Infinite lists

An ordinal is either 0 or an infinite list of ordinals written right to left, which is all 0 after some point
$\gamma = (\ldots, 0, 0, \gamma_n, \ldots, \gamma_0)$, so $\gamma_k = 0$ for all $k > n$
For all i (including n and larger), $\gamma_i < (\ldots, \gamma_{i+3}, \gamma_{i+2}, \gamma_{i+1} + 1, 0, \ldots, 0)$

Finite lists

Leading 0s may be removed from infinite list and length of remaining list put in front: $(n : \gamma_{n-1}, \ldots, \gamma_0)$
Requires construction of naturals first, unlike infinite lists
Still lexicographic since lengths are compared first
No need for base case since list may be empty: could set $(0 :) = 0$, $(1 : (0 :)) = (1 : 0) = 1$, $(1 : 1) = 2$, etc.
This would offset naturals by 1 from infinite list representation, but otherwise identical

Ordinal-indexed lists

"Redundant" finite lists: place index of each element before it: $(n : \gamma_n, \ldots, \gamma_0) \to (n : \gamma_n, \ldots, 0 : \gamma_0)$
"Sparse" finite lists: remove all 0 entries from list (along with their indices),
turn all entries $\alpha \to -1 + \alpha$ to fill gap: $(2 : 1, 1 : 0, 0 : \omega) \to (2 : 0, 0 : \omega)$
Ordinal-indexed lists: allow indices to be ordinals themselves: $(\omega : 0) = \sup(\{(0 : 0), (1 : 0), (2 : 0), \ldots\})$
No longer requires naturals to be constructed first

Formal definition

Ordinal is a finite list of pairs $(\beta_n : \gamma_n, \beta_{n-1} : \gamma_{n-1}, \ldots, \beta_0 : \gamma_0)$, ordered lexicographically
Indices are strictly decreasing: $\beta_n > \beta_{n-1} > \ldots > \beta_0$
For all i, if $i < n$ and $\beta_{i+1} = \beta_i + 1$, then $\gamma_i < (\beta_n : \gamma_n, \ldots, \beta_{i+1} : \gamma_{i+1} + 1)$
Otherwise, $\gamma_i < (\beta_n : \gamma_n, \ldots, \beta_{i+1} : \gamma_{i+1}, \beta_i + 1 : 0)$, specifically $(\beta_n + 1 : 0)$ if $i = n$

In general:

$$\begin{aligned}
() &= 0 \\
(0 : n) &= 1 + n \\
(1 : \gamma) &= \varphi_0(1 + \gamma) \\
(2 + \beta : \gamma) &= \varphi_{1+\beta}(\gamma) \\
(\beta_n : \gamma_n, \ldots, \beta_1 : \gamma_1, 0 : \gamma_0) &= (\beta_n : \gamma_n, \ldots, \beta_1 : \gamma_1) + 1 + \gamma_0 \\
(\beta_n : \gamma_n, \ldots, \beta_1 : \gamma_1, 1 + \beta_0 : \gamma_0) &= \varphi_{\beta_0}((\beta_n : \gamma_n, \ldots, \beta_1 : \gamma_1) + 1 + \gamma_0)
\end{aligned}$$

Examples:

$$\begin{aligned}
(0 : 0) &= 1 \\
(0 : 1) &= 2 \\
(1 : 0) &= \omega \\
(1 : 0, 0 : \gamma) &= \omega + 1 + \gamma, && \gamma < (1 : 1) \\
(1 : 1) &= \omega^2 \\
(1 : \gamma_1, 0 : \gamma_0) &= \omega^{1+\gamma_1} + 1 + \gamma_0, && \gamma_1 < (2 : 0), \gamma_0 < (1 : \gamma_1 + 1) \\
(2 : 0) &= \varepsilon_0 \\
(2 : 0, 1 : \gamma_1) &= \varepsilon_0 \omega^{1+\gamma_1} && \gamma_1 < (2 : 1) \\
(2 : 0, 1 : \gamma_1, 0 : \gamma_0) &= \varepsilon_0 \omega^{1+\gamma_1} + 1 + \gamma_0 && \gamma_1 < (2 : 1), \gamma_0 < (2 : 0, 1 : \gamma_1 + 1) \\
(\omega : 0) &= \varphi_\omega(0)
\end{aligned}$$

The notation goes up to Γ_0, and as seen above, it is easily converted to two-argument Veblen normal form. However, it has the advantage of lexicographic comparison.

Further extensions

Allow colons to separate more entries than pairs

Veblen functions discard initial 0 arguments, but we want the restrictions on values to be only from above

As such, $(0 : 0 : 0) = \Gamma_0$

Lengths of lists within list representing a given ordinal should be in non-strictly decreasing order

Should also allow 1- and 0-entry colon-separated lists

$1 + \beta : \gamma \to \beta : \gamma$, $0 : 1 + \gamma \to \gamma$, $0 : 0 \to$