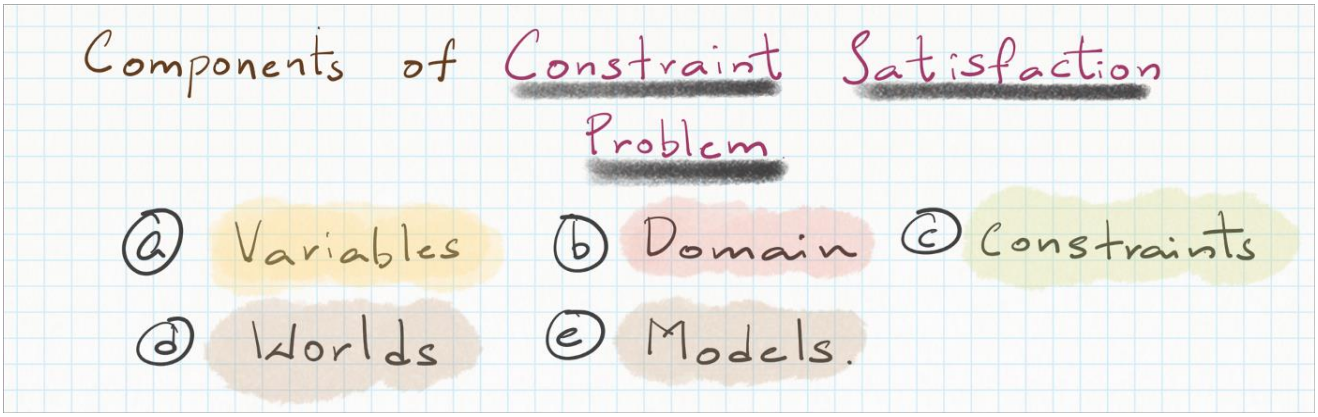
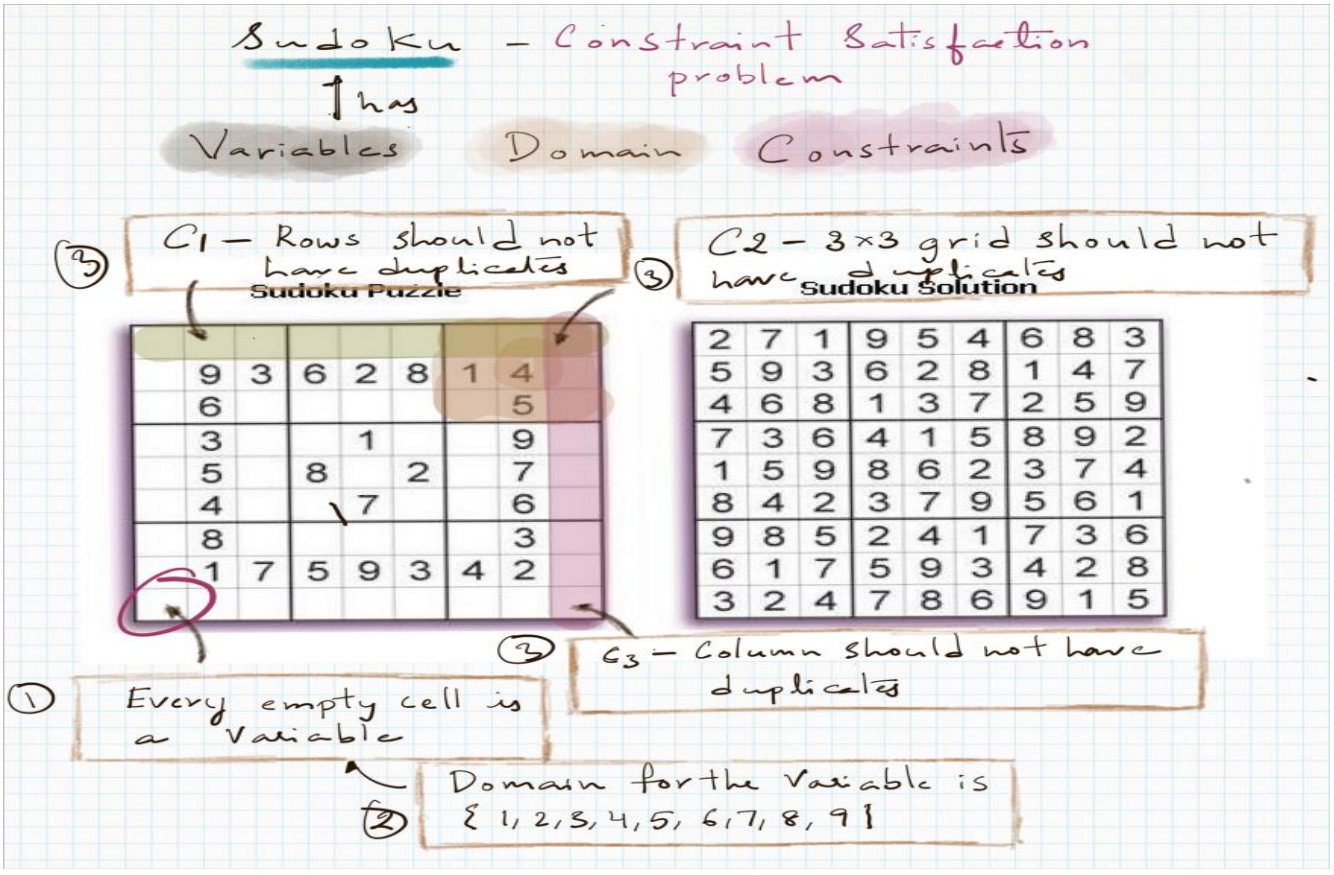
AI: Sudoku problem using Constraint Satisfaction Search

