Gödel's Dialectica Translation

Stephen Mackereth

Types $\mathbb{N}$ is a type. If α, β are types, then $(\alpha \to \beta)$ is a type.

Language of T Terms of T are built up by means of function application from variables of each type and constants 0, succ, K, S, R.

Formulas of T are Boolean combinations of equations between terms of equal type.

Gödel uses definition schemes of abstraction and primitive recursion instead of K, S, R.

Axioms and rules of T Intuitively, T is nothing other than primitive recursive arithmetic generalized to finite types.

Here n, m are variables of type $\mathbb{N}$, and x, y, z, f are variables of any appropriate type.

1. Propositional logic: tautologies, modus ponens
2. Equality: $x = x$, $x = y \to t(x) = t(y)$
3. Successor: $\operatorname{succ} n \neq 0$, $\operatorname{succ} n = \operatorname{succ} m \to n = m$
4. Substitution: from $\varphi(x)$ infer $\varphi(t)$
5. Defined constants: $K(x, y) = x$, $S(x, y, z) = x(z, y(z))$, $R(f, x, 0) = x$, $R(f, x, \operatorname{succ} n) = f(n, R(f, x, n))$
6. Induction: from $\varphi(0)$ and $\varphi(n) \to \varphi(\operatorname{succ} n)$ infer $\varphi(t)$

where $t(y)$ is the result of substituting y for all occs of x in t

where t is of the same type as x

where t is of type $\mathbb{N}$

The translation, at last To each arithmetical formula $A(z)$, we assign a Dialectica translation $A^D = \exists x \, \forall y \, A_D(x, y, z)$. The quantifier-free part $A_D(x, y, z)$ is a formula in the language of T. We often suppress z and simply write $A_D(x, y)$.

Here x, y are (possibly empty) sequences of fresh variables, and z is the (possibly empty) sequence of free variables of A.

- If A is atomic, then $A^D = A_D = A$.
- Suppose that $A^D = \exists x \, \forall y \, A_D(x, y)$ and $B^D = \exists u \, \forall v \, B_D(u, v)$, where x, y, u, v denote sequences of fresh variables that are pairwise distinct from each other and from any other variables in A_D, B_D. Then we define:

$$
\begin{aligned}
(A \wedge B)^D &= \exists xu \; \forall yv \, (A_D(x, y) \wedge B_D(u, v)) \\
(A \vee B)^D &= \exists zxu \; \forall yv \, ((z = 0 \wedge A_D(x, y)) \vee (z = 1 \wedge B_D(u, v))) \\
(\exists z \; A(z))^D &= \exists zx \; \forall y \; A_D(x, y, z) \\
(\forall z \; A(z))^D &= \exists X \, \forall zy \; A_D(X(z), y, z) \\
(A \to B)^D &= \exists UY \, \forall xv \, (A_D(x, Y(xv)) \to B_D(U(x), v)) \\
(\neg A)^D &= \exists Y \; \forall x \; \neg A_D(x, Y(x)).
\end{aligned}
$$

Concatenation of variables denotes concatenation of sequences. A one-element sequence is identified with its element.

If $X = (X_1, \cdots, X_k)$ and $y = (y_1, \cdots, y_m)$, then $X(y)$ denotes the sequence of terms

$$X_1(y_1, \cdots, y_m), \; X_2(y_1, \cdots, y_m), \; \cdots, \; X_k(y_1, \cdots, y_m).$$

In the case where X or y is the empty sequence $\varnothing$, we define $\varnothing(y) = \varnothing$ and $X(\varnothing) = X$.

Dialectica interpretation theorem If $A(z)$ is a theorem of HA, then we can exhibit constants Q such that T proves $A_D(Q(z), y, z)$.

Soundness theorem Classical logic + AC proves $A \leftrightarrow A^D$.