

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—

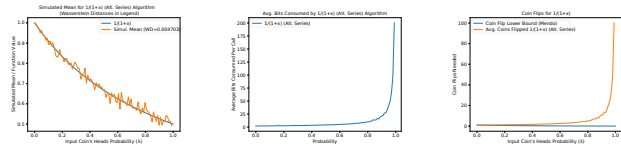
- an ϵ of $1 - (x + c) * 1.001$ was used (or 0.0001 if ϵ would be greater than 1), and
- an ϵ of $(x - c) * 0.9995$ for the subtraction variants.

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.

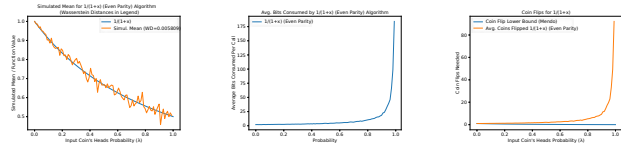
1.1 The Charts

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)			

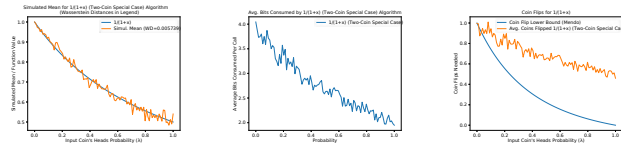
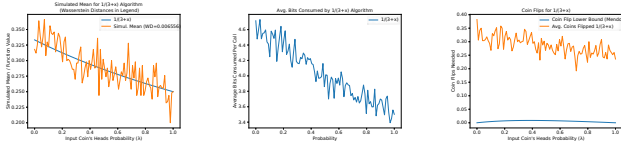
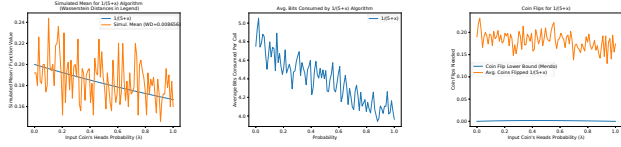
$1/(1+x)$ (Alt. Series)



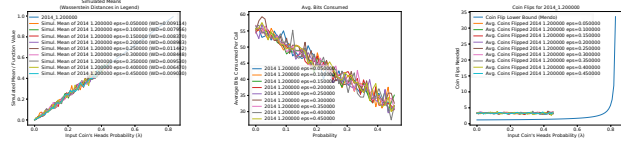
$1/(1+x)$ (Even Parity)



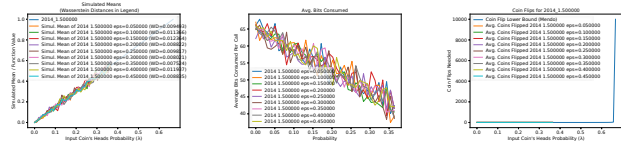
$1/(1+x)$ (Two-Coin Special Case)


