NOTE

THE DISCRIMINATION THEOREM HOLDS FOR COMBINATORY WEAK REDUCTION

R. HINDLEY

University College of Swansea, Singleton Park, Swansea SA2 8PP, Wales.

Communicated by C. Böhm
Received January 1978

In [1] it was proved that for any \( \eta \)-distinct \( \lambda \beta \)-normal forms \( C_1, \ldots, C_n \) and any terms \( X_1, \ldots, X_m \), there exists a \( \lambda \)-term \( D \) such that

\[ DC_i \beta X_i \quad (i = 1, \ldots, n). \]

The authors of [1] remarked that their proof does not extend to combinatory weak reduction. However, once their theorem has been proved for \( \lambda \)-reduction, it can easily be deduced for combinatory weak reduction as follows.

**Corollary.** For any combinatory terms \( C_1, \ldots, C_n \), distinct and in strong normal form, and for any \( X_1, \ldots, X_m \), there exists a \( D \) such that

\[ DC_i \beta X_i \quad (i = 1, \ldots, n), \]

where \( \beta \) is combinatory weak reduction.

**Proof.** It is enough to prove the result with \( X_i = x_i \) variables. By [1] applied to the \( \lambda \)-transforms of \( C_1, \ldots, C_n \), there exists a \( \lambda \)-term \( E \) such that

\[ EC_\lambda \beta x_i \quad (i = 1, \ldots, n). \]

(Note that for \( i \neq j \) we have \( C_\lambda \neq C_\lambda \), because the \( C \)'s are in strong normal form.) Standardize these \( \lambda \beta \)-reductions. A standard reduction to a simple variable must consist entirely of "simple head-steps", i.e. steps of form

\[(\lambda xM)NP_1 \cdots P_k \beta ([N/x]M)P_1 \cdots P_k. \quad (1)\]

Then take the \( H \)-transform of this reduction [2, p. 212]. Each step (1) becomes a series of weak contractions, so we have

\[ EH C_\lambda H \beta x_i \]

But \( X_\lambda H = X \) for all \( X \); hence result.
Remark. The corollary would fail for $C_1, \ldots, C_n$ in weak normal form. For one example take $C_1 = S(KI)I$, $C_2 = I$, $X_1 = x_1$, $X_2 = x_2$. For another, take any two unsolvable terms $U, V$ and take $C_1 = [x] \cdot Ux$, $C_2 = [x] \cdot Vx$, where $[x]$ is here defined by algorithm (fab) of [2, p. 191].

References