

Correctness and Performance Charts

Peter Occil

1 Correctness and Performance Charts

This version of the document is dated 2025-09-30.

The following charts show the correctness of many of the algorithms in “[Bernoulli Factory Algorithms¹](#)” and show their performance in terms of the number of bits they use on average. For each algorithm, and for each of 100 λ values evenly spaced from 0.0001 to 0.9999:

- 500 runs of the algorithm were done. Then...
- The number of bits used by the runs were averaged, as were the return values of the runs (since the return value is either 0 or 1, the mean return value will be in the interval $[0, 1]$). The number of bits used included the number of bits used to produce each coin flip, assuming the coin flip procedure for λ was generated using the `Bernoulli#coin()` method in *bernoulli.py*, which produces that probability in an optimal or near-optimal way.

For each algorithm, if a single run was detected to use more than 5000 bits for a given λ , the entire data point for that λ was suppressed in the charts below.

In addition, for each algorithm, a chart appears showing the minimum number of input coin flips that any fast Bernoulli factory algorithm will need on average to simulate the specified function, based on work by Mendo (2019)[¹]. Note that some functions require a growing number of coin flips as λ approaches 0 or 1. Note that for the 2014, 2016, and 2019 algorithms—

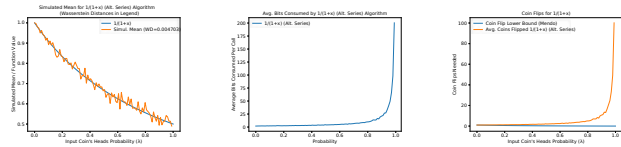
- an ϵ of $1 - (x + c) * 1.001$ was used (or 0.0001 if ϵ would be greater than 1), and
- an ϵ of $(x - c) * 0.9995$ for the subtraction variants.

Points with invalid ϵ values were suppressed. For the low-mean algorithm, an m of $\max(0.49999, x \cdot 1.02)$ was used unless noted otherwise.

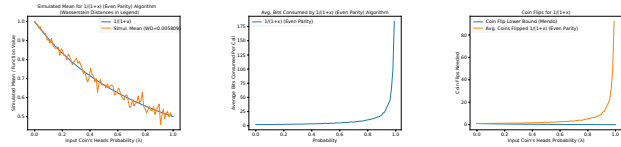
1.1 The Charts

Algorithm	Simulated Mean	Average Bits Consumed	Coin Flips
$(1-x) \cdot \tan(x)$			
$(1-x)/\cos(x)$			
$(1/3)^x / (1 + (1/3)^x)$			
$(2/3)^x / (1 + (2/3)^x)$			
$(3/2)^x / (1 + (3/2)^x)$			
$0.5^x / (1 + 0.5^x)$			
$1 - \ln(1+x)$ (Alt. Series)			

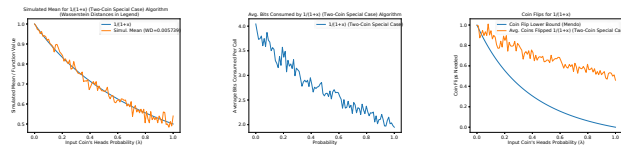
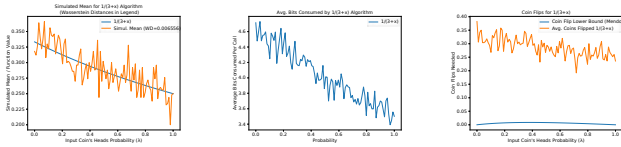
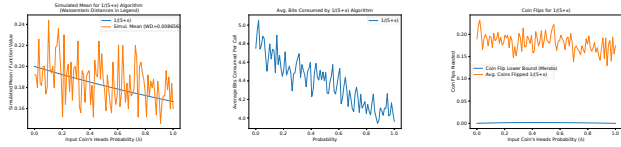
$1/(1+x)$ (Alt. Series)



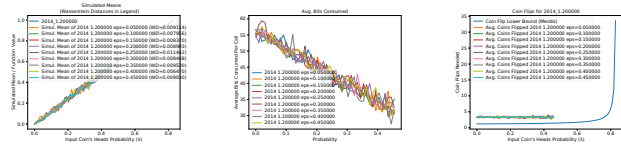
$1/(1+x)$ (Even Parity)



