80. Standard Form in PGO and Transformation
Algorithm: Problem-Solving Machines. II

By Motokiti KONDO* and Haruo MURATA**
(Comm. by Zyoiti SUETUNA, M.J.A., May 19, 1965)

1. Definitions. In the former paper "Problem-Solving Machines, I", we reported on a method to produce the code expression from any natural language sentence of plane geometry, providing the code system PGO, and to retrieve the proof of any given theorem by machine. Now, we provide below some efficient standardization of any logical expression in plane geometry, named the standard form, and give a transformation algorithm.

Definition 1. A formula is a finite sequence of atomic formulas, logical symbols, auxiliary symbols except for comma.

The order of "binding" of logical symbols coincide with that of conventional mathematical usage, that is, in descending order by degree; $\lor$, $\land$, $\neg$.

Definition 2. An atomic formula is a literal; and if $Q$ is an atomic formula then $\neg Q$ is a literal.

Definition 3. Well formed formula (wff): 1. An atomic formula is a wff. 2. If $F$ is a wff, then $\neg F$ is a wff. 3–5. If $E$ and $F$ are wff's, then $E \land F$, $E \lor F$, and $E \rightarrow F$ are wff's. 6. The only wff's are those given by 1–5.

2. Standard form of the formula. Theorems in plane geometry consist of the hypothesis and the conclusion part, that is, let $A$ and $B$ be wff's, the theorem is usually in the form

\[(2.1) A \rightarrow B.\]

Using the disjunctive normal form in the hypothesis $A$ and the conjunctive normal form in the conclusion $B$,

\[(2.2) \bigcup_{i=1}^{m_0} \bigcap_{j=1}^{n_i} A_{ij} \rightarrow \bigcap_{k=1}^{m_1} \bigcup_{l=1}^{n_k} B_{kl},\]

where $A_{ij}$ and $B_{kl}$ are literals.

It is easily seen that the disjunctions in the hypothesis and the conjunctions in the conclusion can be transferred to the front of the formula as conjunctions. Then we have

*) Tokyo Metropolitan University. **) IBM Japan, Tokyo.

1) atomic formula: A string of symbols consisting of a predicate letter followed by $n$ terms.
2) logical symbols: $\lor$, $\land$, $\rightarrow$, which mean conjunction, and, or, and imply.
3) auxiliary symbols: $\neg$, (,) and comma.
Hence, in order to give the proof of the theorem, we have only to execute retrieving the following formula at most \( m_0 \cdot m_1 \) times:

\[
(2.3) \quad \bigcap_{i=1}^{m_0} \left\{ \bigcap_{j=1}^{m_1} A_{ij} \to \bigcup_{k=1}^{n} B_{kl} \right\}.
\]

Now, we put

\[
Q_{ij} = \neg A_{ij}, \quad \text{if } \neg A_{ij} \text{ is atomic},
\]

\[
Q'_{ij} = A_{ij}, \quad \text{otherwise},
\]

\[
Q_{kl} = B_{kl}, \quad \text{if } B_{kl} \text{ is atomic},
\]

\[
Q'_{kl} = \neg B_{kl}, \quad \text{otherwise}.
\]

Then, by suitable permutation, we obtain from (2.4)

\[
(\bigcup Q_{q_1} \cup \bigcup Q_{q_2} \cup \cdots \cup \bigcup Q_{q_p}) \cup (Q_{z_1} \cup \cdots \cup Q_{z_q}).
\]

Therefore, we have

\[
(2.5) \quad \bigcap_{\mu=1}^{p} Q'_{q_\mu} \to \bigcup_{\nu=1}^{q} Q_{z_\nu}.
\]

The formula (2.5) is called the standard form of (2.2).

3. Transformation procedure. Now, a procedure transforming the code expression to the standard form is given.

P0: Enclose each of operands of logical operation in parentheses in order of occurrence from the leftmost, according to the "binding" order (hierarchy) of logical symbols.

Example. (((Q1) \neg(Q2)) \lor (Q3)) \lor ((\neg(Q4) \lor (Q5))) \lor (\neg(Q6)).

Let \( F_1F_2 \cdots F_M \) be a wff \( F \) resulting from P0, where each \( F_i \) is either symbol or atomic formula.

Definition 4. Scope: Let \( F_i \) be one of logical symbols, appearing in a wff \( F \), then \( F_{i+1} \) is the left parenthesis, ‘‘(‘‘, and if \( F_i \) is either of ‘‘ARIABLE,’’ ‘‘\lor’’ or ‘‘\land’’, then \( F_{i-1} \) is the right parenthesis, ‘‘)’’.

