

Ethan Sanchez 70748407

Jeremy Hobson 68795576

Mini Project Analysis and Conclusions

Easy Problem Data:

FC --- --- --- ---	: avg. time = 0.61684	avg. backtracks = 19.2
FC --- LCV --- ---	: avg. time = 0.61454	avg. backtracks = 22.0
FC --- --- NKP ---	: avg. time = 0.53109	avg. backtracks = 15.2
FC --- LCV NKP ---	: avg. time = 0.45933	avg. backtracks = 15.4
FC --- --- --- NKT	: avg. time = 0.50195	avg. backtracks = 20.8
FC --- LCV --- NKT	: avg. time = 0.63545	avg. backtracks = 25.6
FC --- --- NKP NKT	: avg. time = 0.27120	avg. backtracks = 13.6
FC --- LCV NKP NKT	: avg. time = 0.43641	avg. backtracks = 17.8
FC DH --- --- ---	: avg. time = 12.62445	avg. backtracks = 637.6
FC DH LCV --- ---	: avg. time = 10.09034	avg. backtracks = 478.0
FC DH --- NKP ---	: avg. time = 0.72812	avg. backtracks = 29.2
FC DH LCV NKP ---	: avg. time = 0.46473	avg. backtracks = 16.2
FC DH --- --- NKT	: avg. time = 2.12531	avg. backtracks = 105.0
FC DH LCV --- NKT	: avg. time = 0.84180	avg. backtracks = 30.2
FC DH --- NKP NKT	: avg. time = 0.65927	avg. backtracks = 26.4
FC DH LCV NKP NKT	: avg. time = 0.46573	avg. backtracks = 16.2

FC MRV --- --- ---	: avg. time = 1.22367	avg. backtracks = 47.6
FC MRV LCV --- ---	: avg. time = 1.15602	avg. backtracks = 41.8
FC MRV --- NKP ---	: avg. time = 0.60443	avg. backtracks = 16.0
FC MRV LCV NKP ---	: avg. time = 0.53358	avg. backtracks = 13.6
FC MRV --- --- NKT	: avg. time = 1.19925	avg. backtracks = 44.8
FC MRV LCV --- NKT	: avg. time = 1.06896	avg. backtracks = 37.6
FC MRV --- NKP NKT	: avg. time = 0.57561	avg. backtracks = 16.2
FC MRV LCV NKP NKT	: avg. time = 0.46293	avg. backtracks = 12.4
ACP --- --- --- ---	: avg. time = 0.33804	avg. backtracks = 8.6
ACP --- LCV --- ---	: avg. time = 0.36506	avg. backtracks = 9.0
ACP --- --- NKP ---	: avg. time = 0.28040	avg. backtracks = 5.0
ACP --- LCV NKP ---	: avg. time = 0.28441	avg. backtracks = 5.4
ACP --- --- --- NKT	: avg. time = 0.30722	avg. backtracks = 7.8
ACP --- LCV --- NKT	: avg. time = 0.33023	avg. backtracks = 8.4
ACP --- --- NKP NKT	: avg. time = 0.29061	avg. backtracks = 6.2
ACP --- LCV NKP NKT	: avg. time = 0.29861	avg. backtracks = 7.0
ACP DH --- --- ---	: avg. time = 3.83832	avg. backtracks = 212.4
ACP DH LCV --- ---	: avg. time = 3.45765	avg. backtracks = 179.8
ACP DH --- NKP ---	: avg. time = 0.39148	avg. backtracks = 9.8
ACP DH LCV NKP ---	: avg. time = 0.36466	avg. backtracks = 7.4
ACP DH --- --- NKT	: avg. time = 0.96268	avg. backtracks = 39.2
ACP DH LCV --- NKT	: avg. time = 0.36426	avg. backtracks = 7.4