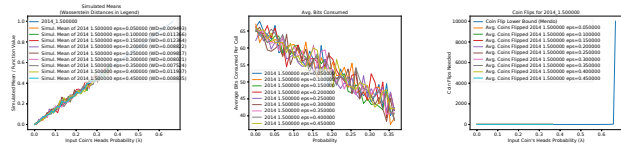
$1/(1+x)$ (Two-Coin Special Case)


$$1/(3+x)$$

$$1/(5+x)$$


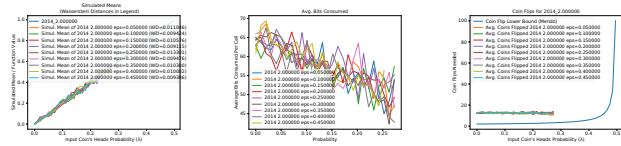
2014 1.200000
eps=0.050000



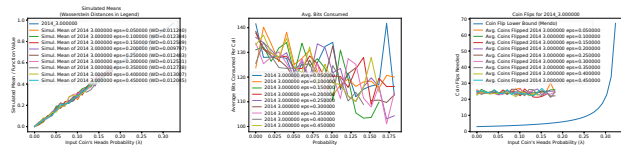
2014 1.500000
eps=0.050000



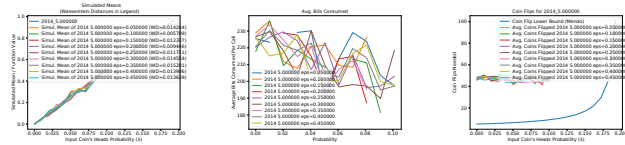
2014 2.000000
eps=0.050000



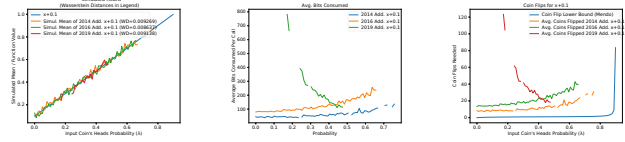
2014 3.000000
eps=0.050000



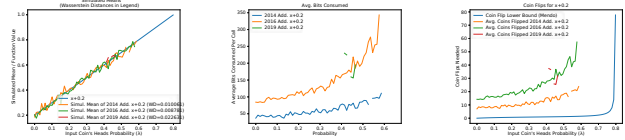
2014 5.000000
eps=0.050000



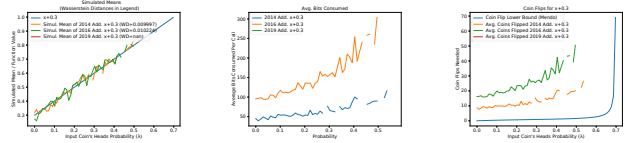
2014 Add. x+0.1



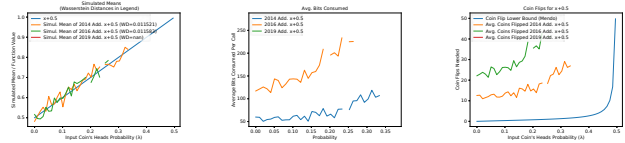
2014 Add. x+0.2



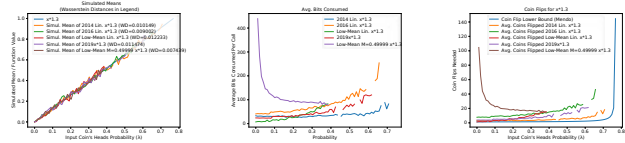
2014 Add. x+0.3



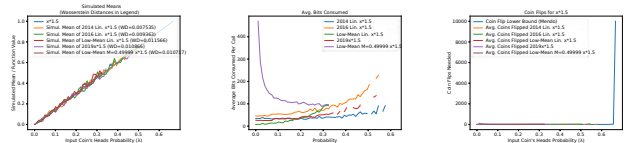
2014 Add. x+0.5



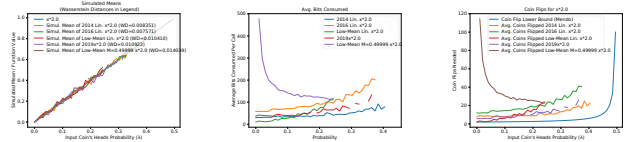
2014 Lin. x*1.3



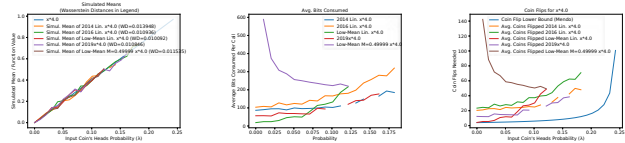
2014 Lin. x*1.5



2014 Lin. x*2.0



2014 Lin. x*4.0



[illegible]

(a) ROC Curve

Legend:

- Proposed (w=0.5)
- Baseline (w=0.0)
- Baseline (w=0.1)
- Baseline (w=0.2)
- Baseline (w=0.3)
- Baseline (w=0.4)
- Baseline (w=0.5)
- Baseline (w=0.6)
- Baseline (w=0.7)
- Baseline (w=0.8)
- Baseline (w=0.9)
- Baseline (w=1.0)

(b) Avg. Size (Commented)

Legend:

- Proposed (w=0.5)
- Baseline (w=0.0)
- Baseline (w=0.1)
- Baseline (w=0.2)
- Baseline (w=0.3)
- Baseline (w=0.4)
- Baseline (w=0.5)
- Baseline (w=0.6)
- Baseline (w=0.7)
- Baseline (w=0.8)
- Baseline (w=0.9)
- Baseline (w=1.0)

(c) CPU Time (Seconds)

Legend:

- Proposed (w=0.5)
- Baseline (w=0.0)
- Baseline (w=0.1)
- Baseline (w=0.2)
- Baseline (w=0.3)
- Baseline (w=0.4)
- Baseline (w=0.5)
- Baseline (w=0.6)
- Baseline (w=0.7)
- Baseline (w=0.8)
- Baseline (w=0.9)
