CMPEN 431 - Project 2
Design Space Exploration
Report
Prapti Panigrahi (pmp5425)

1. **Framework Description:**
   The provided framework allowed rapid exploration of the design space with minimum simulation time. We were introduced to an architectural simulator called Simple Scalar that enabled the exploration of multiple configurations with respect to two metrics - Energy and Performance (time). Correlation of the metrics (edp and execution time) with each parameter from the 18 dimensions was evaluated, observing how the performance (and energy) metrics change with the change in each dimension value. The framework enabled us to design efficient heuristics instead of brute force exhaustive search. Further, pre-marking many design configurations invalid helped quicken the process of exploration.

2. **Design Points Chosen by DSE:**
   a) **Performance (Best Time)**
      i) bestTimeConfig:
         0 0 0 2 0 8 0 0 3 2 0 0 4 3 0 0 0 0

      ii) bestEDPConfig
         0 0 0 2 0 8 0 0 3 1 0 0 4 3 0 0 0 0

   b) **Energy Efficiency**
      i) bestTimeConfig:
         0 0 0 2 0 6 2 0 3 1 0 0 4 3 2 0 0 0

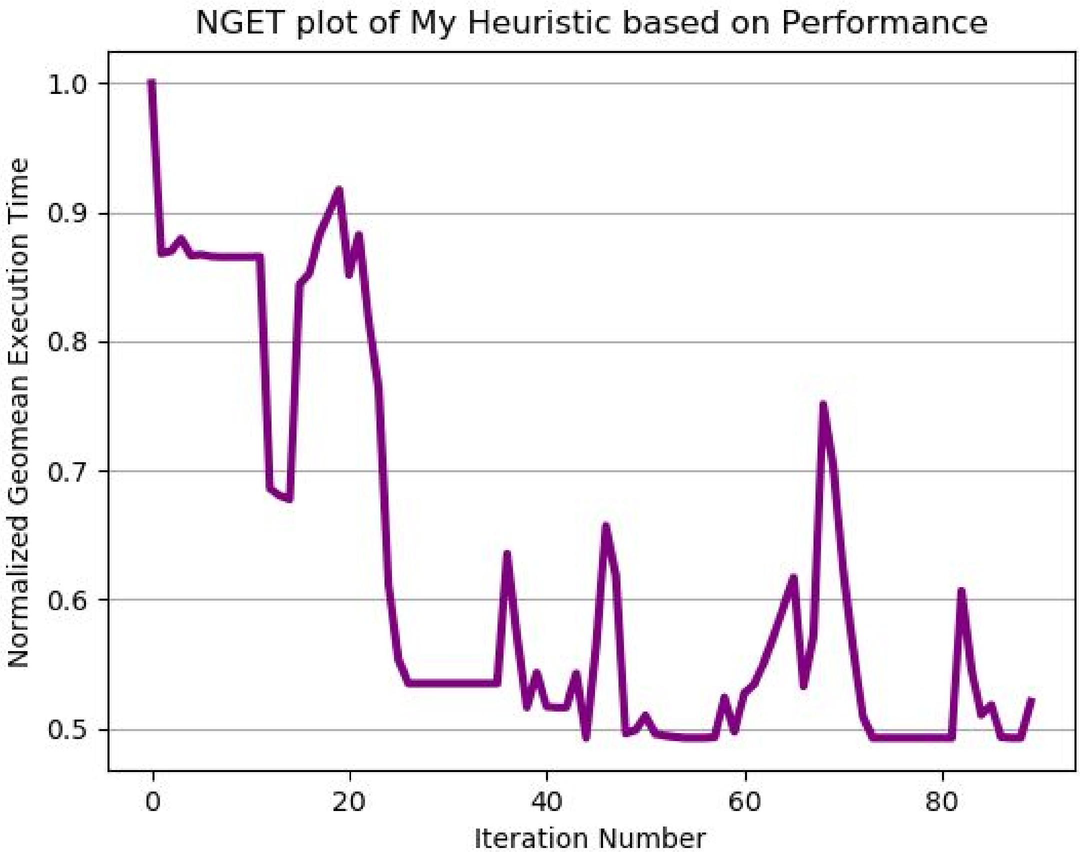
      ii) bestEDPConfig:
         0 0 0 2 0 6 1 0 3 1 0 0 4 3 2 0 0 0

