices} &= \{1, 16, 23, 40, 53, 56, 74, 90, 100, 116, 127\} \end{aligned}$$

and where e_j is the message having zeros everywhere except at the j^{th} position for $j \in \{1, \ldots, 128\}$.

STATISTICAL TEST CHECK

▶ There is an expected number of $\left(2^{15}/256\right) = 128$ elements per cell so the condition for applying the chi-square is fulfilled.

ENCRYPTION ROUTINE CHECK

▶ The encryption package used is at
http://polarssl.org/?page=show_source&type=source&file=aes
and it was double checked with AES calculators on the Internet, Matlab and other packages.

STATISTICAL ROUTINE CHECK

▶ My statistical routine was checked against Pierre L'Écuyer's library of statistical tests TestU01 available at
http://www.iro.umontreal.ca/ simardr/testu01/tu01.html
The name of the test in TestU01 is smultin_MultinomialBits with parameters $N = 1$, $n = 32768$, $L = 8$, $r = 0$, and $s = 32$.

NUMBER OF SAMPLES TESTED

▶ Fewer than 500 millions keys were generated with the HAVEGE generator available within the encryption package so there were fewer than 500 millions samples tested. Again a sample is 32768 bytes long or 262144 bits long.

SELECTED RESULTS

▶ I retained some cases that seem to me very interesting as they gave me p-values (probability of the chi-square under the hypothesis) of the $O(10^{\{-9,-10\}})$ or $1 - O(10^{\{-9,-10\}})$ where the hidden constant in the O notation is the interval $[0, 10)$.

▶ We can think about these cases as a proof with probability almost one that the output is not uniformly distributed.

READING THE TEXT RESULTS

- N is the number of time the tests is applied. In all the cases, $N = 1$ which means only first level tests are done.

- n is the size of the sample (*not to be confused with the number of inputs = number of outputs...*)

- L is the size of the blocks. Blocks can be taken in an independent way i.e. without overlap or in circular dependent way i.e. with overlaps.

READING THE GRAPHICAL RESULTS

▶ The histograms are created from Matlab through the outputs of my routine. Keys and p-values are written at the top of each histogram allowing a comparaison with the text results from TestU01.

▶ Remember that the cell counts are under the hypothesis normally distributed around the mean = 128. (The square of one normalized count is a chi-square with 1 degree of freedom. The sum of all the squared normalized counts is a chi-square with 255 degrees of freedom because the sum of all counts must equal the sample size.)

