Table for Kclause Formula Conversion

A basic approach to derive ϕ converts each constraint into propositional clauses and conjoins them. When multiple constraints restrict the same feature, they may need to be considered semantically together for conversion. For an example, please refer to our paper.

We performed an analysis of `Kconfig` to identify such combinations. Our analysis is not exhaustive; we identified combinations that were used in the `Kconfig` models that we analyzed in this paper.We continue to generalize our analysis as we incrementally develop our tool. When a combination of constraints is encountered that we have not considered (see "indexed" in the next section), `Kclause` raises a warning.

Converting a propositional formula into an equivalent `CNF` formula can increase the number of clauses exponentially. Prior work used ETs to avoid this. We exploited the structure of `Kconfig` constraints, rather than ϕ, to create a compact and equivalent $\phi^{\mathtt{cnf}}$ instead.

Let $\kappa_{\mathtt{f}}$ be one or a combination of constraints restricting a feature `f` and let $\rho_{\mathtt{f}}$ be the conjunction of propositional clauses that represent $\kappa_{\mathtt{f}}$. As ϕ is a conjunction of $\rho_{\mathtt{f}}$ for each feature, we can convert each $\rho_{\mathtt{f}}$ into equivalent `CNF` clauses, $\rho_{\mathtt{f}}^{\mathtt{cnf}}$, and conjoin to get $\phi^{\mathtt{cnf}}$.

As we have defined finite legal combinations of constraints for $\kappa_{\mathtt{f}}$, legal clauses for $\rho_{\mathtt{f}}$ can be predefined. We precomputed $\rho^{\mathtt{cnf}}$ for possible combinations of constraints so that it can be retrieved from a table "indexed" by $\kappa_{\mathtt{f}}$. Doing so, we do not have to use a `CNF` transformation tool when converting an actual `Kconfig` model. For any given $\kappa_{\mathtt{f}}$ from a `Kconfig` model, `Kclause` looks up its precomputed $\rho^{\mathtt{cnf}}$ and substitutes the variables with actual feature names to derive $\rho_{\mathtt{f}}^{\mathtt{cnf}}$.

Then, to make $\phi^{\mathtt{cnf}}$ compact, we simplified $\rho^{\mathtt{cnf}}$ to have as few clauses as possible. Unlike arbitrary formulas, simplifying $\rho^{\mathtt{cnf}}$ is feasible as it has predefined structure and limited number of clauses.

Table 1 shows the transformations. Each row shows possible combinations of constraints restricting a feature `A`, and its propositional clauses for the case when `A` has a prompt or not. The clauses at the right side of the equivalence sign shows the CNF formula of the propositional clauses in its left side. Here are some points considered for the conversion:

- Default values are treated differently for features with prompt and without prompt. Default value of features with prompt are ignored in the formula, as it is considered as possible value of a feature, but not a constraint. Default value of features without prompt are different, as it cannot be selected by user. The formula should take account when no other feature constrains the value of that feature. On the table below, $\mathtt{A_def}$ is the default value of an invisible feature. Its value is defined in `Kconfig` file and considered as a constant.
- Non-Boolean features are treated as Boolean features. Non-Boolean features are selected unless constrained by other features. If other feature constrains the non-Boolean feature to be not selected, its value becomes `0` for integer type and “” (empty quote) for string type. We assume that when non-Boolean features are selected, its default value is used.
- We have yet considered the tristates, as none of the system we have studied used it, and some systems modified the `Kconfig` tool to disable the tristates. Extending `Kclause` for tristates are a future work.

