

Computer Architecture

Project 2 – Branch Predictor Simulator

Introduction:

The purpose of this project was to create a working gshare branch predictor simulator. The code accepts command line inputs of M, N, and the trace file path. The trace files are setup in the format of address and status, for example: *00A4C6FB t*, and is read into the code for processing. The output then reiterates the M and N inputs and then provides the misprediction ratio as a percentage; for an example output:

M: x N: y Misprediction Ratio: z.zz%

Testing and Results:

To test this code, I ran through provided test cases and ensured my code ran and matched the expected outputs. The code ran successfully and yielded the expected results as we can see below:

```
PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 8 5 mcf_trace.txt
M: 8 N: 5 Misprediction Ratio: 7.48%

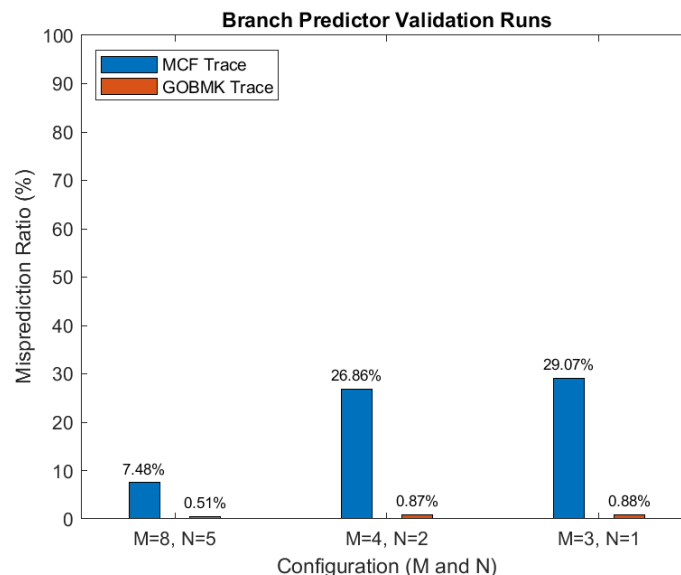
PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 2 mcf_trace.txt
M: 4 N: 2 Misprediction Ratio: 26.86%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 3 1 mcf_trace.txt
M: 3 N: 1 Misprediction Ratio: 29.07%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 8 5 gobmk_trace.txt
M: 8 N: 5 Misprediction Ratio: 0.51%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 2 gobmk_trace.txt
M: 4 N: 2 Misprediction Ratio: 0.87%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 3 1 gobmk_trace.txt
M: 3 N: 1 Misprediction Ratio: 0.88%
```



Following these provided validation cases, I began testing my simulator with varying values of M and N to plot and determine trends of the gshare branch predictor. To start, I fixed M at 4 bits which means there will be 16 table entries (0-15) and then varied N from 1 to 4. The following are my outputs and a plot of the misprediction ratios for both the mcf and gobmk trace files:

```
PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 1 mcf_trace.txt
M: 4    N: 1    Misprediction Ratio: 24.71%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 2 mcf_trace.txt
M: 4    N: 2    Misprediction Ratio: 26.86%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 3 mcf_trace.txt
M: 4    N: 3    Misprediction Ratio: 29.36%

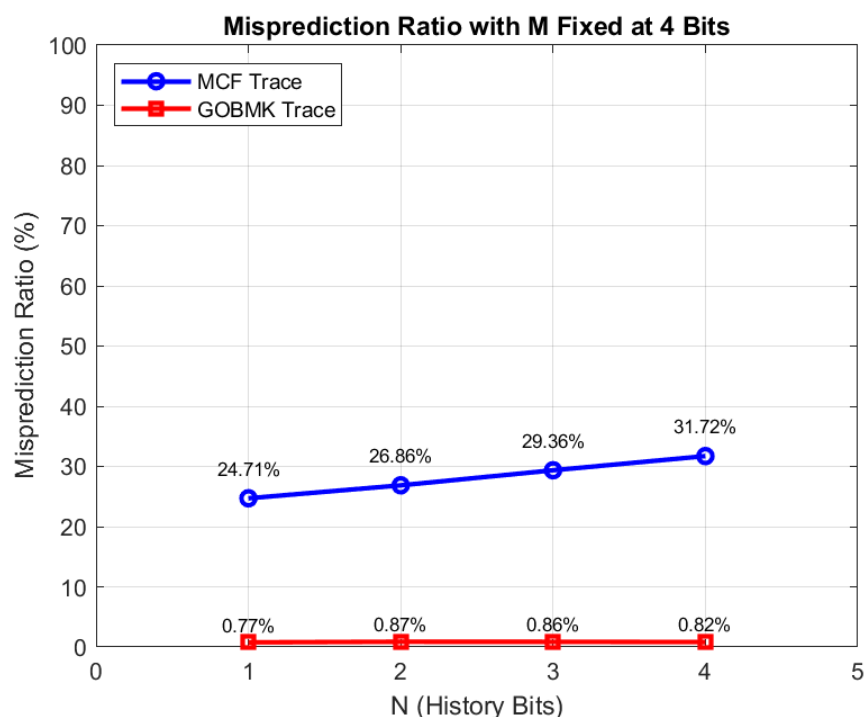
PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 4 mcf_trace.txt
M: 4    N: 4    Misprediction Ratio: 31.72%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 1 gobmk_trace.txt
M: 4    N: 1    Misprediction Ratio: 0.77%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 2 gobmk_trace.txt
M: 4    N: 2    Misprediction Ratio: 0.87%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 3 gobmk_trace.txt
M: 4    N: 3    Misprediction Ratio: 0.86%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 4 gobmk_trace.txt
M: 4    N: 4    Misprediction Ratio: 0.82%
```