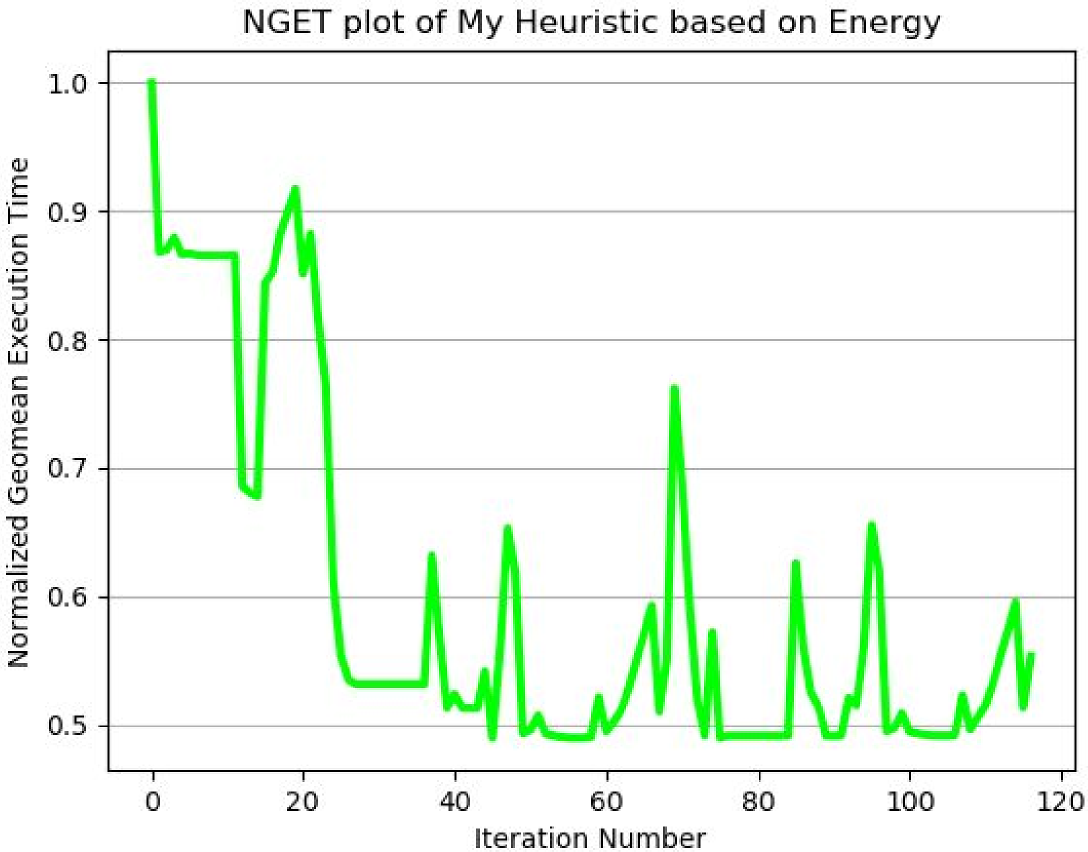
## A1. Table

| Parameter | Performance | EDP |
|---|---|---|
| width | Value = 1<br>Why = Pipelining leads to higher IPC, thus reducing time. Multi-width pipelines have more complications and a lot more data hazard checks | Value = 1<br>Why = Pipelining leads to higher IPC, thus reducing time. Multi-width pipelines have more complications and a lot more data hazard checks |
| scheduling | Value = In-Order<br>Why = OOO leads to a faster execution | Value = In-Order<br>Why = OOO executes every instruction greedily, thus leading to a faster execution |
| l1block | Value = 8<br>Why = Higher block size means lower miss rate. | Value = 8<br>Why = Higher block size means lower miss rate. |
| dl1sets | Value = 128<br>Why = Optimal number of data sets reduces miss rates | Value = 128<br>Why = Optimal number of data sets reduces miss rates |
| dl1assoc | Value = 1<br>Why = Higher data associativity improves Cache miss rates | Value = 1<br>Why = Higher data associativity improves Cache miss rates |
| il1sets | Value = 8192<br>Why = More Instruction Sets increases IPC | Value = 8192<br>Why = More Instruction Sets increases IPC |
| il1assoc | Value = 1<br>Why = Low value chosen as num sets is high | Value = 1<br>Why = Low value chosen as num sets is high |
| ul2sets | Value = 256<br>Why = decreases Global Miss Rate | Value = 256<br>Why = decreases Global Miss Rate |
| ul2block | Value = 128<br>Why = reduces Global AMAT | Value = 128<br>Why = Global AMAT |
| ul2assoc | Value = 4<br>Why = leads to lesser Global miss rates | Value = 2<br>Why = leads to lesser Global miss rates |
| replacepolicy | Value = LRU<br>Why = Multiple data blocks and instructions, once used, would be used again with high probabilities. | Value = LRU<br>Why = Multiple data blocks and instructions, once used, would be used again with high probabilities. |
| fpwidth | Value = 1<br>Why = More ALUs allow extraction of more ILP and increase performance | Value = 1<br>Why = More ALUs allow extraction of more ILP and increase performance |
| branchsettings | Value = -bpred comb -bpred: comb 1024<br>Why = Branch Prediction takes care of control hazards, thus increasing performance. | Value = -bpred comb -bpred: comb 1024<br>Why = Branch Prediction takes care of control hazards, thus increasing performance. |
| ras | Value = 8<br>Why = Higher the value, lower the miss penalty | Value = 8<br>Why = Higher the value, lower the miss penalty |
| btb | Value = (128 – 16)<br>Why = Larger btb has lesser misses | Value = (128 – 16)<br>Why = Larger btb has lesser misses |

