CS 512, Spring 2017, Handout 10

Propositional Logic:

Conjunctive Normal Forms,
Disjunctive Normal Forms,
Horn Formulas,
and other special forms

Assaf Kfoury

5 February 2017

conjunctive normal form & disjunctive normal form

conjunctive normal form & disjunctive normal form

CNF

$L ::= p \mid \neg p$	literal
$D ::= L \mid L \vee D$	disjuntion of literals
$C ::= D \mid D \wedge C$	conjunction of disjunctions

conjunctive normal form & disjunctive normal form

CNF

$L \ ::= p \ \mid \neg p$	literal
$D ::= L \mid L \vee D$	disjuntion of literals
$C ::= D \mid D \wedge C$	conjunction of disjunctions

DNF

$L \ ::= p \ \mid \neg p$	literal
$C ::= L \mid L \wedge C$	conjunction of literals
$D ::= C \mid C \vee D$	disjunction of conjunctions

Why CNF?

Why CNF?

- A disjunction of literals $L_1 \vee \cdots \vee L_m$ is **valid** (equivalently, is a **tautology**) iff there are $1 \leqslant i, j \leqslant m$ with $i \neq j$ such that L_i is $\neg L_j$.

Why CNF?

- A disjunction of literals $L_1 \vee \cdots \vee L_m$ is **valid** (equivalently, is a **tautology**) iff there are $1 \leqslant i, j \leqslant m$ with $i \neq j$ such that L_i is $\neg L_j$.
- A conjunction of disjunctions $D_1 \wedge \cdots \wedge D_n$ is **valid** (equivalently, is a **tautology**) iff for every $1 \leqslant i \leqslant n$ it is the case that D_i is valid.

Why CNF?

- A disjunction of literals $L_1 \vee \cdots \vee L_m$ is **valid** (equivalently, is a **tautology**) iff there are $1 \leqslant i, j \leqslant m$ with $i \neq j$ such that L_i is $\neg L_j$.
- A conjunction of disjunctions $D_1 \wedge \cdots \wedge D_n$ is **valid** (equivalently, is a **tautology**) iff for every $1 \leqslant i \leqslant n$ it is the case that D_i is valid.
- **CNF** allows for a fast and easy syntactic test of **validity**.

Why CNF?

- A disjunction of literals $L_1 \vee \cdots \vee L_m$ is **valid** (equivalently, is a **tautology**) iff there are $1 \leqslant i, j \leqslant m$ with $i \neq j$ such that L_i is $\neg L_j$.
- A conjunction of disjunctions $D_1 \wedge \cdots \wedge D_n$ is **valid** (equivalently, is a **tautology**) iff for every $1 \leqslant i \leqslant n$ it is the case that D_i is valid.
- **CNF** allows for a fast and easy syntactic test of **validity**.
- Unfortunately, conversion into **CNF** may lead to exponential blow-up:

$$(x_1 \wedge y_1) \vee (x_2 \wedge y_2) \vee \cdots \vee (x_n \wedge y_n) \quad \text{becomes}$$
$$(x_1 \vee \cdots \vee x_{n-1} \vee x_n) \wedge (x_1 \vee \cdots \vee x_{n-1} \vee y_n) \wedge \cdots \wedge (y_1 \vee \cdots \vee y_{n-1} \vee y_n)$$

  i.e., the initial WFF of size $\mathcal{O}(n)$ becomes an equivalent WFF of size $\mathcal{O}(2^n)$, because each clause in the latter contains either x_i or y_i for every i.

- Converting a WFF into an equivalent WFF in **CNF**, preserving **validity**, is NP-hard!

  (However, converting a WFF into another WFF, not necessarily equivalent, preserving **satisfiability** can be carried out in linear time – more in a later handout.)

Why DNF?

Why DNF?

- A conjunction of literals $L_1 \wedge \cdots \wedge L_m$ is **satisfiable** iff $\{L_1, \ldots, L_m\}$ does not include both a propositional atom P and its negation $\neg P$.

Why DNF?

- A conjunction of literals $L_1 \wedge \cdots \wedge L_m$ is **satisfiable** iff $\{L_1, \ldots, L_m\}$ does not include both a propositional atom P and its negation $\neg P$.
- A disjunction of conjunctions $C_1 \vee \cdots \vee C_n$ is **satisfiable** iff there is some $1 \leqslant i \leqslant n$ such that C_i is satisfiable.

Why DNF?

- A conjunction of literals $L_1 \wedge \cdots \wedge L_m$ is **satisfiable** iff $\{L_1, \ldots, L_m\}$ does not include both a propositional atom P and its negation $\neg P$.
- A disjunction of conjunctions $C_1 \vee \cdots \vee C_n$ is **satisfiable** iff there is some $1 \leqslant i \leqslant n$ such that C_i is satisfiable.
- **DNF** allows for a fast and easy syntactic test of **satisfiability**.

