15-399 Supplementary Notes
Normalization

Robert Harper

February 10, 2003

1 Existence of Normal Forms

A term M is in *normal form* iff it is irreducible: there is no M' such that $M \Rightarrow M'$. A term M is *normalizable* iff there exists a normal form M' such that $M \Rightarrow^* M'$. It is natural to ask whether every term has a normal form. More precisely, if $\Gamma \vdash M : A$, then does M have a normal form? The answer is "yes", but the proof is far from obvious. We will give a proof of this fact for the small fragment of type theory consisting only of the types 1 and $A \to B$. The proof is quite challenging even for this small fragment.

To get started let us first note the importance of types for the termination argument. If we ignore types, then it is easy to show that there are terms that do not have normal forms. For example, let $\omega = \lambda x.x(x)$ and let $\Omega = \omega(\omega)$. It is easy to see that $\Omega \Rightarrow \Omega$, and hence Ω does not have a normal form. Moreover, since ω is already a normal form, this observation shows that normal forms are not closed under application — even if M and N are in normal form, then $M(N)$ need not be. However, note that this example relies on the use of untypable terms![1]

Well-Typed Terms Are Hereditarily Normalizable

Our goal is to prove a *normalization theorem* for well-typed terms. We write $\mathsf{Norm}(M)$ to mean that M is a normal form term of type A in Γ.

Theorem 1.1 (Normalization) *If* $\Gamma \vdash M : A$*, then* $\mathsf{Norm}(M)$.

The obvious first attempt is to proceed by induction on typing. And indeed, the first few cases work quite well:

Proof (attempt): We proceed by induction on the typing rules.

- Case 1-I: $\Gamma \vdash \langle\rangle : 1$. Clearly $\langle\rangle$ is in normal form, so $\mathsf{Norm}(\langle\rangle)$.

[1] It is easy to see that ω is untypable. For if it were to have a type A, then A would have to be a function type $B \to C$ such that $B = B \to C$, which is impossible.

- Case Var: $\Gamma \vdash x : A$. Again, x is in normal form, so $\mathsf{Norm}(x)$.
- Case $\rightarrow$-I: $\Gamma \vdash \lambda x{:}A.M : A \rightarrow B$, because $\Gamma, x{:}A \vdash M : B$. By induction $\mathsf{Norm}(M)$, and hence $\mathsf{Norm}(\lambda x{:}A.M)$.
- Case $\rightarrow$-E: $\Gamma \vdash M(N) : B$ because $\Gamma \vdash M : A \rightarrow B$ and $\Gamma \vdash N : A$. By induction we have $\mathsf{Norm}(M)$ and $\mathsf{Norm}(N)$. We wish to show that $\mathsf{Norm}(M(N))$. But there is no obvious way to proceed! As we remarked above, knowing that M and N are normalizable is not by itself sufficient to show that $M(N)$ is normalizable.

□

This shows that we need to know more about terms than simply their normalizability if we are to make the proof go through. To this end let us look more closely at the problem of showing $\mathsf{Norm}(M(N))$ given that $\mathsf{Norm}(M)$ and $\mathsf{Norm}(N)$. We are to show that there exists a normal form P such that $M(N) \Rightarrow^* P$. We know that there exists normal forms M' and N' such that $M \Rightarrow^* M'$ and $N \Rightarrow^* N'$. So we have $M(N) \Rightarrow^* M'(N')$. But is $M'(N')$ normalizable? If $M' = \lambda x.M''$, then $M'(N') \Rightarrow [N'/x]M''$, so it is sufficient to show that $[N'/x]M''$ is normalizable. But we cannot obtain this knowing only that M' and N' are normalizable.

Our strategy is to *strengthen the induction hypothesis*. This means that we try to establish a property of well-typed terms that implies normalization. This stronger property, called *hereditary normalizability*, will take advantage of types as part of its definition, so it is written $\mathsf{HNorm}_A(M)$. The proof proceeds in two steps:

1. Show that if $\Gamma \vdash M : A$, then $\mathsf{HNorm}_A(M)$.
2. Show that if $\mathsf{HNorm}_A(M)$, then $\mathsf{Norm}(M)$.

