Charles Hathaway
2013-10-14
CSCI-6140 – Computer Operating Systems

Homework 2

1)

Previously submitted.

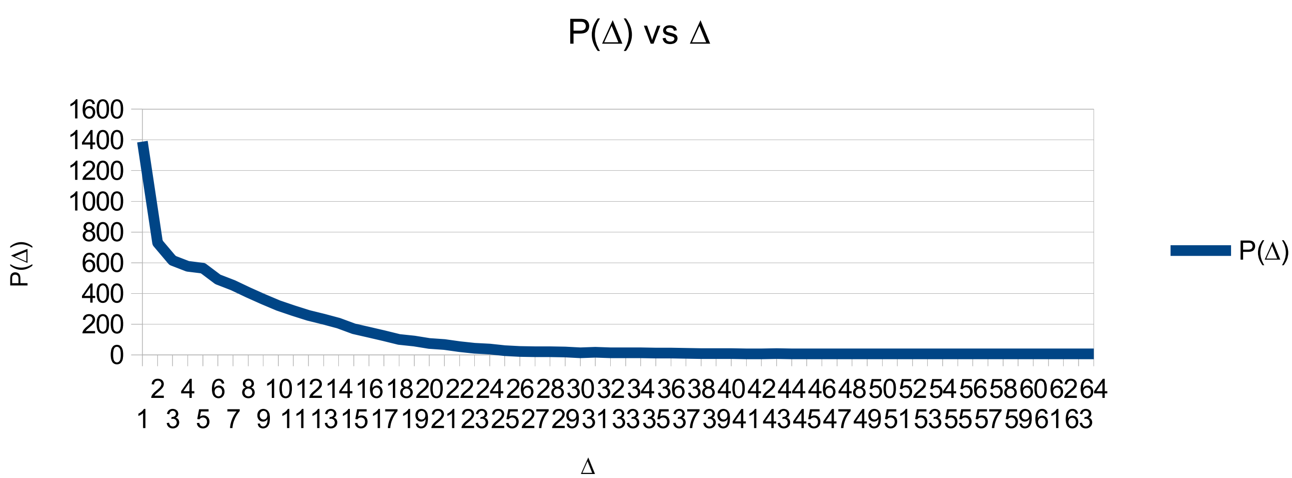
2)

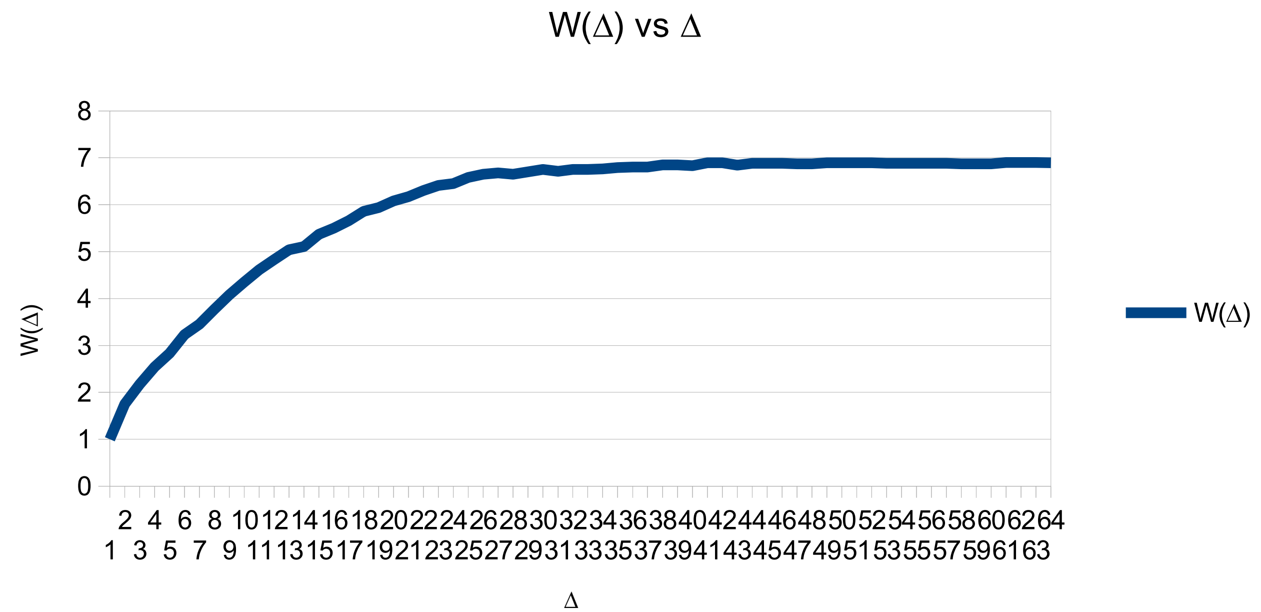
The program is included in the tarball of this file. The output is as follows:

