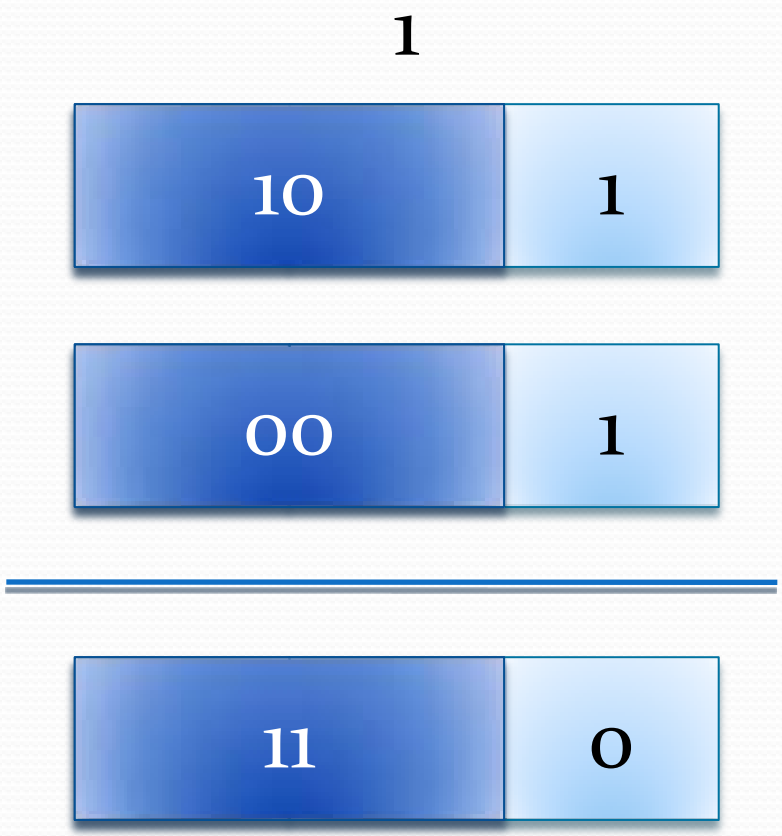
Pseudo-Linear Approximations for ARX Ciphers:

With Application to Threefish

Kerry A. McKay and Poorvi L. Vora

Approach

- Instead of individual bits, use larger partitions of state and key
- Window is a group of contiguous bits
- For window size w, approximate with addition mod 2^w instead of 2^n
  - If addends uniformly distributed
    - Probability of correctness ≈ ½
      - Independent of w
    - Success of a random guess is 2^{-w}
    - Bias is ½ - 2^{-w}
      - Significant when w > 1



Approach (cont.)

- We can form expressions for ARX functions with windows
- Look like (a $\oplus$ b) $\boxplus$ (c $\oplus$ (d $\boxplus$ e))
  - a, b, c, d, and e are windows
- Linearity
  - Addition is linear in GF(2^n)
  - Exclusive-or is linear in GF(2)
  - Combination not linear in the traditional sense
    - We call these expressions pseudo-linear

Properties of Addition

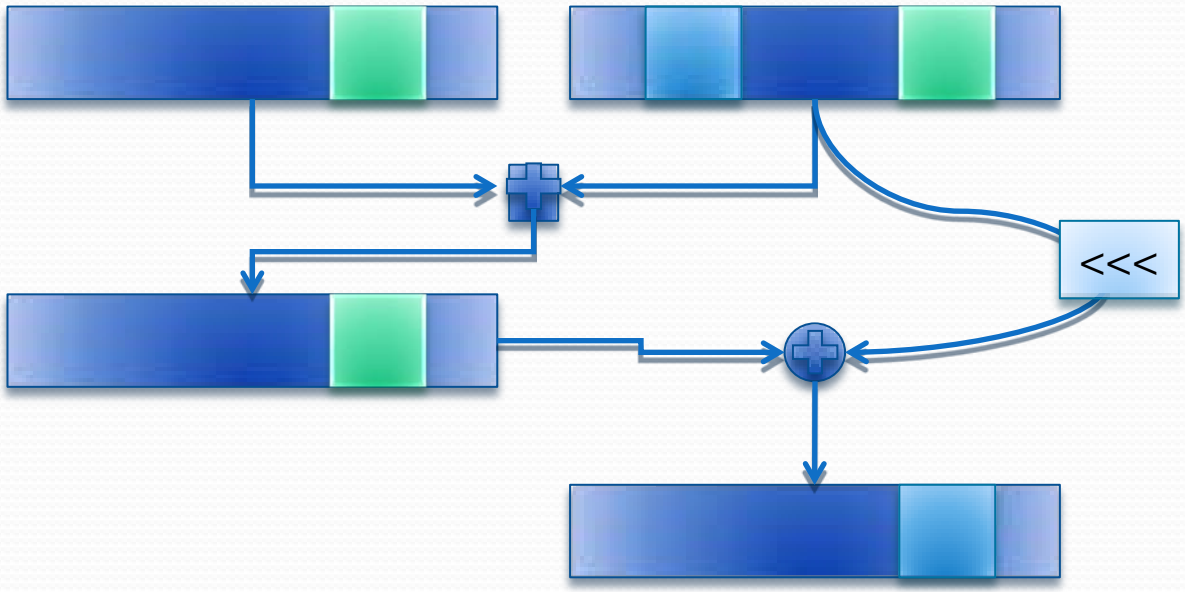
- Carry function biased for *random* m-bit addends
  - Pr[no carry] = ½ + 2^{-m-1}
- Carry more likely for larger m
  - Always occurs with probability < ½
  - Probability dependent on position only
    - Independent of window size
- When addends not random, difficult to predict carry with high bias without knowing pdf of input

