Satisfiability Encodings

Introduction to
Artificial Intelligence
COS302
Michael L. Littman
Fall 2001
Administration

Questions?

Clarify 3-CNF conversion?
RISC AI

Much design and implementation work has gone into creating SAT solvers, like chaff.

By “compiling” problems to SAT, we take advantage of this effort.
3-Coloring as SAT

Arguably fastest way to solve coloring problems.

We are given a graph with a set of nodes $N$ and edges $E$.

One encoding:

For each node $i$, three Boolean variables $R_i, B_i, G_i$ representing color.
Coloring Clauses

No conflicting colors:

- All \((i, j)\):
  \((\neg R_i + \neg R_j)(\neg B_i + \neg B_j)(\neg G_i + \neg G_j)\)

No uncolored nodes:

- All \(i\): \((R_i + B_i + G_i)\)

No multicolored nodes:

- All \(i\): \((\neg R_i + \neg B_i)(\neg R_i + \neg G_i)(\neg B_i + \neg G_i)\)
General CSPs

CSP: V, D, C

For all v in V and d in D, Boolean variable A(v,d) that says we are assigning variable v value d.

For each entry e in each legal assignment list, a variable L(e) that says we’ve chosen the corresponding legal assignment.
CSP Clauses

Must assign each variable:
- All \( v \): \((A(v,d1)+A(v,d2)+...\))

Can’t assign multiple values:
- All \( v, d1, d2 \): \((\sim A(v,d1)+\sim A(v,d2))\)

Must pick a legal assignment:
- All constraints: \((L(e1)+L(e2)+...\))

Apply chosen assignment:
- All \( e, v \): \((\sim L(e)+A(v,d(e,v)))\)
Battleships

Fleet hiding in a 10x10 grid (section of ocean).

1  ◀  □ □  ▶  Battleship
2  ◀  □  ▶  Cruisers
3  ◀  ▶  ▶  Destroyers
4  ●  ▶  Submarines
Battleship Rules

Ships can be oriented horizontally or vertically.

Ships may not intersect or share adjacent grid cells, even diagonally.

Digits in rows and columns represent total number of grid cells occupied by vessels.
Example Grid
Objects

Rows: $i \in \{1, \ldots, 10\}$
Columns: $j \in \{1, \ldots, 10\}$
Parts: $\{ \triangleleft, \triangleright, \uparrow, \downarrow, \square, \bullet \}$
Direction: across (0), down (1)
Ship No.: $k \in \{1, 2, 3, 4\}$
Variables

Filled: \( f(i,j) = i,j \) contains ship
Part: \( p_a(i,j) = \text{part a appears at } i,j \)
Battleship: \( b_i(i), b_j(j), b_d \) encodes the position & direction of ship
Cruisers: \( c_i(k,i), c_j(k, j), c_d(k) \) ditto
Ditto for Destroyers and Submarines. No. of vars?
### Initial Board

**Ship parts encoded via pa, pb, pc:**

<table>
<thead>
<tr>
<th>pa</th>
<th>pb</th>
<th>pc</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Initial Board Clauses

Example:
\[ f(5,8) \sim pa(5,8) \sim pb(5,8) \sim pc(5,8) \]
\[ f(7,2) \sim pa(7,2) \quad pb(7,2) \quad pc(7,2) \]
\[ f(10,5) \sim pa(10,5) \sim pb(10,5) \sim pc(10,5) \]

Total constraints: Less than 10x10x4 (400). More like 30.
Footprints

Destroyer #2 at 6,5 going across means no ship part at 5,4.

di(2,6) dj(2,5) ~dd(2) → ~f(5,4)

~di(2,6) + ~dj(2,5) + dd(2) + ~f(5,4)

1x2x7x10x18+ 2x2x8x10x15+
3x2x9x10x12+ 4x2x10x10x9

= 17400 constraints
Row, Col. Constraints

Each row and column (20) must have the number of filled squares sum to the number indicated. We can introduce a set of variables (3x20=60) to represent the sum. Then a set of clauses (3x20=60) match the actual total to the desired total.
Computing a Sum

Boolean variables $x_1, \ldots, x_n$.

$S(i, t)$ represents if $x_1 + \ldots + x_i = t$.

$S(1, 1) \equiv x_1$, $S(1, 0) \equiv \neg x_1$, $\neg S(1, t)$

for $1 < t \leq n$. For $i > 1$, $t$

$S(i-1, t) \; x_i \rightarrow \; S(i, t+1)$

$S(i-1, t) \; \neg x_i \rightarrow \; S(i, t)$

$\neg S(i-1, t) \; x_i \rightarrow \; \neg S(i, t+1)$

$\neg S(i-1, t) \; \neg x_i \rightarrow \; \neg S(i, t)$
Other Options

Leads to $20 \times 8 \times 9 = 1440$ new variables and $20 \times 4 \times 8 \times 9 = 5760$ new constraints.

Could implement an efficient arithmetic circuit (Boolean gates).

If $n$ small, map directly to count.
All Else Empty

How can you say that any place that’s not a boat is water?

Don’t need to: If there were extra boat pieces, the row/col sums would be off.
Practical Considerations

Ultimately, the formula has 2586 vars and 150k constraints. Zoinks!

Michail Lagoudakis at Duke simplified it to 306 vars. 28k constraints. Tough, but much easier.
Other Things to Try

15 puzzle
* Mastermind
Cross sums
N-queens (HW)
What to Learn

Some tricks for generating SAT encodings of search problems.
Homework 3 (due 10/10)

1. Let \( f = \sim(x + \sim y(\sim x + z)) \). 
   (a) Write out the truth table for \( f \). 
   (b) Convert the truth table to CNF. 
   (c) Show the series of steps DPLL makes while solving the resulting formula. 
   Assume variables chosen for splitting in the order \( x, y, z \). 

2. Using the same \( f \) from the first part, follow the 3-CNF conversion algorithm to create an equivalent 3-CNF formula. 

3. Describe how to encode \( n \)-queens as a SAT problem.