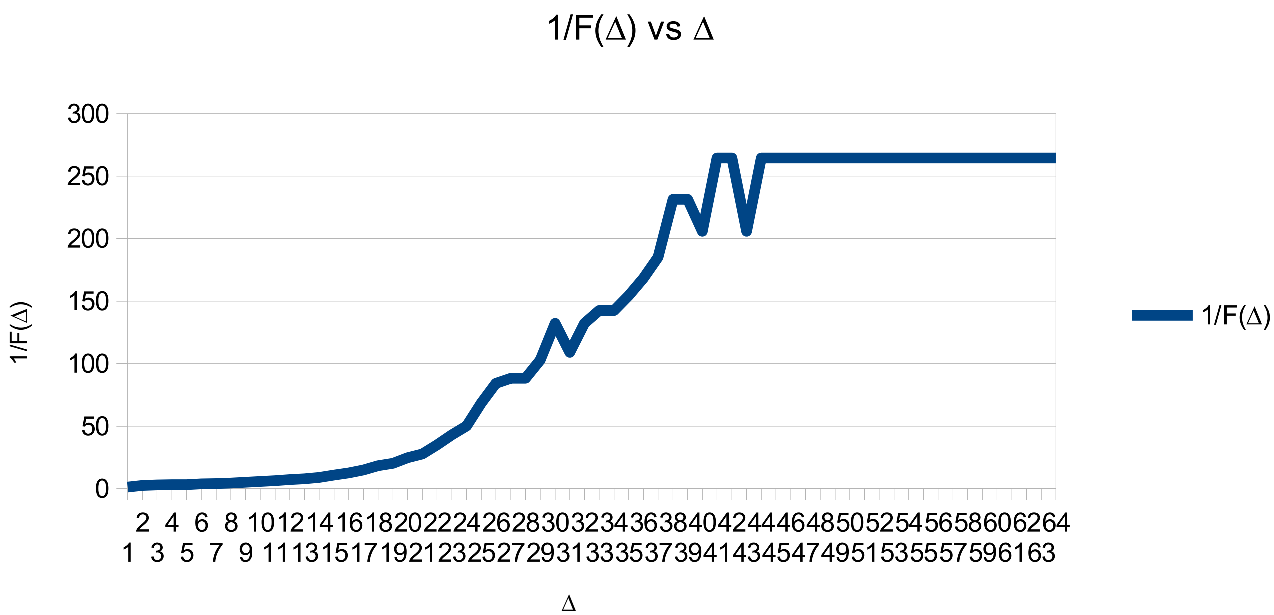
Δ	P(Δ)	W(Δ)	1/F(Δ)
1	1389	1	1.33
2	7301.75		2.54
3	6152.17		3.01
4	5762.54		3.22
5	5642.83		3.28
6	4923.23		3.76
7	4543.45		4.08
8	4063.77		4.56
9	3634.08		5.1
10	3214.35		5.77
11	2894.61		6.41
12	2574.83		7.21
13	2335.04		7.95
14	2075.11		8.95
15	1705.37		10.89
16	1475.5		12.6
17	1255.66		14.82
18	1015.86		18.34
19	91	5.94	20.35
20	75	6.08	24.69
21	67	6.17	27.64
22	53	6.3	34.94
23	43	6.41	43.07
24	37	6.45	50.05
25	27	6.58	68.59
26	22	6.65	84.18
27	21	6.68	88.19
28	21	6.65	88.19
29	18	6.7	102.89
30	14	6.75	132.29
31	17	6.71	108.94
32	14	6.75	132.29
33	13	6.75	142.46
34	13	6.76	142.46

