Propositional Satisfiability Solver (DPLL)

1. Table for number of Nodes searched for different problems

	Backtracking Alone	Backtracking with Unit Clause heuristic	Backtracking with Unit Clause and Pure Symbol heuristics
Abstract Boolean Problem (example1.kb)	55	33	13
Builder agent problem with argument - "painter sander gluer joiner"	574	38	35
Builder agent problem with argument - "cutter welder painter joiner recharger"	69	15	17

2. Transcripts

Example1.kb with Backtracking, Pure Symbol and Unit Clause heuristics:

Initial Clauses:

0: (-a v g v -f)

1: (-a v -h v -b)

2: (a v c)

3: (a v -i v -l)

4: (a v -k v -j)

5: (b v d)

6: (b v g v -n)

7: (n v k v b v -f)

8: (-c v k)

9: (-i v -c v -k v l)

```
10: ( h v c v -m v n )
11: ( c v l )
12: ( -k v d v l )
13: ( d v -g v l )
14: ( o v -g v n )
15: ( h v -j v -o v n )
16: ( -i v j )
17: ( -m v -d v -l )
18: ( m v -e v -n )
19: ( i v h v -f )
The Input Facts
Props: ['a', 'c', 'b', 'e', 'd', 'g', 'f', 'i', 'h', 'k', 'j', 'm', 'l', 'o', 'n']
------------------------
Pure Symbol on e= False
Model= {'e': False}
Pure Symbol on f= False
Model= {'e': False, 'f': False}
Pure Symbol on i= False
Model= {'i': False, 'e': False, 'f': False}
Pure Symbol on j= False
Model= {'i': False, 'j': False, 'e': False, 'f': False}
Pure Symbol on m= False
Model= {'i': False, 'm': False, 'j': False, 'e': False, 'f': False}
Pure Symbol on d= True
Model= {'e': False, 'd': True, 'f': False, 'i': False, 'j': False, 'm': False}
Pure Symbol on h= False
Model= {'e': False, 'd': True, 'f': False, 'i': False, 'h': False, 'j': False, 'm': False}
Pure Symbol on a= True
Model= {'a': True, 'e': False, 'd': True, 'f': False, 'i': False, 'h': False, 'j': False, 'm': False}
```

```
Pure Symbol on b= True

Model= {'a': True, 'b': True, 'e': False, 'd': True, 'f': False, 'i': False, 'h': False, 'j': False, 'm': False}

Pure Symbol on g= False

Model= {'a': True, 'b': True, 'e': False, 'd': True, 'g': False, 'f': False, 'i': False, 'h': False, 'j': False, 'm': False}

Pure Symbol on k= True

Model= {'a': True, 'b': True, 'e': False, 'd': True, 'g': False, 'f': False, 'i': False, 'h': False, 'k': True, 'j': False,
'm': False}

Pure Symbol on c= True

Model= {'a': True, 'c': True, 'b': True, 'e': False, 'd': True, 'g': False, 'f': False, 'i': False, 'h': False, 'k': True,
'j': False, 'm': False}

{'a': True, 'c': True, 'b': True, 'e': False, 'd': True, 'g': False, 'f': False, 'i': False, 'h': False, 'k': True, 'j': False,
'm': False}

------------------------

Nodes Searched: 13

Full Solution

a = True

c = True

b = True

e = False

d = True

g = False

f = False

i = False

h = False

k = True

j = False

m = False

l = True/False

o = True/False

n = True/False
```

Multi-agent Task-assignment problem with input as "painter sander gluer joiner" and with backtracking, pure symbol and unit clause heuristics

Initial Clauses:

0: (a v c v e v g v -painter)

1: (a v b v e v g v f v -stapler)

2: (a v g v d v -recharger v f)

3: (a v b v e v d v f v -welder)

4: (h v c v b v -cutter)

5: (-sander v b v d)

6: (h v -gluer v g)

7: (-joiner v f)

8: (-a v -c v -b v -d)

9: (-a v -c v -b v -e)

10: (-a v -c v -b v -f)

11: (-a v -c v -b v -g)

12: (-a v -h v -c v -b)

13: (-a v -b v -e v -d)

14: (-a v -b v -d v -f)

15: (-a v -b v -d v -g)

16: (-a v -h v -b v -d)

17: (-a v -b v -e v -f)

18: (-a v -b v -e v -g)

19: (-a v -h v -b v -e)