Take a moment to convince yourself that if we can prove both of these properties, then the proof of the normalization theorem is complete.

The key to the definition of $\mathsf{HNorm}_A(M)$ is to make it strong enough that we can push through the case of application, but weak enough that we can show that λ-abstractions are hereditarily normalizable. We may satisfy the former requirement by defining $\mathsf{HNorm}_A(M)$ by *induction on types* as follows:

1. $\mathsf{HNorm}_1(M)$ iff $\mathsf{Norm}(M)$.
2. $\mathsf{HNorm}_{A \rightarrow B}(M)$ iff $\mathsf{HNorm}_A(N)$ implies $\mathsf{HNorm}_B(M(N))$.

That is, at base type a term is hereditarily normalizable iff it is normalizable. At function type, however, we require that M map hereditarily normalizable arguments to hereditarily normalizable results. This ensures that hereditary normalizability is closed under application, which is crucial for showing that well-typed terms are hereditarily normalizable.

So let's try to prove that well-typed terms are hereditarily normalizable.

Lemma 1.1 *If* $\Gamma \vdash M : A$*, then* $\mathsf{HNorm}_A(M)$*.*

Proof (attempt): We proceed as before by induction on typing.

1. Case 1-I: $\mathsf{HNorm}_1(\langle\rangle)$ since $\mathsf{Norm}(\langle\rangle)$.
2. Variable: Suppose that $\Gamma \vdash x : A$. What to do?
3. Case $\to$-E: We have by induction $\mathsf{HNorm}_{A\to B}(M)$ and $\mathsf{HNorm}_A(N)$. But then $\mathsf{HNorm}_B(M(N))$ by definition of hereditary normalizability.
4. Case $\to$-I: We have by induction $\mathsf{HNorm}_B(M)$, and we are to show that $\mathsf{HNorm}_{A\to B}(\lambda x{:}A.M)$. To do so we suppose that $\mathsf{HNorm}_A(N)$, and show that $\mathsf{HNorm}_B((\lambda x{:}A.M)(N))$. But it is not clear how to proceed.

□

The problem is that $(\lambda x{:}A.M)(N) \Rightarrow [N/x]M$, so we must prove something about the behavior of terms under substitution of hereditarily normalizable terms. To do so we strengthen the induction hypothesis one more time.

Lemma 1.2 *Let* $\Gamma = x_1{:}A_1, \ldots, x_k{:}A_k$*. If* $\Gamma \vdash M : A$ *and* $\mathsf{HNorm}_{A_1}(M_1)$*, ...,* $\mathsf{HNorm}_{A_k}(M_k)$*, then* $\mathsf{HNorm}_A([M_i/x_i]_{i=1}^k(M))$*.*

The lemma states that all substitution instances of well-typed terms by hereditarily normalizable terms are themselves hereditarily normalizable. We will show later that variables are hereditarily normalizable, so that by taking $M_i = x_i$ we obtain that M is hereditary normalizable.

Once again, let's try the proof. This time we will succeed, up to some lemmas that we will prove later.

Proof: By induction on typing derivations.

1. Case 1-I: as above.
2. Case variable x_j: we are to show that $\mathsf{HNorm}_{A_j}(M_j)$ under that very assumption.
3. Case $\to$-E: as above, using the fact that substitution distributes over application.
4. Case $\to$-I: Assume $\mathsf{HNorm}_A(N)$. By induction we know that $\mathsf{HNorm}_B([M_i/x_i]_{i=1}^k[N/x](M))$. We are to show that

$\mathsf{HNorm}_B(([M_i/x_i](\lambda x{:}A.M))(N))$. This follows from *closure under head expansion*, which we will prove shortly.

This completes the proof. □

What is "closure under head expansion"?