AES 256 **********
key : F4DD8829F13DBDBF168E91880A4F91752075193C834BFC7C27B08DBA6F2758CA

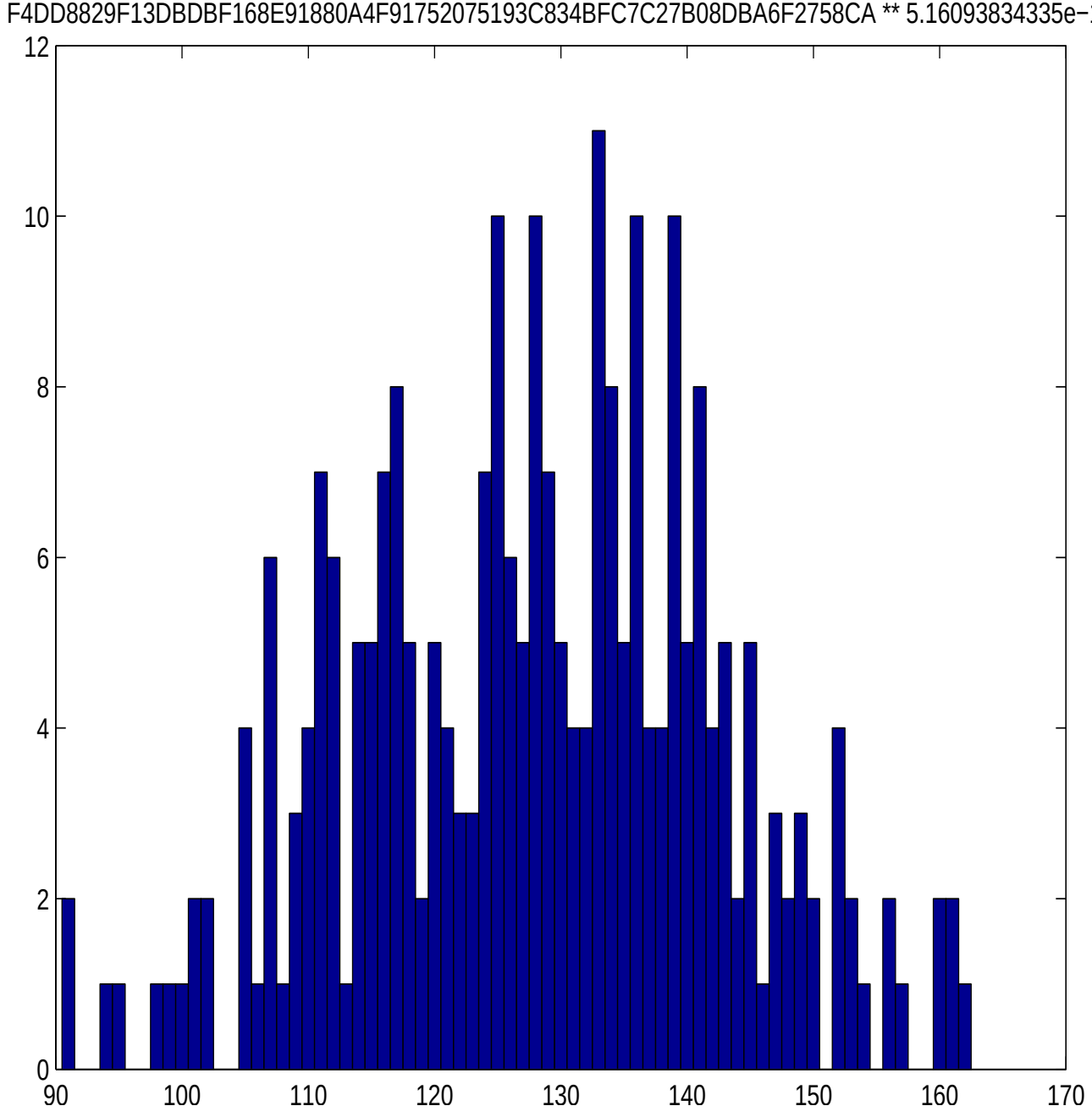
*** WITHOUT OVERLAP - INDEPENDENT ***

N = 1 * n = 262144 * L = 1 * p-value = 0.103333
N = 1 * n = 131072 * L = 2 * p-value = 0.214802
N = 1 * n = 65536 * L = 4 * p-value = 0.284705
N = 1 * n = 32768 * L = 8 * p-value = **5.15836e-10** **********

*** WITH OVERLAPS - DEPENDENT ***

N = 1 * n = 262144 * L = 2 * p-value = 0.216131
N = 1 * n = 262144 * L = 3 * p-value = 0.173693
N = 1 * n = 262144 * L = 4 * p-value = 0.0570323
N = 1 * n = 262144 * L = 5 * p-value = 0.00576847
N = 1 * n = 262144 * L = 6 * p-value = 0.0462922
N = 1 * n = 262144 * L = 7 * p-value = 0.0482701
N = 1 * n = 262144 * L = 8 * p-value = 0.0257433
N = 1 * n = 262144 * L = 9 * p-value = 0.0620828
N = 1 * n = 262144 * L = 10 * p-value = 0.0744195
N = 1 * n = 262144 * L = 11 * p-value = 0.144411
N = 1 * n = 262144 * L = 12 * p-value = 0.319223

Oscillation on the corresponding histogram:


F4DD8829F13DBDBF168E91880A4F91752075193C834BFC7C27B08DBA6F2758CA ** 5.16093834335e−10
12
10
8
6
4
2
0
90
100
110
120
130
140
150
160
170