- Baseline (w=1.0)

Figure 1 consists of three subplots. Subplot (a) is a scatter plot titled 'Scatterplot Results (Proposed Model)' showing 'Actual Values' on the x-axis and 'Predicted Values' on the y-axis, both ranging from 0.0 to 1.0. It includes data for 2018, 2019, and 2020, with a legend for each year. Subplot (b) is a line graph titled 'Avg. RMSE by Dataset' showing 'Avg. RMSE' on the y-axis (0.6 to 0.9) versus 'Dataset' on the x-axis. The legend lists various datasets with their corresponding years and models. Subplot (c) is a confusion matrix titled 'Conf. Matrix for 2018, 2020000' showing 'Conf. Matrix' on the y-axis (0.0 to 1.0) versus 'Actual Values' on the x-axis (0.0 to 1.0). The legend indicates 'Conf. Matrix' and 'Actual Values' for 2018 and 2020000.

Figure 1 consists of three subplots. Subplot (a) is a line graph titled 'Training Loss' showing the loss on the y-axis (0.0 to 0.6) against Epochs on the x-axis (0 to 100). It includes a legend for 'Train Loss' and 'Valid Loss' for years 2014-2018. All lines show a decreasing trend, with training loss generally lower than validation loss. Subplot (b) is a line graph titled 'F1 Score' showing the average F1 score on the y-axis (0.0 to 1.0) against Epochs on the x-axis (0 to 100). It includes a legend for 'F1 Score' for years 2014-2018. The lines fluctuate between 0.5 and 0.9. Subplot (c) is a line graph titled 'Count vs. F1 Score' showing the count on the y-axis (0 to 100) against F1 Score on the x-axis (0.0 to 1.0). It includes a legend for 'Count' for years 2014-2018. The counts are mostly concentrated between 0.5 and 0.8 F1 scores.

