Performance Impact of Library Cell Size Quantization

David Harris & Dan Hartman

Good algorithms exist to size gates in a path in order to minimize delay, area, power, or various other metrics. However, the algorithms generally assume a continuous range of allowable gate sizes. Gates are often selected from a standard cell library which has a discrete set of gate sizes. Mapping from the continuous range of sizes produced by the sizing algorithm onto the discrete set may result in a suboptimal design, especially if the set of gate sizes is very limited. This paper examines the performance impact of such a mapping.

1.0 Problem Formulation

Consider the problem of assigning sizes to gates in order to minimize the delay through a path. For gates with the same input capacitance on all inputs, the size can be expressed as the input capacitance of any input. Various sizing algorithms are capable of minimizing delay through the path as long as a continuous range of gate sizes are available. We call this delay t_{oc}, for the optimal delay allowing continuous gate sizes.

In semi-custom design, only a discrete range of gate sizes are available. Integer programming could be used to assign the best sizes, resulting in an optimal delay t_{od} greater or equal to t_{oc}. Integer programming may be computationally infeasible, so we may instead modify the continuous algorithm to round gate sizes to the nearest available library cell, giving a time t_{nd}. For purposes of analysis, we may also consider truncating the gate size downward instead of rounding to nearest. The delay of such a truncated path t_{td} is even more conservative:

$$t_{oc} \le t_{od} \le t_{nd} \le t_{td} \qquad \text{(EQ 1)}$$

We are interested in the penalty we pay for discrete gate sizes, which can be conservatively bounded at t_{td}/t_{oc}. The penalty will be a function of the set of discrete gate sizes available; as a more complete set of sizes is introduced, the path delay should approach t_{oc}.

2.0 Analysis

This section analyzes the quantization penalty, both in closed form and through Monte-Carlo simulation.

2.1 Inverter Chains

To quantify the penalty t_{td}/t_{oc}, let us consider a chain of N inverters driving a total fanout f^N. The first inverter has size 1 and the load is f^N. Let the library have inverter sizes k^s, for all non-negative integers s and some constant k. Let the delay of each inverter be a + $b*C_{out}/C_{in}$ for some constants a and b.

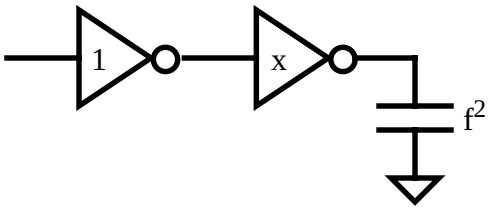
Small k indicates a more complete library. The problem approaches continuous sizing as k approaches 1. A good industrial library has k around 1.5. A less complete library might have k of 2 or greater. The HP26 0.8 micron process has a=98 ps and b=81 ps.

The penalty will vary with f, N, and k. We can consider two limits analytically: N=2 and N->inf. Only the N=2 analysis is presented here.

2.1.1 N=2

Consider the following inverter chain shown below. The delay through the chain is:

$$a + bx + a + f^2/x \qquad \text{(EQ 2)}$$



The optimal delay given continuous sizing is t_{oc} = 2a + bf*2. The sizing assigned in the continuous case depends on k and f. If we truncate down to a discrete size, the size selected will be k^i for some integer i and j in the range [0,1)[1] such that $f = k^{i+j}$.

Thus, the delay is $t_{td} = 2a + b(k^1 + f^2/k^i)$, or can be rewritten as $2a + bf(k^j + k^{-j})$. In this form, it is clear that the delay becomes more and more suboptimal as j approaches 1 (in other words, when the quantized gate size is furthest from the required gate size).

We find the quantization performance penalty:

$$\frac{t_{td}}{t_{oc}} = \frac{1 + \frac{bf}{2a}\left(k^j + k^{-j}\right)}{1 + \frac{bf}{a}} \qquad \text{(EQ 3)}$$

This penalty is bounded at j = 1; we can plot curves showing the worst case ratio as k increases for various values of bf/a (the ratio of load-dependent delay to parasitic delay).

1. Assuming f >1 (otherwise, there's no point building an inverter chain!)

bf/a is around 4 in a well-designed path; penalty becomes largest as bf/a approaches infinity.

