ECE 4100
Project 2 Report
By Sho Ko

To start with, we try to find the best (g, c) pair within our budget. The total budget is 64KB, which is $64 \times 1024 \times 8 = 2^{19} = 524288$ bits. We can calculate the total bits in both BTB and cache using the following equation: Total Bits $= 2^g \times (64 + 2) + g + (64 - c + 1 + 1) \times 2^{(c - 6)} + 2^c \times 8 = 2^g \times 66 + g + (66 - c) \times 2^{(c - 6)} + 2^{(c+3)}$

We then find out the absolute maximal value for g and c.
$2^{12} \times 66 + 12 = 270348$ bits < 524288 bits
$2^{13} \times 66 + 12 = 540684$ bits > 524288 bits
So the maximal value for g is 12.

$(66 - 15) \times 2^{(15 - 6)} + 2^{(15 + 3)} = 288256$ bits < 524288 bits
$(66 - 16) \times 2^{(16 - 6)} + 2^{(16 + 3)} = 575488$ bits > 524288 bits
So the maximal value for c is 15.

When g is at maximal 12, c can be as large as 14.
$2^{12} \times 66 + 12 + (66 - 14) \times 2^{(14 - 6)} + 2^{(14 + 3)} = 414732$ bits < 524288 bits

When c is at maximal 15, g can be as large as 11.
$2^{11} \times 66 + 11 + (66 - 15) \times 2^{(15 - 6)} + 2^{(15 + 3)} = 423435$ bits < 524288 bits

Then we exhaustively run every combination of (f, k, l, m, r) to find the best configuration for all four traces.
bwaves603_2M:
f = 4, k = 3, l = 3, m = 3, r = 4, g = 11, c = 15, IPC = 2.774749

gcc602_2M:
f = 2, k = 3, l = 1, m = 3, r = 4, g = 11, c = 15, IPC = 1.715862

mcf605_2M:
f = 4, k = 3, l = 3, m = 3, r = 4, g = 12, c = 14, IPC = 1.645092

perlbench600_2M:
f = 3, k = 3, l = 3, m = 3, r = 4, g = 11, c = 15, IPC = 1.748915

Then for each trace, we find out the minimal value for k, l, m, and r which achieves more than 95% of the best performance.

The number of functional units is $k + l + m$ and the number of reservation stations is $r \times (k + l + m)$. More reservation stations will cost the delay to be longer because searching through the scheduling queue will takes more time. More functional units will require more hardware and cost more power and area.

Assume the relative importance of RS and FU is 1:1, we can use the equation: Cost $= (k + l + m) + r \times (k + l + m) = (1 + r) \times (k + l + m)$ to choose the best (k, l, m, r) configuration.

After choosing the best (k, l, m, r), we will try to minimize (g, c) to minimize total bits.

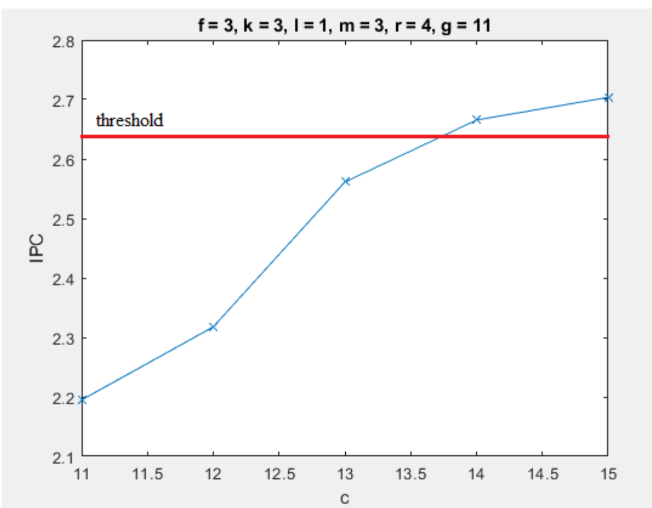
bwaves603_2M.trace

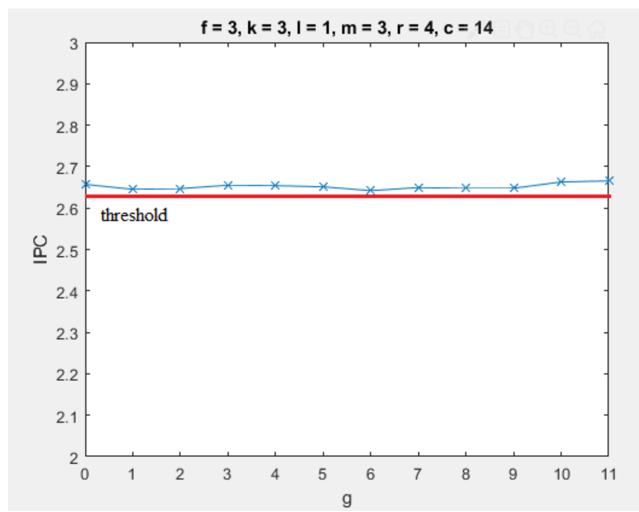
Threshold IPC: 2.774749 x 0.95 = 2.63601155