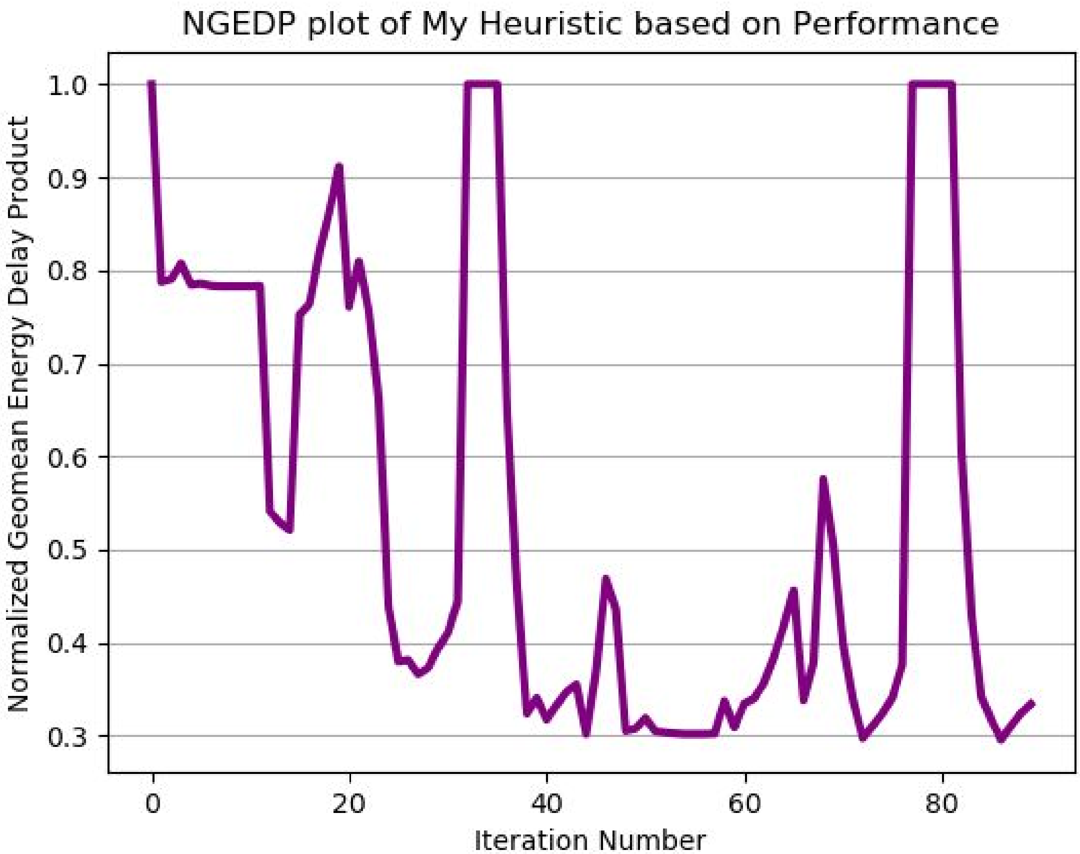
**A2. Plots**

**A.**

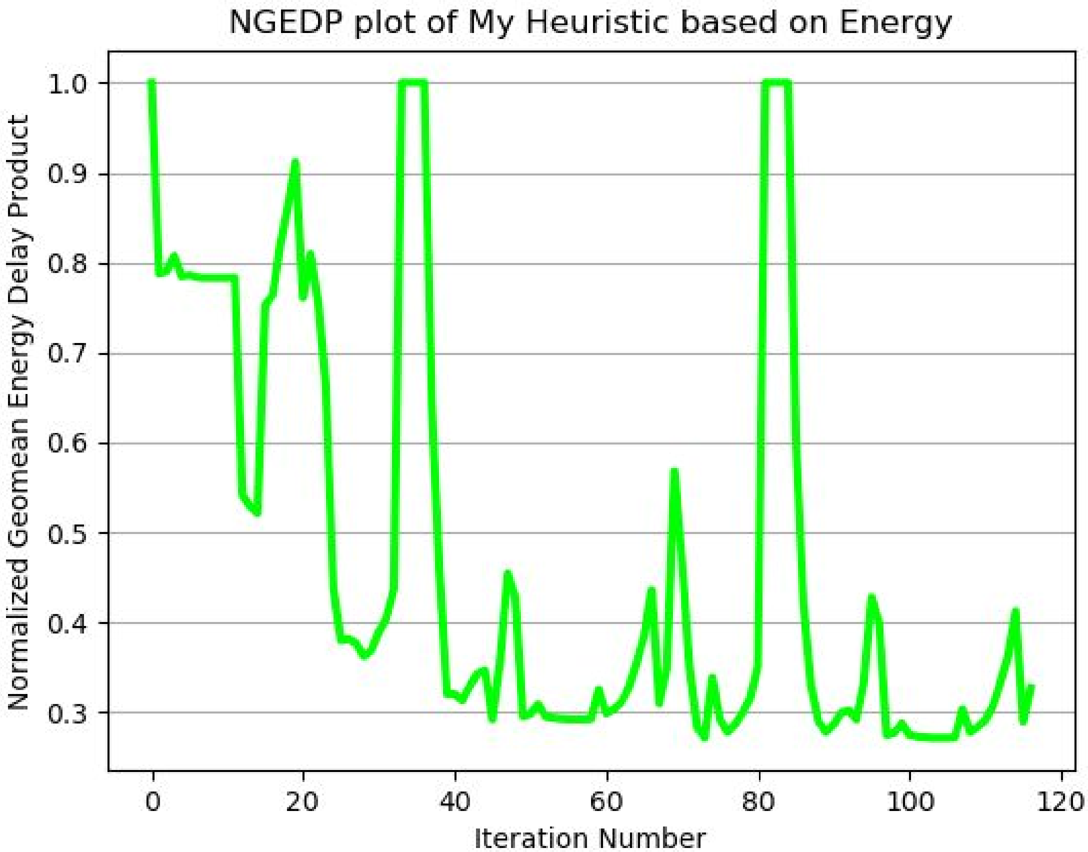

NGET plot of My Heuristic based on Performance
1.0
0.9
0.8
0.7
0.6
0.5
Normalized Geomean Execution Time
0
20
40
60
80
Iteration Number


NGET plot of My Heuristic based on Energy
Normalized Geomean Execution Time
Iteration Number
1.0
0.9
0.8
0.7
0.6
0.5
0
20
40
60
80
100
120