Next, I then fixed N at 4 which means there is a history of the last 4 results and varied M from 4 to 7. As you can see below, these are my results and a plot comparing the trace files. As we can see, as the table size increases, the misprediction ratio decreases. The most drastic difference we can observe here is a change of 19.32%!

```
PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 4 .\mcf_trace.txt
M: 4    N: 4    Misprediction Ratio: 31.72%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 5 4 .\mcf_trace.txt
M: 5    N: 4    Misprediction Ratio: 26.56%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 6 4 .\mcf_trace.txt
M: 6    N: 4    Misprediction Ratio: 19.81%

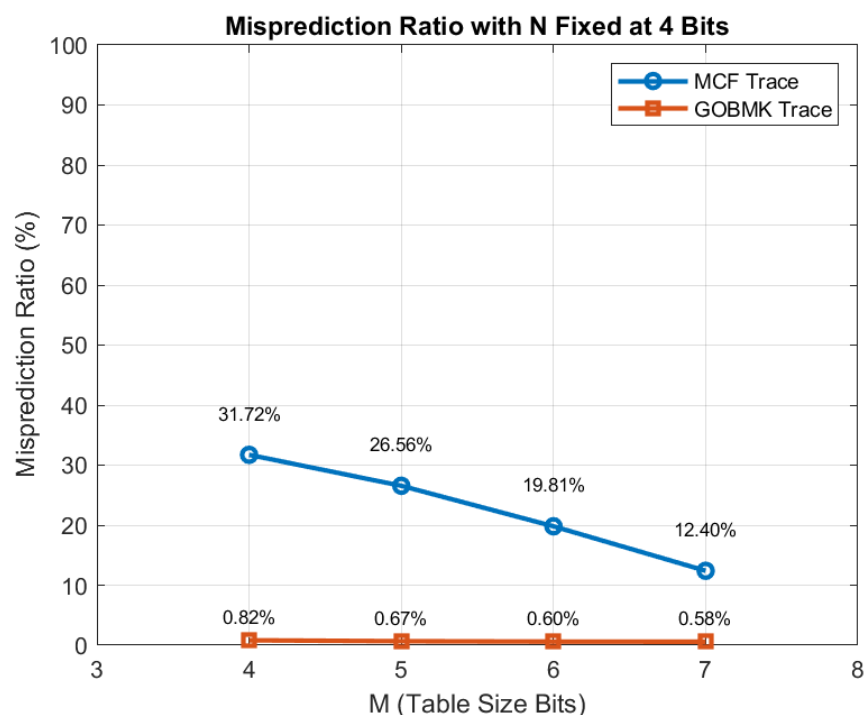
PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 7 4 .\mcf_trace.txt
M: 7    N: 4    Misprediction Ratio: 12.40%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 4 gobmk_trace.txt
M: 4    N: 4    Misprediction Ratio: 0.82%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 5 4 gobmk_trace.txt
M: 5    N: 4    Misprediction Ratio: 0.67%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 6 4 gobmk_trace.txt
M: 6    N: 4    Misprediction Ratio: 0.60%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 7 4 gobmk_trace.txt
M: 7    N: 4    Misprediction Ratio: 0.58%
```