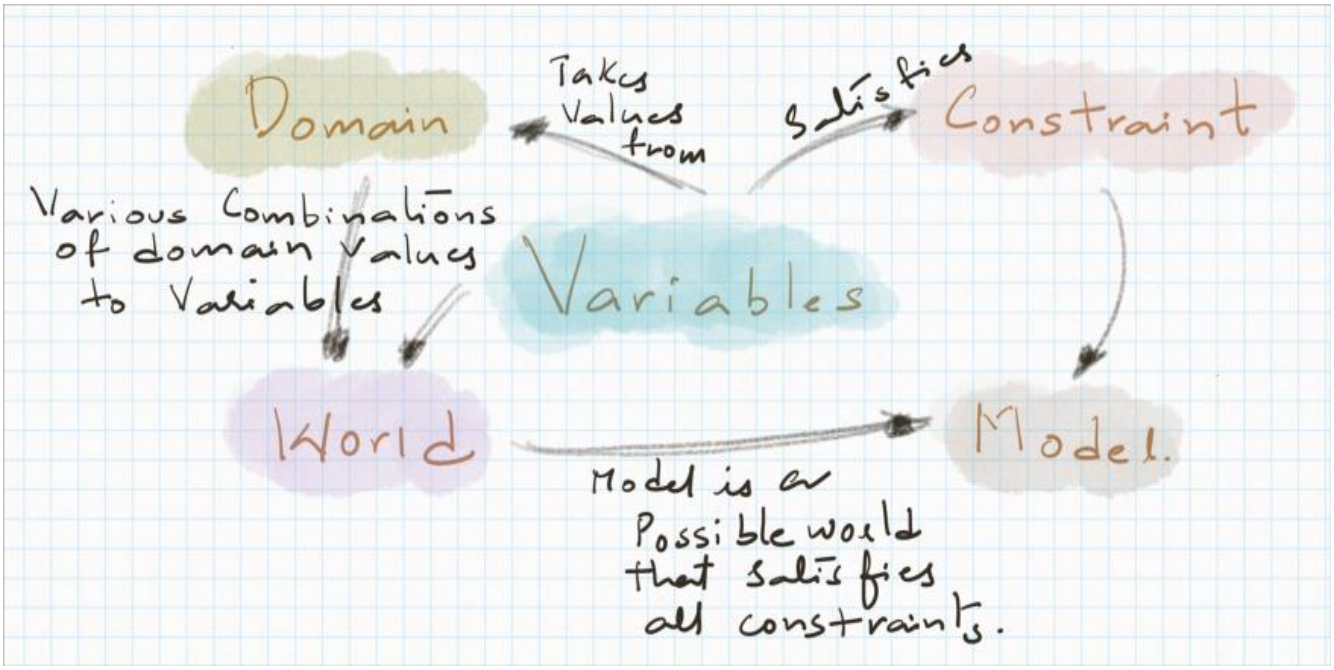
Team Members: Dipneet Kaur (12) & Drishti Malhotra (13)

Components of Constraint Satisfaction Problem (CSP)



Using Sudoku and identifying these components:





Variables:

Rows are indexed alphabetically starting with {A, B, C, D .. and so on.}
Columns are indexed by digits starting with {1,2,3 ,4 .. and so on.}
For 9*9 sudoku boards, variables are {A1, A2…A9, B1, B2 ,..., I1,I2,...I9.}
At max 81 variables can be present.