AES 256 **********
key : 97B76C3E57E548D183A54D071AA14F313C5E50608D966C4D5DF26BE03043D24D

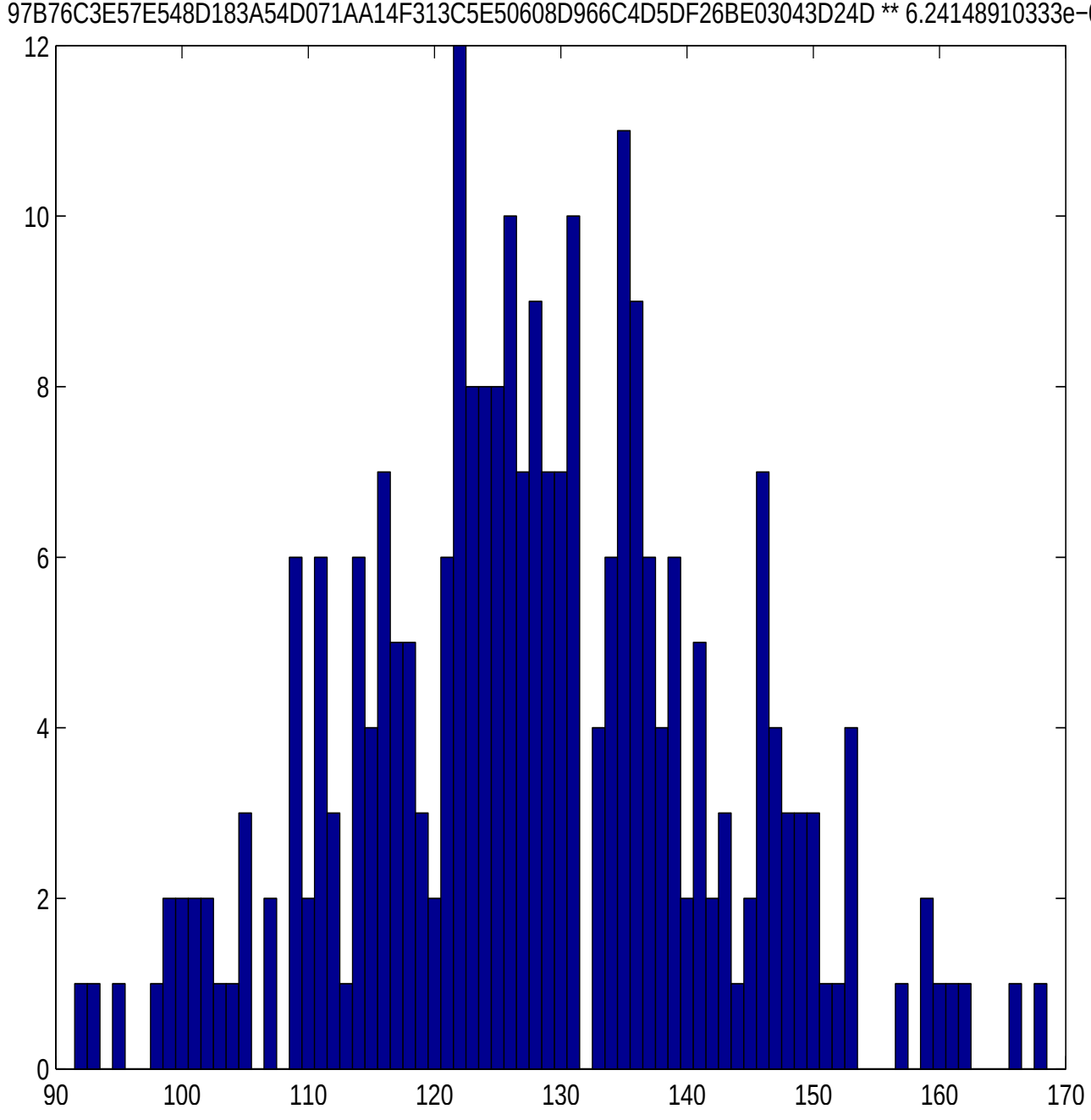
*** WITHOUT OVERLAP - INDEPENDENT ***

N = 1 * n = 262144 * L = 1 * p-value = 0.0856597
N = 1 * n = 131072 * L = 2 * p-value = 0.121402
N = 1 * n = 65536 * L = 4 * p-value = 0.424792
N = 1 * n = 32768 * L = 8 * p-value = **6.23875e-09** **********

*** WITH OVERLAPS - DEPENDENT ***

N = 1 * n = 262144 * L = 2 * p-value = 0.226948
N = 1 * n = 262144 * L = 3 * p-value = 0.487313
N = 1 * n = 262144 * L = 4 * p-value = 0.596476
N = 1 * n = 262144 * L = 5 * p-value = 0.692821
N = 1 * n = 262144 * L = 6 * p-value = 0.31934
N = 1 * n = 262144 * L = 7 * p-value = 0.0767956
N = 1 * n = 262144 * L = 8 * p-value = 0.363232
N = 1 * n = 262144 * L = 9 * p-value = 0.682192
N = 1 * n = 262144 * L = 10 * p-value = 0.583882
N = 1 * n = 262144 * L = 11 * p-value = 0.447798
N = 1 * n = 262144 * L = 12 * p-value = 0.777747