And finally, I tested my code by fixing N to be 0. Again, I then varied M from 4 to 7 and plotted the results. This test was much like the previous test case, and my results are below:

```
PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 0 mcf_trace.txt
M: 4    N: 0    Misprediction Ratio: 23.76%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 5 0 mcf_trace.txt
M: 5    N: 0    Misprediction Ratio: 20.83%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 6 0 mcf_trace.txt
M: 6    N: 0    Misprediction Ratio: 15.07%

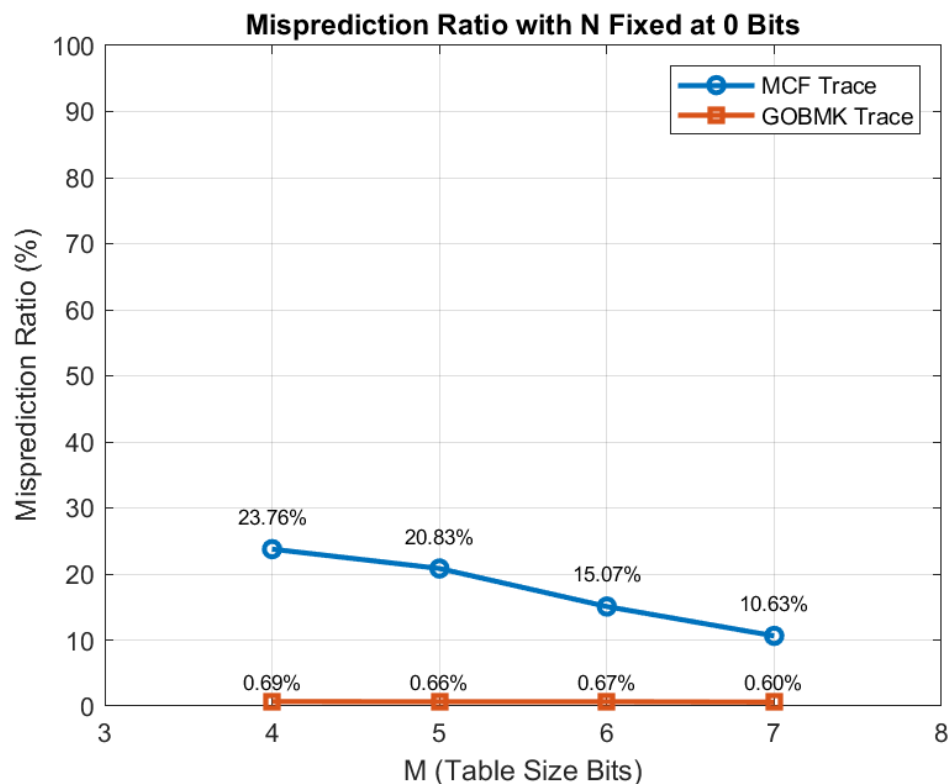
PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 7 0 mcf_trace.txt
M: 7    N: 0    Misprediction Ratio: 10.63%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 4 0 gobmk_trace.txt
M: 4    N: 0    Misprediction Ratio: 0.69%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 5 0 gobmk_trace.txt
M: 5    N: 0    Misprediction Ratio: 0.66%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 6 0 gobmk_trace.txt
M: 6    N: 0    Misprediction Ratio: 0.67%

PS C:\Users\andre\Desktop\VSC-Projects\CompArch\BranchPredictorSimulator> ./sim.exe 7 0 gobmk_trace.txt
M: 7    N: 0    Misprediction Ratio: 0.60%
```



As we can see above, as the table size increases, we see a noticeable difference in the misprediction ratio decreasing, despite the value of N. In this case with N equal to 0, we see a lower misprediction ratio overall when compared to N being fixed at 4.

Overall, with all the results from above, we can observe that the larger the table, the better (i.e., lower) the misprediction ratio is. This stayed true when M was varied larger despite the N value as seen when it was fixed at N equals 4 and N equals 0. In contrast to this, we observed that as N grew larger with a fixed M value, we saw worse (i.e., higher) misprediction ratios. This may be because we were overflowing or disregarding necessary bit information that would help differentiate and predict the branches better.

Deliverables:

With this report I have also submitted the .C file for my code and the necessary readme and makefile documents. This will allow you to run my code and understand how the inputs and outputs are formatted. To briefly show the compile and input format see below; this is assuming you are in the proper directory:

Compile →	<code>gcc sim.c -o sim.exe</code>
Input format →	<code><.exe file> <M> <N> <path/to/trace/file></code>
Input example →	<code>./sim.exe 7 0 gobmk_trace.txt</code>

Along with this, all documentation is available on my GitHub at: <https://github.com/Fugate-A/BranchPredictorSimulator>