Figure 1 consists of three subplots. The left subplot, titled 'Uncorrelated Results', shows the 'True Mean Value' on the y-axis (ranging from 0.0 to 1.0) versus the 'Estimated Mean' on the x-axis (ranging from 0.0 to 1.0). It displays data for various models and datasets, including 'Data 1, Dataset 1', 'Data 1, Dataset 2', 'Data 1, Dataset 3', 'Data 1, Dataset 4', 'Data 1, Dataset 5', 'Data 1, Dataset 6', 'Data 1, Dataset 7', 'Data 1, Dataset 8', 'Data 1, Dataset 9', 'Data 1, Dataset 10', 'Data 1, Dataset 11', 'Data 1, Dataset 12', 'Data 1, Dataset 13', 'Data 1, Dataset 14', 'Data 1, Dataset 15', 'Data 1, Dataset 16', 'Data 1, Dataset 17', 'Data 1, Dataset 18', 'Data 1, Dataset 19', 'Data 1, Dataset 20', 'Data 1, Dataset 21', 'Data 1, Dataset 22', 'Data 1, Dataset 23', 'Data 1, Dataset 24', 'Data 1, Dataset 25', 'Data 1, Dataset 26', 'Data 1, Dataset 27', 'Data 1, Dataset 28', 'Data 1, Dataset 29', 'Data 1, Dataset 30', 'Data 1, Dataset 31', 'Data 1, Dataset 32', 'Data 1, Dataset 33', 'Data 1, Dataset 34', 'Data 1, Dataset 35', 'Data 1, Dataset 36', 'Data 1, Dataset 37', 'Data 1, Dataset 38', 'Data 1, Dataset 39', 'Data 1, Dataset 40', 'Data 1, Dataset 41', 'Data 1, Dataset 42', 'Data 1, Dataset 43', 'Data 1, Dataset 44', 'Data 1, Dataset 45', 'Data 1, Dataset 46', 'Data 1, Dataset 47', 'Data 1, Dataset 48', 'Data 1, Dataset 49', 'Data 1, Dataset 50', 'Data 1, Dataset 51', 'Data 1, Dataset 52', 'Data 1, Dataset 53', 'Data 1, Dataset 54', 'Data 1, Dataset 55', 'Data 1, Dataset 56', 'Data 1, Dataset 57', 'Data 1, Dataset 58', 'Data 1, Dataset 59', 'Data 1, Dataset 60', 'Data 1, Dataset 61', 'Data 1, Dataset 62', 'Data 1, Dataset 63', 'Data 1, Dataset 64', 'Data 1, Dataset 65', 'Data 1, Dataset 66', 'Data 1, Dataset 67', 'Data 1, Dataset 68', 'Data 1, Dataset 69', 'Data 1, Dataset 70', 'Data 1, Dataset 71', 'Data 1, Dataset 72', 'Data 1, Dataset 73', 'Data 1, Dataset 74', 'Data 1, Dataset 75', 'Data 1, Dataset 76', 'Data 1, Dataset 77', 'Data 1, Dataset 78', 'Data 1, Dataset 79', 'Data 1, Dataset 80', 'Data 1, Dataset 81', 'Data 1, Dataset 82', 'Data 1, Dataset 83', 'Data 1, Dataset 84', 'Data 1, Dataset 85', 'Data 1, Dataset 86', 'Data 1, Dataset 87', 'Data 1, Dataset 88', 'Data 1, Dataset 89', 'Data 1, Dataset 90', 'Data 1, Dataset 91', 'Data 1, Dataset 92', 'Data 1, Dataset 93', 'Data 1, Dataset 94', 'Data 1, Dataset 95', 'Data 1, Dataset 96', 'Data 1, Dataset 97', 'Data 1, Dataset 98', 'Data 1, Dataset 99', 'Data 1, Dataset 100'. The middle subplot, titled 'Avg. Bias Coefficient', shows the 'Avg. Bias Coefficient (in %)' on the y-axis (ranging from -10.0 to 10.0) versus the 'Avg. Bias Coefficient' on the x-axis (ranging from -10.0 to 10.0). The right subplot, titled 'Con. Bias Coefficient', shows the 'Con. Bias Coefficient' on the y-axis (ranging from 0.0 to 1.0) versus the 'Con. Bias Coefficient' on the x-axis (ranging from 0.0 to 1.0).