FIGURE 1. Delay penalty vs. k (worst case)

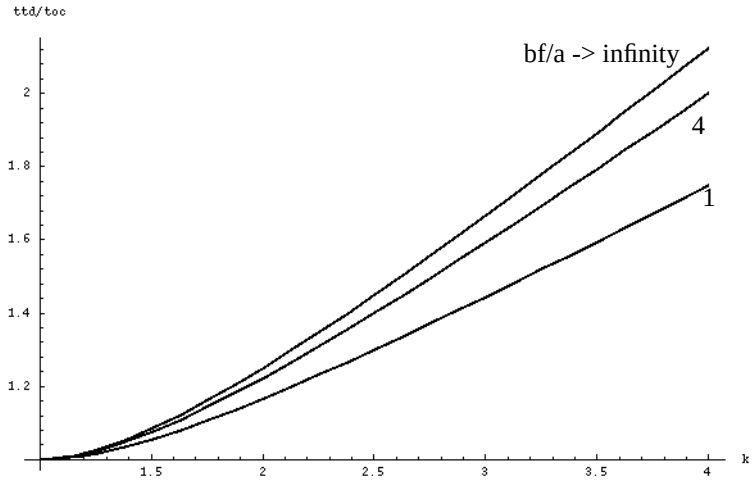
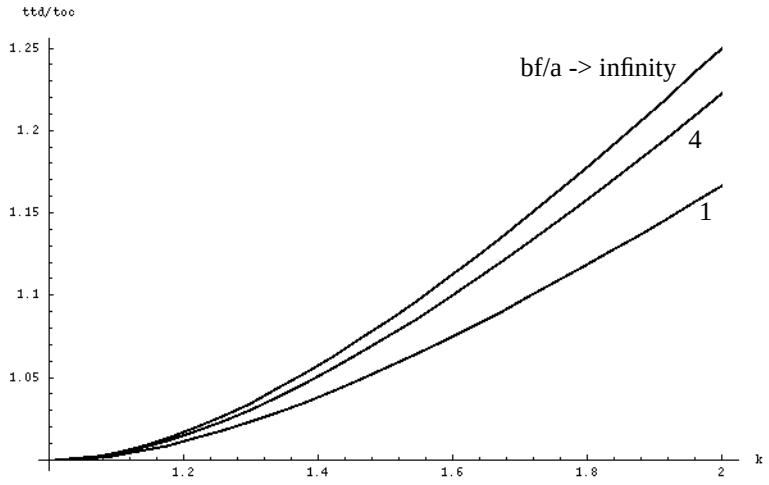


FIGURE 2. Zoom in from Figure 1 for 1 < k < 2



The average case quantization penalty is less than the worst case. Consider the expected delay when f is a uniformly distributed random variable. We can write $f = k^i * r$ for some integer i and r in the range [1, k). The expected delay is:

$$\frac{1}{k-1}\int_{1}^{k}\left[2a + bf\left(r + \frac{1}{r}\right)\right]dr \;=\; 2a + bf\left(\frac{k+1}{2} + \frac{\ln k}{k-1}\right) \qquad \text{(EQ 4)}$$

We can make similar plots of t_{td}/t_{oc} for the average case:

FIGURE 3. Delay penalty vs. k (average case)

