Defn: A Boolean expression is in DNF (disjunctive normal form) if it can be written as $D_1 \vee D_2 \vee D_3 \cdots\cdots \vee D_n$, where each D_i is conjunction of literals.

Example

$x_1 x_2 \vee x_3 \vee \neg x_4 x_5$

Exercise Show that every Boolean expression can be converted to equivalent DNF expression.

All conversions discussed so far hold in every BA, (because we used rules which hold in every BA in these conversions)

* In CS, we usually consider BA 2, the BA with the elements $\{0, 1\}$ only.

$e = e_1 \vee e_2 \rightarrow$ trivial

$e = e_1 \wedge e_2$

$$= \bigvee_{i,j} (A_i \wedge B_j)$$ (Prove by induction)

Example

Given a Boolean expression convert it into sum of products.

x_1	x_2	x_3	. - - - - - .	x_n	$f(x_1, x_2, x_3 \ldots x_n)$
0				0	1
0				1	0
.	1	.		.	0
.					.
1	1				.
1		1		1	.
.		1		1	.
	1			1	.
1	1	1		1	0

2^n complexity.

* In what follows, we will also restrict to 2 element BA

Def^n A formula ϕ is said to be satisfiable iff $\exists$ valuation

$v : [[\cdot]]_2 : var \longrightarrow \{0, 1\}$ s.t.

$v(\phi) = [[\phi]]_2 = 1$

OBSERVATION ϕ is unsatisfiable iff $\neg\phi$ is valid.

⇒ In general, we don't know better method than brute force checking a formula for every valuation, for solving satisfiability problem.

⇒ Satisfiability checking problem is of interest from CS perspective

⇒ There are many other problems which can be solved efficiently iff SAT can be solved efficiently.

for example

(i) Hamiltonian cycle problem in graph.

ie existance of a cycle in the input graph that goes through each vertex exactly once.

(ii) K-clique problem

(G, K) if G has a complete subgraph of size K.

⇒ these are decision problems : Answer is Yes/No.

And efficiently solvable ≡ worst case time complexity is polynomial in size of input.

CLASS NP (Non deterministic polynomial time)

A problem in class NP

Input x

output is yes iff ∃

$\underbrace{|y| \le P(x)}_{\text{guess}} \quad \underbrace{[g(x, y)]}_{\text{Boolean condition (verify)}}$

$|x|$ is size

SAT, HAM, Clique are all hardest problems in NP, in a sense.

NP-complete problems

$P \stackrel{?}{=} NP$

Special case of SAT can be solved efficiently...

Example input ϕ is a DNF

ϕ is unsatisfiable iff every disjunct D, there is a variable p such that $p, \neg p \in D$

$\Longrightarrow$ Proof by contraposition

If there is a disjunct D_i s.t. for no P, it has $p, \neg p \in D_i$

Define a valuation v s.t.

$$v(i) = \begin{cases} 1 & \text{if } q \in D_i \\ 0 & \text{if } \neg q \in D_i \end{cases}$$

v satisfies D_i or ϕ.

CNF satisfiability is NP complete
3-CNF (each clause can have atmost 3-literals) is also NP complete

---

Horn clause satisfiability is efficiently solvable $O(n)$

A Horn clause is of the form

$p_1 \wedge p_2 \wedge p_3 \ldots\ldots \wedge p_n \longrightarrow p$

p_i's and p are variables (positive literals)

Rule based systems $\Longrightarrow$

$p_1, p_2, p_3 \ldots\ldots p_n \longrightarrow p$

$q_1, q_2, q_3 \ldots\ldots q_m \longrightarrow q$

is the set of rules consistent?