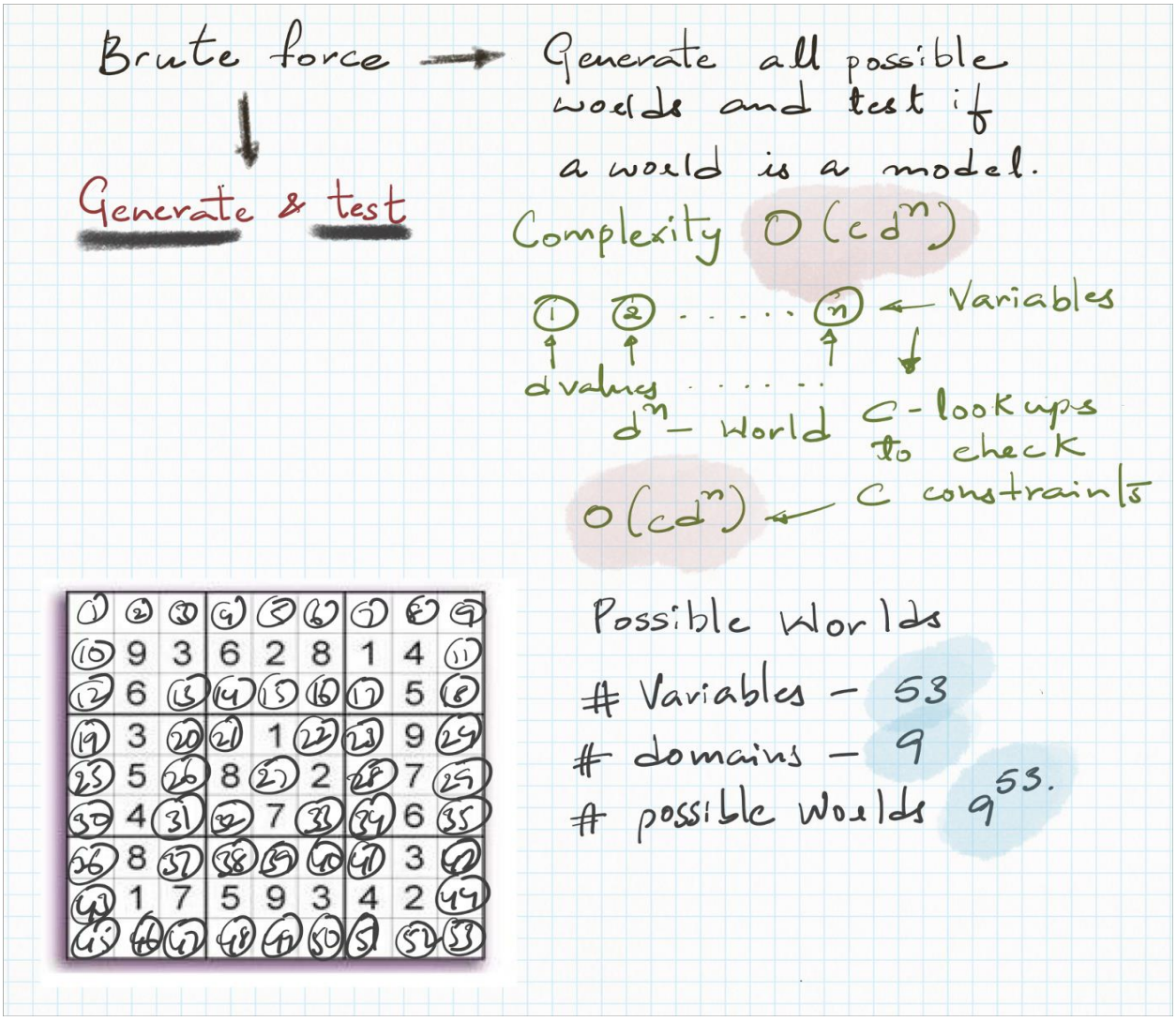
Domain:

For each variable, the value lies b/w [1,9]