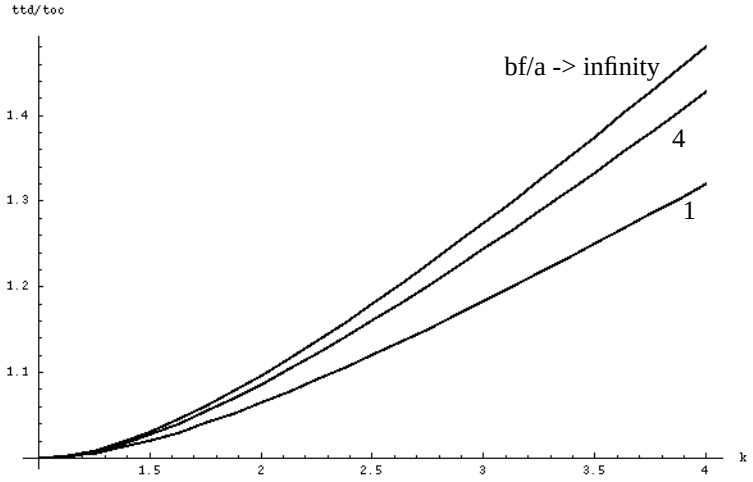
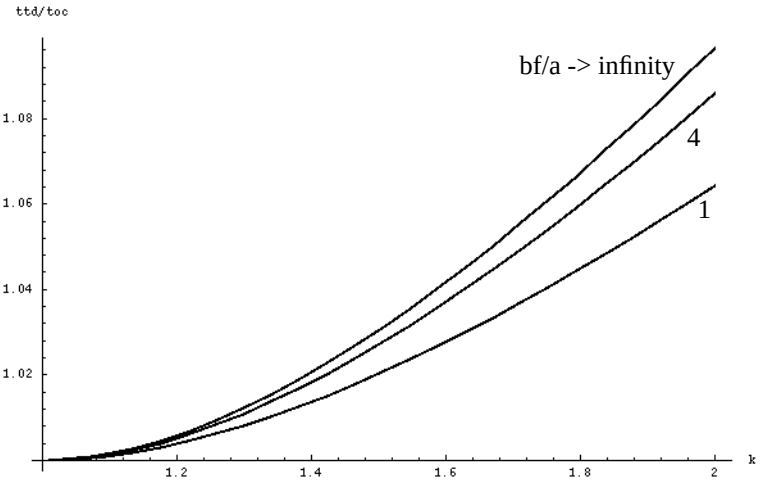
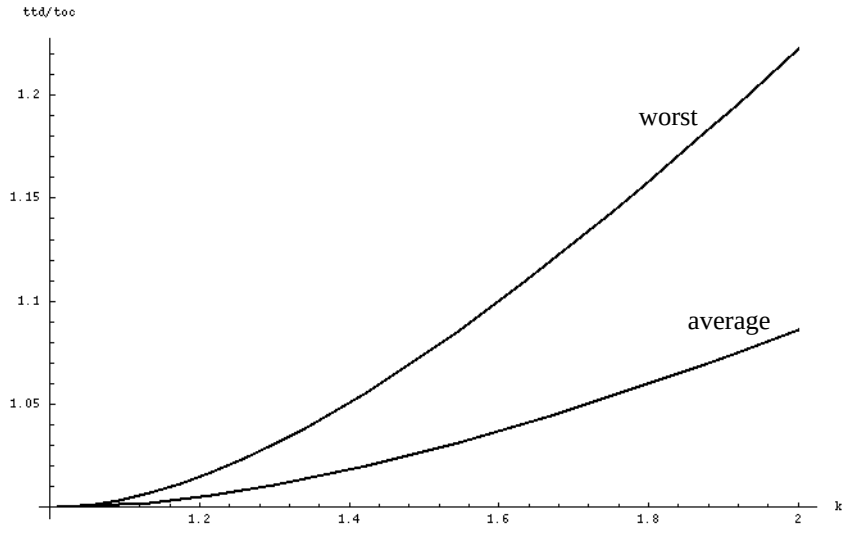


FIGURE 4. Zoom in on Figure 3 1 < k < 2



Finally, we can compare the average and worst case behavior in Figure 5.

FIGURE 5. Comparison of average and worst case quantization penalty (bf/a = 4)



We can improve the results by “geometrically” rounding instead of truncating. Suppose we map gates of size k^{i+j} to k^i for j in [-1/2, 1/2) (rounding), rather than [0, 1) (truncating). The worst case is still described by EQ 3, but is now bounded at by j=1/2. The k scale is effectively compressed to k’ = sqrt(k), as plotted below:

FIGURE 6. Delay penalty vs. k (worst case)