20: (-a v -b v -g v -f)

21: (-a v -h v -b v -f)

22: (-a v -h v -b v -g)

23: (-a v -c v -e v -d)

24: (-a v -c v -d v -f)

25: (-a v -c v -d v -g)

26: (-a v -h v -c v -d)

27: (-a v -c v -e v -f)

28: (-a v -c v -e v -g)

29: (-a v -h v -c v -e)

30: (-a v -c v -g v -f)

31: (-a v -h v -c v -f)

32: (-a v -h v -c v -g)

33: (-a v -e v -d v -f)

34: (-a v -e v -d v -g)

35: (-a v -h v -e v -d)

36: (-a v -d v -g v -f)

37: (-a v -h v -d v -f)

38: (-a v -h v -d v -g)

39: (-a v -e v -g v -f)

40: (-a v -h v -e v -f)

41: (-a v -h v -e v -g)

42: (-a v -h v -g v -f)

43: (-c v -b v -e v -d)

44: (-c v -b v -d v -f)

45: (-c v -b v -d v -g)

46: (-h v -c v -b v -d)

47: (-c v -b v -e v -f)

48: (-c v -b v -e v -g)

49: (-h v -c v -b v -e)

50: (-c v -b v -g v -f)

51: (-h v -c v -b v -f)

52: (-h v -c v -b v -g)

53: (-b v -e v -d v -f)

54: (-b v -e v -d v -g)

55: (-h v -b v -e v -d)

56: (-b v -d v -g v -f)

57: (-h v -b v -d v -f)

58: (-h v -b v -d v -g)

59: (-b v -e v -g v -f)

60: (-h v -b v -e v -f)

61: (-h v -b v -e v -g)

62: (-h v -b v -g v -f)

63: (-c v -e v -d v -f)

64: (-c v -e v -d v -g)

65: (-h v -c v -e v -d)

66: (-c v -d v -g v -f)

67: (-h v -c v -d v -f)

68: (-h v -c v -d v -g)

69: (-c v -e v -g v -f)

70: (-h v -c v -e v -f)

71: (-h v -c v -e v -g)

72: (-h v -c v -g v -f)

73: (-e v -d v -g v -f)

74: (-h v -e v -d v -f)

75: (-h v -e v -d v -g)

76: (-h v -d v -g v -f)

77: (-h v -e v -g v -f)

The Input Facts

78: (painter)

79: (sander)