35	12	6.79	154.33
36	11	6.8	168.36
37	10	6.8	185.2
38	8	6.85	231.5
39	8	6.85	231.5
40	9	6.83	205.78
41	7	6.89	264.57
42	7	6.89	264.57
43	9	6.84	205.78
44	7	6.88	264.57
45	7	6.88	264.57
46	7	6.88	264.57
47	7	6.87	264.57
48	7	6.87	264.57
49	7	6.89	264.57
50	7	6.89	264.57
51	7	6.89	264.57
52	7	6.89	264.57
53	7	6.88	264.57
54	7	6.88	264.57
55	7	6.88	264.57
56	7	6.88	264.57
57	7	6.88	264.57
58	7	6.87	264.57
59	7	6.87	264.57
60	7	6.87	264.57
61	7	6.9	264.57
62	7	6.9	264.57
63	7	6.9	264.57
64	7	6.89	264.57

And here are the requested charts:







3)

The last graph I believe to be somewhat erroneous. There are “spikes” that I can't make sense of. It appears that the number of page faults seem to spike for some unknown reason (perhaps luck?), thusly causing the F(Δ) to plummet (and 1/F(Δ) to spike). However, we can still see at least two very clear knees, at ~26 and ~34. If we assume the drops to be anomalies, we can find four knees at ~26, ~30, ~38, ~42.

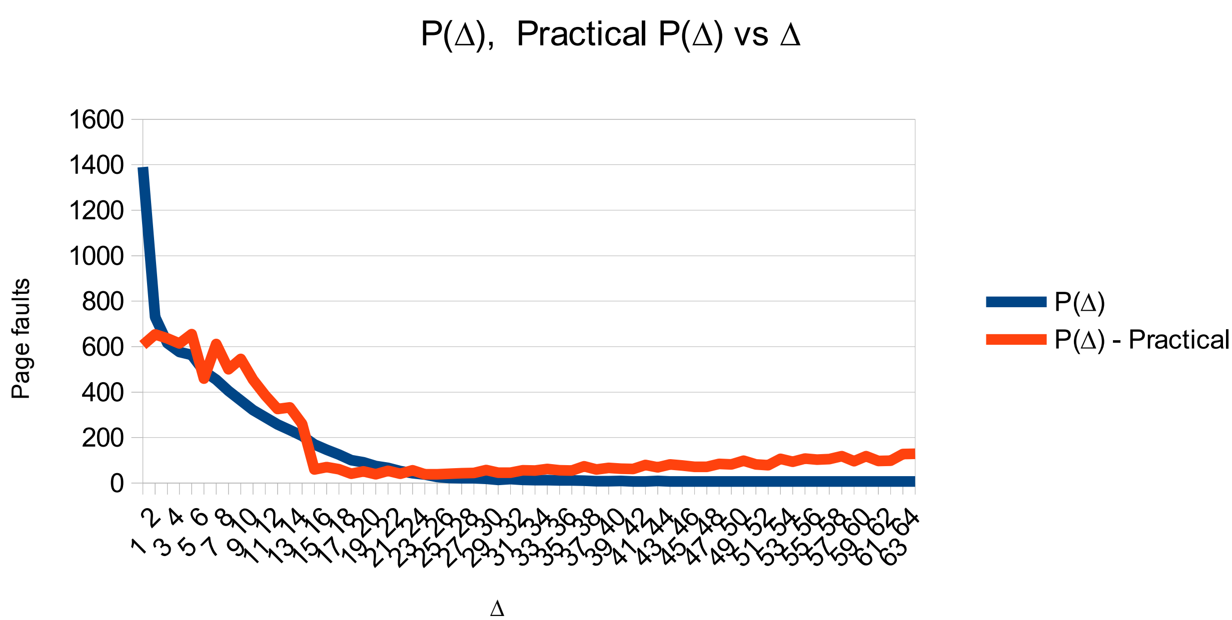
These knees occur because the size of Δ increases past a point that causes a significant change in the way the code behaves. One of the most frequently executed pieces of code in this example is the embedded for loop. It happens to contain 26 instructions, which lines up perfectly with the knee. The knee near 42 could be caused by the entire second loop (including the embedded loop) being able to run without losing any pages. 30 would be enough for us to retain the initialization data and execute the first loop (once), and 38 would allow us to keep the initialization data from the second loop.

