Algorithm inspired by [Ignatiev et al., 2015].

Algorithm 1: OMUS($\mathcal{F}, \mathcal{H}_0, f_o$)

input : $\mathcal{F}$ *a CNF formula*
input : $\mathcal{H}_0$ *set of disjoint Minimum Correction Sets*
input : f *a cost function*
begin
  $\mathcal{H} \leftarrow \emptyset$;
  while *true* **do**
    `// minimum hitting set with cost function`
    $h \leftarrow$ `OMinHS`$(\mathcal{H}, f)$;
    $\mathcal{F}' \leftarrow \{c_i | e_i \in h\}$;
    if ***not*** *SAT*$(\mathcal{F}')$ **then**
      return $\mathcal{OMUS} \leftarrow \mathcal{F}'$;
    else
      `// written as grow(`$\mathcal{F}'$`) which is Minimum Correction Set of` $\mathcal{F}'$
      $\mathcal{C} \leftarrow \mathcal{F} \setminus grow(\mathcal{F}', \mathcal{F}, f)$;
    end
    $\mathcal{H} \leftarrow \mathcal{H} \cup \{\mathcal{C}\}$;
  end
end

Definition 1. *Let* Γ *be a collection of sets and MHS(*Γ*) the set of all minimal hitting sets on* Γ *and let* f *be an objective function with input a set of constraints. Then a hitting set* $h \in \Gamma$ *is said to be an* ***minimum cost*** *hitting set if* $\forall h' \in MHS(\Gamma)$ *we have that* $|h| \leq |h'|$ *and* $f(h) \leq f(h')$ *[Davies and Bacchus, 2011].*

Property 1. *The* ***minimum cost*** *hitting set of a collection of sets* Γ *is denoted by* $OMHS(\Gamma)$*.*

1. Parallelize OMUS with parallel calls to OMHS
2. Ideas from [Davies and Bacchus, 2011] and from [De La Banda et al., 2014]:
   (a) Formalize : Dissociate SAT from from minimum cost hitting set using Design model
   (b) OMHS can be written as a Integer Linear Program:
3. Ideas from [Ignatiev et al., 2015]:
   (a) Reducing the number of SAT Calls

OMHS ILP

$$min_{x_i} \sum_{i \in [1..|C|]} cost(c_i) \times x_i \tag{1}$$

x_i is a boolean decision variable if constraint/clause c_i is chosen or not.

$$x_i = \{0, 1\} \tag{2}$$

Every constraint has to appear at least once in the hitting set.

$$\sum_{i \in [1..|w_j|]} x_i \times w_{ij} \geq 1 \tag{3}$$

References

[Davies and Bacchus, 2011] Davies, J. and Bacchus, F. (2011). Solving maxsat by solving a sequence of simpler sat instances. In *International conference on principles and practice of constraint programming*, pages 225–239. Springer.

[De La Banda et al., 2014] De La Banda, M. G., Stuckey, P. J., Van Hentenryck, P., and Wallace, M. (2014). The future of optimization technology. *Constraints*, 19(2):126–138.

[Ignatiev et al., 2015] Ignatiev, A., Previti, A., Liffiton, M., and Marques-Silva, J. (2015). Smallest mus extraction with minimal hitting set dualization. In *International Conference on Principles and Practice of Constraint Programming*, pages 173–182. Springer.