Why DNF?

- A conjunction of literals $L_1 \wedge \cdots \wedge L_m$ is **satisfiable** iff $\{L_1, \ldots, L_m\}$ does not include both a propositional atom P and its negation $\neg P$.
- A disjunction of conjunctions $C_1 \vee \cdots \vee C_n$ is **satisfiable** iff there is some $1 \leqslant i \leqslant n$ such that C_i is satisfiable.
- **DNF** allows for a fast and easy syntactic test of **satisfiability**.
- Unfortunately, conversion into **DNF** may lead to exponential blow-up:

$$(x_1 \vee y_1) \wedge (x_2 \vee y_2) \wedge \cdots \wedge (x_n \vee y_n) \quad \text{becomes}$$
$$(x_1 \wedge \cdots \wedge x_{n-1} \wedge x_n) \vee (x_1 \wedge \cdots \wedge x_{n-1} \wedge y_n) \vee \cdots \vee (y_1 \wedge \cdots \wedge y_{n-1} \wedge y_n)$$

  i.e., the initial WFF of size $\mathcal{O}(n)$ becomes an equivalent WFF of size $\mathcal{O}(2^n)$, because each clause in the latter contains either x_i or y_i for every i.

- Converting a WFF into an equivalent WFF in **DNF**, preserving **satisfiability**, is NP-hard!

further comments on CNF and DNF, summing up:

- propositional WFF's can be partitioned into three disjoint subsets:
  1. tautologies, or **unfalsifiable** WFF's
  2. contradictions, or **unsatisfiable** WFF's
  3. WFF's that are both **satisfiable** and **falsifiable**
- satisfiability of:
  - **CNF** is in NP
  - **DNF** is in P
- tautology of:
  - **CNF** is in P
  - **DNF** is in co-NP
- falsifiability of:
  - **CNF** is in P
  - **DNF** is in NP

other special forms of propositional WFF's:

- One such form is that of the WFF's in **negation normal form** (**NNF**): the negation operator ($\neg$) is only applied to variables, and the only logical operators are conjunction ($\wedge$) and disjunction ($\vee$).

other special forms of propositional WFF's:

- One such form is that of the WFF's in **negation normal form** (**NNF**): the negation operator ($\neg$) is only applied to variables, and the only logical operators are conjunction ($\wedge$) and disjunction ($\vee$).
- More formally:

$$
\begin{aligned}
L &::= \quad p \quad | \quad \neg p \\
\varphi &::= \quad L \quad | \quad \varphi \wedge \psi \quad | \quad \varphi \vee \psi
\end{aligned}
$$

other special forms of propositional WFF's:

- One such form is that of the WFF's in **negation normal form** (**NNF**): the negation operator ($\neg$) is only applied to variables, and the only logical operators are conjunction ($\wedge$) and disjunction ($\vee$).

- More formally:

$$\begin{array}{rcccccc} L & ::= & p & \mid & \neg p & & \\ \varphi & ::= & L & \mid & \varphi \wedge \psi & \mid & \varphi \vee \psi \end{array}$$

- **Fact**: Every WFF in **CNF** or in **DNF** is also in **NNF**,
  but the converse is not true in general. See next slide for an example.

- **Fact**: **NNF** is **not** a canonical form, in contrast to **CNF** and **DNF**.

  Example: $x \wedge (y \vee \neg z)$ and $(x \wedge y) \vee (x \wedge \neg z)$ are equivalent and both in **NNF**.

- **Fact**: Every propositional WFF φ can be translated in linear time into an equivalent propositional WFF ψ in **NNF** such that $\left|\psi\right| < (3/2) \cdot \left|\varphi\right|$.

  Proof. Left to you.

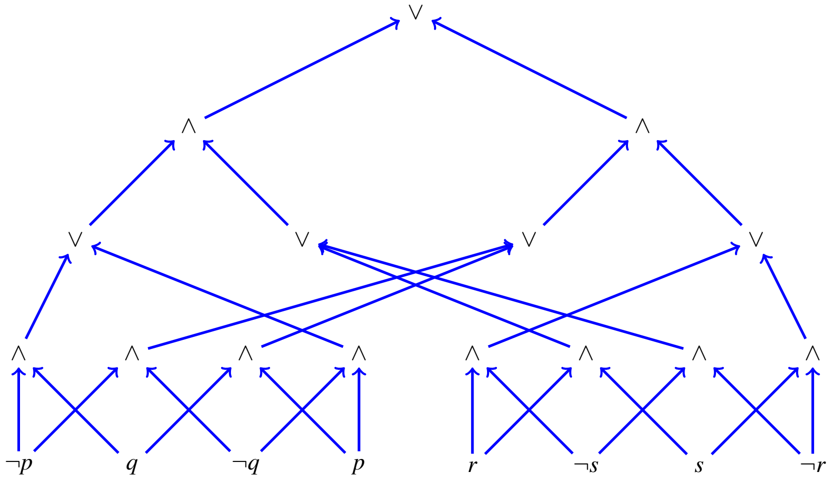
example of a WFF in **NNF**, which is neither in **CNF** nor in **DNF**