Let \( F_p \) and \( F_q \) be the conjugate parenthesis of \( F_{i-1} \) and \( F_{i+1} \), respectively.

Then, the scope of either of ‘‘\lor’’ or ‘‘\land’’ is the following strings; \( F_p \to F_{i-1} \to F_{i+1} \to F_q \) and \( F_{i-1} \to F_{i+1} \to F_q \) and the scope of ‘‘\lor’’ is the string \( F_{i-1} \to F_{i+1} \to F_q \). In particular, the left scope of either logical symbol ‘‘\lor’’ or ‘‘\land’’ is \( F_p \to F_{i-1} \to F_q \), and the right scope of it is \( F_{i-1} \to F_q \).

Pl. ‘‘\lor’’-elimination: Let \( F_i \) be the symbol ‘‘\lor’’, appearing first from the leftmost in a wff \( F \), \( 3 \leq i \leq M-3 \). Eliminate both the scope of \( F_i \) and \( F_i \) itself, and produce \((\neg F_p \to F_{p+1} \cdots F_{i-1} \to F_{i+1} \to F_{i+2} \cdots F_q) \lor F_{i+1} \to F_{i+2} \cdots F_q \) in that place. Henceforth, the production like this is to be written as ‘‘PROD’’:

\[
F_1 \cdots F_p \cdots F_q \cdots F_M \text{ PROD } F_1 \cdots F_{p+1} \cdots F_{i-1} \lor F_{i+1} \cdots F_M.
\]

4) This algorithm exists; cf. [3].
Repeat this procedure until all "(" symbols in each of the hypothesis $A$ and the conclusion $B$ are eliminated.

P2. Let $F_i F_{i+1} F_{i+2}$ be either string "$(o r (f l))$", appearing first from the leftmost of a wff $F$.

Case 1. The scope of $F_i$ is $(F)$: where $F$ is a string, $F$ PROD $F_1 \cdots F_{i-1} F_{i+2} F_{i+4} \cdots F_{q-1} F_q \cdots F_M$.

Case 2. The scope of $F_i$ is (($F$) $\cap (\varnothing)$), where $\varnothing$ and $\varnothing$ are strings: $F$ PROD $F_1 \cdots F_{i-1} (F) \cup (F) F_{q+1} \cdots F_M$.

Case 3. The scope of $F_i$ is (($F$) $\cap (\varnothing)$): $F$ PROD $F_1 \cdots F_{i-1} (F) \cap (F) F_{q+1} \cdots F_M$.

Repeat this procedure until all such strings are handled in both of the hypothesis $A$ and conclusion $B$.

P3--A. Application of the 1st distributive law for the hypothesis $A$: Let $A_i$ be the "$\cap$" in either string "$(o r (f l))$ $\cap (\varnothing)$", appearing first from the leftmost of the hypothesis $A$.

Case 1. The scope of $A_i$ is $(F) A_i (\varnothing) (\varnothing)$, where each of $F$, $\varnothing$, and $\varnothing$ is a string: $A$ PROD $A_1 \cdots A_{p-1} (F) \cup (\varnothing) \cup (F) \cup (\varnothing) A_{q+1} \cdots A_M$.

Case 2. The scope of $A_i$ is $(\varnothing) \cup (\varnothing) A_i (F)$: $A$ PROD $A_1 \cdots A_{p-1} (\varnothing) \cup (F) \cup (\varnothing) \cup (F) A_{q+1} \cdots A_M$.

P3--B. Application of the 2nd distributive law for the conclusion $B$: Let $B_i$ be the "$\cup$", appearing first from the leftmost of a wff $B$.

Case 1. The scope of $B_i$ is $(F) B_i (\varnothing)$: $B$ PROD $B_1 \cdots B_{p-1} (F) \cup (F) \cup (F) B_{q+1} \cdots B_M$.

Case 2. The scope of $B_i$ is $(\varnothing) \cup (F) B_i (F)$: $B$ PROD $B_1 \cdots B_{p-1} (\varnothing) \cup (F) \cup (F) B_{q+1} \cdots B_M$.

Repeat these procedures P3--A and B until all such strings are completed.

P4. Eliminate "$(o r (f l))$", if "$(o r (f l))$" is followed by another "$(o r (f l))$".

The result of P0--P4 is the form of (2.2).