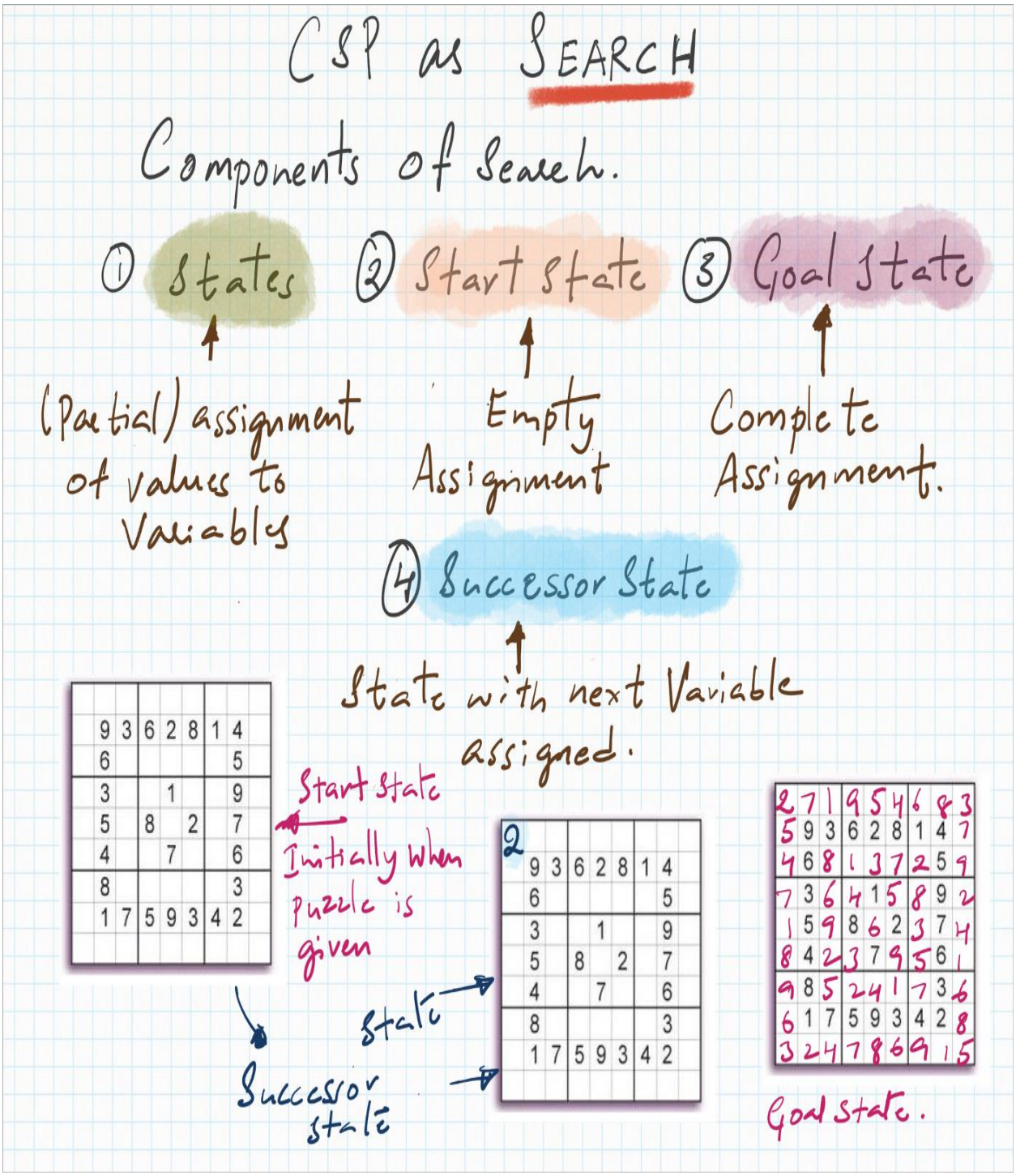
Constraints:

Each row consists of 9 distinct values ∈ [1,9].
I.e. <(A1,A2..A9), (all should be distinct)> and so on.
Each column consists of 9 distinct values ∈ [1,9].
I.e. <(A1,B1,...,I1), (all should be distinct)>
For each 3*3 blocks (no overlapping blocks) starting from A1 cell, consists of the 9 distinct values ∈ [1,9].
I.e. <(A1,A2,A3,B1,B2,B3,C1,C2,C3), (all should be distinct)>

Brute force to get all possibilities:



Reducing the Search space:



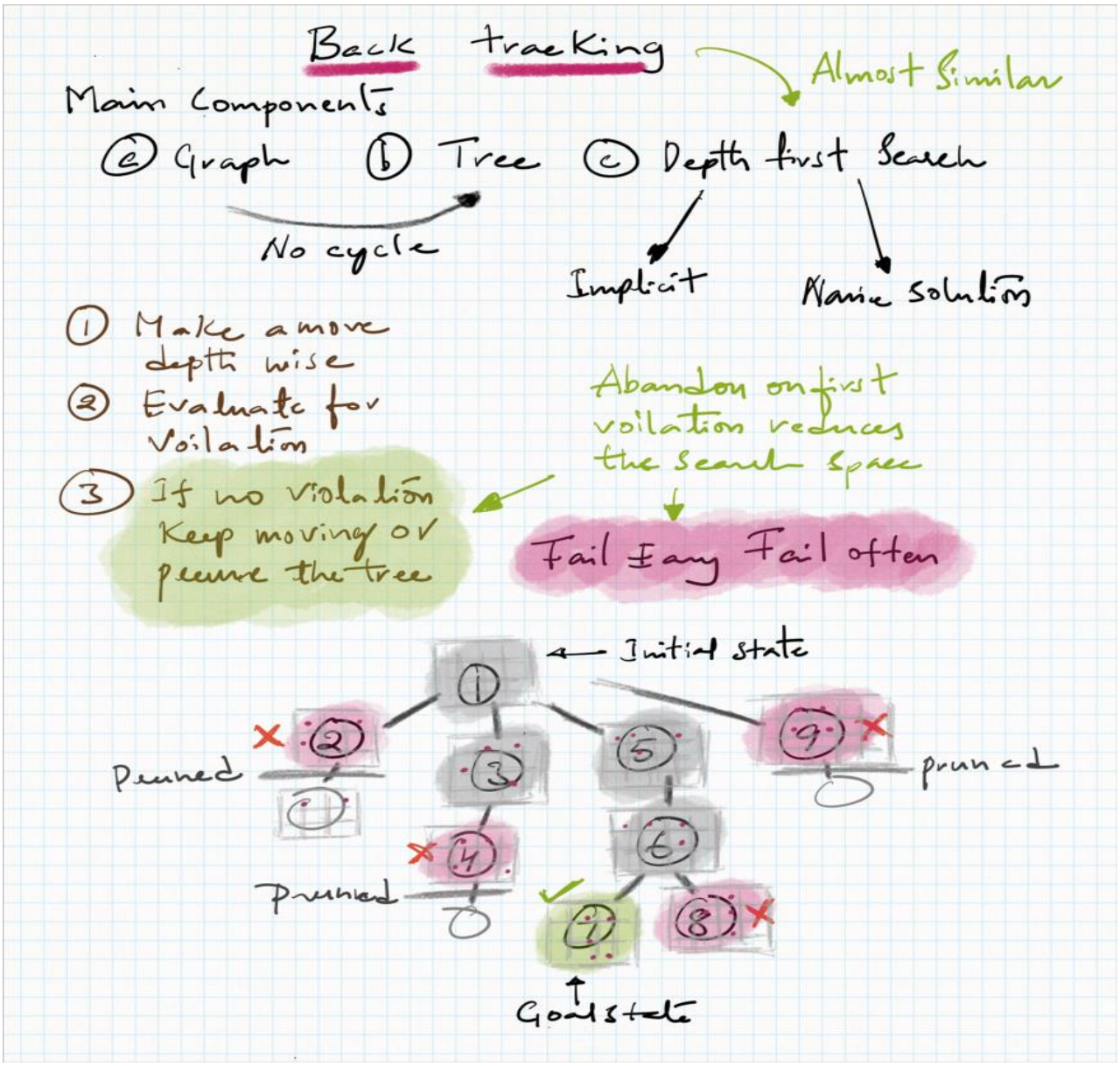
CSP Search using Backtracking:

Methodology:

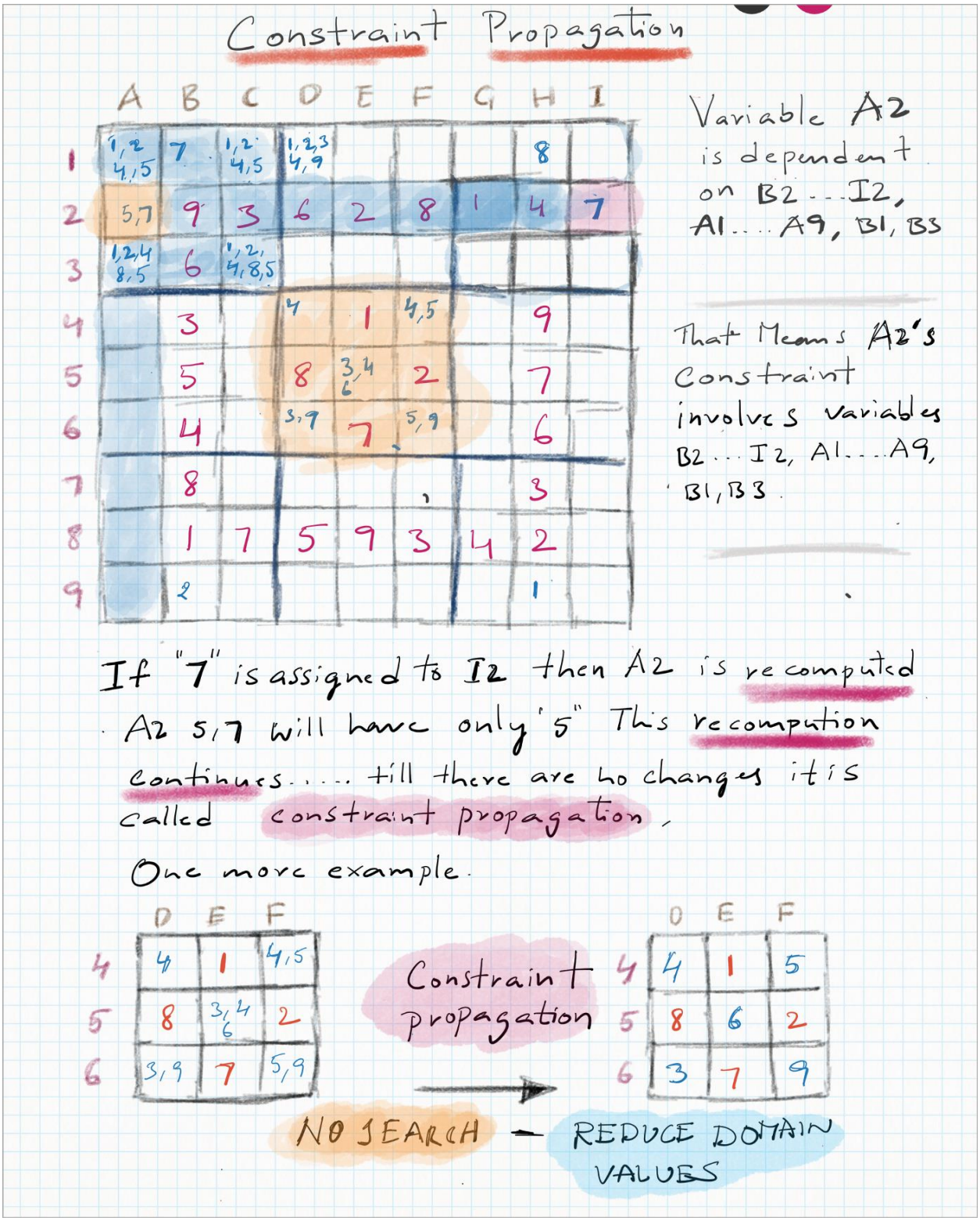
For each row, column and block of 3*3 We maintain a set which will tell us about the values which are already taken by the particular row, column and block.

When we assign a value to a cell, we find out a value which is not included in the underlying row, column and block, and assign that value to the cell.
In the same manner we try to assign values to all the cells until all cells are filled or a constraint violation has taken place.

In case of a constraint violation we backtrack and try to assign other values.
We keep on repeating these steps until all the cells are filled.



Using Constraint Propagation to solve CSP:



Generalization

For sudoku of sizes 16*16, 25*25, 36*36, n*n in general

We would need to assume n as a perfect square for making smaller blocks,
Let m = sqrt(n)

Variables will increase depending on the board size (Number of variable = n*n - #(given values))
At max n*n variables can be present.

Domain for each cell will become [1,n]

Constraints:

For each row we need distinct values for each cell ∈ [1,n]
For each column we need distinct values for each cell ∈ [1,n]
For each block of m*m we need distinct values for each cell ∈ [1,n]

Methodology will be the same as previous.

References:

www.medium.com

Thank You.