cs228_Assignment1

Srinivas Dorisala(24B1000),Charan Bothamolla(24B0966)

August 2025

1 Question1:SAT-Based Sudoku Solver

OBJECTIVE: The objective of solution is to provide a formulae which is "TRUE" iff the given sudoku is solvable.
For finding the solution of the sudoku is same as giving a single assignment to each box of the sudoku(considering 9×9),by obeying the rules of sudoku.

RULES: let $\mathcal{S}$ be a set {1,2,3,4,5,6,7,8,9}

1. Each row of the sudoku must contain all numbers from the set $\mathcal{S}$
2. Each column of the sudoku must contain all numbers from the set $\mathcal{S}$
3. sudoku is divided into 9 boxes of each $3 * 3$ which are $\{(row(1,2,3) \times column(1,2,3)), (row(4,5,6) \times column(1,2,3)), (row(7,8,9) \times column(1,2,3)), (row(1,2,3) \times column(4,5,6)), (row(4,5,6) \times column(4,5,6)), (row(7,8,9) \times column(4,5,6)), (row(7,8,9) \times column(1,2,3)), (row(7,8,9) \times column(4,5,6)), (row(7,8,9) \times column(7,8,9))\}$ each should also contain every number from the set $\mathcal{S}$
4. Each box should contain only one element .

Sudoku is said satisfiable if the given assignment of numbers follow this rules
So, we will encode this rules into the SAT solver to get the satisfiable assignments

VARIABLE ID's or VARIABLE CHOICES:
The chosen variable is defined as $P_{(i,j,k)}$ which is true when the (i,j) box is assigned the value k.
We considered variable ID for $P_{(i,j,k)}$ as $(k * 100 + i * 10 + j)$
ENCODING

rule 1 rule 1 is that all numbers from the set $\mathcal{S}$ are present in the row .For that we write like a number k is present in one of the box of the row i.e; at least one $P_{(i,j,k)}$ is true for (i,j) belongs to a row . This is encoded using or($\vee$). here it is iterated over every row .

rule 2 rule 2 is similar as rule 1 but here it is iterated over every column.

rule 3 rule 3 is similar as rule 1,2 but here it is for some particular box of 3×3 . which is iterated over a set of numbers {1,4,7},boxes of the form $i \in$ x + {1,2,3} , $j \in$ x+ {1,2,3},($x \in$ {1,4,7}).

rule 4 encoding rule 4 is like the each box should not contain any combination of 2 numbers $\wedge_{k1,k2 \in \mathcal{S}, k1 \neq k2}(\neg P_{(i,j,k1)} \vee \neg P_{(i,j,k2)})$. It is iterated over each box of the sodoku.

CONTRIBUTIONS:
Srinivas(24B1000) : encoding of rule 1,rule 4.
charan(24B0966) : encoding of rule 2,3.
Combined work : decoding.

2 Question 2:Grading Assignments Gone Wrong(sokoban game

OBJECTIVE: The objective here is to find satisfiability of the given position of the player , boxes and Goals such that every box will be in the position of goal at the end of T moves of the player.
VARIABLE ID's or VARIABLE CHOICES:
The chosen variable is defined as $P_{(t,k,i,j)}$ which implies
if k==1 at time t player at i,j grid is true.
if k==1+b at time t bth box at i,j grid is true .
we considered variable ID for $P_{(t,k,i,j)}$ as t*1000+k*100+i*10+j.(given the no.of rows and columns < 10 . so , $i, j \in \{0, 9\}$ for any no.of rows i, j cannot disturb the values of other digits (no coincidences of variables)
ENCODING:
In encoding there are 5 types of conditions

initial conditions: The initial conditions are like assigning truth and false values to the initial positions of the players ,boxes.

player movements and box movements: The player movements are like the possible steps for the player when he is at (i,j) which can be encoded as $P_{(t,k,i,j)} \rightarrow P_{(t+1,k,i,j)} \vee P_{(t+1,k,i+1,j)} \vee P_{(t+1,k,i-1,j)} \vee P_{(t+1,k,i,j+1)} \vee P_{(t+1,k,i,j-1)}$ similarly same for box movement(but there will extra constraints for which i gave in pushing logic)
Note if i,j went out of grid then we assign it variable 800 which is always false

pushing logic: when a particular box(k) at position i,j+1 at time t+1 is true and box at position i,j at time t is true(similarly in four directions) then pushing occured.$B_{(t+1,k,i,j+1)} \wedge B_{(t,k,i,j)} \rightarrow P_{(t,k,i,j-1)} \wedge P_{(t+1,k,i,j)}$ (similarly other directions) now convert this into cnf form and append it to the sat solver.

overlapping constraints
1. player and all boxes variables are false at all time if they are at wall position it is implemented in cnf by just appending -x where x is player if it is at wall pos and x is box if it is wall pos at i,j,t in self.cnf
2. both players and boxes variables cannot be true at same position at same time it is implemented in cnf by appending [-p(player),-b(box)] at every i,j,t in self.cnf
3. different boxes variables can't be true at same position at time t it is implemented in cnf by appending [-b1,-b2] for each pair of boxes at every i,j,t in self.cnf
4. player and each labeled box variables are at only one position in grid at a time t it is implemented in cnf by appending [-p1,-p2] p1 is player/box at position a and p2 is same player/box at position b(different from a) for every pair of a,b at every t in self.cnf

Goal condition each position in Goals at time T(final time) there exist at least one box in that position it is implemented in cnf by for each pos in goals appending $[b_1, b_2, .., b_n]$ where b1,b2,.. are boxes at goal pos at time T in self.cnf

DECODING:

step1 collect all player and boxes variables which are true at all instances of time

step2 we need to find at what minimum steps goal condition is satisfied .This can be done by checking the no.of goal conditions satisfied by the boxes and find at which minimum steps will the count be equal to no.of total goals.then store that no.of steps.

step 3 now we need to extract the player positions till the steps of minimum satisfiability which we found in step 2.

step4 from the above extracted players we need to append the moves of player in to a list according to change of position from time t to t+1.and return that list

CONTRIBUTIONS

Srinivas(24B1000) did Pushing logic, Overlapping Constraints, Initial assignment
Charan(24B0966) did Decoder, Player movement, Box movement, Only at one position player exits at time