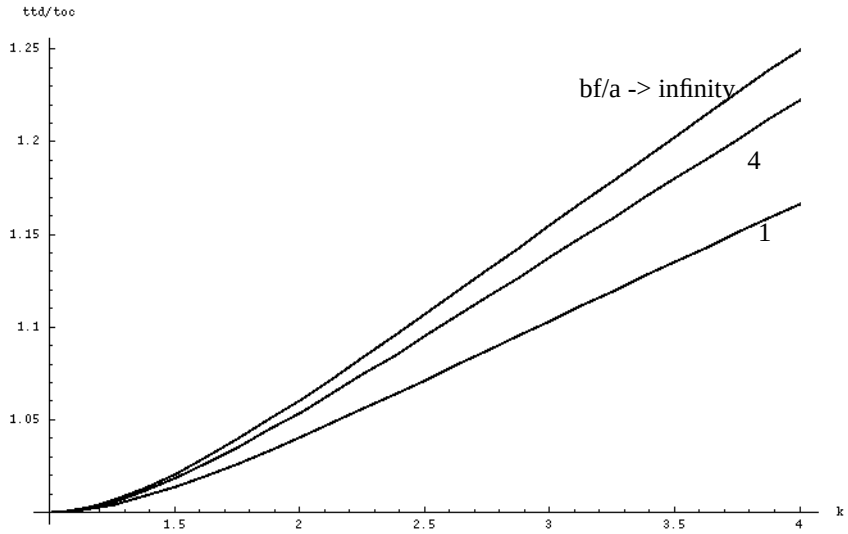
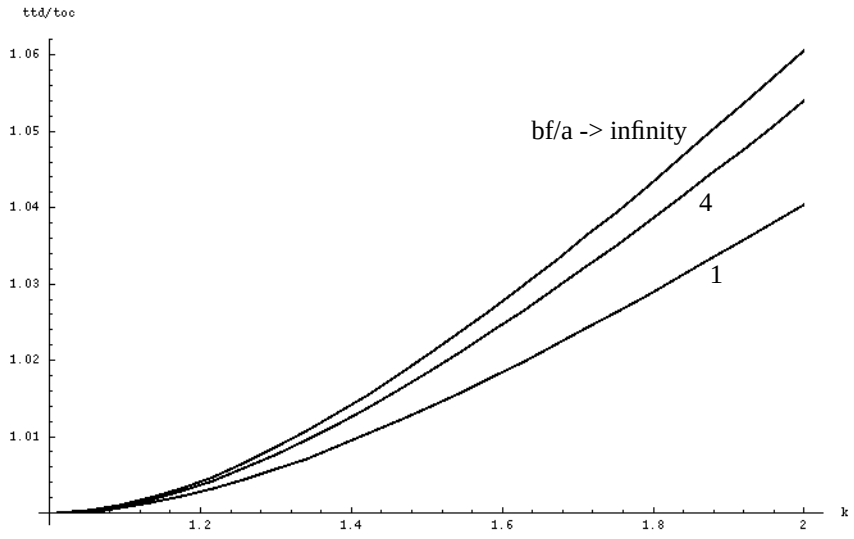


FIGURE 7. Zoom in on Figure 6 1 < k < 2



The average quantization penalty can be found by modifying EQ 4 such that r is in the range [1/sqrt[k], sqrt[k]).

$$\frac{1}{\sqrt{k}-1/\sqrt{k}}\int_{1/\sqrt{k}}^{\sqrt{k}}\left[2a+bf\left(r+\frac{1}{r}\right)\right]dr \;=\; 2a+bf\left(\frac{\sqrt{k}+1/\sqrt{k}}{2}+\frac{\ln k}{\sqrt{k}-1/\sqrt{k}}\right) \qquad \text{(EQ 5)}$$

Thus, the average case is better than worst case, but doesn't see as large an improvement as it did during truncation because the worst case is already better.

FIGURE 8. Delay penalty vs. k (average case)

