

Unit propagation

Let S be a set of clauses. **Unit propagation**: repeatedly performing the following transformation: if S contains a unit clause, i.e. a clause consisting of one literal L , then

1. remove from S every clause of the form $L \vee C'$;
2. replace in S every clause of the form $\tilde{L} \vee C'$ by the clause C' .

DLL

```
procedure  $DLL(S)$   
input: set of literals  $S$   
output: satisfiable or unsatisfiable  
parameters: function select_literal  
begin  
   $S := propagate(S)$   
  if  $S$  is empty then return satisfiable  
  if  $S$  contains  $\square$  then return unsatisfiable  
   $L := select\_literal(S)$   
  if  $DLL(S \cup \{L\}) = satisfiable$   
    then return satisfiable  
    else return  $DLL(S \cup \{\tilde{L}\})$   
end
```

Two optimizations

Tautology: $p \vee \neg p \vee C$.

Pure literal L in S : if S contains no clauses of the form $\tilde{L} \vee C$.