The configurations that exceed the threshold IPC and are close to minimal cost:

F	K	L	M	R	G	C	IPC	#RS	#FU	Cost
3	3	1	3	4	11	15	2.703624	28	7	35
3	3	1	3	4	12	14	2.671899	28	7	35
3	3	3	3	3	11	15	2.690579	27	9	36
3	3	3	3	3	12	14	2.646483	27	9	36
4	3	1	3	4	11	15	2.699161	28	7	35
4	3	1	3	4	12	14	2.649898	28	7	35
4	3	3	3	3	11	15	2.689691	27	9	36
4	3	3	3	3	12	14	2.648340	27	9	36

We choose (k, l, m, r) to be (3, 1, 3, 4) to minimize both RS and FU. Then we need to figure out f and minimal g and c (smaller fetch rate can result in less hardware resources). As shown in the left graph, c has a great impact on IPC, and we can only reduce c to 14. Then we find out that g doesn't affect IPC much, so we can reduce g to as low as 0, as shown in the right graph.





Then we try different fetch rate:

F	K	L	M	R	G	C	IPC	Total Bits
2	3	1	3	4	0	14	1.996199	144450
3	3	1	3	4	0	14	2.657104	144450
4	3	1	3	4	0	14	2.658856	144450

As a result, we choose f = 4, k = 3, l = 1, m = 3, r = 4, g = 0, c = 14, IPC = 2.658856, total bits = 144450, cost = 35.

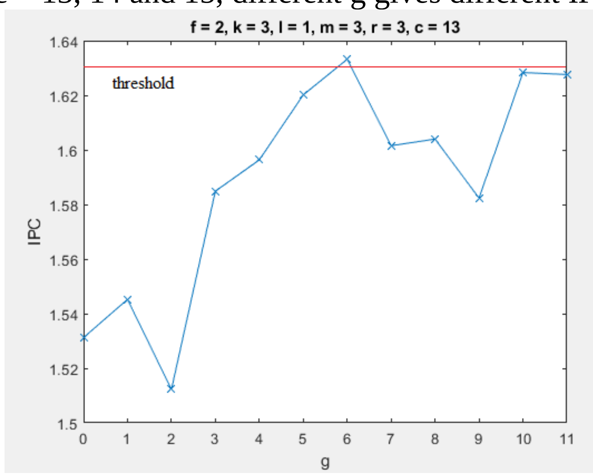
gcc602_2M.trace

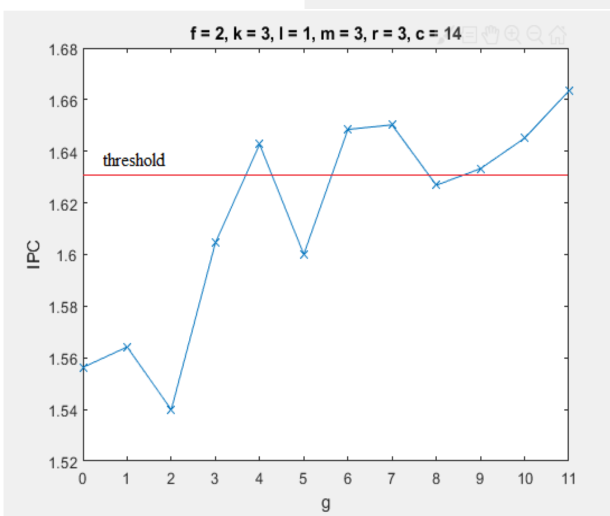
Threshold IPC: 1.715862 x 0.95 = 1.6300689

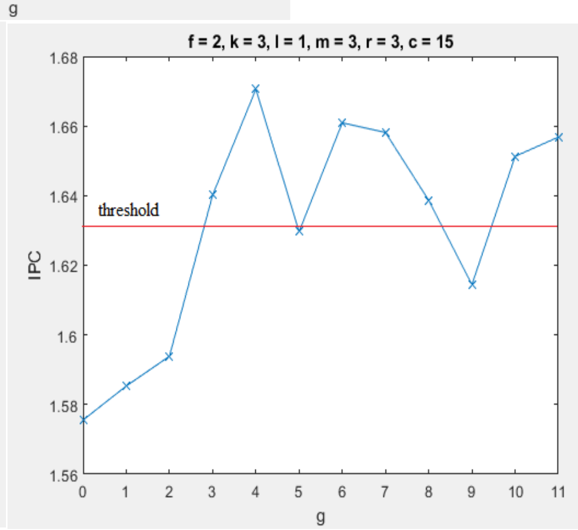
The configurations that exceed the threshold and are close to minimal cost:

F	K	L	M	R	G	C	IPC	#RS	#FU	Cost
2	3	1	3	3	11	15	1.656828	21	7	28
2	3	1	3	3	12	14	1.662477	21	7	28
2	3	1	3	4	11	15	1.715862	28	7	35
2	3	1	3	4	12	14	1.688748	28	7	35

We choose (k, l, m, r) to be (3, 1, 3, 3) to minimize both RS and FU. Then we need to figure out f and minimal g and c (smaller fetch rate can result in less hardware resources). For this trace, both g and c have a great impact on IPC. The graphs show when c = 13, 14 and 15, different g gives different IPC.







To summarize, the possible configurations are:

F	K	L	M	R	G	C	IPC	Total Bits
2	3	1	3	3	6	13	1.633196	76550
2	3	1	3	3	4	14	1.642689	145444
2	3	1	3	3	3	15	1.640201	288787

As a result, we choose f = 2, k = 3, l = 1, m = 3, r = 3, g = 6, c = 13, IPC = 1.633196, total bits = 76550, cost = 28.

mcf605_2M.trace

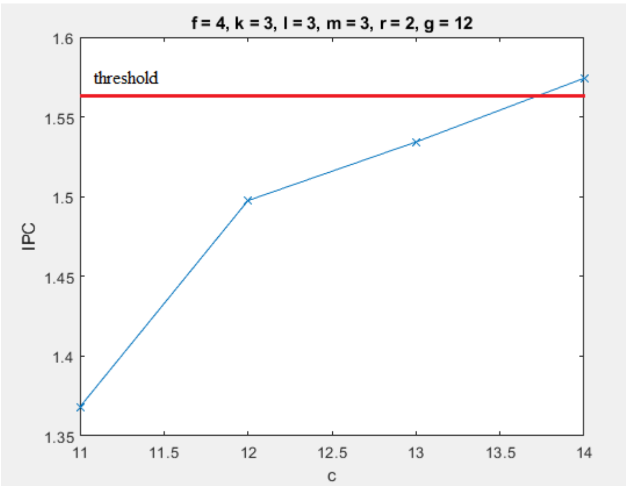
Threshold IPC: 1.645092 x 0.95 = 1.5628374

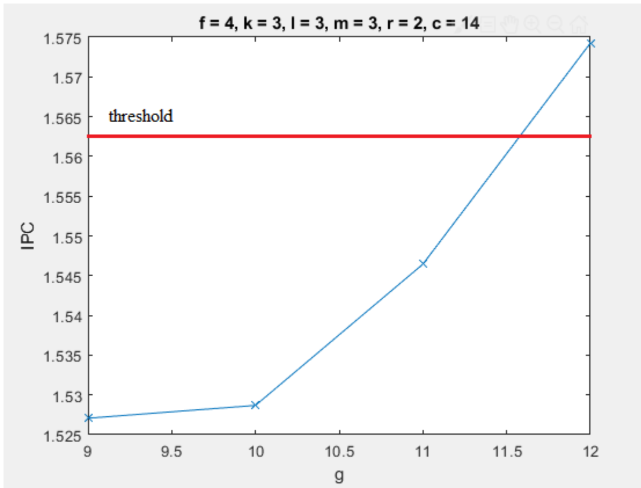
The configurations that exceed the threshold and are close to minimal cost:

F	K	L	M	R	G	C	IPC	#RS	#FU	Cost
4	3	3	3	2	12	14	1.574245	18	9	27
2	3	1	3	3	11	15	1.568807	21	7	28
2	3	1	3	3	12	14	1.576403	21	7	28
3	3	1	3	3	11	15	1.576993	21	7	28
3	3	1	3	3	12	14	1.587317	21	7	28
4	3	1	3	3	11	15	1.579122	21	7	28
4	3	1	3	3	12	14	1.597269	21	7	28

We choose (k, l, m, r) to be (3, 3, 3, 2) to minimize both RS and FU. Then we need to figure out f and minimal g and c (smaller fetch rate can result in less hardware resources).

We first try to reduce c without changing g, but even reducing c by 1 will make IPC lower than the threshold IPC, as the left graph shows. Similarly, reducing g by 1 will also make IPC lower than the threshold IPC, as the right graph shows, so we need to keep (g, c) at (12, 14).