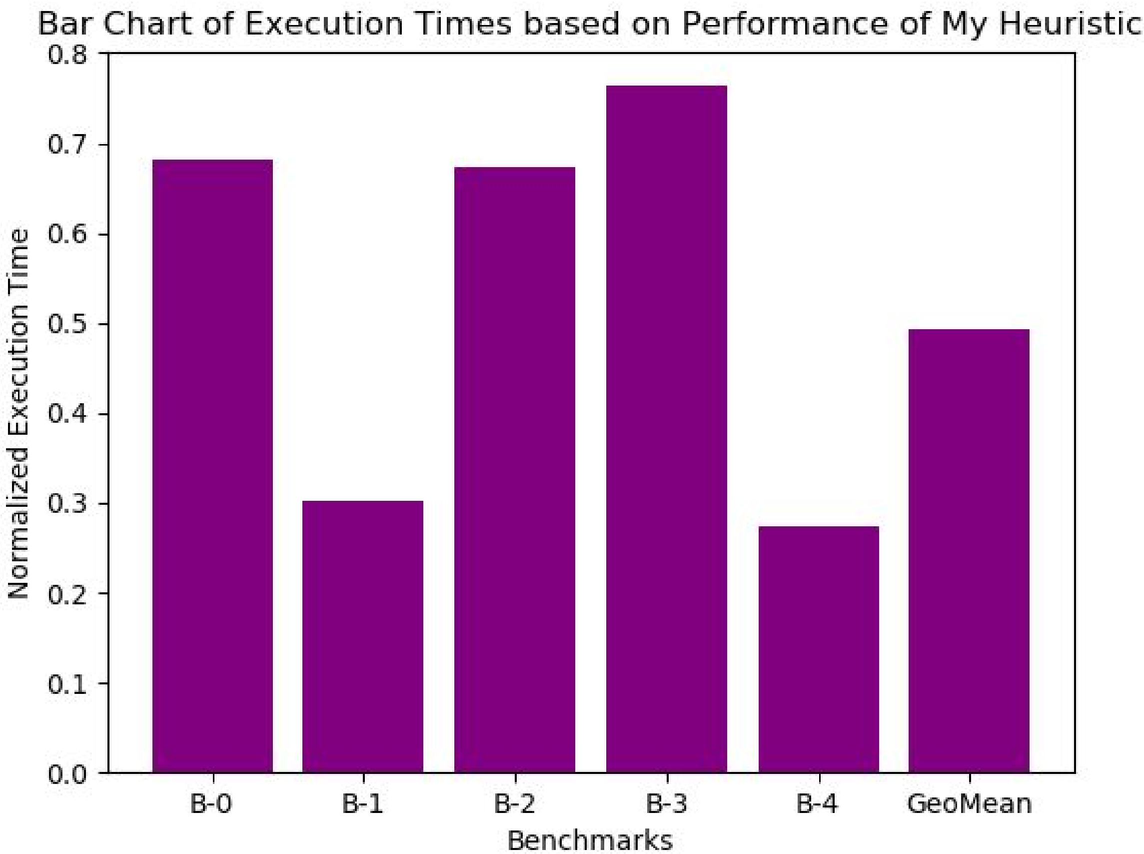
**B**.