80: (gluer)

```
81: ( joiner )

Props: ['a', 'sander', 'c', 'b', 'stapler', 'd', 'g', 'f', 'h', 'gluer', 'joiner', 'recharger', 'e', 'welder', 'painter',
'cutter']

------------------------

Pure Symbol on stapler= False

Model= {'stapler': False}

Pure Symbol on welder= False

Model= {'welder': False, 'stapler': False}

Pure Symbol on recharger= False

Model= {'welder': False, 'stapler': False, 'recharger': False}

Pure Symbol on cutter= False

Model= {'welder': False, 'stapler': False, 'recharger': False, 'cutter': False}

Unit Clause on painter gives  painter=True

Model= {'welder': False, 'stapler': False, 'painter': True, 'recharger': False, 'cutter': False}

Unit Clause on sander gives  sander=True

Model= {'sander': True, 'stapler': False, 'recharger': False, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on gluer gives  gluer=True

Model= {'sander': True, 'stapler': False, 'gluer': True, 'recharger': False, 'welder': False, 'painter': True,
'cutter': False}

Unit Clause on joiner gives  joiner=True

Model= {'sander': True, 'joiner': True, 'stapler': False, 'gluer': True, 'recharger': False, 'welder': False,
'painter': True, 'cutter': False}

Unit Clause on -joiner v f gives  f=True

Model= {'sander': True, 'joiner': True, 'stapler': False, 'f': True, 'gluer': True, 'recharger': False, 'welder':
False, 'painter': True, 'cutter': False}

trying a = True

Model= {'a': True, 'sander': True, 'welder': False, 'stapler': False, 'f': True, 'gluer': True, 'painter': True,
'joiner': True, 'recharger': False, 'cutter': False}

Pure Symbol on c= False

Model= {'a': True, 'sander': True, 'c': False, 'welder': False, 'stapler': False, 'f': True, 'gluer': True, 'painter':
True, 'joiner': True, 'recharger': False, 'cutter': False}
```

```
Pure Symbol on e= False

Model= {'a': True, 'sander': True, 'c': False, 'welder': False, 'stapler': False, 'f': True, 'gluer': True, 'painter':
True, 'e': False, 'joiner': True, 'recharger': False, 'cutter': False}

trying b = True

Model= {'a': True, 'sander': True, 'c': False, 'joiner': True, 'stapler': False, 'f': True, 'gluer': True, 'b': True,
'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Pure Symbol on d= False

Model= {'a': True, 'sander': True, 'c': False, 'joiner': True, 'stapler': False, 'd': False, 'f': True, 'gluer': True,
'b': True, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on -a v -b v -g v -f gives  g=False

Model= {'a': True, 'sander': True, 'c': False, 'joiner': True, 'stapler': False, 'd': False, 'g': False, 'f': True,
'gluer': True, 'b': True, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on h v -gluer v g gives  h=True

Model= {'a': True, 'sander': True, 'c': False, 'joiner': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'h':
True, 'gluer': True, 'b': True, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Clause: -a v -h v -b v -f Fails

backtracking

trying b = False

Model= {'a': True, 'sander': True, 'c': False, 'joiner': True, 'stapler': False, 'f': True, 'gluer': True, 'b': False,
'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on -sander v b v d gives  d=True

Model= {'a': True, 'sander': True, 'c': False, 'joiner': True, 'stapler': False, 'd': True, 'f': True, 'gluer': True,
'b': False, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on -a v -d v -g v -f gives  g=False

Model= {'a': True, 'sander': True, 'c': False, 'joiner': True, 'stapler': False, 'd': True, 'g': False, 'f': True,
'gluer': True, 'b': False, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on h v -gluer v g gives  h=True

Model= {'a': True, 'sander': True, 'c': False, 'joiner': True, 'stapler': False, 'd': True, 'g': False, 'f': True, 'h':
True, 'gluer': True, 'b': False, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Clause: -a v -h v -d v -f Fails

backtracking

trying a = False
```

```
Model= {'a': False, 'sander': True, 'welder': False, 'stapler': False, 'f': True, 'gluer': True, 'painter': True,
'joiner': True, 'recharger': False, 'cutter': False}

trying b = True

Model= {'a': False, 'sander': True, 'joiner': True, 'stapler': False, 'f': True, 'gluer': True, 'b': True,
'recharger': False, 'welder': False, 'painter': True, 'cutter': False}

Pure Symbol on d= False

Model= {'a': False, 'sander': True, 'joiner': True, 'stapler': False, 'd': False, 'f': True, 'gluer': True, 'b': True,
'recharger': False, 'welder': False, 'painter': True, 'cutter': False}

trying c = True

Model= {'a': False, 'sander': True, 'c': True, 'b': True, 'stapler': False, 'd': False, 'f': True, 'gluer': True,
'painter': True, 'welder': False, 'joiner': True, 'recharger': False, 'cutter': False}

Pure Symbol on e= False

Model= {'a': False, 'sander': True, 'c': True, 'b': True, 'stapler': False, 'd': False, 'f': True, 'gluer': True,
'painter': True, 'welder': False, 'e': False, 'joiner': True, 'recharger': False, 'cutter': False}

Unit Clause on -c v -b v -g v -f gives  g=False

Model= {'a': False, 'sander': True, 'c': True, 'b': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'gluer':
True, 'painter': True, 'welder': False, 'e': False, 'joiner': True, 'recharger': False, 'cutter': False}

Unit Clause on h v -gluer v g gives  h=True

Model= {'a': False, 'sander': True, 'c': True, 'b': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'h': True,
'gluer': True, 'painter': True, 'welder': False, 'e': False, 'joiner': True, 'recharger': False, 'cutter': False}

Clause: -h v -c v -b v -f Fails

backtracking

trying c = False

Model= {'a': False, 'sander': True, 'c': False, 'b': True, 'stapler': False, 'd': False, 'f': True, 'gluer': True,
'painter': True, 'welder': False, 'joiner': True, 'recharger': False, 'cutter': False}

trying e = True

Model= {'a': False, 'sander': True, 'c': False, 'b': True, 'stapler': False, 'd': False, 'f': True, 'gluer': True,
'joiner': True, 'recharger': False, 'e': True, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on -b v -e v -g v -f gives  g=False

Model= {'a': False, 'sander': True, 'c': False, 'b': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'gluer':
True, 'joiner': True, 'recharger': False, 'e': True, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on h v -gluer v g gives  h=True
```

```
Model= {'a': False, 'sander': True, 'c': False, 'b': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'h': True, 'gluer': True, 'joiner': True, 'recharger': False, 'e': True, 'welder': False, 'painter': True, 'cutter': False}

Clause: -h v -b v -e v -f Fails

backtracking

trying e = False

Model= {'a': False, 'sander': True, 'c': False, 'b': True, 'stapler': False, 'd': False, 'f': True, 'gluer': True, 'joiner': True, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Unit Clause on a v c v e v -painter v g gives  g=True

Model= {'a': False, 'sander': True, 'c': False, 'b': True, 'stapler': False, 'd': False, 'g': True, 'f': True, 'gluer': True, 'joiner': True, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

Pure Symbol on h= False

Model= {'a': False, 'sander': True, 'c': False, 'b': True, 'stapler': False, 'd': False, 'g': True, 'f': True, 'h': False, 'gluer': True, 'joiner': True, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

{'a': False, 'sander': True, 'c': False, 'b': True, 'stapler': False, 'd': False, 'g': True, 'f': True, 'h': False, 'gluer': True, 'joiner': True, 'recharger': False, 'e': False, 'welder': False, 'painter': True, 'cutter': False}

------------------------

Nodes Searched: 35

Full Solution

a = False

sander = True

c = False

b = True

stapler = False

d = False

g = True

f = True

h = False

gluer = True

joiner = True
```

```
recharger = False
e = False
welder = False
painter = True
cutter = False
------------------------
Agent Team: b g f
```