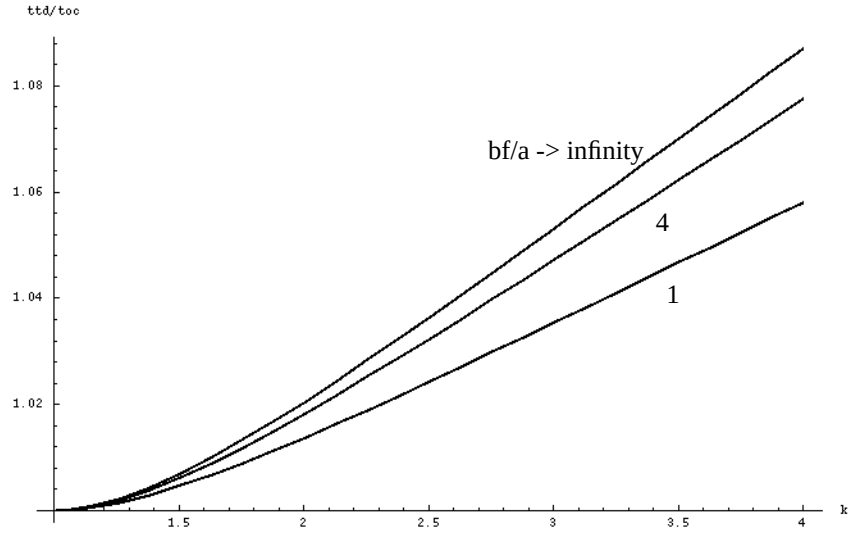


FIGURE 9. Zoom in on Figure 8 1 < k < 2

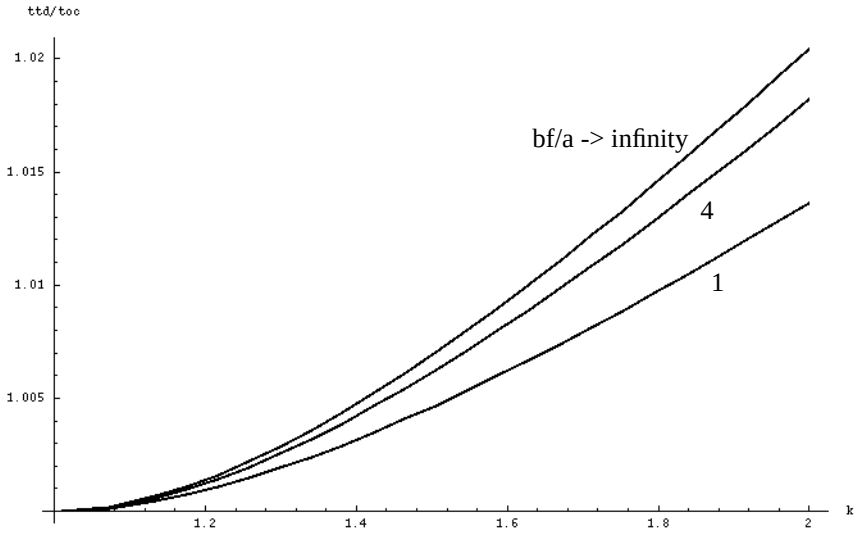
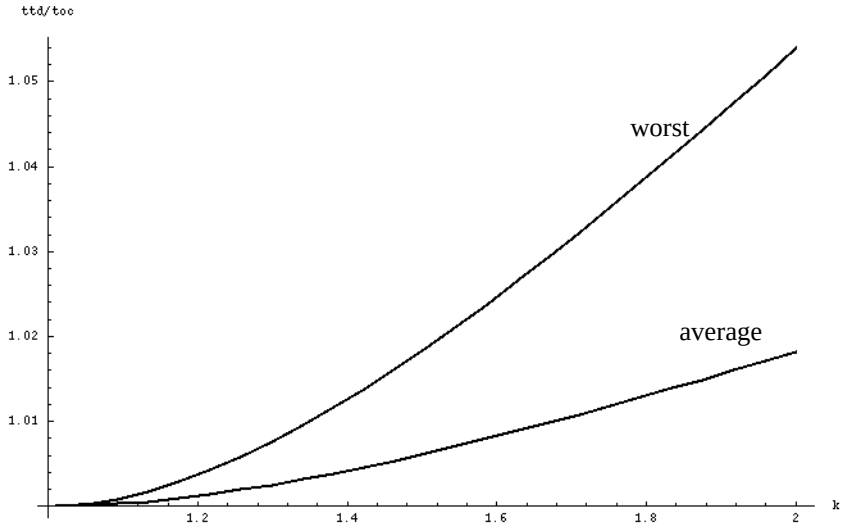


FIGURE 10. Comparison of average and worst case quantization penalty (bf/a = 4)



2.2 Monte-Carlo Analysis

The general analysis of quantization penalty gets nasty, so we turn to Monte-Carlo simulation to approximate the worst and average case behavior of a number of problems.

Consider a path of N inverters in which the first inverter is sized s_0 and the load is $s_0 f^N$. The optimal gain (fanout) per stage is f and the optimal delay is N(a+bf).

How do the average and worst case delays of a discrete sized path compare? To be conservative, we let bf/a approach infinity. As we saw earlier, changing bf/a to 4 (typical for a real path) slightly reduces the quantization penalty.

In addition, let the logical effort vary in the range of [1, 2) for each gate and allow some assorted fixed capacitance loads after each gate. We created 4000 paths (varying f, size of the first gate, fixed capacitance side loads, and logical efforts) for eight values of k from 1.1 (a very fine-grained library) to 4 (a very coarse-grained library). The number of gates to size in the path (N-1) was varied from 1 to 10. The tables below summarizes the average and worst case ratio of actual delay to optimal delay for each value of k and N-1. They are sorted by N & k, by worst average delay, and by worst maximal delay. Both rounding to the nearest legal gate size and truncating to the next smaller gate are considered.