Lemma 1.3 *If* $\mathsf{HNorm}_A(M')$ *and* $M \Rightarrow M'$*, then* $\mathsf{HNorm}_A(M)$*.*

Proof: By induction on typing.

1. Case $A = 1$: We have $\mathsf{Norm}(M')$, which is to say that there is a normal form P such that $M' \Rightarrow P$. We are to show $\mathsf{Norm}(M)$. But this is obvious, since $M \Rightarrow M' \Rightarrow^* P$.

2. Case $A = B \rightarrow C$: Suppose that $\mathsf{HNorm}_B(N)$; we are to show that $\mathsf{HNorm}_C(M(N))$. Since $\mathsf{HNorm}_{B\rightarrow C}(M')$, we have $\mathsf{HNorm}_C(M'(N))$. Since $M(N) \Rightarrow M'(N)$, it follows by induction that $\mathsf{HNorm}_C(M(N))$.

□

This completes the proof that hereditarily normalizable substitution instances of well-typed terms are hereditarily normalizable.

Hereditarily Normalizable Terms are Normalizable

We must now show that hereditarily normalizable terms are normalizable. At base type 1 this is immediate from the definition. But what happens at function type? Suppose $\mathsf{HNorm}_{A\rightarrow B}(M)$; we wish to show $\mathsf{Norm}(M)$. How can we exploit the assumption to get the result? The only way is to come up with a hereditarily normalizable term that we can pass as argument to M, then apply induction. The "trick" is to choose a *variable* as argument. For suppose that $\mathsf{HNorm}_A(x)$. Then $\mathsf{HNorm}_B(M(x))$, so by induction $\mathsf{Norm}(M(x))$, and hence $\mathsf{Norm}(M)$. The last step requires a lemma:

Lemma 1.4 *If* $\mathsf{Norm}(M(x))$*, then* $\mathsf{Norm}(M)$*.*

Proof: Suppose that M does not have a normal form, but that $M(x)$ does, say $M(x) \Rightarrow^* P$ with P in normal form. The reduction starting from $M(x)$ can take one of two forms:

1. $M(x) \Rightarrow^* M'(x) = P$, where $M \Rightarrow^* M'$ and M' is in normal form. (Clearly M' cannot be a λ-abstraction.)

2. $M(x) \Rightarrow^* M'(x) = (\lambda x.M'')(x) \Rightarrow M'' \Rightarrow P$, where $M \Rightarrow^* M'$. But since $M' = \lambda x.M''$ and M'' is normalizable, then so is M'.

□

OK, but why are variables hereditarily normalizable? We used this fact in the preceding argument, so it is essential that we prove it. For variables of base type, this is immediate because variables are obviously normalizable. Suppose x is a variable of type $A \to B$. We wish to show that $\mathsf{HNorm}_{A\to B}(x)$. To do so we assume $\mathsf{HNorm}_A(M)$, and show that $\mathsf{HNorm}_B(x(M))$. But what to do? We seem to be in an even worse position: to show that x is hereditarily normalizable, we seem to have to show that $x(M)$ is too, and to show that we have to show that $x(M)(N)$ is, and so on. Where does it stop? Once we get to base type, at which point we have to show that $x(M_1)\cdots(M_n)$ is normalizable, given that each M_i is hereditarily normalizable. Ah, but since hereditarily normalizable terms are normalizable, we simply have to argue that $x(M_1)\cdots(M_n)$ is normalizable, given that the M_i's are. It is easy to see that this is the case.

But wait a minute. To show that hereditarily normalizable terms are normalizable, we needed to know that variables are hereditarily normalizable. To show that variables are hereditarily normalizable, we needed to show that all applications $x(M_1)\cdots(M_n)$ are normalizable, given that the M_i's are hereditarily normalizable. And to do this we used the fact that they are normalizable.

Each lemma refers to the other ...so we prove both properties *simultaneously*, by induction on types.

