Decision Procedures for BS

3.15.3 Definition (Bernays-Schoenfikel Fragment (BS))

A formula of the Bernays-Schoenfikel fragment has the form $\exists \vec{x}. \forall \vec{y}. \phi$ such that ϕ does not contain quantifiers nor non-constant function symbols.

3.15.4 Theorem (BS is decidable)

Unsatisfiability of a BS clause set is decidable.

$$1 : \neg R(x, y) \vee \neg R(y, z) \vee R(x, z)$$

$$2 : R(x, y) \vee R(y, x)$$

Non-Redundant Clause Learning (NRCL)

NRCL lifts CDCL to the BS class (and further). The idea is to compute modulo a trail of ground literals, but with respect to first-order inferences.
Similar to a CDCL state, an *NRCL state* is a five tuple $(\Gamma; N; U; j; C)$, where Γ is a (partial) model assumption build from ground literals, N the initial BS clause set, U the set of learned BS clauses, j the current level and C is either $\top$, $\bot$ or a BS clause.

Literals $L \in \Gamma$ are either annotated with a number, a level, i.e., they have the form L^k meaning that L is the $k-th$ guessed decision literal, or they are annotated with a pair consisting of a clause and a (ground) substitution $L\sigma^{(C \vee L)\cdot\sigma}$ that forced the literal to become true. A pair $(C \cdot \sigma)$ is called a *closure*.

Propagate $(\Gamma; N; U; k; \top) \Rightarrow_{\mathrm{NRCL}}$
$(\Gamma L\sigma^{(C\setminus\{L_1\ldots,L_n\}\vee L)\delta\cdot\sigma}; N; U; k; \top)$

provided $C \vee L \in (N \cup U)$, $C\sigma$ is false under Γ for some grounding substitution σ, $L\sigma$ is undefined in Γ, let $L_1\sigma, \ldots, L_n\sigma$ be all copies of $L\sigma$ in $C\sigma$ and δ be the mgu of the $L_1, \ldots, L_n$

Decide $(\Gamma; N; U; k; \top) \Rightarrow_{\mathrm{NRCL}} (\Gamma, L^{k+1}; N; U; k+1; \top)$

provided L is a ground literal undefined under Γ

Conflict $(\Gamma; N; U; k; \top) \Rightarrow_{\mathrm{NRCL}} (\Gamma; N; U; k; D \cdot \sigma)$

provided $D \in (N \cup U)$, $D\sigma$ false in Γ for grounding σ

Factorize $(\Gamma; N; U; k; (D \vee L \vee L') \cdot \sigma) \Rightarrow_{\text{NRCL}}$
$(\Gamma; N; U; k; ((D \vee L)\tau) \cdot \sigma)$
provided $L\sigma = L'\sigma$, $\tau = \text{mgu}(L, L')$

Resolve $(\Gamma L\delta^{(C \vee L) \cdot \delta}; N; U; k; (D \vee L') \cdot \sigma) \Rightarrow_{\text{NRCL}}$
$(\Gamma; N; U; k; ((D \vee C)\gamma) \cdot \sigma\delta)$
provided $D\sigma$ is of level k, $L'\sigma \notin D\sigma$, $L\delta = \text{comp}(L'\sigma)$, and
$\gamma = \text{mgu}(L, \text{comp}(L'))$

Skip $(\Gamma L\delta^{(C \vee L) \cdot \delta}; N; U; k; D \cdot \sigma) \Rightarrow_{\text{NRCL}} (\Gamma; N; U; k; D \cdot \sigma)$
provided $\text{comp}(L\delta)$ does not occur in $D\sigma$

Backtrack $(\Gamma K^{i+1}\Gamma'; N; U; k; (D \vee L)\cdot\sigma) \Rightarrow_{\text{NRCL}}$
$(\Gamma L\sigma^{(D\vee L)\cdot\sigma}; N; U \cup \{D \vee L\}; i; \top)$
provided $L\sigma$ is of level k and $D\sigma$ is of level i.

3.16.3 Theorem (NRCL Overall Properties)

NRCL is sound, complete and terminating on a set of BS clauses.

Instance Generation (InstGen)

The idea of InstGen is to rely on a SAT solver. But instead of doing an overall grounding, a single constant is substituted for all variables and a satisfiability result of the SAT solver turned into an interpretation for the overall clause set. This interpretation either satisfies the clause st or triggers an inference via instantiation, analogous to superposition.

