($\forall_i$) **if** $r(a,b) \in \mathcal{A}$ **then** $\texttt{ExtendLabel}(b, \mathsf{Trans}(\mathit{Label}(a), r))$;

($\forall$) **if** x is reachable from Δ_0 and $\mathit{Next}(x, \exists R.C) = y$ **then**
$\mathit{Next}(x, \exists R.C) :=$
$\texttt{Find}(\mathit{Label}(y) \cup \mathsf{Satr}(\mathsf{Trans}(\mathit{Label}(x), R)))$;

($\forall_I$) **if** x is reachable from Δ_0 and $\langle x, R, y\rangle \in \mathit{Edges}$ **then**
$\texttt{ExtendLabel}(x, \mathsf{Trans}(\mathit{Label}(y), \overline{R}))$;

($\exists$) **if** x is reachable from Δ_0, $\exists R.C \in \mathit{Label}(x)$, $R \in \mathbf{R}$ and
$\mathit{Next}(x, \exists R.C)$ is not defined **then** $\mathit{Next}(x, \exists R.C) :=$
$\texttt{Find}(\mathsf{Satr}(\{C\} \cup \mathsf{Trans}(\mathit{Label}(x), R)) \cup \mathcal{T}')$;

($\sqsubseteq$) **if** x is reachable from Δ_0, $(C \sqsubseteq D) \in \mathit{Label}(x)$ and
$\texttt{CheckPremise}(x, C)$ **then** $\texttt{ExtendLabel}(x, \{D\})$;

Table 1: Expansion rules for Horn-$\mathcal{R}eg^I$ graphs.

Function Find(X)

if *there exists* $z \in \Delta \setminus \Delta_0$ *with* $\mathit{Label}(z) = X$ **then**
    return z
else
    add a new element z to Δ with $\mathit{Label}(z) := X$;
    return z

Procedure ExtendLabel(z, X)

if $X \subseteq \mathit{Label}(z)$ **then return**;
if $z \in \Delta_0$ **then** $\mathit{Label}(z) := \mathit{Label}(z) \cup \mathsf{Satr}(X)$
else
    $z_* := \texttt{Find}(\mathit{Label}(z) \cup \mathsf{Satr}(X))$;
    foreach y, R, C *such that* $\mathit{Next}(y, \exists R.C) = z$ **do**
        $\mathit{Next}(y, \exists R.C) := z_*$

Function CheckPremise(x, C)

if $C = \top$ **then return** *true*
else let $C = C_1 \sqcap \ldots \sqcap C_k$;
foreach $1 \leq i \leq k$ **do**
    if $C_i = A$ *and* $A \notin \mathit{Label}(x)$ **then return** *false*
    else if $C_i = \forall\exists R.A$ *and (*$\exists R.\top \notin \mathit{Label}(x)$ *or* $\mathit{Next}(x, \exists R.\top)$ *is not defined or* $A \notin \mathit{Label}(\mathit{Next}(x, \exists R.\top))$*)* **then**
        return *false*
    else if $C_i = \exists R.A$ *and* $\langle \mathbf{A}_R\rangle A \notin \mathit{Label}(x)$ **then**
        return *false*
return *true*

Algorithm 1: checking satisfiability in Horn-$\mathcal{R}eg^I$

Input: a clausal Horn-$\mathcal{R}eg^I$ knowledge base $\langle \mathcal{R}, \mathcal{T}, \mathcal{A}\rangle$ and the RIA-automaton-specification $\mathbf{A}$ of $\mathcal{R}$.
Output: *true* if $\langle \mathcal{R}, \mathcal{T}, \mathcal{A}\rangle$ is satisfiable, or *false* otherwise.

let Δ_0 be the set of all individuals occurring in $\mathcal{A}$;
if $\Delta_0 = \emptyset$ **then** $\Delta_0 := \{\tau\}$;
$\Delta := \Delta_0$, $\mathcal{T}' := \mathsf{Satr}(\mathcal{T})$, empty the mapping Next;
foreach $a \in \Delta_0$ **do**
    $\mathit{Label}(a) := \mathsf{Satr}(\{A \mid A(a) \in \mathcal{A}\}) \cup \mathcal{T}'$
while *some rule in Table 1 can make changes* **do**
    choose such a rule and execute it;
    `// any strategy can be used`
    if *there exists* $x \in \Delta$ *such that* $\bot \in \mathit{Label}(x)$ **then**
        return *false*
return *true*