1 Explanation Generation

Algorithm 1: $\textsc{ExplainCSP}(\mathcal{C}, U, f, I)$

Input : *$\mathcal{C}$ a CNF $\mathcal{C}$ over a vocabulary V*
Input : *U a user vocabulary $U \subseteq V$*
Input : *f, a cost function $f : 2^{\mathcal{G}} \rightarrow \mathbb{N}$ over a CNF $\mathcal{G}$*
Input : *I, a partial interpretation over U*
Output : *E, a sequence of explanation steps as implications $I_{expl} \implies N_{expl}$*

$\textsc{sat} \leftarrow \textsc{InitSat}(\mathcal{C})$
$E \leftarrow \langle\rangle$
$I_{end} \leftarrow \textsc{OptimalPropagate}(U, I)$
while $I \neq I_{end}$ **do**
  $X \leftarrow \textsc{bestStep}(\mathcal{C}, f, I_{end}, I)$
  $I_{best} \leftarrow I \cap X$
  $N_{best} \leftarrow \textsc{OptimalPropagate}(U, I_{best}) \setminus I$
  add $\{I_{best} \implies N_{best}\}$ to E
  $I \leftarrow I \cup N_{best}$
end
return E

Algorithm 2: $\textsc{OptimalPropagate}(\mathcal{U}, I)$

Input : *U a user vocabulary $U \subseteq V$*
State : *$\textsc{sat}$, a $\textsc{sat}$ solver.*
Optional: *I, a set of assumption literals.*
Output : *The projection onto U of the intersection of all models of U*

$sat?, \mu \leftarrow \textsc{sat}(I)$
$\mu \leftarrow \{l \mid l \in \mu : \texttt{var}(l) \in U\}$
$b_i \leftarrow$ a new blocking variable
while *true* **do**
  ~~$\mathcal{C} \leftarrow \mathcal{C} \wedge (\neg b_i \bigvee_{l \in \mu} \neg l)$~~
  add $(\neg b_i \bigvee_{l \in \mu} \neg l)$ to $\textsc{sat}$ solver
  $sat?, \mu' \leftarrow \textsc{sat}(I \wedge \{b_i\})$
  if *not sat?* **then**
    add clause $(\neg b_i)$ to $\textsc{sat}$ solver
    return μ
  end
  $\mu \leftarrow \mu \cap \{l' \mid l' \in \mu' : \texttt{var}(l') \in U\}$
end

Algorithm 3: $\textsc{bestStep–c-OUS}(\mathcal{C}, f, I_{end}, I)$

Input : *$\mathcal{C}$, a CNF.*
Input : *f, a cost function $f : 2^{\mathcal{G}} \rightarrow \mathbb{N}$ over CNF $\mathcal{G}$.*
Input : *I_{end}, the cautious consequence, the set of literals that hold in all models.*
Input : *I, a partial interpretation s.t. $I \subseteq I_{end}$.*
Output : *a single best explanation step*

$A \leftarrow I \cup (\overline{I_{end}} \setminus \bar{I})$ `// Optimal US is subset of A`
set $p \triangleq \sum_{l \in \overline{I_{end}}} l = 1$ i.e. exactly one of $\overline{I_{end}}$ is present in the hitting set
return $\textsc{c-OUS}(\mathcal{C}, f, p, A)$

Algorithm 4: c-OUS$(\mathcal{C}, f, p, A)$

Input : $\mathcal{C}$, *a CNF.*
Input : f, *a cost function* $f : 2^{\mathcal{G}} \rightarrow \mathbb{N}$ *over CNF* $\mathcal{G}$.
Input : p, *a predicate* $p : 2^{\mathcal{G}} \rightarrow \{t, f\}$ *over CNF* $\mathcal{G}$.
Input : A, *a set of assumption literals, s.t.* $\mathcal{C} \cup A$ *is unsatisfiable.*
Output : *a p-constrained f-optimal unsatisfiable subset* $(p, f) - OUS$.

$\mathcal{H} \leftarrow \emptyset$
while *true* **do**
  $A' \leftarrow \textsc{CondOptHittingSet}(f, p, A, \mathcal{H})$
  if $\neg\text{SAT}(\mathcal{C} \cup A')$ **then**
    return A'
  end
  ~~$A'' \leftarrow \textsc{Grow}(\mathcal{C}, f, p, A', A)$~~
  Optional Grow, if the sat solver can provide a provide a good model, we can skip the expensive call to the grow procedure. Needs to be checked experimentally!
  ~~$\mathcal{H} \leftarrow \mathcal{H} \cup \{A \setminus A''\}$~~
  $\mathcal{H} \leftarrow \mathcal{H} \cup \{A \setminus A'\}$
end

2 MIP model

We define a set of user variables U defined over a vocabulary V of the CNF $\mathcal{C}$ as $U \subseteq V$. Given an initial assignment I, where $vars(I) \subseteq U$, I_{end} is as the cautious consequence (the set of literals that hold in all models) of U.

The Mixed Integer Programming model for computing c-OUSes has many similarities with a set covering problem. The $\textsc{CondOptHittingSet}$ computes the optimal hitting set over a p-constrained collection of weighted sets $\mathcal{H}$.

In practice, to ensure that MIP model takes advantage of the incrementality of the problem, namely across different c-OUS calls, the specification is defined on the full set of literals of I_{end}. The constrained optimal hitting set is described by

- $x_l = \{0, 1\}$ is a boolean decision variable if the literal is selected or not.
- $w_l = f(l)$ is the cost assigned to deriving the literal or using the derived literal (∞ otherwhise).
- $c_{ij} = \{0, 1\}$ is 1 (0) if the literal l is (not) present in hitting set j.

$$\min_{x} \sum_{l\ \in I_{end} \cup \overline{I_{end}}} w_l \cdot x_l \tag{1}$$

$$\sum_{l\ \in I_{end} \cup \overline{I_{end}}} x_l \cdot w_{lj} \geq 1, \ \forall j \in \{1..|hs|\} \tag{2}$$

$$\sum_{l\ \in \overline{I_{end}} \setminus \bar{I}} x_l \geq 1, \ \forall j \in \{1..|hs|\} \tag{3}$$