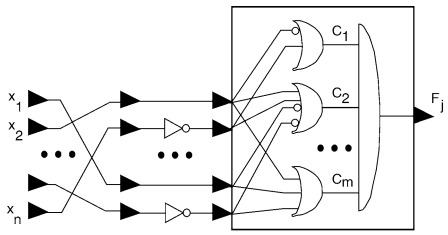
(a) On generating a CNF formula instance in an equivalence class.



The dag with inputs $\{x_1, \ldots, x_n\}$, a *permutation/complementation* network, and a 'black box' with output F_j, depicts an instance from an equivalence class $\mathcal{F}$ of Boolean functions, analyzed for their invariants in [16]. Here, the black box represents a *reference CNF formula*.

(b) Samples of CNF formula instances from four equivalence classes.

The 6-variable, 12-clause *reference formula* v06_0004 below has four satisfying assignments. Subject to the four re-writing rules described in the text, each randomly chosen instance from each of the four equivalence classes not only maintains exactly four satisfying assignments, it also preserves the syntactic structure of the reference formula.

Reference formula:	{−1 −2} {−1 −3} {−1 −4} {−1 −5} {−1 −6} {−2 −3} {−2 −4} {−3 −4} {−5 −6} {1 2 3} {4 5 6} {−1 2 −3 4}
solution vectors:	(001001, 001010, 010001, 010010)
I-class formula:	{−1 −5} {−4 −2} {−1 −3} {−1 −2} {−4 −1} {2 3 1} {−2 −3} {5 4 6} {2 −1 4 −3} {−4 −3} {−5 −6} {−1 −6}
solution vectors:	(001001, 001010, 010001, 010010)
transformations:	none, solutions *are* reference formula solutions.
P-class formula:	{−4 −2} {−5 −6} {−3 −1} {−5 −4} {−6 −2} {1 3 5} {−4 −1} {6 2 4} {5 −4 −2 6} {−4 −3} {−5 −2} {−4 −6}
solution vectors:	(011000, 110000, 001001, 100001)
transformations:	variable permutation 462513 is applied to reference formula solutions.
C-class formula:	{4 −5 6} {1 −4} {−6 1} {−2 −3} {−6 5} {−4 −2} {2 −3 4 1} {−1 3 2} {1 −3} {−3 −4} {5 1} {1 −2}
solution vectors:	(101011, 101000, 110011, 110000)
transformations:	variable complementation of (1, 5) is applied to reference formula solutions.
PC-class formula:	{−5 4} {−6 5 2 −4} {2 −3} {−6 2} {2 −5} {4 2} {−6 −5} {−2 6 −4} {−1 2} {−3 −1} {5 3 1} {4 −6}
solution vectors:	(110101, 011101, 110000, 011000)
transformations:	variable permutation 246531 and complementation of (2, 4) is applied to reference solutions.

Figure 5. A method to generate instances of CNF formulas in four equivalence classes.

- PC-class (permutation and complement) – variable names are permuted randomly *and* variables are complemented randomly (as described under the P-class and C-class); clauses and literals within a clause are randomly permuted.