Multi-agent Task-assignment problem with input as "cutter welder painter joiner recharger" and with backtracking, pure symbol and unit clause heuristics

```
 Initial Clauses:
0: ( a v c v e v g v -painter )
1: ( a v b v e v g v f v -stapler )
2: ( a v g v d v -recharger v f )
3: ( a v b v e v d v f v -welder )
4: ( h v c v b v -cutter )
5: ( -sander v b v d )
6: ( h v -gluer v g )
7: ( -joiner v f )
8: ( -a v -c v -b v -d )
9: ( -a v -c v -b v -e )
10: ( -a v -c v -b v -f )
11: ( -a v -c v -b v -g )
12: ( -a v -h v -c v -b )
13: ( -a v -b v -e v -d )
14: ( -a v -b v -d v -f )
15: ( -a v -b v -d v -g )
16: ( -a v -h v -b v -d )
17: ( -a v -b v -e v -f )
```

18: (-a v -b v -e v -g)

19: (-a v -h v -b v -e)

20: (-a v -b v -g v -f)

21: (-a v -h v -b v -f)

22: (-a v -h v -b v -g)

23: (-a v -c v -e v -d)

24: (-a v -c v -d v -f)

25: (-a v -c v -d v -g)

26: (-a v -h v -c v -d)

27: (-a v -c v -e v -f)

28: (-a v -c v -e v -g)

29: (-a v -h v -c v -e)

30: (-a v -c v -g v -f)

31: (-a v -h v -c v -f)

32: (-a v -h v -c v -g)

33: (-a v -e v -d v -f)

34: (-a v -e v -d v -g)

35: (-a v -h v -e v -d)

36: (-a v -d v -g v -f)

37: (-a v -h v -d v -f)

38: (-a v -h v -d v -g)

39: (-a v -e v -g v -f)

40: (-a v -h v -e v -f)

41: (-a v -h v -e v -g)

42: (-a v -h v -g v -f)

43: (-c v -b v -e v -d)

44: (-c v -b v -d v -f)

45: (-c v -b v -d v -g)

46: (-h v -c v -b v -d)

47: (-c v -b v -e v -f)

48: (-c v -b v -e v -g)

49: (-h v -c v -b v -e)

50: (-c v -b v -g v -f)

51: (-h v -c v -b v -f)

52: (-h v -c v -b v -g)

53: (-b v -e v -d v -f)

54: (-b v -e v -d v -g)

55: (-h v -b v -e v -d)

56: (-b v -d v -g v -f)

57: (-h v -b v -d v -f)

58: (-h v -b v -d v -g)

59: (-b v -e v -g v -f)

60: (-h v -b v -e v -f)

61: (-h v -b v -e v -g)

62: (-h v -b v -g v -f)

63: (-c v -e v -d v -f)