A substitution σ is a *proper instantiator* with respect to a literal L (clause C), if for some variable $x \in \mathrm{vars}(L)$ ($x \in \mathrm{vars}(C)$), $x\sigma$ is not a variable. Let $\succ$ be a well-founded closure ordering satifying $C \cdot \sigma \succ D \cdot \gamma\tau$ if
(i) $C\sigma = D\gamma\tau$ but $C\rho = D$ for some proper instantiator ρ, or,
(ii) $D\gamma \subset C$, or
(iii) $D\gamma = C$ where γ is not a renaming, nor a proper instantiator for D.

The candidate model $\mathcal{I}_N$ is inductively defined over the well-founded closure ordering $\succ$ with respect to a ground model $\mathcal{I}_{N_\alpha}$ of the grounded clause set N_α. The ground clause set N_α is constructed by mapping all variables in N to a distinguished single constant α. Then if N_α is unsatisfiable, so is N. If N_α is satisfiable, N is not necessarily satisfiable and $\mathcal{I}_N$ lifts the model $\mathcal{I}_{N_\alpha}$ of N_α to a candidate model for N. Satisfiability of the clause set N_α can be more efficiently decided by a procedure for SAT (NP $\neq$ NEXPTIME), e.g., a CDCL-based SAT solver.

Similar to the model construction for superposition, suppose the sets $\delta D \cdot \sigma$ have been defined for all closures $D \cdot \sigma$ smaller than $C \cdot \gamma$.

$$
\begin{array}{rcl}
\mathcal{I}_{C\cdot\gamma} & := & \bigcup_{D\cdot\sigma \prec C\cdot\gamma} \delta_{D\cdot\sigma} \\[2ex]
\delta_{C\cdot\gamma} & := & \left\{ \begin{array}{ll} \{L\gamma\} & \text{if } C\gamma \text{ is false in } \mathcal{I}_{C\cdot\gamma} \\ & C\cdot\gamma \text{ is the minimal representation of } C\gamma \text{ in } N \\ & L \in C \text{ and } L\gamma \text{ undefined in } \mathcal{I}_{C\cdot\gamma} \text{ and } L_\alpha \in \mathcal{I}_{N_\alpha} \\ \emptyset & \text{otherwise} \end{array} \right. \\
\mathcal{I}_N & := & \bigcup_{C\in N} \delta_{C\cdot\gamma}
\end{array}
$$

The inference rules are:

Falsify $(N, \top) \Rightarrow_{\text{IGEN}} (N, M)$
where $M = \bot$ if N_α is unsatisfiable and $M = \{L_1, \dots, L_n\}$ if $\{L_1, \dots, L_n\}$ is a model for N_α

Instantiate $(N \uplus \{C \vee A, D \vee \neg B\}, M) \Rightarrow_{\text{IGEN}}$
$(N \uplus \{C \vee A, D \vee \neg B, (C \vee A)\sigma, (D \vee \neg B)\sigma\}, \top)$
where $M = \{L_1, \dots, L_n\}$, $\sigma = \text{mgu}(A, B)$, and σ is a proper instantiator of A or B

It is important that the grounding of N_α is obtained by substituting the same constant α for all variables, for otherwise the calculus becomes incomplete. For example, the two unit clauses $P(x, y); \neg P(x, x)$ are unsatisfiable. A grounding $P(a, b); \neg P(a, a)$ results in the model $\mathcal{I}_{N_\alpha} = \{P(a, b); \neg P(a, a)\}$ but Instantiate is not applicable, because the unifier $\{x \mapsto y\}$ is not a proper instantiater for both literals.
The model M is actially not used in rule Instantiate. The proof of the Theorem 3.16.4 shows that it is sufficient to consider a minimal false clause $C \vee A$ or $D \vee \neg B$ with respect to $\mathcal{I}_N$, for the inference.

3.16.4 Theorem (Completeness of InstGen)

Let $(N, \top) \Rightarrow^*_{\text{IGEN}} (N', M)$ and let (N', M) be a final state. If N is satisfiable then $M \neq \bot$ and $\mathcal{I}_{N'} \models N'$.

Redundancy can be defined analogoulsy to superposition as well. A ground closure $C \cdot \sigma$ is *redundant* in a clause set N, if there are closures $C_1 \cdot \sigma_1, \ldots, C_n \cdot \sigma_n$ from clauses $C_1, \ldots, C_n$ from N such that $C_i \cdot \sigma_i \prec C \cdot \sigma$ for all i and $C_1\sigma_1, \ldots, C_n\sigma_n \models C\sigma$. A clause C from N is *redundant* if all its ground closures $C \cdot \sigma$ are redundant.