Hole near the mean and oscillation:


97B76C3E57E548D183A54D071AA14F313C5E50608D966C4D5DF26BE03043D24D ** 6.24148910333e−09
12
10
8
6
4
2
0
90
100
110
120
130
140
150
160
170

AES 256 **********
key : 8028B2B179C07F06FD71CE8C228A7F32A9B889C46A5E42B357ABE66B6BC577F5

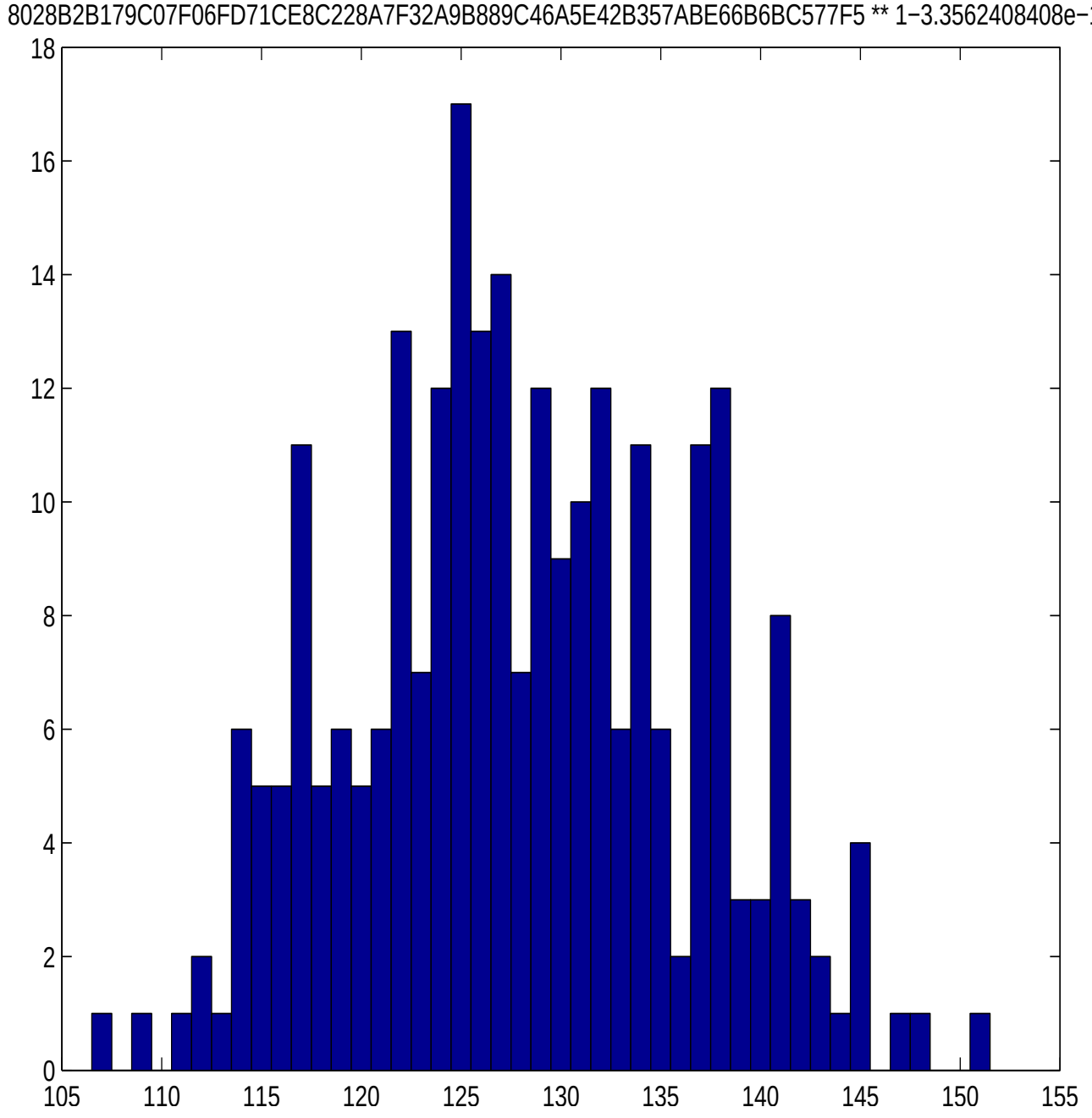
*** WITHOUT OVERLAP - INDEPENDENT ***

N = 1 * n = 262144 * L = 1 * p-value = 0.73692
N = 1 * n = 131072 * L = 2 * p-value = 0.724621
N = 1 * n = 65536 * L = 4 * p-value = 0.821153
N = 1 * n = 32768 * L = 8 * p-value = **1-3.35373e-10 ********** **********

*** WITH OVERLAPS - DEPENDENT ***

N = 1 * n = 262144 * L = 2 * p-value = 0.449732
N = 1 * n = 262144 * L = 3 * p-value = 0.708637
N = 1 * n = 262144 * L = 4 * p-value = 0.649967
N = 1 * n = 262144 * L = 5 * p-value = 0.47087
N = 1 * n = 262144 * L = 6 * p-value = 0.767975
N = 1 * n = 262144 * L = 7 * p-value = 0.704057
N = 1 * n = 262144 * L = 8 * p-value = 0.694797
N = 1 * n = 262144 * L = 9 * p-value = 0.401099
N = 1 * n = 262144 * L = 10 * p-value = 0.481183
N = 1 * n = 262144 * L = 11 * p-value = 0.956451
N = 1 * n = 262144 * L = 12 * p-value = 0.669379

Too concentrated around the mean i.e. excessive uniformity:


8028B2B179C07F06FD71CE8C228A7F32A9B889C46A5E42B357ABE66B6BC577F5 ** 1−3.3562408408e−10
18
16
14
12
10
8
6
4
2
0
105
110
115
120
125
130
135
140
145
150
155

CONCLUSION

- Tests failed only for the 8-bits independent blocks.

WHY THE CONCLUSION?

- I cannot answer since AES is still a blackbox for me!

- The only thing I know about AES that could potentially explain this behaviour is that AES is byte-designed so it may have problem with substreams of 8 bits; it has at least on the examples shown. More investigations would be needed.

REFERENCES

(1) Various references from the NIST especially
http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22b.pdf and