$$1/(3+x)$$

$$1/(5+x)$$


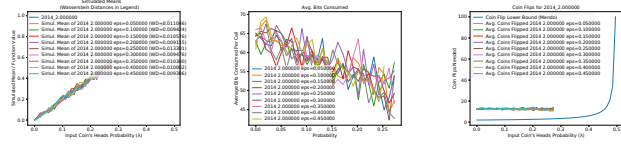
2014 1.200000
eps=0.050000



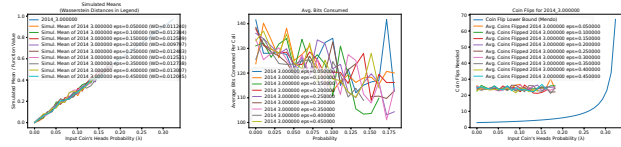
2014 1.500000
eps=0.050000



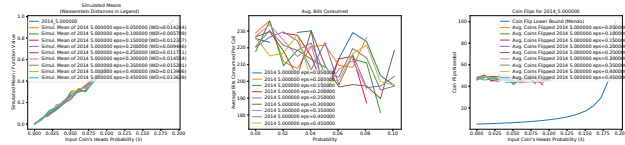
2014 2.000000
eps=0.050000



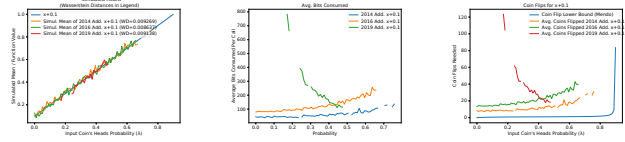
```
2014 3.000000
eps=0.050000
```



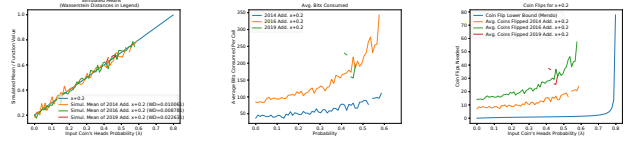
2014 5.000000
eps=0.050000



2014 Add. x+0.1



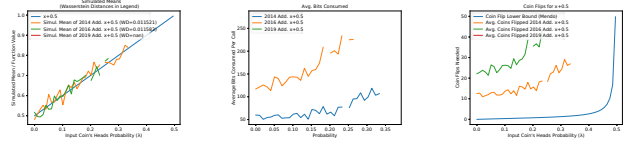
2014 Add. x+0.2



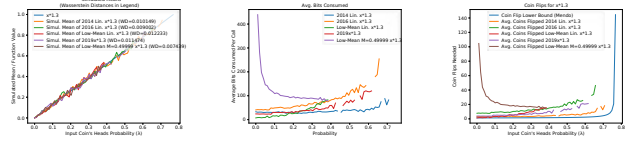
2014 Add. x+0.3



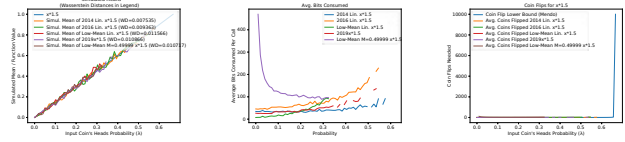
2014 Add. x+0.5



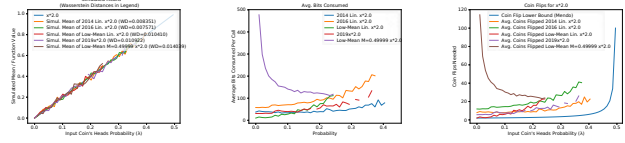
2014 Lin. x*1.3



2014 Lin. x*1.5



2014 Lin. x*2.0



2014 Lin. x*4.0

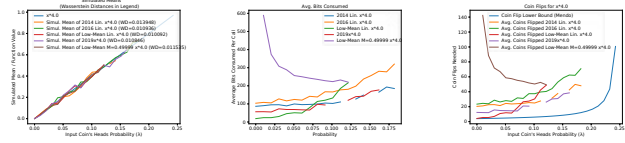


Figure 1 consists of three subplots. Subplot (a) is a scatter plot titled 'Simulated Results' showing 'True vs. Predicted values for the proposed model'. The x-axis is 'Input (Randomly Generated)' and the y-axis is 'Output (Randomly Generated)'. Both axes range from 0.0 to 1.0. A diagonal line represents the identity function. Several data series are plotted, each corresponding to a different model and its regression coefficients. Subplot (b) is a line plot titled 'Reg. Coeff. Comparison' showing 'Coefficients and Correlation' on the y-axis (ranging from 0.0 to 1.0) versus 'Parameter' on the x-axis (ranging from 0.0 to 1.0). The parameters are β_0 , β_1 , β_2 , β_3 , β_4 , β_5 , β_6 , β_7 , β_8 , β_9 , β_{10} , β_{11} , β_{12} , β_{13} , β_{14} , β_{15} , β_{16} , β_{17} , β_{18} , β_{19} , β_{20} , β_{21} , β_{22} , β_{23} , β_{24} , β_{25} , β_{26} , β_{27} , β_{28} , β_{29} , β_{30} , β_{31} , β_{32} , β_{33} , β_{34} , β_{35} , β_{36} , β_{37} , β_{38} , β_{39} , β_{40} , β_{41} , β_{42} , β_{43} , β_{44} , β_{45} , β_{46} , β_{47} , β_{48} , β_{49} , β_{50} , β_{51} , β_{52} , β_{53} , β_{54} , β_{55} , β_{56} , β_{57} , β_{58} , β_{59} , β_{60} , β_{61} , β_{62} , β_{63} , β_{64} , β_{65} , β_{66} , β_{67} , β_{68} , β_{69} , β_{70} , β_{71} , β_{72} , β_{73} , β_{74} , β_{75} , β_{76} , β_{77} , β_{78} , β_{79} , β_{80} , β_{81} , β_{82} , β_{83} , β_{84} , β_{85} , β_{86} , β_{87} , β_{88} , β_{89} , β_{90} , β_{91} , β_{92} , β_{93} , β_{94} , β_{95} , β_{96} , β_{97} , β_{98} , β_{99} , β_{100} . Subplot (c) is a line plot titled 'Coeff. Error for $w=0.5$ ' showing 'Coeff. Error' on the y-axis (ranging from 0.0 to 1.0) versus 'Input (Randomly Generated)' on the x-axis (ranging from 0.0 to 1.0). The plot compares the proposed model with several existing models: 'Proposed Model', 'Proposed Model with $w=0.5$ ', 'Proposed Model with $w=0.7$ ', 'Proposed Model with $w=0.9$ ', 'Proposed Model with $w=1.0$ ', 'Proposed Model with $w=1.1$ ', 'Proposed Model with $w=1.2$ ', 'Proposed Model with $w=1.3$ ', 'Proposed Model with $w=1.4$ ', 'Proposed Model with $w=1.5$ ', 'Proposed Model with $w=1.6$ ', 'Proposed Model with $w=1.7$ ', 'Proposed Model with $w=1.8$ ', 'Proposed Model with $w=1.9$ ', 'Proposed Model with $w=2.0$ ', 'Proposed Model with $w=2.1$ ', 'Proposed Model with $w=2.2$ ', 'Proposed Model with $w=2.3$ ', 'Proposed Model with $w=2.4$ ', 'Proposed Model with $w=2.5$ ', 'Proposed Model with $w=2.6$ ', 'Proposed Model with $w=2.7$ ', 'Proposed Model with $w=2.8$ ', 'Proposed Model with $w=2.9$ ', 'Proposed Model with $w=3.0$ ', 'Proposed Model with $w=3.1$ ', 'Proposed Model with $w=3.2$ ', 'Proposed Model with $w=3.3$ ', 'Proposed Model with $w=3.4$ ', 'Proposed Model with $w=3.5$ ', 'Proposed Model with $w=3.6$ ', 'Proposed Model with $w=3.7$ ', 'Proposed Model with $w=3.8$ ', 'Proposed Model with $w=3.9$ ', 'Proposed Model with $w=4.0$ ', 'Proposed Model with $w=4.1$ ', 'Proposed Model with $w=4.2$ ', 'Proposed Model with $w=4.3$ ', 'Proposed Model with $w=4.4$ ', 'Proposed Model with $w=4.5$ ', 'Proposed Model with $w=4.6$ ', 'Proposed Model with $w=4.7$ ', 'Proposed Model with $w=4.8$ ', 'Proposed Model with $w=4.9$ ', 'Proposed Model with $w=5.0$ ', 'Proposed Model with $w=5.1$ ', 'Proposed Model with $w=5.2$ ', 'Proposed Model with $w=5.3$ ', 'Proposed Model with $w=5.4$ ', 'Proposed Model with $w=5.5$ ', 'Proposed Model with $w=5.6$ ', 'Proposed Model with $w=5.7$ ', 'Proposed Model with $w=5.8$ ', 'Proposed Model with $w=5.9$ ', 'Proposed Model with $w=6.0$ ', 'Proposed Model with $w=6.1$ ', 'Proposed Model with $w=6.2$ ', 'Proposed Model with $w=6.3$ ', 'Proposed Model with $w=6.4$ ', 'Proposed Model with $w=6.5$ ', 'Proposed Model with $w=6.6$ ', 'Proposed Model with $w=6.7$ ', 'Proposed Model with $w=6.8$ ', 'Proposed Model with $w=6.9$ ', 'Proposed Model with $w=7.0$ ', 'Proposed Model with $w=7.1$ ', 'Proposed Model with $w=7.2$ ', 'Proposed Model with $w=7.3$ ', 'Proposed Model with $w=7.4$ ', 'Proposed Model with $w=7.5$ ', 'Proposed Model with $w=7.6$ ', 'Proposed Model with $w=7.7$ ', 'Proposed Model with $w=7.8$ ', 'Proposed Model with $w=7.9$ ', 'Proposed Model with $w=8.0$ ', 'Proposed Model with $w=8.1$ ', 'Proposed Model with $w=8.2$ ', 'Proposed Model with $w=8.3$ ', 'Proposed Model with $w=8.4$ ', 'Proposed Model with $w=8.5$ ', 'Proposed Model with $w=8.6$ ', 'Proposed Model with $w=8.7$ ', 'Proposed Model with $w=8.8$ ', 'Proposed Model with $w=8.9$ ', 'Proposed Model with $w=9.0$ ', 'Proposed Model with $w=9.1$ ', 'Proposed Model with $w=9.2$ ', 'Proposed Model with $w=9.3$ ', 'Proposed Model with $w=9.4$ ', 'Proposed Model with $w=9.5$ ', 'Proposed Model with $w=9.6$ ', 'Proposed Model with $w=9.7$ ', 'Proposed Model with $w=9.8$ ', 'Proposed Model with $w=9.9$ ', 'Proposed Model with $w=10.0$ '.