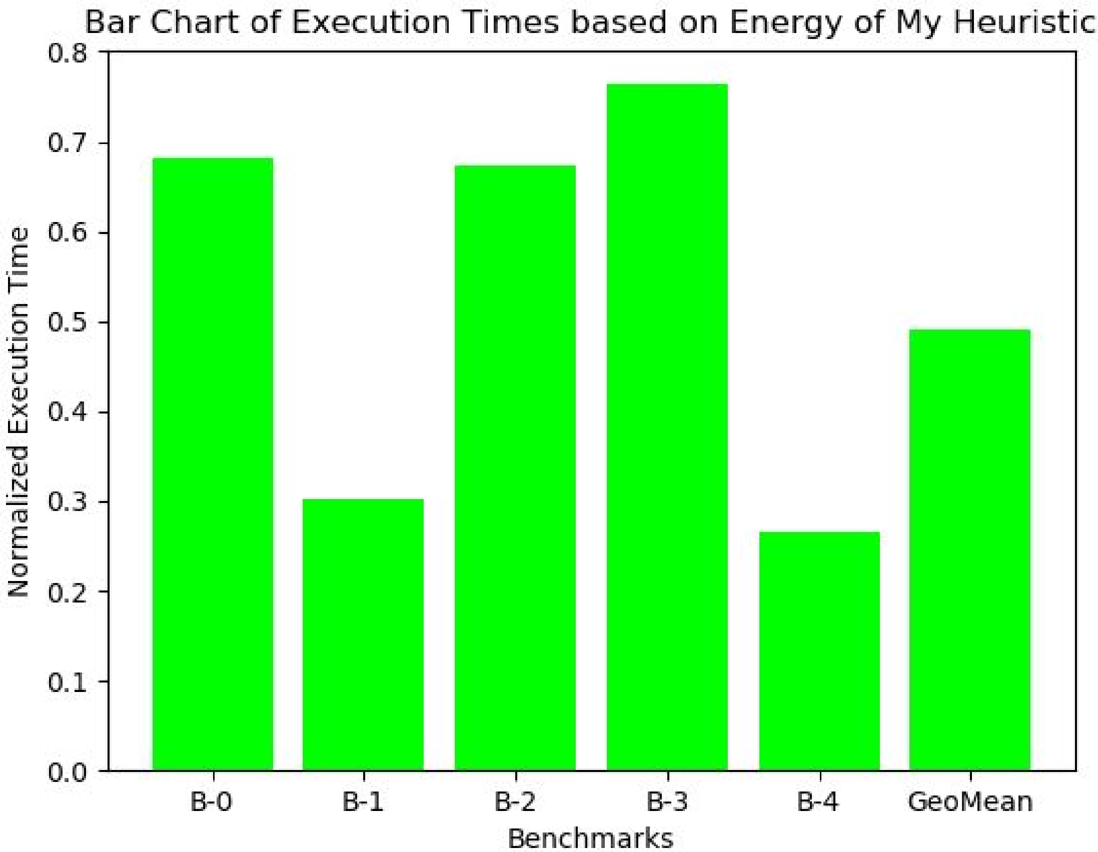


NGEDP plot of My Heuristic based on Performance
Normalized Geomean Energy Delay Product
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0
20
40
60
80
Iteration Number

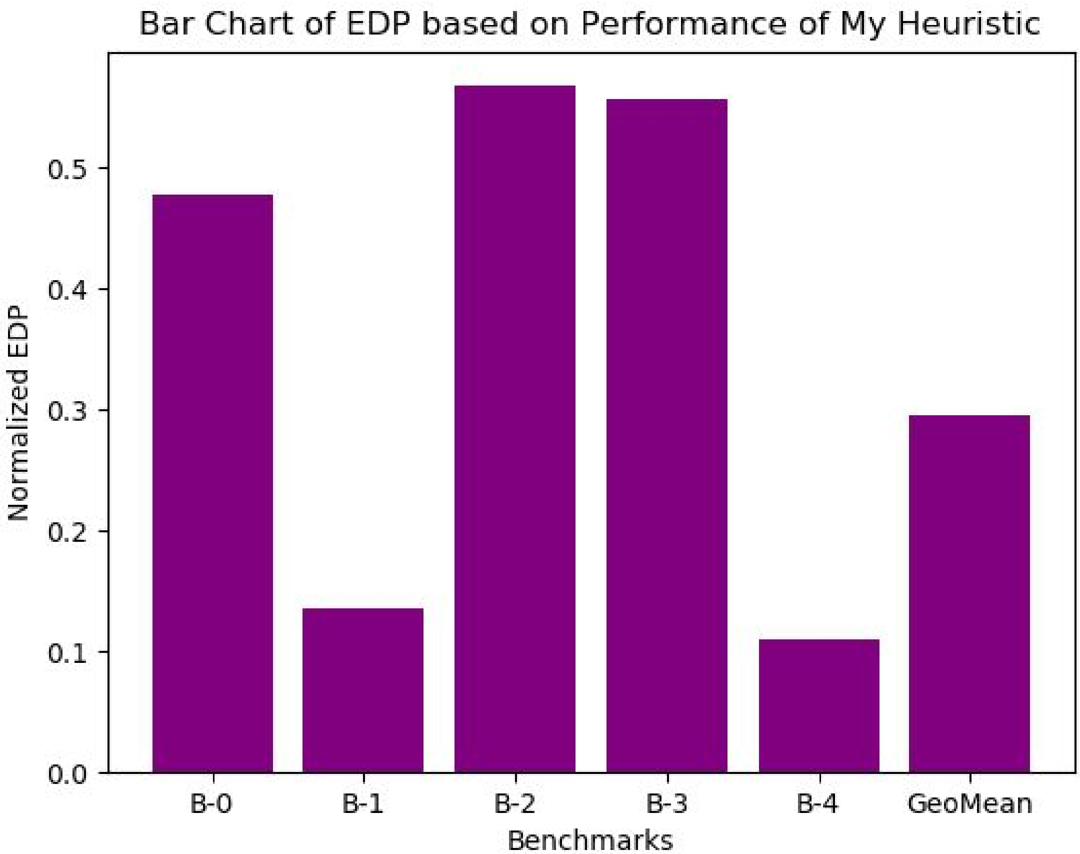

NGEDP plot of My Heuristic based on Energy
Normalized Geomean Energy Delay Product
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0
20
40
60
80
100
120
Iteration Number