ACP DH --- NKP NKT	: avg. time = 0.41730	avg. backtracks = 10.0
ACP DH LCV NKP NKT	: avg. time = 0.33144	avg. backtracks = 6.0
ACP MRV --- --- ---	: avg. time = 0.48635	avg. backtracks = 11.4
ACP MRV LCV --- ---	: avg. time = 0.42630	avg. backtracks = 9.0
ACP MRV --- NKP ---	: avg. time = 0.35685	avg. backtracks = 5.0
ACP MRV LCV NKP ---	: avg. time = 0.36086	avg. backtracks = 4.8
ACP MRV --- --- NKT	: avg. time = 0.49555	avg. backtracks = 11.4
ACP MRV LCV --- NKT	: avg. time = 0.46033	avg. backtracks = 8.6
ACP MRV --- NKP NKT	: avg. time = 0.34605	avg. backtracks = 4.8
ACP MRV LCV NKP NKT	: avg. time = 0.36546	avg. backtracks = 4.6

* It should be noted that each of these combinations successfully completes for each of the easy problems (i.e., they have a 100% success rate).

From our results from solving the easy the problems with every logical heuristic combination (i.e., we did not run tests with *only* backtracking search), it appears that solving with ACP-MRV-LCV-NKP-NKT results in the fewest backtracks, with an average of 4.6 backtracks. However, the combination FC-NKP-NKT runs significantly faster (.37 seconds versus .27 seconds), with an average of 13.6 backtracks. It is our conclusion then, that for the easy problems FC-NKP-NKT is the optimal combination of heuristics to use for the 9 x 9 easy sudoku problems..

Medium Problem Data (Best Combinations)

FC --- LCV NKP NKT	: avg. time = 16.90966	avg. backtracks = 328.0
FC MRV LCV NKP NKT	: avg. time = 32.99692	avg. backtracks = 528.5
ACP --- LCV NKP NKT	: avg. time = 8.05930	avg. backtracks = 136.0
ACP MRV LCV NKP NKT	: avg. time = 4.34843	avg. backtracks = 41.25

* It should be noted that each of these combinations successfully completed for problems 1, 2, 3, and 5 of the medium problem set. It seemed that for problem 4, the program gets stuck in a loop.

The four combinations above are what we considered to be the best combinations for solving the medium problems. We noticed that the degree heuristic was consistently producing solutions slower when used instead of MRV, which is why it is not included in the combinations above. This may be because, unlike MRV, the degree heuristic does not necessarily choose a node which will minimize the depth of our backtracks.

From the data, the combination ACP-MRV-LCV-NKP-NKT is the winner, with an average time of 4.35 seconds and an average of 41.25 backtracks. Both of the combinations using ACP are faster than the combinations using FC.

It is interesting to note that certain combinations which use MRV find the solution slower than their counterparts. Combinations using MRV also require more backtracks (as is expected). It can be seen in the data for the easy and hard problems that this is also the case. One notable exception is the combination ACP-MRV-LCV-NKP-NKT for the medium problems. This

combination is actually the *most* efficient in terms of both time and backtracks. However, this appears to be the exception rather than the rule. Our conclusion then, is that while MRV excels at finding incorrect values quickly, the sheer volume of backtracks it produces severely impacts the efficiency of solving difficult problems (i.e., ones which have few initial values).

Hard Problem Data (Best Combinations)

FC --- LCV NKP NKT : avg. time = 2299.98037 avg. backtracks = 18120.4

FC MRV LCV NKP NKT : avg. time = 6009.32965 avg. backtracks = 53760.0

ACP --- LCV NKP NKT : avg. time = 689.65092 avg. backtracks = 7744.8

ACP MRV LCV NKP NKT : avg. time = 997.97031 avg. backtracks = 8635.2

* It should be noted that each of these combinations successfully completes for each of the hard problems (i.e., they have a 100% success rate).

The four combinations above are what we considered to be the best combinations for solving the hard problems. These are the same combinations that were used for the medium problems. We again noticed that the degree heuristic was slower when used instead of MRV, for the same reasons.

From the data, the combination ACP-LCV-NKP-NKT is, again, the clear winner, with an average time of 689.6 seconds ~ 12.5 minutes and an average of 7745 backtracks. Indeed, both combinations using ACP are much, much faster than the combinations using FC.

Yet again, we noticed that the inclusion of MRV in the heuristics for these problems severely impacts their efficiency, for the same reasons as explained above.