P5. Let $F$ be the left scope of the "$\cup$", appearing first from the leftmost of the hypothesis part $A$. Let $\varnothing$ be the left scope of the "$\cap$", appearing first from the leftmost of the conclusion part $B$. Then

$I = (A_1) \cap (A_2) \cap \cdots \cap (A_q), \quad J = (B_1) \cup (B_2) \cup \cdots \cup (B_m)$,

where each of $A_j$ and $B_i$ is a literal. Produce the following formula:

5) 1st distributive law: $X \cap (Y \cup Z) \Rightarrow (X \cap Y) \cup (X \cap Z)$,

$\Rightarrow (X \cup Y) \cap Z \Rightarrow (X \cap Z) \cup (Y \cap Z)$.

6) 2nd distributive law: $X \cup (Y \cap Z) \Rightarrow (X \cup Y) \cap (X \cup Z)$,

$\Rightarrow (X \cap Y) \cup Z \Rightarrow (X \cup Z) \cap (Y \cup Z)$.
And apply P1, P2, and P4 for this formula. Now, the result is in the form
\[ E_1 \cup E_2 \cup \cdots \cup E_N, \]
where each of \( E_i \) is a parenthesized literal, that is, either \((Q_i)\) or \((\neg(Q_i))\), where \( Q_i \) is atomic.

P6. Let \((Q_1)\) be the scope of \( \neg \), appearing first from the leftmost of (3.2):
\[ E_1 \cup \cdots \cup E_{j-1} \cup (\neg(Q_j)) \cup E_{j+1} \cup \cdots \cup E_N, \]
Then P0 is applied for the resulting formula. Repeat this procedure until all of negation symbols in (3.2) is transformed into
\[ (Q_{i1}) \cap (Q_{i2}) \cap \cdots \cap (Q_{iz}) \cup (Q_{i2}) \cup \cdots \cup (Q_{iz}), \]
then produce the \( \neg \) in that place.

P7. Eliminate the \( \cup \), appearing first from the leftmost of (3.3), and then produce the \( \neg \) in that place.

P8. Eliminate all parentheses except for proper ones within an atomic formula, and then permute elements in each of the left scope and the right scope of \( \neg \) in the resulting formula of P7 in lexicographical order.

The resulting formula is in the form of
\[ \neg(Q_{i1} \cap Q_{i2} \cap \cdots \cap Q_{iz}) \cup Q_{i1} \cup Q_{i2} \cup \cdots \cup Q_{iz}. \]
The formula (3.4) is the final result of P0–P8, and this formula is the standard form in PGO.

4. Proof Retrieval. Using the standard form of the input formula as an index to the thesaurus, retrieval of the proof is executed by machine. If there exists the matching solution, the input formula is provable. Otherwise, the following procedures are taken.

P9. Eliminate \( \neg \) in the conclusion part in the resulting formula of P4 and return to P5. Repeat this procedure until the conclusion part becomes empty.

P10. Eliminate \( \neg \) in the hypothesis part in the resulting formula of P4, produce \( B = \bigcup_{k=1}^{m} (B_{k1} \cap \cdots \cap B_{kn}) \cap (B_{k1} \cup \cdots \cup B_{kn}) \) in place of its conclusion part and return to P5.

The procedures P9–P10 are taken at most \( m_0 \cdot m_1 \) times.

5. Examples. Let \( a, b, c, \) and \( d \) be points, respectively. \( a=b \) means that one point \( a \) equals to one point \( b \). \( G_r(a, b, c) \) means that three points \( a, b, \) and \( c \) lie on one straight line. By the code system PGO, these are represented as follows: \( a; 00000, b; 00001, c; 00002, d; 00003, =; 710, \) and \( G_r; 170 \).

\( a=b \rightarrow (a=c \rightarrow b=c) \)
The standard form of (5.1) is \( a=b \cap a=c \rightarrow b=c \), and its code expression is 710(00000,00001) AND 710(00000,00002) → 710(00001,00002).
(5.2) \( (G,(a, b, c) \cap G,(a, b, d) \cap \neg a=b) \rightarrow G,(a, c, d) \).

The standard form of (5.2) is \( G,(a, b, c) \cap G,(a, b, d) \rightarrow a=b \cup G,(a, c, d) \), and its code expression is

\[
\begin{align*}
&170(00000,00001,00002) \text{ AND } 170(00000,00001,00003) \rightarrow \\
&170(00000,00002,00003) \text{ OR } 710(00000,00001).
\end{align*}
\]

Furthermore, redundant formulas in each of the hypothesis and conclusion parts are negligible. That is, let \( \Delta'(I') \) be the proper partial conclusion (hypothesis) in the standard form of a formula \( I' \rightarrow \Delta \). Suppose \( I' \rightarrow \Delta' \Rightarrow I' \rightarrow \Delta \), then \( I' \rightarrow \Delta \) is called reducible. We have only to make a dictionary for retrieval consisting of only irreducible standard formulas. This problem will be argued in the next paper.

References