Properties of Addition Windows

- Rotation and shift operations can easily be accounted for
  - Do not change the structure of the window, only the start and end indices
    - For shift/rotate left by c, subtract c from indices
    - For shift/rotate right by c, add c from indices

Making Approximations

- Base approximation
  - Trace windows through the cipher
    - Use cipher operations for addition and xor
  - Assume no carry into any window



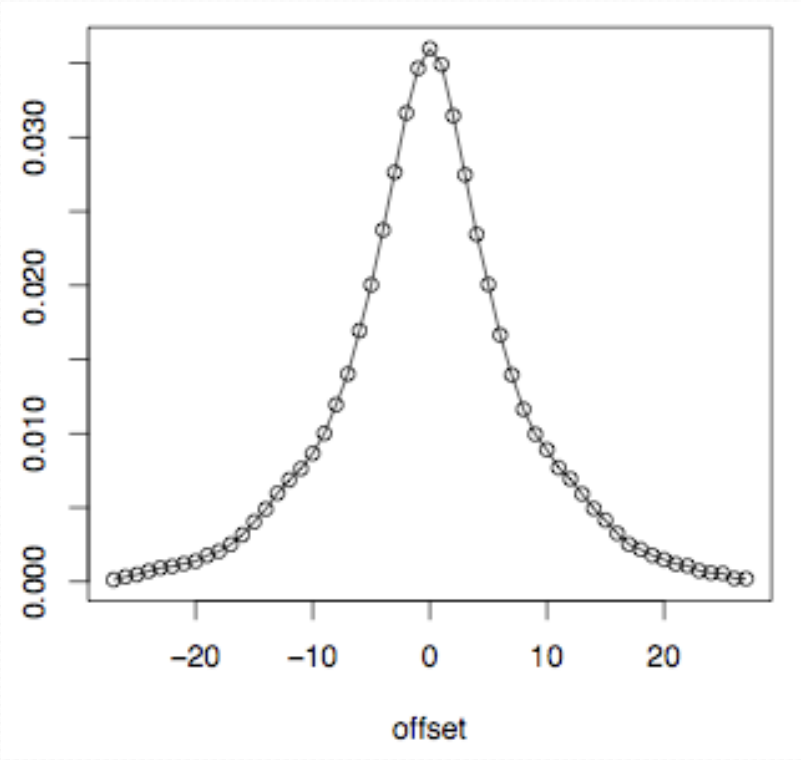
Making Approximations (cont.)

- Carry patterns
  - Overlay the base approximation with a sequence of 0's and 1's
    - A 0 or 1 for each approximated addition represents whether or not we guess a carry into the window
- Each carry pattern + base approximation represents a different equation

Making Approximations (cont.)

- Offsets
  - Loosen the meaning of "equal"
  - Instead of y = x constituting a correct guess, let a correct guess be

    (x-offset) ≤ y ≤ (x+offset)
  - Allow us to make up for incorrect carry predictions