Observations:

- Rounding is very important and produces much better results than truncating. An integer programming algorithm which finds the best gate will be even better (by an unknown amount) than rounding.
- For k =2, the worst case penalty was always less than 14% and the average case penalty is less than 4%! Thus, a finer grain library has very little performance benefit.
- The truncated case is always worst for N=2 (N-1 = 1) because the largest truncation error occurs when there is only one stage. The rounded case worst error peaks with at a larger number of stages because worst case misrounding takes an elaborate combination of stages. This matches my experience that computing an analytic expression for rounded case worst behavior is difficult
- The average case performance was not very sensitive to the range of f which we considered. Therefore, the data is fairly robust over design styles

TABLE 1. Penalty sorted by rounded average case behavior

k	N-1	Round Avg	Round Worst	Trunc Avg	Trunc Worst
1.10	1	1.0004	1.0011	1.0013	1.0045
1.10	2	1.0004	1.0021	1.0011	1.0029
1.10	6	1.0005	1.0021	1.0010	1.0035
1.10	4	1.0005	1.0018	1.0011	1.0032
1.10	5	1.0005	1.0024	1.0012	1.0042
1.10	3	1.0005	1.0024	1.0011	1.0038
1.10	7	1.0005	1.0019	1.0010	1.0032
1.10	8	1.0006	1.0020	1.0009	1.0028
1.10	10	1.0006	1.0019	1.0010	1.0024
1.10	9	1.0006	1.0017	1.0010	1.0028
1.20	1	1.0014	1.0041	1.0056	1.0165
1.20	2	1.0019	1.0070	1.0043	1.0104
1.20	5	1.0020	1.0070	1.0043	1.0121
1.20	3	1.0020	1.0077	1.0044	1.0150
1.20	4	1.0022	1.0068	1.0040	1.0118
1.20	6	1.0022	1.0065	1.0038	1.0115
1.20	7	1.0023	1.0079	1.0038	1.0100
1.20	9	1.0023	1.0073	1.0037	1.0090
1.20	8	1.0024	1.0078	1.0036	1.0093
1.20	10	1.0024	1.0076	1.0036	1.0081
1.41	1	1.0049	1.0150	1.0182	1.0605
1.41	2	1.0065	1.0284	1.0170	1.0398
1.41	3	1.0069	1.0292	1.0160	1.0536

TABLE 1. Penalty sorted by rounded average case behavior

k	N-1	Round Avg	Round Worst	Trunc Avg	Trunc Worst
1.41	5	1.0075	1.0238	1.0153	1.0447
1.41	4	1.0078	1.0314	1.0151	1.0412
1.50	2	1.0079	1.0385	1.0230	1.0478
1.41	7	1.0082	1.0245	1.0134	1.0397
1.50	1	1.0083	1.0201	1.0298	1.0775
1.41	9	1.0084	1.0261	1.0135	1.0344
1.41	6	1.0084	1.0274	1.0138	1.0351
1.41	10	1.0088	1.0272	1.0128	1.0324
1.41	8	1.0089	1.0272	1.0125	1.0331
1.50	4	1.0102	1.0328	1.0186	1.0447
1.50	5	1.0106	1.0413	1.0223	1.0496
1.50	10	1.0108	1.0305	1.0201	1.0428
1.50	9	1.0108	1.0263	1.0186	1.0418
1.50	6	1.0109	1.0306	1.0191	1.0477
1.50	8	1.0112	1.0332	1.0167	1.0397
1.50	7	1.0114	1.0314	1.0229	1.0502
1.50	3	1.0123	1.0387	1.0189	1.0438
2.00	1	1.0182	1.0594	1.1043	1.2470
2.00	2	1.0252	1.1233	1.0759	1.1581
2.00	4	1.0289	1.1126	1.0572	1.1771
2.00	3	1.0308	1.0976	1.0608	1.1960
2.00	6	1.0325	1.1245	1.0538	1.1901
2.00	5	1.0326	1.1043	1.0620	1.1787
2.00	7	1.0329	1.1000	1.0565	1.1591
2.50	1	1.0331	1.1066	1.1287	1.4465
2.00	9	1.0332	1.0897	1.0596	1.1601
2.00	8	1.0347	1.0970	1.0547	1.1500
2.00	10	1.0362	1.1366	1.0556	1.1193
2.50	3	1.0468	1.2282	1.1132	1.3847
2.50	2	1.0516	1.2421	1.1220	1.2923
2.50	4	1.0530	1.2002	1.1156	1.3254
2.50	9	1.0539	1.1849	1.0935	1.2710
2.50	7	1.0563	1.1614	1.1004	1.2595
2.50	5	1.0574	1.1991	1.0960	1.3327
2.50	10	1.0581	1.2068	1.0892	1.2215
2.50	8	1.0585	1.2268	1.0876	1.2536
4.00	1	1.0614	1.2391	1.3194	2.1180
3.00	2	1.0615	1.3406	1.1713	1.4154
2.50	6	1.0617	1.1829	1.0935	1.2828
3.00	1	1.0634	1.1532	1.1715	1.6606
3.00	9	1.0688	1.2969	1.1504	1.3433
3.00	3	1.0728	1.3389	1.1401	1.4920
3.00	4	1.0742	1.3400	1.1608	1.3902
3.00	10	1.0828	1.2655	1.1375	1.3276
3.00	5	1.0856	1.2507	1.1258	1.4232
3.00	7	1.0859	1.2810	1.1525	1.3497
3.00	8	1.0860	1.3042	1.1287	1.3879
3.00	6	1.0891	1.2909	1.1349	1.3383
4.00	5	1.0897	1.4501	1.2836	1.8838
4.00	2	1.1023	1.5106	1.2428	1.7312
4.00	9	1.1293	1.4145	1.2067	1.5109