Table 1: Conversion of Kconfig Constraints into CNF Clauses

Constraints	A with prompt	A without prompt
none	none	$A \leftrightarrow A_{def} \Leftrightarrow (\neg A \lor A_{def}) \land (A \lor \neg A_{def})$
A if C	$A \rightarrow C \Leftrightarrow (\neg A \lor C)$	$(\neg C \land \neg A) \lor (C \land (A \leftrightarrow A_{def}))$ $\Leftrightarrow (\neg A \lor C) \land (C \lor A \lor \neg\neg A_{def}) \land (\neg A \lor A_{def})$
A prompt if V	$(V \land (A \lor \neg A)) \lor (\neg V \land (A \leftrightarrow A_{def})) \Leftrightarrow (V \lor \neg A \lor A_{def}) \land (V \lor A \lor A_{def})$	
A depends on D	$A \rightarrow D \Leftrightarrow (\neg A \lor D)$ (if A non-Boolean, $A \leftrightarrow D$)	$A \leftrightarrow (D \land A_{def})$ $\Leftrightarrow (D \lor \neg A) \land (\neg D \lor A \lor \neg A_{def}) \land (\neg A \lor A_{def})$
S select A	$S \rightarrow A \Leftrightarrow (\neg S \lor A)$	$A \leftrightarrow (S_{\lor} A_{def})$ $\Leftrightarrow (\neg S \lor A) \land (S \lor \neg A \lor A_{def}) \land (A \lor \neg A_{def})$
$S_1 \cdots S_n$ select A	$(\bigvee_{i=1}^{n} S_i) \rightarrow A \Leftrightarrow \bigwedge_{i=1}^{n}((\neg S_i \lor A))$	$A \leftrightarrow ((\bigvee_{i=1}^{n} S_i) \lor A_{def})$ $\Leftrightarrow \bigwedge_{i=1}^{n}(\neg S_i \lor A) \land ((\bigvee_{i=1}^{n} S_i) \lor \neg A \lor A_{def}) \land (A \lor \neg A_{def})$
S select A if C	$(S \land C) \rightarrow A \Leftrightarrow (\neg C \lor \neg S \lor A)$	$(C \land (A \leftrightarrow (S \lor A_{def}))) \lor (\neg C \land (A \leftrightarrow A_{def}))$ $\Leftrightarrow (\neg C \lor \neg S \lor A) \land (C \lor \neg A \lor A_{def}) \land (S \lor \neg A \lor A_{def}) \land (A \lor \neg A_{def})$
(S_1 select A if C_1), $\cdots$, (S_n select A if C_n)	$(\bigvee_{i=1}^{n}(S_i \land C_i)) \rightarrow A \Leftrightarrow \bigwedge_{i=1}^{n}(\neg S_i \lor \neg C_i \lor A)$	$A \leftrightarrow ((\bigvee_{i=1}^{n}(S_i \land C_i)) \lor A_{def})$ $\Leftrightarrow \bigwedge_{i=1}^{n}(\neg S_i \lor \neg C_i \lor A) \land (\bigvee_{i=1}^{n}(S_i \land C_i) \lor \neg A \lor A_{def}) \land (A \lor \neg A_{def})$
choice $A_1 \cdots A_n$	$\bigwedge_{i=1}^{n} \bigwedge_{j=1}^{n}(A_i \rightarrow \neg A_j) \land (\bigvee_{i=1}^{n} A_i) \Leftrightarrow \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{n}(\neg A_i \lor \neg A_j) \land (\bigvee_{i=1}^{n} A_i)$	
(choice $A_1 \cdots A_n$), (choice {depends on C, if C})	$\bigwedge_{i=1}^{n} \bigwedge_{j=1}^{n}(A_i \rightarrow \neg A_j) \land ((C \land \bigvee_{i=1}^{n} A_i) \lor (\neg C \land \neg \bigvee_{i=1}^{n} A_i))$ $\Leftrightarrow \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{n}(\neg A_i \lor \neg A_j) \land (\bigvee_{i=1}^{n} A_i \lor \neg C) \land \bigvee_{i=1}^{n}(\neg A_i \lor C)$	
(S select A), (A depends on D)	$(S \land A) \lor (\neg S \land (A \rightarrow D))$ $\Leftrightarrow (\neg S \lor A) \land (S \lor \neg A \lor D)$	$(S \land A) \lor (\neg S \land (A \leftrightarrow (D \land A_{def})))$ $\Leftrightarrow (\neg S \lor A) \land (S \lor D \lor \neg A) \land (S \lor \neg A \lor A_{def}) \land (\neg D \lor A \lor \neg A_{def})$
(S select A if C), (A depends on D)	$(C \land ((S \land A) \lor (\neg S \land (A \rightarrow D)))) \lor (\neg C \land (A \rightarrow D)))$ $\Leftrightarrow (\neg C \lor \neg S \lor A) \land (C \lor \neg A \lor D) \land (S \lor \neg A \lor D)$	$(C \land ((S \land A) \lor (\neg S \land (A \leftrightarrow (D \land A_{def}))))) \lor (\neg C \land (A \leftrightarrow (S \lor A_{def})))$ $\Leftrightarrow (\neg S \lor A) \land (\neg C \lor S \lor \neg A \lor D) \land (S \lor \neg A \lor A_{def})$ $\land (A \lor \neg D \lor \neg A_{def}) \land (C \lor A \lor \neg A_{def})$
(A prompt if V), (A depends on D)	$(V \land (A \rightarrow D)) \lor (\neg V \land (A \leftrightarrow (D \land A_{def}))) \Leftrightarrow (\neg A \lor D) \land (V \lor \neg D \lor A \lor \neg A_{def}) \land (V \lor \neg A \lor A_{def})$	
(A prompt if V), (S select A)	$(V \land (S \rightarrow A)) \lor (\neg V \land (A \leftrightarrow (S \lor A_{def}))) \Leftrightarrow (\neg S \lor A) \land (V \lor S \lor \neg A \lor A_{def}) \land (V \lor A \lor \neg A_{def})$	
(A prompt if V), (S select A if C)	$(V \land ((C \land (S \rightarrow A)) \lor \neg C)) \lor (\neg V \land ((C \land (A \leftrightarrow (S \lor A_{def}))) \lor (\neg C \land (A \leftrightarrow A_{def}))))$ $\Leftrightarrow (\neg A \lor V \lor C \lor A_{def}) \land (\neg A \lor V \lor S \lor A_{def}) \land (A \lor B \lor \neg A_{def}) \land (A \lor \neg C \lor \neg S)$	
(A prompt if V), (S select A if C), (A depends on D)	$(V \land ((C \land ((S \land A) \lor (\neg S \land (A \rightarrow D)))) \lor (\neg C \land (A \rightarrow D)))) \lor (\neg V \land ((C \land ((S \land A) \lor (\neg S \land (A \leftrightarrow (D \land A_{def}))))) \lor (\neg C \land (A \leftrightarrow (S \lor A_{def})))))$ $\Leftrightarrow (\neg A \lor \neg C \lor \neg D \lor \mathrm{negV}) \land (\neg A \lor \neg C \lor D \lor S) \land (\neg A \lor D \lor \neg S \lor \neg V) \land (A \lor \neg C \lor \neg S) \land (A \lor \neg S \lor V) \land (\neg A_{def} \lor C \lor S \lor V) \land (\neg A_{def} \lor \neg D \lor S \lor V)$	