[illegible]

Figure 1 consists of three subplots labeled (a), (b), and (c). Each subplot shows the performance of the proposed model in terms of ROC curve, AUC, and Confusion Matrix.

(a) ROC curve for the proposed model. The x-axis is 'True Class vs. False Positivity' and the y-axis is 'True Class vs. False Negativity'. The curve is a solid blue line, and the AUC is 0.95. The legend indicates the model is 'Proposed Model'.

(b) ROC curve for the proposed model. The x-axis is 'True Class vs. False Positivity' and the y-axis is 'True Class vs. False Negativity'. The curve is a solid blue line, and the AUC is 0.95. The legend indicates the model is 'Proposed Model'.

(c) ROC curve for the proposed model. The x-axis is 'True Class vs. False Positivity' and the y-axis is 'True Class vs. False Negativity'. The curve is a solid blue line, and the AUC is 0.95. The legend indicates the model is 'Proposed Model'.

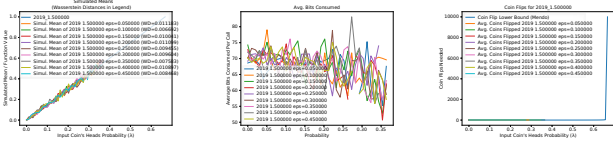
Figure 1 consists of three subplots. Subplot (a) is a scatter plot titled 'Scatterplot Results' showing 'True Rank (Rank)' on the x-axis and 'Estimated Rank (Rank)' on the y-axis, both ranging from 0.0 to 1.0. A diagonal line represents the identity function. Data points are labeled with year and rank, such as '2016 - Rank 1' and '201600000 - Rank 1000000'. Subplot (b) is a line plot titled 'Avg. Rank Comparison' showing 'Avg. Rank' on the y-axis (42 to 100) against 'Rank' on the x-axis (0.0 to 1.0). It compares '2016' (blue line) and '201600000' (orange line) across various ranks. Subplot (c) is a line plot titled 'Cost Ratio for 2016 & 201600000' showing 'Cost Ratio' on the y-axis (0.0 to 10000) against 'Rank' on the x-axis (0.0 to 1.0). It compares '2016' (blue line) and '201600000' (orange line) across various ranks.