TABLE 1. Penalty sorted by rounded average case behavior

k	N-1	Round Avg	Round Worst	Trunc Avg	Trunc Worst
4.00	4	1.1349	1.5006	1.2237	1.7586
4.00	6	1.1355	1.5395	1.2660	1.5884
4.00	10	1.1366	1.5040	1.1962	1.4966
4.00	8	1.1381	1.4642	1.1853	1.6171
4.00	7	1.1435	1.4920	1.2323	1.6597
4.00	3	1.1526	1.5156	1.2688	1.6868

TABLE 2. Penalty sorted by rounded worst case behavior

k	N-1	Round Avg	Round Worst	Trunc Avg	Trunc Worst
1.10	1	1.0004	1.0011	1.0013	1.0045
1.10	9	1.0006	1.0017	1.0010	1.0028
1.10	4	1.0005	1.0018	1.0011	1.0032
1.10	10	1.0006	1.0019	1.0010	1.0024
1.10	7	1.0005	1.0019	1.0010	1.0032
1.10	8	1.0006	1.0020	1.0009	1.0028
1.10	6	1.0005	1.0021	1.0010	1.0035
1.10	2	1.0004	1.0021	1.0011	1.0029
1.10	3	1.0005	1.0024	1.0011	1.0038
1.10	5	1.0005	1.0024	1.0012	1.0042
1.20	1	1.0014	1.0041	1.0056	1.0165
1.20	6	1.0022	1.0065	1.0038	1.0115
1.20	4	1.0022	1.0068	1.0040	1.0118
1.20	2	1.0019	1.0070	1.0043	1.0104
1.20	5	1.0020	1.0070	1.0043	1.0121
1.20	9	1.0023	1.0073	1.0037	1.0090
1.20	10	1.0024	1.0076	1.0036	1.0081
1.20	3	1.0020	1.0077	1.0044	1.0150
1.20	8	1.0024	1.0078	1.0036	1.0093
1.20	7	1.0023	1.0079	1.0038	1.0100
1.41	1	1.0049	1.0150	1.0182	1.0605
1.50	1	1.0083	1.0201	1.0298	1.0775
1.41	5	1.0075	1.0238	1.0153	1.0447
1.41	7	1.0082	1.0245	1.0134	1.0397
1.41	9	1.0084	1.0261	1.0135	1.0344
1.50	9	1.0108	1.0263	1.0186	1.0418
1.41	10	1.0088	1.0272	1.0128	1.0324
1.41	8	1.0089	1.0272	1.0125	1.0331
1.41	6	1.0084	1.0274	1.0138	1.0351
1.41	2	1.0065	1.0284	1.0170	1.0398
1.41	3	1.0069	1.0292	1.0160	1.0536
1.50	10	1.0108	1.0305	1.0201	1.0428
1.50	6	1.0109	1.0306	1.0191	1.0477
1.41	4	1.0078	1.0314	1.0151	1.0412
1.50	7	1.0114	1.0314	1.0229	1.0502
1.50	4	1.0102	1.0328	1.0186	1.0447
1.50	8	1.0112	1.0332	1.0167	1.0397
1.50	2	1.0079	1.0385	1.0230	1.0478
1.50	3	1.0123	1.0387	1.0189	1.0438
1.50	5	1.0106	1.0413	1.0223	1.0496

