

Sudoku Final AI Report

Team number: 52

Member #1 (name/id): Zhanhao Ruan 17883412

Member #2 (name/id): Feiyang Jin 92569824

I. Minimal AI

I.A. Briefly describe your Minimal AI algorithm. What did you do that was fun, clever, or creative?

We implement Forward Checking(FC) for our Minimal AI algorithm. Forward Checking will pour out some available domain values for neighbor's variables once a variable has been assigned. Forward checking will return false when one variable has no available value in its domain and let the solver know it needs to backtrack.

We optimize the FC algorithm a little bit to improve its efficiency by passing the assigned variable to the function. So instead of iterating through the constraints network to find out the modified constraints, we let the solver to pass the modified variable into the FC function. So our FC function will only check the constraints involving with this modified variable. Iterating the whole constraints is not necessary because we do FC ever since we assign a value, so we only need to pour out values in domains that are involved with this assigned value.

However, the BT class will do Forward Checking(If we choose to use this) once it has been created before any values are assigned, so we implement one-time only iterating through the constraints network to get an initial global forward checking. Once that has been done, we will only looking at constraints that are involved with the assigned variable instead of looking through the whole network.

I.B Describe your Minimal AI algorithm's performance:

E.g. Generate around 60 boards of different difficulties and run your Minimal AI algorithm. Then provide a few words and a table like the following:

We generate 65 boards in total and distribute more easy boards than hard boards because Minimal AI algorithm has poor ability to solve relative hard boards. We don't waste too much time on hard and expert boards. The average number of backtracks is weighted by the number of sample boards.

Board Size	Sample Size (n)	Boards Solved	Average # of backtracks
9x9 (easy)	30	30	344.1
12x12 (intermediate)	20	20	3261.65
16x16 (hard)	10	9	569289.0
25x25 (Expert)	5	0	1699008.0

Total Summary	65	59	219438.25(Weighted)
---------------	----	----	---------------------

II. Final AI

II.A. How did integrating advanced techniques (LCV, MRV, MAD, or NOR) into the Final AI change its solving strategy compared to Minimal AI?

The advanced techniques greatly improve algorithm's ability in solving boards. By picking good values to proceed and more comprehensive constraints pouring techniques, the Final AI will save much computational power and reduce backtracks.

The MRV will, instead of choosing the first available variable to assign value in Minimal AI, choose the variable with the minimal remaining values. Then it will assign the first available value in the domain of that variable.

The MRD will do one more step further than MRV. It will choose the variable with the most constraints associated. This step will happen when there are multiple variables with minimal remaining values and be a tie breaker.

The LCV is used to choose which value to assign to of a picked variable. So the Final AI will choose the value that is associated with least constraints with its neighbor variables.

The NOR is one more step further than FC. After the neighbor constraints have been poured, the NOR will automatically assign variables with only one values in the domains. And then it will do constraints pouring recursively until no variables could be assigned.

II.B. Which of the advanced heuristics (LCV, MRV, MAD, or NOR) had the most significant impact on the performance, and why do you think that was?

We think MRV and MAD had the most significant impact on the performance because first, from the test, an algorithm with MRV or MAD will have much less backtracks than an algorithm without MRV and MAD. And second, in theory, MRV and MAD will help the algorithm to pick a suitable variable to proceed, which will help to save algorithm much computational power on proceeding to not promising variables and reduce many backtracks. MAD apparently will have further impact than MRV since

II.C Describe your Final AI algorithm's performance:

E.g. Use the same generated 60 boards from earlier and run your Final AI algorithm. Compare your results with Minimal AI performance, then provide a few words and a table like the following:

So we have generated 65 boards in total for testing, and we distribute more hard boards than Minimal AI since the Final AI will have more ability in solving hard boards and easy boards can be solved quickly. The average number of backtracks is weighted by the number of sample boards.

Board Size	Sample Size (n)	Boards Solved	Average # of backtracks
9x9 (easy)	20	20	0.25
12x12 (intermediate)	20	20	14.5
16x16 (hard)	15	15	41.93333333333333
25x25 (Expert)	10	8	87520.6
Total Summary	65	63	13478.92(Weighted)

III. Has this project altered your interest or perspective towards artificial intelligence? If so, how?

Yes, our interest in artificial intelligence is sparked by the project. When we were young, we used this Sudoku problem to train our logical thinking so we could better solve mathematical problems. However, now we can use this algorithm for us to solve the same problem. It gets us thinking, there are got to be many unsolvable problems that are hard for humans but easy with artificial intelligence. We want to dive down try to know and try to develop our own algorithm that can benefits our mankind.

III. In about 1/4 page of text or less, provide suggestions for improving this project (*this section does NOT count as past of your two-page total limit.*)

This is definitely a fun project. It is interesting to see how we can use our fresh knowledge we have learned in the class to implement algorithm in solving this Sudoku problem, which when we were young, we used to play around. We felt achieved with the outcome. In terms of improving, we think it will be better to if we can use more knowledge we have learned in the late quarter in this project. So as for now, the project requires us to use the knowledge before week4(CSP). But the project will be more fun if we can use implement some Neural network and machine learning theory in the algorithm. Also, if we can create two or more algorithm competing against each other, it will be more fun.