Parallelizing SAT Solver

6.338 Applied Parallel Computing
Hank Huang
5/13/2009

- The SAT problem:
- Given a formula made of boolean variables and operators, find an assignment to the variables that makes it true:
  - i.e. (P v Q) ^ (~P v R)
  - A solution: {P = false, Q = true, R = false}
- SAT is NP-Complete (Hard)

What is SAT?

- Conjunctive Normal Form is a set of clauses, each containing a set of literals
- Clauses are “Anded” with one another
- Literals are “Ored” with one another in a clause
- An efficient SAT solver takes a formula in CNF and returns an assignment as solution or says none exists

SAT Solver

More on SAT Solver

- Naïve Solve:
  - Enumerate assignments and check formula for each assignment
  - For k variables, 2^k assignments!
- DPLL/Davis-Putnam-Logemann-Loveland
  - Back-tracking
  - Unit propagation
  - Pure literal elimination

DPLL Algorithm

- function DPLL(Φ, Env):
  - if Φ is empty
    - return Env;
  - if Φ contains an empty clause
    - return null;
  - for every unit clause l in Φ
    - Φ=unit-propagate(Φ,Env);
  - for every literal l that occurs pure in Φ
    - Φ=pure-literal-assign(l, Φ, Env);
  - l := choose-literal(Φ);
  - Env2 = DPLL(Φ,assign(l, Env));
  - If Env2 is null
    - return DPLL(Φ,assign(not(l), Env));
  - return Env2
- Env is a map of assignments

5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

A sudoku puzzle...

5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

...and its solution numbers marked in red

Sudoku Puzzle

- Rules of the game
  - No two same digit can appear in a single column
  - No two same digit can appear in a single row
  - No two same digit can appear in a single sub-grid
  - Exactly one number must occupy each cell

Sudoku Puzzle

- For each column, row, sub-grid, cell:
  - An At-least clause
  - An At-most clause
  - Together enforces exactly once behavior
- Example:
  - An At-Least clause: [119 129 139 149 159 169 179 189 199].
  - An associated series of At-Most clause:
  - [-119 -129], [-119 -139], [-119 -149]…[-119 -199]
  - [-129 -139],[-129 -139]…[-129 -199]
  - …
  - [-189 -199]
  - These 37 clauses together ensure that “9” appears exactly once in row 1.

Reduction

Operations

- Unit Propagation:
  - (X)(X v Y v Z)(~X v Y)(Y Z ~W) (~Z W) →(Y)(Y Z ~W)(~Z W)
  - (Y)(Y Z ~W)(~Z W) → (~Z W)
  - Recursively searches the problem for a unit clause then simplify the problem accordingly in serial
- Pure Literal Elimination
  - Tracks whether a literal has become pure at each step of recursion
  - If so, simplify the problem accordingly

Parallelization

- Spread the problem across nodes
- Have each operations performed on sub-problem in parallel
- Obtain new sub-problem, and recurse

Performance Comparison:

Puzzle Number	Number of Recursion Steps	Performance in milliseconds		
		Java	Matlab Serial	Matlab Star-P
1	145	76	89	4105
2	219	88	101	3809
3	732	148	168	5672
4	2644	465	481	12691
5	4052	858	912	19780
6	8234	1485	1674	24536

Performance

- Sudoku is too small, too easy
  - Computation requires is light
  - Does not justify for communication cost
- Bigger problem more likely to see benefits of parallelization

Conclusion