Figure 1 consists of three subplots. Subplot (a) is a scatter plot titled 'Normalized Mean Squared Error' showing NMSE on the y-axis (0.0 to 0.6) versus Input Correlation Coefficient (ρ) on the x-axis (0.0 to 0.5). It contains 10 data series for different SNR values (0 dB to 20 dB) and two methods: 'Proposed' (blue lines) and 'ML' (red lines). Subplot (b) is a line plot titled 'Avg. Squ. Error' showing ASE on the y-axis (0.00 to 0.04) versus Probability on the x-axis (0.00 to 0.25). It shows ASE for the proposed model and ML for SNR values from 0 dB to 20 dB. Subplot (c) is a line plot titled 'Cost Function' showing Cost Function on the y-axis (0 to 10) versus Input Correlation Coefficient (ρ) on the x-axis (0.0 to 0.5). It shows the cost function for the proposed model and ML for SNR values from 0 dB to 20 dB.

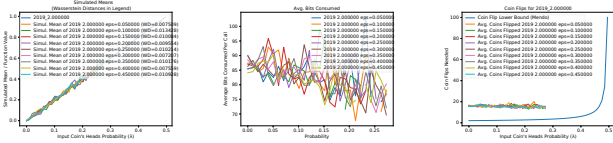
[illegible]

Figure 1 consists of three subplots. The left subplot is an ROC curve for the 2014-2015 dataset, showing the True Positive Rate (TPR) on the y-axis (0.0 to 1.0) versus the False Positive Rate (FPR) on the x-axis (0.00 to 0.60). The proposed model (red line) shows a significantly better performance than the baseline (black line). The middle subplot is an ROC curve for the 2015-2016 dataset, showing the True Positive Rate (TPR) on the y-axis (0.0 to 1.0) versus the False Positive Rate (FPR) on the x-axis (0.00 to 0.10). The proposed model (red line) shows a significantly better performance than the baseline (black line). The right subplot is a Cost Ratio plot for the 2014-2015 dataset, showing the Cost Ratio on the y-axis (0.00 to 1.00) versus the Cost Ratio on the x-axis (0.00 to 0.70). The proposed model (red line) maintains a cost ratio near 1.0 across all cost ratios.