4)

$$F(\Delta)=W(\Delta+1)-W(\Delta)$$
$$\frac{(P(\Delta))}{|\omega|}=W(\Delta+1)-W(\Delta)$$

If ω is really big, the value of the first half of the equation will be really small. For most cases, increasing the Δ by one will not have a big impact on how the large ω gets executed (that is, jumping from a page from 1 to 2 will not increase the number of page faults by a lot as long as the pages don't simply go "01010101010101010101010101010101010101". In this case, the impact would be very extreme and by very obvious). As there is not a big difference between W(Δ+1) and W(Δ), there mathematical difference would be very small. Hence, there will not be a huge difference.

5)

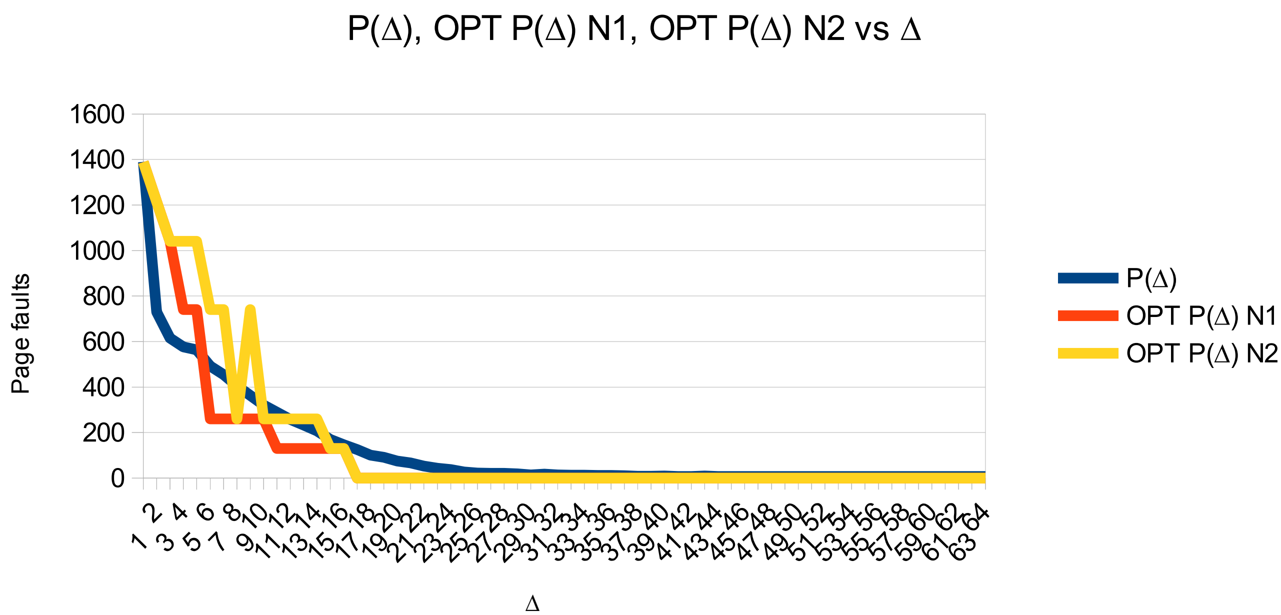


P(Δ)
  Average 128.32
  Max 1389

Practical P(Δ)
  Average 166.48
  Max 655

The average size of the practical implementation is much higher than the clean one, however, the max is much lower. This suggests to me that there be certain circumstances where one is preferred over the other.

Bonus:



Aside from the one anomaly, it is pretty straightforward to observe that the dirty page replacement lags behind the clean one (in terms of max working set sizes) by about 2 deltas. However, they both reach the point where there are no page faults at the same time. It only takes 7 (the ceiling of the N's) to get a situation where there are no page faults, so one could use this method to figure out how many frames would be ideal in the ideal situation.