Work: Trying to mine patterns from gem5 still

Progress: only got it to work with some files, ran into issues

Challenge: working with odd indices, slow performance, low acceptance ratio

Message file groups:

('cpu0', 'dcache0', (10, 19, 20, 23, 26, 28, 37, 38, 39, 40, 42, 58, 64))

('cpu0', 'icache0', (0, 9))

('cpu1', 'dcache1', (49, 52, 53, 57, 61, 69, 70, 71, 72, 77, 83))

('cpu1', 'icache1', (44, 48))

('cpu2', 'dcache2', (90, 93, 94, 97, 98, 101, 102, 103, 104, 107, 111))

('cpu2', 'icache2', (86, 89))

('dcache0', 'l2bus', (12, 18, 21, 22, 27, 30, 41, 43, 55, 59, 60, 65, 67, 74, 76))

('dcache1', 'l2bus', (50, 51, 54, 56, 66, 68, 73, 78, 79, 80, 84, 85))

('dcache2', 'l2bus', (91, 92, 95, 96, 99, 100, 105, 108, 109, 110, 112, 113))

('dram', 'membus', (3, 5, 13, 15, 34))

('icache0', 'l2bus', (2, 8, 25, 36, 45))

('icache1', 'l2bus', (46, 47, 81, 82))

('icache2', 'l2bus', (87, 88, 106, 114))

('l2bus', 'l2cache', (1, 7, 11, 17, 24, 29, 31, 33, 63, 75))

('l2cache', 'membus', (4, 6, 14, 16, 32, 35, 62))

Attempt:

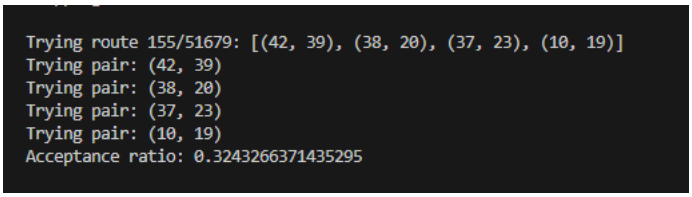
-before generating routes it checks the trace for what indices are used, usually the traces end up using every single one

-Tried making it so it would generate all the pairings, and then there are unused indices, and for each unused index you get all the causal pairs and add that to each route (fail)

Ex: ('dram', 'membus', (3, 5, 13, 15, 34)), get an even amount of pairs, for the unused index try all combinations with that

Problem: doesn't work for ('cpu0', 'dcache0', (10, 19, 20, 23, 26, 28, 37, 38, 39, 40, 42, 58, 64))

Tried just generating routes that didn't use all the indices but could see atleast some patterns, then maybe from the remaining ones more could be mined but



In cases like

('cpu0', 'dcache0', (10, 19, 20, 23, 26, 28, 37, 38, 39, 40, 42, 58, 64))

There were too many routes to try, it took too long.

But it works for:

('dram', 'membus', (3, 5, 13, 15, 34)) for example. So we could look at the remaining numbers and make guesses.

File: unsliced-dram-membus.txt,

Group: dram-membus Indices: (3, 5, 13, 15, 34)

BinaryPatterns: ((13, 15), (3, 5)), Acceptance Ratio: 0.9983530468633028

BinaryPatterns: ((15, 13), (3, 5)), Acceptance Ratio: 0.9980536008384489

BinaryPatterns: ((34, 15), (3, 5)), Acceptance Ratio: 0.8025153466087738

BinaryPatterns: ((15, 34), (3, 5)), Acceptance Ratio: 0.8025153466087738

BinaryPatterns: ((3, 5),), Acceptance Ratio: 0.8007186704596496

Worked for other files, failed for some

File: unsliced-cpu1-icache1.txt,

Group: cpu1-icache1 Indices: (44, 48)

BinaryPatterns: ((44, 48),), Acceptance Ratio: 1.0

File: unsliced-cpu2-icache2.txt,

Group: cpu2-icache2 Indices: (86, 89)

BinaryPatterns: ((86, 89),), Acceptance Ratio: 1.0

File: unsliced-dram-membus.txt,

Group: dram-membus Indices: (3, 5, 13, 15, 34)

BinaryPatterns: ((13, 15), (3, 5)), Acceptance Ratio: 0.9983530468633028

BinaryPatterns: ((15, 13), (3, 5)), Acceptance Ratio: 0.9980536008384489

BinaryPatterns: ((34, 15), (3, 5)), Acceptance Ratio: 0.8025153466087738

BinaryPatterns: ((15, 34), (3, 5)), Acceptance Ratio: 0.8025153466087738

BinaryPatterns: ((3, 5),), Acceptance Ratio: 0.8007186704596496

File: unsliced-icache2-l2bus.txt,

Group: icache2-l2bus Indices: (87, 88, 106, 114)

BinaryPatterns: ((106, 114), (87, 88)), Acceptance Ratio: 0.9958847736625515

BinaryPatterns: ((114, 106), (87, 88)), Acceptance Ratio: 0.9465020576131687

BinaryPatterns: ((87, 88),), Acceptance Ratio: 0.9465020576131687

File: unsliced-l2cache-membus.txt,

Group: l2cache-membus Indices: (4, 6, 14, 16, 32, 35, 62)

BinaryPatterns: ((4, 6), (14, 16)), Acceptance Ratio: 0.9797237731413458

BinaryPatterns: ((4, 6), (16, 14)), Acceptance Ratio: 0.9794299147810756

BinaryPatterns: ((16, 32), (4, 6)), Acceptance Ratio: 0.8172200999118425

BinaryPatterns: ((4, 6), (32, 16)), Acceptance Ratio: 0.8101674992653541

Issues:

Issue: For some reason is taking too long on 0,9 (cpu0-icache0)

These are failing too because they have low acceptance ratios, will examine why

Group: icache0-l2bus Indices: (2, 8, 25, 36, 45)

Group: icache1-l2bus Indices: (46, 47, 81, 82)