Obvious Positions

A smaller set of positions from ϕ, called *obvious positions*, is still preventing the explosion and given by the rules:

(i) p is an obvious position if $\phi|_p$ is an equivalence and there is a position $q < p$ such that $\phi|_q$ is either an equivalence or disjunctive in ϕ or

(ii) pq is an obvious position if $\phi|_{pq}$ is a conjunctive formula in ϕ, $\phi|_p$ is a disjunctive formula in ϕ and for all positions r with $p < r < pq$ the formula $\phi|_r$ is not a conjunctive formula.

A formula $\phi|_p$ is conjunctive in ϕ if $\phi|_p$ is a conjunction and $\text{pol}(\phi, p) \in \{0, 1\}$ or $\phi|_p$ is a disjunction or implication and $\text{pol}(\phi, p) \in \{0, -1\}$.

Analogously, a formula $\phi|_p$ is disjunctive in ϕ if $\phi|_p$ is a disjunction or implication and $\text{pol}(\phi, p) \in \{0, 1\}$ or $\phi|_p$ is a conjunction and $\text{pol}(\phi, p) \in \{0, -1\}$.

Polarity Dependent Equivalence Elimination

ElimEquiv1 $\chi[(\phi \leftrightarrow \psi)]_p \Rightarrow_{\text{ACNF}} \chi[(\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)]_p$
provided $\text{pol}(\chi, p) \in \{0, 1\}$

ElimEquiv2 $\chi[(\phi \leftrightarrow \psi)]_p \Rightarrow_{\text{ACNF}} \chi[(\phi \wedge \psi) \vee (\neg\phi \wedge \neg\psi)]_p$
provided $\text{pol}(\chi, p) = -1$

Extra $\top, \bot$ Elimination Rules

ElimTB7	$\chi[\phi \rightarrow \bot]_p$	$\Rightarrow_{\text{ACNF}}$	$\chi[\neg\phi]_p$
ElimTB8	$\chi[\bot \rightarrow \phi]_p$	$\Rightarrow_{\text{ACNF}}$	$\chi[\top]_p$
ElimTB9	$\chi[\phi \rightarrow \top]_p$	$\Rightarrow_{\text{ACNF}}$	$\chi[\top]_p$
ElimTB10	$\chi[\top \rightarrow \phi]_p$	$\Rightarrow_{\text{ACNF}}$	$\chi[\phi]_p$
ElimTB11	$\chi[\phi \leftrightarrow \bot]_p$	$\Rightarrow_{\text{ACNF}}$	$\chi[\neg\phi]_p$
ElimTB12	$\chi[\phi \leftrightarrow \top]_p$	$\Rightarrow_{\text{ACNF}}$	$\chi[\phi]_p$

where the two rules ElimTB11, ElimTB12 for equivalences are applied with respect to commutativity of $\leftrightarrow$.

Advanced CNF Algorithm

Algorithm: 3 acnf(ϕ)

Input : A formula ϕ.
Output: A formula ψ in CNF satisfiability preserving to ϕ.

whilerule *(***ElimTB1**(ϕ),. . .,**ElimTB12**(ϕ)*)* **do** ;
SimpleRenaming(ϕ) on obvious positions;
whilerule *(***ElimEquiv1**(ϕ),**ElimEquiv2**(ϕ)*)* **do** ;
whilerule *(***ElimImp**(ϕ)*)* **do** ;
whilerule *(***PushNeg1**(ϕ),. . .,**PushNeg3**(ϕ)*)* **do** ;
whilerule *(***PushDisj**(ϕ)*)* **do** ;
return ϕ;

Propositional Resolution

The propositional resolution calculus operates on a set of clauses and tests unsatisfiability.

Recall that for clauses I switch between the notation as a disjunction, e.g., $P \vee Q \vee P \vee \neg R$, and the multiset notation, e.g., $\{P, Q, P, \neg R\}$. This makes no difference as we consider $\vee$ in the context of clauses always modulo AC. Note that $\bot$, the empty disjunction, corresponds to $\emptyset$, the empty multiset. Clauses are typically denoted by letters C, D, possibly with subscript.

Resolution Inference Rules

Resolution $(N \uplus \{C_1 \vee P, C_2 \vee \neg P\}) \Rightarrow_{\text{RES}}$
$(N \cup \{C_1 \vee P, C_2 \vee \neg P\} \cup \{C_1 \vee C_2\})$

Factoring $(N \uplus \{C \vee L \vee L\}) \Rightarrow_{\text{RES}}$
$(N \cup \{C \vee L \vee L\} \cup \{C \vee L\})$