$$\begin{aligned}
&\Big(\big((\neg p \wedge q) \vee (\neg q \wedge p)\big) \wedge \big((r \wedge s) \vee (\neg s \wedge \neg r)\big)\Big) \\
\vee \quad &\Big(\big((\neg p \wedge \neg q) \vee (q \wedge p)\big) \wedge \big((r \wedge \neg s) \vee (s \wedge \neg r)\big)\Big)
\end{aligned}$$

example of a WFF in **NNF**, which is neither in **CNF** nor in **DNF**

$$\begin{aligned} &\Big(\big((\neg p \wedge q) \vee (\neg q \wedge p)\big) \wedge \big((r \wedge s) \vee (\neg s \wedge \neg r)\big)\Big) \\ \vee\quad &\Big(\big((\neg p \wedge \neg q) \vee (q \wedge p)\big) \wedge \big((r \wedge \neg s) \vee (s \wedge \neg r)\big)\Big) \end{aligned}$$

and its **parse tree** after merging identical leaf nodes, turning it into a more compact **dag**:



another special form of propositional WFFs:

Decomposable Negation Normal Form (*DNNF*)

another special form of propositional WFFs: *Decomposable Negation Normal Form* (*DNNF*)

A propositional WFF φ is a **decomposable negation normal form** (**DNNF**) if it is a **NNF** satisfying the **decomposability property**:

for every conjunction $\psi = \psi_1 \wedge \cdots \wedge \psi_n$ which is a sub-WFF of φ, no propositional variable/atom is shared by any two distinct conjuncts of ψ:

$$\mathsf{Var}(\psi_i) \cap \mathsf{Var}(\psi_j) = \varnothing \quad \text{for every} \quad i \neq j$$

another special form of propositional WFFs: *Decomposable Negation Normal Form* (*DNNF*)

A propositional WFF φ is a **decomposable negation normal form** (**DNNF**) if it is a **NNF** satisfying the **decomposability property**:

for every conjunction $\psi = \psi_1 \wedge \cdots \wedge \psi_n$ which is a sub-WFF of φ, no propositional variable/atom is shared by any two distinct conjuncts of ψ:

$$\mathsf{Var}(\psi_i) \cap \mathsf{Var}(\psi_j) = \varnothing \quad \text{for every} \quad i \neq j$$

Example: The **NNF** shown on page 19 is in fact a **DNNF**.

another special form of propositional WFFs: *Decomposable Negation Normal Form* (*DNNF*)

A propositional WFF φ is a **decomposable negation normal form** (**DNNF**) if it is a **NNF** satisfying the **decomposability property**:

for every conjunction $\psi = \psi_1 \wedge \cdots \wedge \psi_n$ which is a sub-WFF of φ, no propositional variable/atom is shared by any two distinct conjuncts of ψ:

$$\mathsf{Var}(\psi_i) \cap \mathsf{Var}(\psi_j) = \varnothing \quad \text{for every} \quad i \neq j$$

Example: The **NNF** shown on page 19 is in fact a **DNNF**.

Fact: Satisfiability of WFF in **DNNF** is decidable in linear time.

an important restricted class: *Horn formulas*

an important restricted class: *Horn formulas*

$$
\begin{array}{rcllll}
P & ::= & \bot & \mid \ \top & \mid \ p & \\
A & ::= & P & \mid \ P \wedge A & & \\
C & ::= & A \to P & & & \text{Horn clause} \\
H & ::= & C & \mid \ C \wedge H & & \text{Horn formula}
\end{array}
$$

an important restricted class: *Horn formulas*

$$
\begin{array}{lcllll}
P & ::= & \bot & \mid & \top & \mid \; p \\
A & ::= & P & \mid & P \wedge A & \\
C & ::= & A \to P & & & \text{Horn clause} \\
H & ::= & C & \mid & C \wedge H & \text{Horn formula}
\end{array}
$$

Fact: Satisfiability of Horn clauses is decidable in linear time.

Proof: To see this, rewrite a Horn clause into an equivalent disjunction of literals:
$L_1 \wedge \cdots \wedge L_n \to L \quad \equiv \quad \neg L_1 \vee \cdots \vee \neg L_n \vee L.$

Fact: Satisfiability of Horn formulas is decidable in linear time.

Exercise Search the Web to identify one or two applications, or areas of computer science, where each of the following forms are encountered:

1. Propositional WFF's in **NNF**.
2. Propositional WFF's in **DNNF**.
3. Propositional WFF's that are **Horn** formulas.