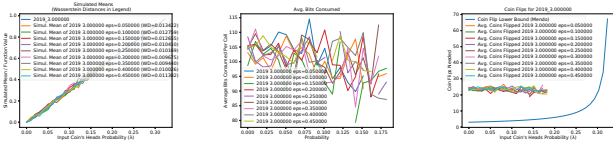
$\text{eps}=0.050000$



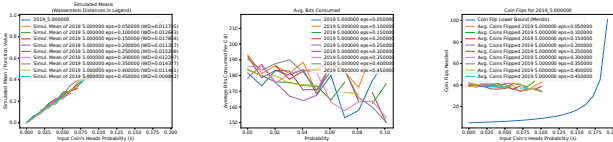
2019 2.000000
 $\text{eps}=0.050000$



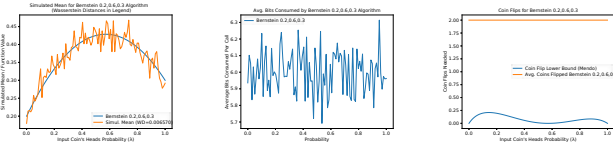
2019 3.000000
 $\text{eps}=0.050000$



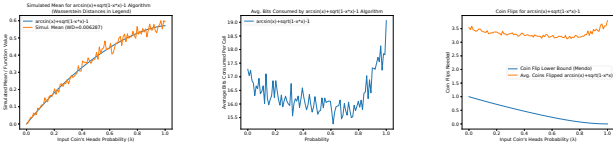
2019 5.000000
 $\text{eps}=0.050000$



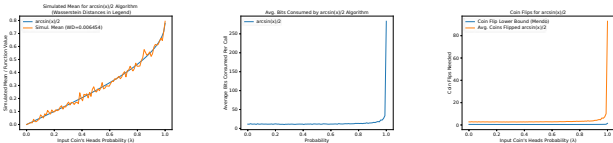
Bernstein
0.2,0.6,0.3



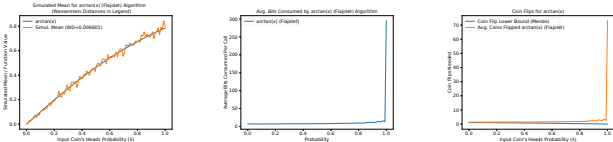
$\arcsin(x)+\sqrt{1-x^2}-1$



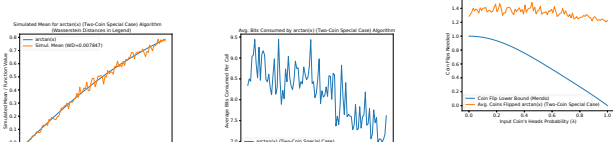
$\arcsin(x)/2$



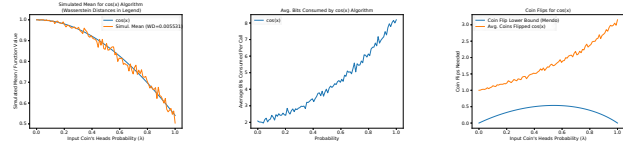
$\arctan(x)$
(Flajolet)



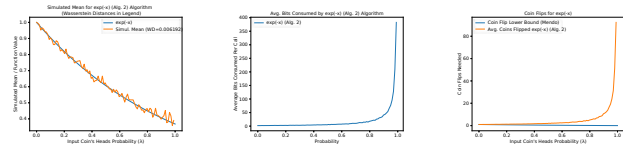
$\arctan(x)$ (Two-Coin Special Case)



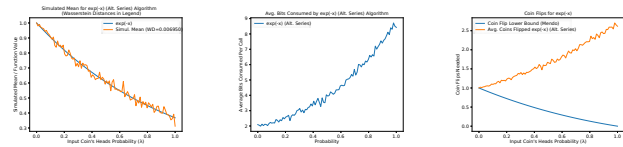
$\cos(x)$



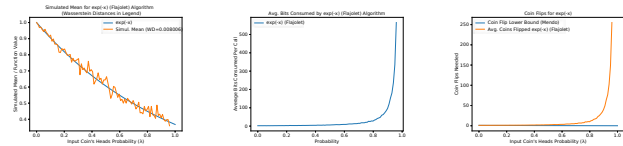
$\exp(-x)$ (Alg. 2)



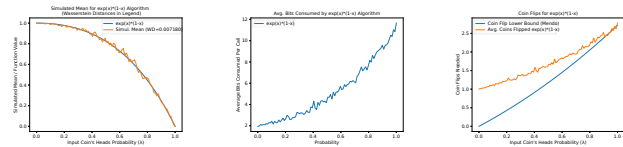
$\exp(-x)$ (Alt. Series)



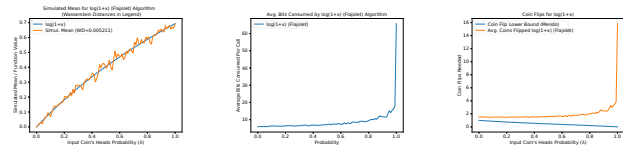
$\exp(-x)$ (Flajolet)