Figure 1 consists of three subplots illustrating the performance of the proposed model on the CIFAR-100 dataset.

Left Subplot: Generalization Results
 This plot shows True Accuracy (Y-axis, 0.0 to 0.8) versus True Calibration (X-axis, 0.00 to 0.75). The legend includes:

- Blue line: Data & Model
- Red line: Data & Model + Histogram
- Green line: Data & Model + Histogram + Kernel
- Orange line: Data & Model + Histogram + Kernel + GMM
- Purple line: Data & Model + Histogram + Kernel + GMM + GMM
- Yellow line: Data & Model + Histogram + Kernel + GMM + GMM + GMM
- Light blue line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM
- Light green line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM
- Light orange line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM
- Light purple line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM
- Light yellow line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM + GMM

 The plot shows that the proposed model (Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM + GMM) achieves the highest true accuracy across the calibration range.

Middle Subplot: Risk-Step Comparison
 This plot shows Average Risk (Y-axis, 0.00 to 0.25) versus Step (X-axis, 0.00 to 0.75). The legend includes:

- Blue line: Data & Model
- Red line: Data & Model + Histogram
- Green line: Data & Model + Histogram + Kernel
- Orange line: Data & Model + Histogram + Kernel + GMM
- Purple line: Data & Model + Histogram + Kernel + GMM + GMM
- Yellow line: Data & Model + Histogram + Kernel + GMM + GMM + GMM
- Light blue line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM
- Light green line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM
- Light orange line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM
- Light purple line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM
- Light yellow line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM + GMM

 The plot shows that the proposed model (Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM + GMM) maintains a low average risk across the calibration range.

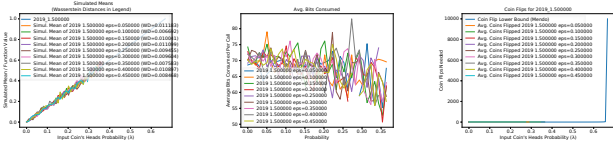
Right Subplot: Cal-Step Comparison
 This plot shows CoV Risk (Y-axis, 0.00 to 0.05) versus Step (X-axis, 0.00 to 0.75). The legend includes:

- Blue line: Data & Model
- Red line: Data & Model + Histogram
- Green line: Data & Model + Histogram + Kernel
- Orange line: Data & Model + Histogram + Kernel + GMM
- Purple line: Data & Model + Histogram + Kernel + GMM + GMM
- Yellow line: Data & Model + Histogram + Kernel + GMM + GMM + GMM
- Light blue line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM
- Light green line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM
- Light orange line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM
- Light purple line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM
- Light yellow line: Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM + GMM

 The plot shows that the proposed model (Data & Model + Histogram + Kernel + GMM + GMM + GMM + GMM + GMM + GMM + GMM + GMM) maintains a low CoV risk across the calibration range.

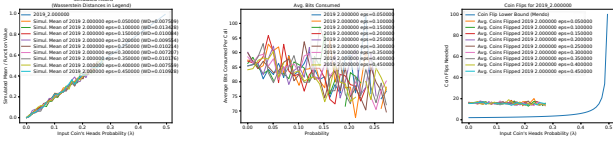
[illegible]