Then we try different fetch rate:

F	K	L	M	R	G	C	IPC	Total Bits
2	3	3	3	2	12	14	1.555378	414732
3	3	3	3	2	12	14	1.562077	414732
4	3	3	3	2	12	14	1.574245	414732

As a result, we choose f = 4, k = 3, l = 3, m = 3, r = 2, g = 12, c = 14, IPC = 1.574245, total bits = 414732, cost = 27.

perbench600_2M.trace

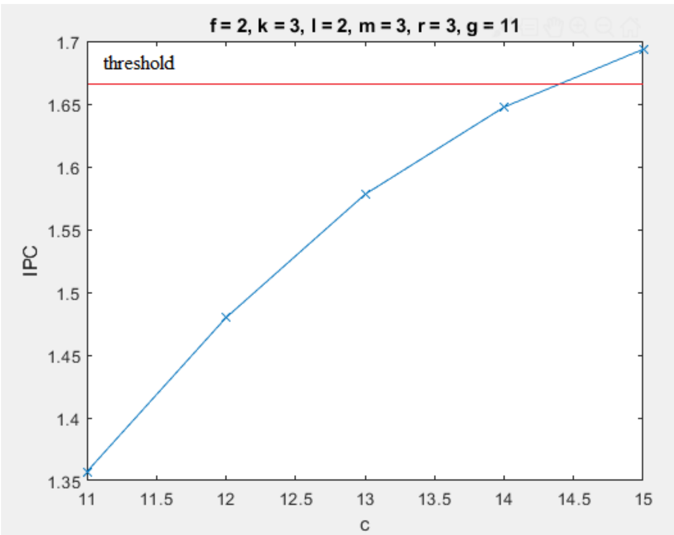
Threshold IPC: 1.748915 x 0.95 = 1.66146925

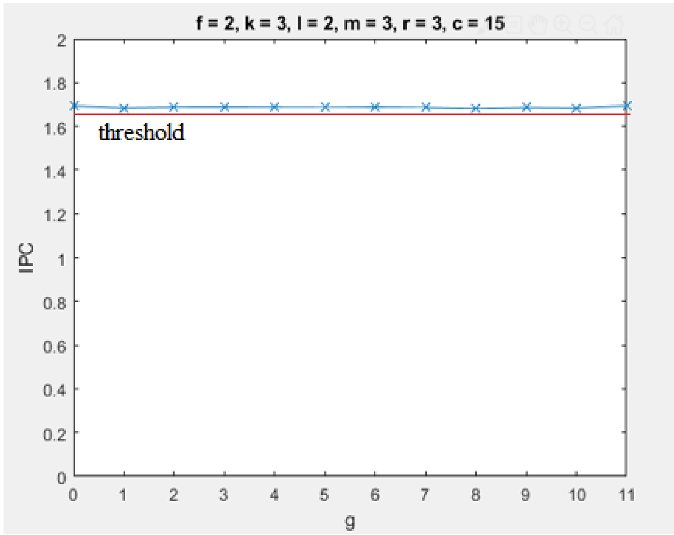
The configurations that exceed the threshold and are close to minimal cost:

F	K	L	M	R	G	C	IPC	#RS	#FU	Cost
2	3	1	3	4	11	15	1.715390	28	7	35
2	3	1	3	4	12	14	1.669332	28	7	35
2	3	2	3	3	11	15	1.693514	24	8	32
3	3	1	3	4	11	15	1.711351	28	7	35
3	3	1	3	4	12	14	1.668312	28	7	35
3	3	2	3	3	11	15	1.684514	24	8	32
4	3	1	3	4	11	15	1.693741	28	7	35
4	3	2	3	3	11	15	1.675290	24	8	32

We choose (k, l, m, r) to be (3, 2, 3, 3) to minimize both RS and FU. Then we need to figure out f and minimal g and c (smaller fetch rate can result in less hardware resources).

We first try to reduce c without changing g, but even reducing c by 1 will make IPC lower than the threshold IPC, as the left graph shows. Therefore, we need to keep c at 15. Then, the size of BTB doesn't affect IPC much, so we can make g to be as low as 0, as shown by the right graph.





Then we try different fetch rate:

F	K	L	M	R	G	C	IPC	Total Bits
2	3	2	3	3	0	15	1.693068	288322
3	3	2	3	3	0	15	1.678447	288322
4	3	2	3	3	0	15	1.669271	288322

As a result, we choose f = 2, k = 3, l = 2, m = 3, r = 3, g = 0, c = 15, IPC = 1.693068, total bits = 288322, cost = 32.