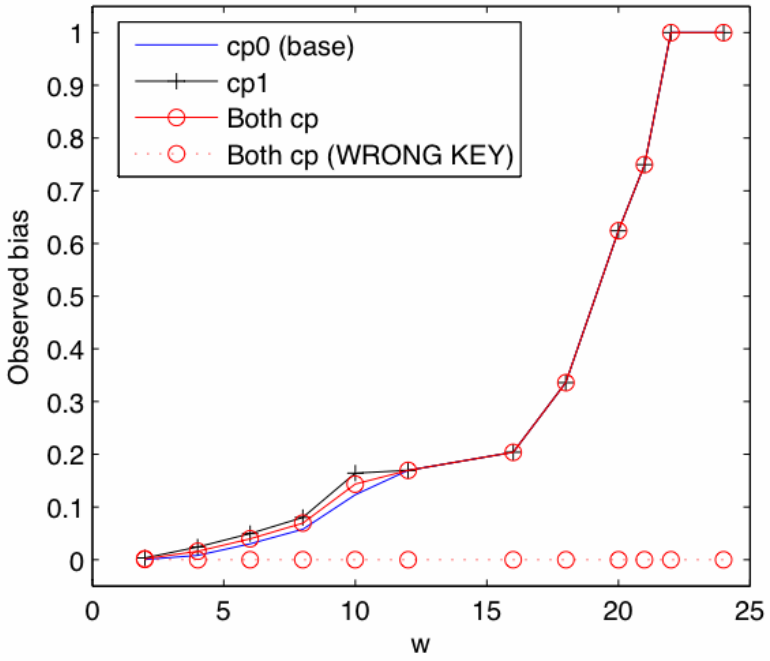


Experiments

- Applied to modified Threefish-256 (round 1)
  - Addition mod 2^{64}, xor, rotate, permute
- Set tweak schedule to 0
- No whitening key
- Generate random (key, data) pairs
- Run each pair through encryption and approximations

Some Empirical Results

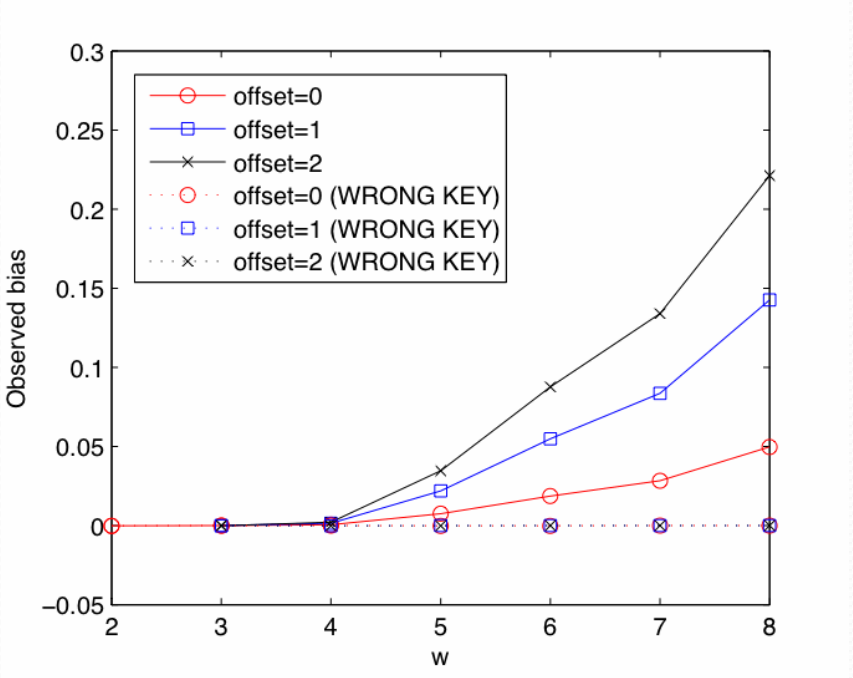
- Rounds 4-8
- 2 different carry patterns
  - Base (all zero)
  - Every other round
  - Both
- Can distinguish 4 rounds of Threefish-256 with w ≥ 3
  - A key recovery attack would require 58 bits of key to be guessed
  - Without approximation, 256 bits
- Choosing multiple patterns doesn't appear to offer any benefit over a single, stronger pattern



Some Empirical Results

- Rounds 4-12
- Carry assumed for every other addition
- Meet-in-the-middle at round 10, word 0, window 0

- Need at least w=5 to distinguish correct key
- Offsets can improve the distinguisher
- This distinguisher requires 232 key bits to be guessed for key recovery



Threefish-256 Diffusion

- The rotation constants in Threefish are well-chosen
  - Defend against this type of attack
  - Windows quickly rotate from least significant half to most significant half of words
- In the 4-8 round approximation, with w=3
  - Only 2 bits of overlap between windows
  - 1 pair of adjacent windows
  - We don’t get to reuse the same bits much
- In the 4-12 round approximation, with w=5
  - There is more overlap, but over 90% of key bits are involved

Conclusions

- This technique is currently not sufficient to break Threefish-256, but is interesting to consider
  - Threefish diffusion forces us to guess too many bits
- The best results are achieved with larger window sizes, but smaller ones can be sufficient for key recovery
- Has potential as a key recovery technique for ARX ciphers under the known-plaintext model

- Future Work
  - Improve technique
  - Variations
  - Apply to other ARX constructions

Thank you!