eps=0.050000



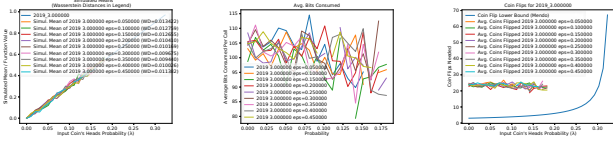
2019 2.000000

eps=0.050000



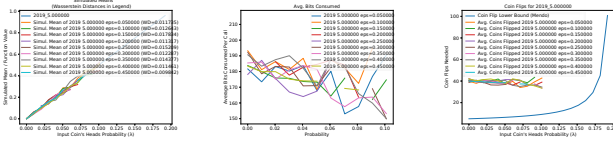
2019 3.000000

eps=0.050000



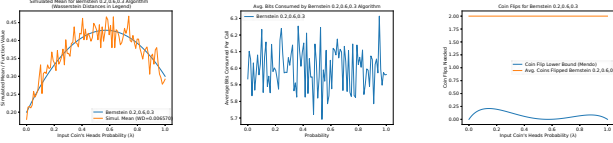
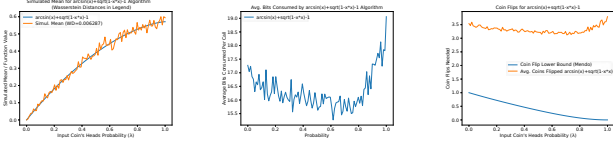
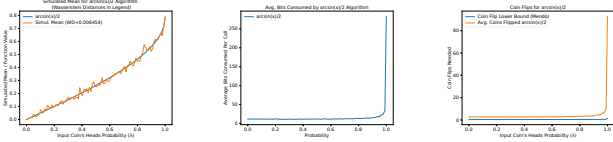
2019 5.000000

eps=0.050000

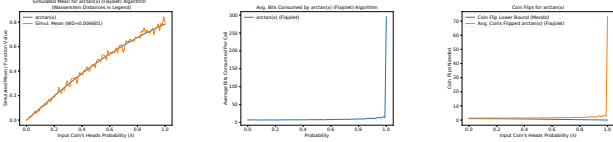


Bernstein

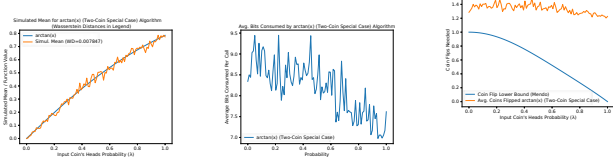
0.2,0.6,0.3


$$\arcsin(x) + \sqrt{1-x^2} - 1$$
 $\arcsin(x)/2$  $\arctan(x)$

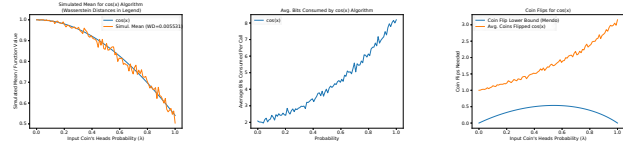
(Flajolet)



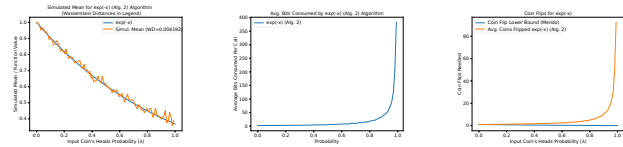
arctan(x) (Two-Coin Special Case)



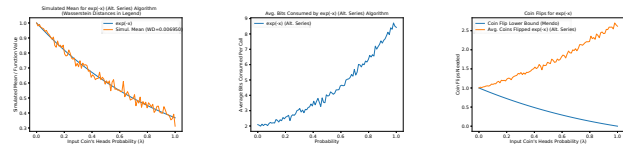
$\cos(x)$



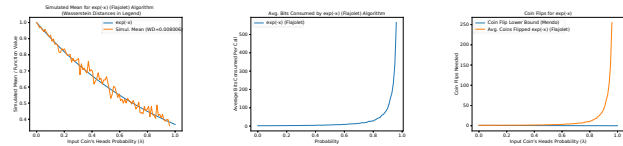
$\exp(-x)$ (Alg. 2)



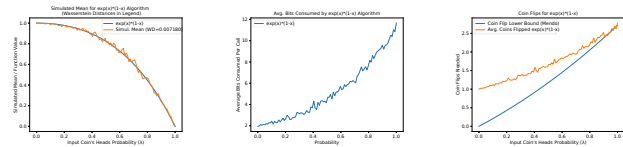
$\exp(-x)$ (Alt. Series)



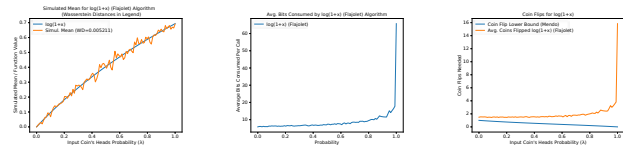
$\exp(-x)$ (Flajolet)