Lemma 1.5 *1. If* $\mathsf{HNorm}_A(M)$*, then* $\mathsf{Norm}(M)$*.*

2. If x has type $A_1 \to \cdots \to A_n \to A$, and $\mathsf{Norm}(M_1)$ *and ... and* $\mathsf{Norm}(M_n)$*, then* $\mathsf{HNorm}_A(x(M_1)\cdots(M_n))$*.*

Proof: The proof is by simultaneous induction on typing. First suppose that $A = 1$.

1. Immmediate, because hereditarily normalizable terms of type 1 are normalizable by definition.

2. Here $n = 0$, and we need only show $\mathsf{Norm}(x)$, which is obvious.

Now assume both parts of the theorem for types B and C, and show them for the type $A = B \to C$.

1. By induction part (2) we have $\mathsf{HNorm}_B(x)$, so $\mathsf{HNorm}_C(M(x))$. By induction part (1) we have $\mathsf{Norm}(M(x))$, so $\mathsf{Norm}(M)$.

2. Suppose that $\mathsf{HNorm}_B(N)$. We are to show $\mathsf{HNorm}_C(x(M_1)\cdots(M_n)(N))$. By induction part (1) we have $\mathsf{Norm}(N)$, so the result follows by induction part (2).

□

We are now one small step to the finish line! Suppose that $\Gamma \vdash M : A$. All substitution instances of M by hereditarily normalizable terms are hereditarily normalizable. Since variables are hereditarily normalizable, we may substitute each variable in M by itself to obtain that M is hereditarily normalizable. But then M is normalizable.

A Detail Concerning Variables

There is a small, correctable technical problem with the above argument. When defining $\mathsf{HNorm}_A(M)$ we are implicitly assuming that M has type A. But with respect to what context? That is, what free variables are allowed to occur in M, and what are their types? In the literature it is customary to gloss over this issue, but for the sake of completeness let us consider how we might account for the declaration of variables.

There are two main methods.

1. Construct an infinite *saturated* context Γ_∞ that declares infinitely many variables of each type. Define $\mathsf{HNorm}_A(M)$ so that $\Gamma_\infty \vdash M : A$ — that is, there exists a finite subset $\Gamma \subseteq \Gamma_\infty$ such that $\Gamma \vdash M : A$. We have implicitly used this method in the foregoing development.

2. Parameterize the hereditary normalization predicate by a context that represents the "current" set of available variables, and restrict attention to well-typed terms over this context. We re-defined hereditary normalizability to take a context as parameter, and arrange that $\mathsf{HNorm}^{\Gamma}_A(M)$ is defined only when $\Gamma \vdash M : A$.

   To make the proof go through we must strengthen the definition of hereditary normalization to take account of all possible expansions of the set of available variables. Specifically, we define $\mathsf{HNorm}^{\Gamma}_{A \to B}(M)$ to hold iff for every $\Gamma' \supseteq \Gamma$, if $\mathsf{HNorm}^{\Gamma'}_A(N)$, then $\mathsf{HNorm}^{\Gamma'}_B(M(N))$. Given this, it is easy to check that if $\mathsf{HNorm}^{\Gamma}_A(M)$ and $\Gamma' \supseteq \Gamma$, then $\mathsf{HNorm}^{\Gamma'}_A(M)$.

   The main lemmas needed for the proof are re-stated as follows:

   (a) If $x_1{:}A_1, \ldots, x_n{:}A_n \vdash M : A$ and $\mathsf{HNorm}^{\Gamma}_{A_i}(M_i)$ $(1 \leq i \leq n)$, then $\mathsf{HNorm}^{\Gamma}_A([M_i/x_i]_{i=1}^n M)$.

   (b) If $\mathsf{HNorm}^{\Gamma}_A(M)$, then $\mathsf{Norm}(M)$, and if $\Gamma \vdash x(M_1)\cdots(M_n) : A$ with $\mathsf{Norm}(M_i)$ $(1 \leq i \leq n)$, then $\mathsf{HNorm}^{\Gamma}_A(x(M_1)\cdots(M_n))$.