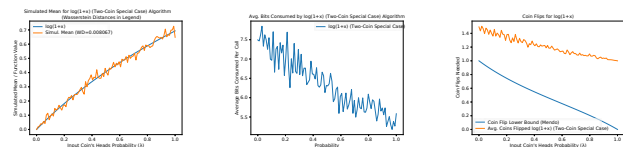
$\exp(x)*(1-x)$



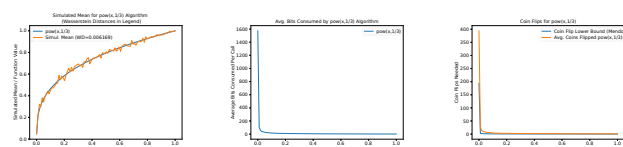
$\ln(1+x)$ (Flajolet)



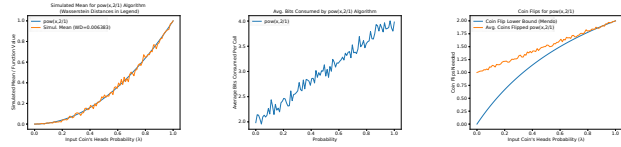
$\ln(1+x)$ (Two-Coin Special Case)



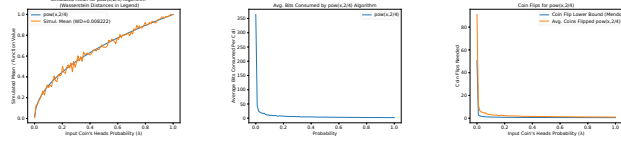
$\text{pow}(x, 1/3)$



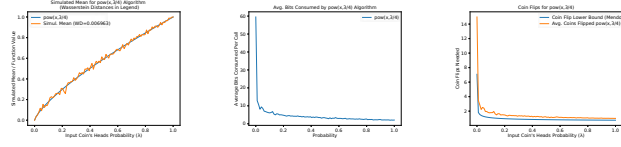
pow(x,2/1)



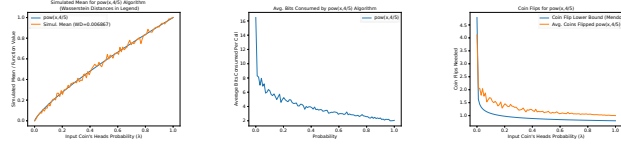
pow(x,2/4)



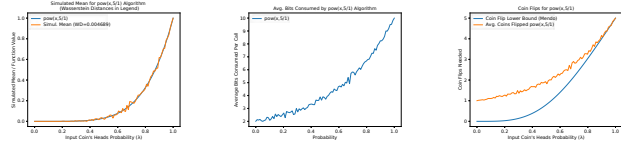
pow(x,3/4)



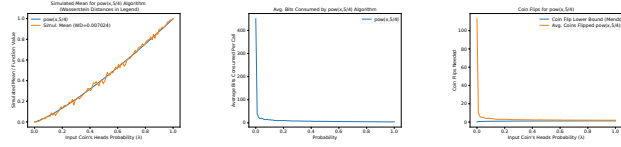
pow(x,4/5)



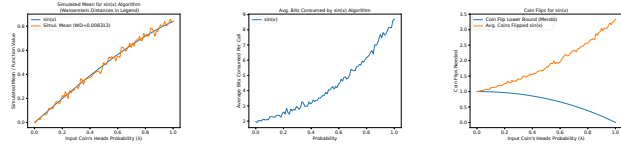
pow(x,5/1)



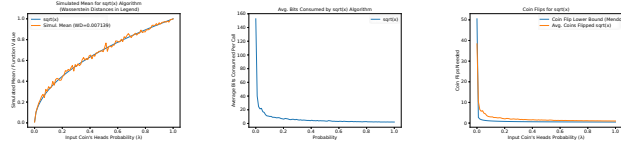
pow(x,5/4)



sin(x)



sqrt(x)



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