**c.**

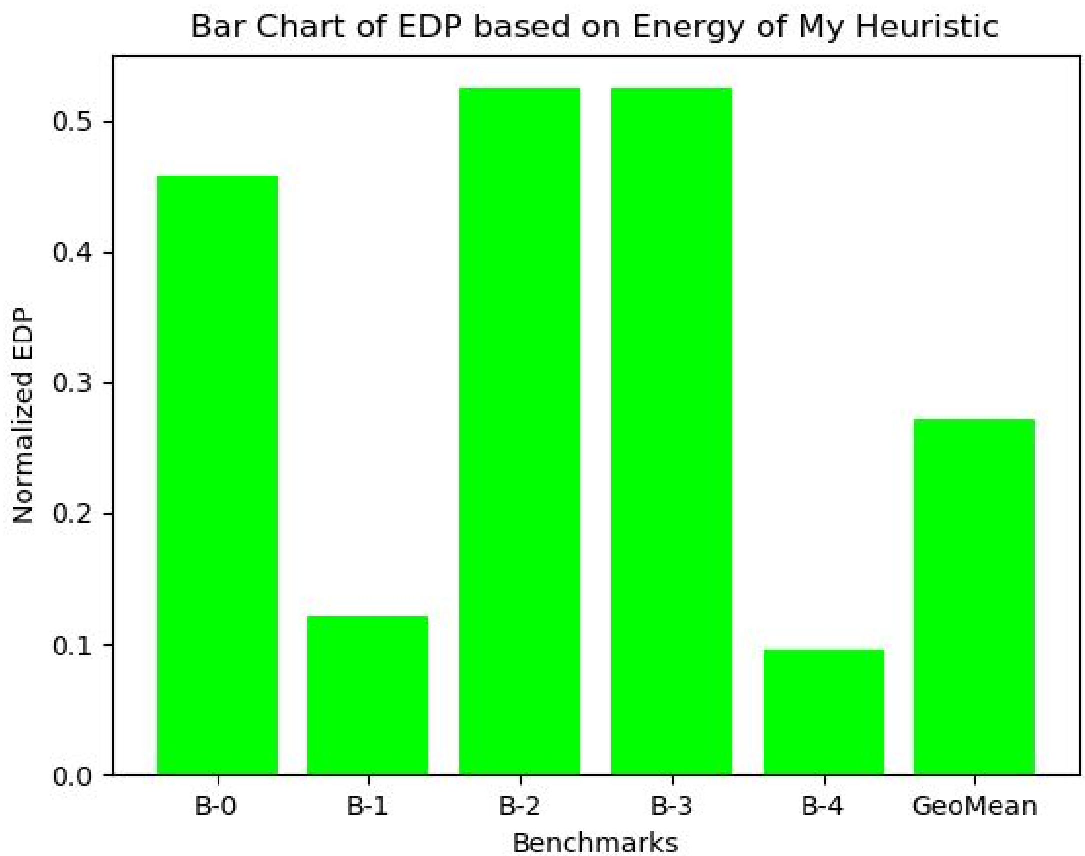

Bar Chart of Execution Times based on Performance of My Heuristic
Normalized Execution Time
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
B-0
B-1
B-2
B-3
B-4
GeoMean
Benchmarks


Bar Chart of Execution Times based on Energy of My Heuristic
Normalized Execution Time
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
B-0
B-1
B-2
B-3
B-4
GeoMean
Benchmarks

**D.**



Bar Chart of EDP based on Performance of My Heuristic
Normalized EDP
0.0
0.1
0.2
0.3
0.4
0.5
B-0
B-1
B-2
B-3
B-4
GeoMean
Benchmarks




**5. A more sophisticated heuristic:**

In our current heuristic, we change one parameter at a time and see its effect on a global metric. The receptive field of this parameter (i.e., the effect it has on the global metric might be unnoticable). So, a better and more sophisticated heuristic might be to change two or more parameters at a time, based on a pre-defined prior, which groups the parameters that have the highest priority of bringing maximum change in the global heuristic. The convergence of such a method might be faster.

**6. Two new insights gained:**

a) I got to learn about SimpleScalar simulator and how it is efficiently used in design space exploration
b) How to choose a heuristic to efficiently perform design space exploration instead of an exhaustive search.