TABLE 2. Penalty sorted by rounded worst case behavior

k	N-1	Round Avg	Round Worst	Trunc Avg	Trunc Worst
2.00	1	1.0182	1.0594	1.1043	1.2470
2.00	9	1.0332	1.0897	1.0596	1.1601
2.00	8	1.0347	1.0970	1.0547	1.1500
2.00	3	1.0308	1.0976	1.0608	1.1960
2.00	7	1.0329	1.1000	1.0565	1.1591
2.00	5	1.0326	1.1043	1.0620	1.1787
2.50	1	1.0331	1.1066	1.1287	1.4465
2.00	4	1.0289	1.1126	1.0572	1.1771
2.00	2	1.0252	1.1233	1.0759	1.1581
2.00	6	1.0325	1.1245	1.0538	1.1901
2.00	10	1.0362	1.1366	1.0556	1.1193
3.00	1	1.0634	1.1532	1.1715	1.6606
2.50	7	1.0563	1.1614	1.1004	1.2595
2.50	6	1.0617	1.1829	1.0935	1.2828
2.50	9	1.0539	1.1849	1.0935	1.2710
2.50	5	1.0574	1.1991	1.0960	1.3327
2.50	4	1.0530	1.2002	1.1156	1.3254
2.50	10	1.0581	1.2068	1.0892	1.2215
2.50	8	1.0585	1.2268	1.0876	1.2536
2.50	3	1.0468	1.2282	1.1132	1.3847
4.00	1	1.0614	1.2391	1.3194	2.1180
2.50	2	1.0516	1.2421	1.1220	1.2923
3.00	5	1.0856	1.2507	1.1258	1.4232
3.00	10	1.0828	1.2655	1.1375	1.3276
3.00	7	1.0859	1.2810	1.1525	1.3497
3.00	6	1.0891	1.2909	1.1349	1.3383
3.00	9	1.0688	1.2969	1.1504	1.3433
3.00	8	1.0860	1.3042	1.1287	1.3879
3.00	3	1.0728	1.3389	1.1401	1.4920
3.00	4	1.0742	1.3400	1.1608	1.3902
3.00	2	1.0615	1.3406	1.1713	1.4154
4.00	9	1.1293	1.4145	1.2067	1.5109
4.00	5	1.0897	1.4501	1.2836	1.8838
4.00	8	1.1381	1.4642	1.1853	1.6171
4.00	7	1.1435	1.4920	1.2323	1.6597
4.00	4	1.1349	1.5006	1.2237	1.7586
4.00	10	1.1366	1.5040	1.1962	1.4966
4.00	2	1.1023	1.5106	1.2428	1.7312
4.00	3	1.1526	1.5156	1.2688	1.6868
4.00	6	1.1355	1.5395	1.2660	1.5884

The following tables show how the average and worst case delays are impacted by varying logical effort, size of the first gate, and fixed capacitance side loads. They combine all the values of M. The average case behavior doesn’t change much between a simple inverter chain and a chain of random gates with different first sizes and different fixed loads. The worst case behavior shows somewhat more variation. This is partly due to the fact that

more random paths were simulated without fixed side loads, so a worse worst case was found.

TABLE 3. Average case penalty

	k=1.414	k=2	k=4
Inverters Only	1.008	1.032	1.145
Vary size of first gate	1.007	1.030	1.130
Vary size & LE	1.007	1.030	1.117
Inverters with fixed loads	1.010	1.044	1.191
Vary size, LE, fixed load	1.009	1.036	1.152

TABLE 4. Worst case penalty

	k=1.414	k=2	k=4
Inverters Only	1.018	1.077	1.345
Vary size of first gate	1.032	1.130	1.660
Vary size & LE	1.035	1.148	1.639
Inverters with fixed loads	1.025	1.099	1.355
Vary size, LE, fixed load	1.031	1.137	1.540

3.0 Conclusion

We studied the ratio of delay through a path using a limited set of cell library elements to the delay of a path using unrestricted gate sizes. We find analytic expressions for simple cases which match well with Monte Carlo simulations of more general cases. We find that for cell libraries with a scale factor of 2 (i.e. with gates of size 1, 2, 4, 8, ...) paths are never more than 15% slower than optimal and on average are within about 4% of optimal. Therefore, there is very little speed benefit of creating a finer-grained cell library. When the scale factor is increased to 4 (gates of size 1, 4, 16, 64, ...), the worst case penalty is about 66% and the average case penalty is about 15%. Hence, such crude libraries are not appropriate for high performance designs.