http://csrc.nist.gov/archive/aes/round1/conf2/NIST-randomness-testing.pdf

(2) On the Statistical Testing of Block Ciphers, Richard J. De Moliner, PhD thesis at ETH Zurich, especially chapter 4, 1999, 118 pages

(3) Fundamental Concepts in the Design of Experiments, Charles R. Hicks and Kenneth V. Turner Jr., Fifth Edition, Oxford University Press, 1999, 565 pages

RELATED WORK

(1) Distinguisher and Related-Key Attack on the Full AES-256, *Alex Biryukov, Dmitry Khovratovich, and Ivica Nikolić*, University of Luxembourg, IACR, CRYPTO 2009, LNCS 5677, pp. 231-249, 2009

ACKNOWLEDGEMENTS

I wish to thank Gilles Brassard, my PhD supervisor. I wish to thank also Richard Simard for helping me with the TestU01 library, and also for having looked at my own code. I also thank Pierre L'Écuyer, Louis Salvail and Sorana Froda for all the discussions we had.

HEURISTIC FOR CHOICE OF INPUTS

▶ The inputs were selected using a heuristic. The heuristic consists of repeating the following procedure until a maximum number of iterations, say M, is reached or until "convergence" is reached with "convergence" explains soon.

▶ The procedure is to select a set of 11 random indices i_j without repetition for $j \in \{1, \ldots, 11\}$ and for $i_j \in \{1, \ldots, 128\}$, take the span of the standard messages e_{i_j}, put a level of confidence 10 times fewer than the number of generated samples (i.e. a level of $\alpha = 10^{-5}$ for say 1000 samples), and keep the set of indices iff a p-value out of the 1000 p-values was smaller than α or greater than $1 - \alpha$.

▶ It should not happen very often that a set of indices is kept, but when one is kept, we hope it is a good one.

▶ When the maximal number of iterations is reached, select the mode for each first, second, up to the eleventh index. Alternatively, one can stop until convergence for each index happens after a certain number of iterations of the procedure.