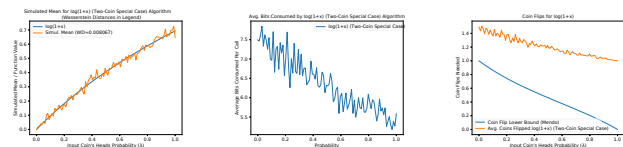
$\exp(x)*(1-x)$



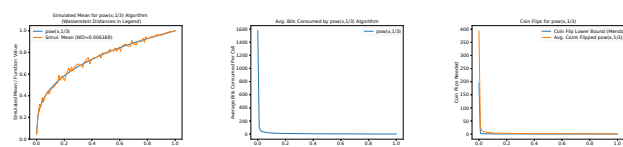
$\ln(1+x)$ (Flajolet)



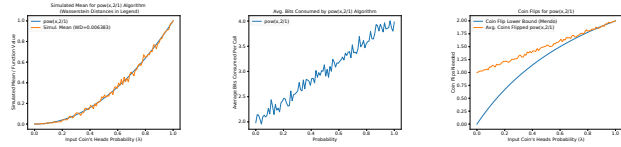
$\ln(1+x)$ (Two-Coin Special Case)



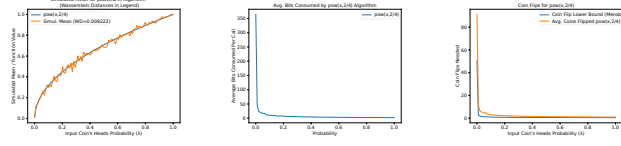
$\text{pow}(x, 1/3)$



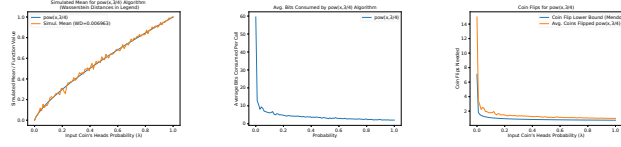
pow(x,2/1)



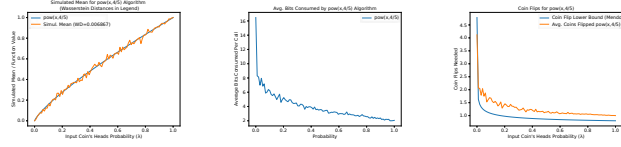
pow(x,2/4)



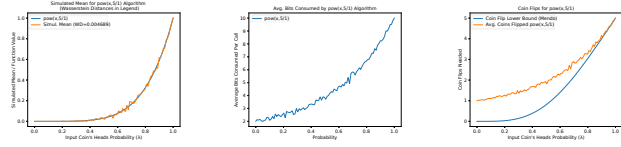
pow(x,3/4)



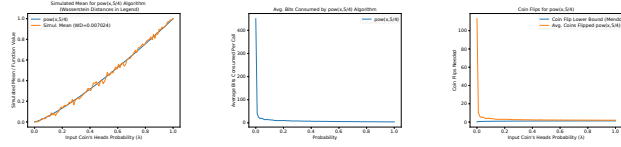
pow(x,4/5)



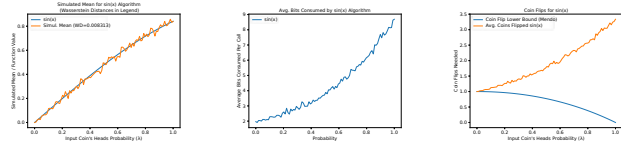
pow(x,5/1)



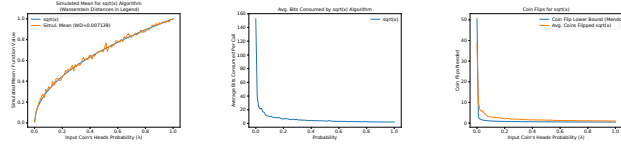
pow(x,5/4)



sin(x)



sqrt(x)



1. <https://peteroupc.github.io/bernoulli.md>