64: (-c v -e v -d v -g)

65: (-h v -c v -e v -d)

66: (-c v -d v -g v -f)

67: (-h v -c v -d v -f)

68: (-h v -c v -d v -g)

69: (-c v -e v -g v -f)

70: (-h v -c v -e v -f)

71: (-h v -c v -e v -g)

72: (-h v -c v -g v -f)

73: (-e v -d v -g v -f)

74: (-h v -e v -d v -f)

75: (-h v -e v -d v -g)

```
76: ( -h v -d v -g v -f )

77: ( -h v -e v -g v -f )

The Input Facts

78: ( cutter )

79: ( welder )

80: ( painter )

81: ( joiner )

82: ( recharger )

Props: ['a', 'sander', 'c', 'b', 'stapler', 'd', 'g', 'f', 'h', 'gluer', 'joiner', 'recharger', 'e', 'welder', 'painter', 'cutter']

------------------------

Pure Symbol on sander= False

Model= {'sander': False}

Pure Symbol on stapler= False

Model= {'sander': False, 'stapler': False}

Pure Symbol on gluer= False

Model= {'sander': False, 'gluer': False, 'stapler': False}

Unit Clause on cutter gives  cutter=True

Model= {'sander': False, 'gluer': False, 'stapler': False, 'cutter': True}

Unit Clause on welder gives  welder=True

Model= {'sander': False, 'gluer': False, 'welder': True, 'stapler': False, 'cutter': True}

Unit Clause on painter gives  painter=True

Model= {'sander': False, 'stapler': False, 'gluer': False, 'welder': True, 'painter': True, 'cutter': True}

Unit Clause on joiner gives  joiner=True

Model= {'sander': False, 'joiner': True, 'stapler': False, 'gluer': False, 'welder': True, 'painter': True, 'cutter': True}

Unit Clause on -joiner v f gives  f=True

Model= {'sander': False, 'joiner': True, 'stapler': False, 'f': True, 'gluer': False, 'welder': True, 'painter': True, 'cutter': True}

Pure Symbol on d= False
```

```
Model= {'sander': False, 'joiner': True, 'stapler': False, 'd': False, 'f': True, 'gluer': False, 'welder': True,
'painter': True, 'cutter': True}

Pure Symbol on recharger= True

Model= {'sander': False, 'joiner': True, 'stapler': False, 'd': False, 'f': True, 'gluer': False, 'recharger': True,
'welder': True, 'painter': True, 'cutter': True}

trying a = True

Model= {'a': True, 'sander': False, 'welder': True, 'stapler': False, 'd': False, 'f': True, 'gluer': False,
'painter': True, 'joiner': True, 'recharger': True, 'cutter': True}

Pure Symbol on e= False

Model= {'a': True, 'sander': False, 'welder': True, 'stapler': False, 'd': False, 'f': True, 'gluer': False,
'painter': True, 'e': False, 'joiner': True, 'recharger': True, 'cutter': True}

Pure Symbol on g= False

Model= {'a': True, 'sander': False, 'welder': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'gluer':
False, 'painter': True, 'e': False, 'joiner': True, 'recharger': True, 'cutter': True}

trying b = True

Model= {'a': True, 'sander': False, 'joiner': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'gluer': False,
'b': True, 'recharger': True, 'e': False, 'welder': True, 'painter': True, 'cutter': True}

Pure Symbol on h= False

Model= {'a': True, 'sander': False, 'joiner': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'h': False,
'gluer': False, 'b': True, 'recharger': True, 'e': False, 'welder': True, 'painter': True, 'cutter': True}

Pure Symbol on c= False

Model= {'a': True, 'sander': False, 'c': False, 'joiner': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'h':
False, 'gluer': False, 'b': True, 'recharger': True, 'e': False, 'welder': True, 'painter': True, 'cutter': True}

{'a': True, 'sander': False, 'c': False, 'joiner': True, 'stapler': False, 'd': False, 'g': False, 'f': True, 'h': False,
'gluer': False, 'b': True, 'recharger': True, 'e': False, 'welder': True, 'painter': True, 'cutter': True}

------------------------

Nodes Searched: 17

Full Solution

a = True

sander = False

c = False

joiner = True
```

```
stapler = False

d = False

g = False

f = True

h = False

gluer = False

b = True

recharger = True

e = False

welder = True

painter = True

cutter = True

------------------------

Agent Team: a f b
```