# Ordinalzahl-Registermaschinen und Modelle der Mengenlehre

by Peter Koepke
Universität Bonn

Abstract:

Wir verallgemeinern gewöhnliche Registermaschinen auf natürlichen Zahlen zu Maschinen, deren Register beliebige Ordinalzahlen enthalten können. Eine Ordinalzahl-Registermaschine kann ein rekursives beschränktes Wahrheitsprädikat auf den Ordinalzahlen berechnen. Die Klasse der Mengen von Ordinalzahlen, die sich aus dem Wahrheitsprädikat ablesen lassen, erfüllt eine natürliche Theorie SO. SO ist die Theorie der Ordinalzahlmengen in einem Modell der Mengenlehre. Daraus folgt eine neuartige Charakterisierung der konstruktiblen Mengen: eine Menge von Ordinalzahlen ist Ordinalzahl-berechenbar genau dann, wenn sie Element des Gödelschen Universums L ist.

a) the *zero instruction* $Z(n)$ changes the contents of $R_n$ to 0, leaving all other registers unaltered;

b) the *successor instruction* $S(n)$ increases the ordinal contained in $R_n$, leaving all other registers unaltered;

c) the *transfer instruction* $T(m, n)$ replaces the contents of $R_n$ by the ordinal $r_m$ contained in $R_m$ , leaving all other registers unaltered;

d) the *jump instruction* $J(m, n, q)$ is carried out within the program $P$ as follows: the contents $r_m$ and $r_n$ of the registers $R_m$ and $R_n$ are compared, but all the registers are left unaltered; then, if $r_m = r_n$ , the ORM proceeds to the $q$th instruction of $P$; if $r_m \neq r_n$, the ORM proceeds to the next instruction in $P$.

Let $P = I_0, I_1, \ldots, I_{s-1}$ be a program. A triple

$$I: \theta \rightarrow \omega, R: \theta \rightarrow ({}^{\omega}\mathrm{Ord})$$

is an *(ordinal register)* computation by $P$ if the following hold:

a) ...

b) ...

c) If $t < \theta$ and $I(t) \in \mathrm{state}(P)$ then $t + 1 < \theta$; the next configuration is determined by the instruction $I_{I(t)}$:

   i. if $I_{I(t)}$ is the zero instruction $Z(n)$ then let $I(t + 1) = I(t) + 1$ and define $R(t+1): \omega \rightarrow \mathrm{Ord}$ by

$$R_k(t+1) = \begin{cases} 0, & \text{if } k = n \\ R_k(t), & \text{if } k \neq n \end{cases}$$

   ii. ...

   iii. ...

d) If $t < \theta$ is a limit ordinal, the machine constellation at $t$ is determined by taking inferior limits:

$$\begin{aligned} \forall k \in \omega \; R_k(t) &= \liminf_{r \rightarrow t} R_k(r); \\ I(t) &= \liminf_{r \rightarrow t} I(r). \end{aligned}$$

```
Ordinal addition, computing gamma = alpha + beta:
0  alpha':=0
1  beta':=0
2  gamma:=0
3  if alpha=alpha' then go to 7
4  alpha':=alpha'+1
5  gamma:=gamma+1
6  go to 3
7  if beta=beta' then STOP
8  beta':=beta'+1
9  gamma:=gamma+1
10 go to 7
```

```
Goedel pairing, computing gamma = G(alpha,beta):
0  alpha':=0
1  beta':=0
2  eta:=0
3  flag:=0
3  gamma:=0
4  if alpha=alpha' and beta=beta' then STOP
5  if alpha'=eta and and beta'=eta and flag=0 then
      alpha'=0, flag:=1, go to 4 fi
6  if alpha'=eta and and beta'=eta and flag=1 then
      eta:=eta+1, alpha'=eta, beta'=0, gamma:=gamma+1, go
to 4 fi
7  if beta'<eta and flag=0 then
      beta':=beta'+1, gamma:=gamma+1, go to 4 fi
8  if alpha'<eta and flag=1 then
      alpha':=alpha'+1, gamma:=gamma+1, go to 4 fi
```

**Theorem 1.** *Let $H$ be an ordinal register computable function. Define $F\colon \mathrm{Ord}\rightarrow 2$ recursively:*

$$F(\alpha)=\begin{cases} 1 & \text{iff } \exists\nu<\alpha\ H(\alpha,\nu,F(\nu))=1 \\ 0 & \text{else} \end{cases}$$

Then $F$ is ordinal register computable.

```
  value:=2                        %% set value to undefined
MainLoop:
  nu:=last(stack)
  alpha:=llast(stack)
  if nu = alpha then
1:  do
    remove_last_element_of(stack)
    value:=0                      %% set value equal to 0
    goto SubLoop
    end_do
  else
2:  do
    stack:=stack + 1              %% push the ordinal 0
onto the stack
    goto MainLoop
    end_do
SubLoop:
  nu:=last(stack)
  alpha:=llast(stack)
  if alpha = UNDEFINED then STOP
  else
    do
    if H(alpha,nu,value)=1 then
3:    do
      remove_last_element_of(stack)
      value:=1
      goto SubLoop
      end_do
    else
4:    do
      stack:=stack + 2*(3**y)      %% push y+1
      value:=2                     %% set value to
undefined
      goto MainLoop
      end_do
    end_do
```

A bounded truth predicate:

$$T(\alpha) \;\text{ iff }\; \alpha \text{ is a bounded } L_T\text{-sentence and } (\alpha, <, G, T \cap \alpha) \vDash \alpha.$$

$$\chi_T(\alpha) = \begin{cases} 1 \text{ iff } \exists \nu < \alpha \; H(\alpha, \nu, \chi_T(\nu)) = 1 \\ 0 \text{ else} \end{cases}$$

where

$$\begin{aligned} H(\alpha,\nu,\chi) = 1 \;\text{ iff }\; & \alpha \text{ is an } L_T\text{-sentence and} \\ & \big(\exists \xi, \zeta < \alpha \, (\alpha = c_\xi \equiv c_\zeta \wedge \xi = \zeta) \\ \text{or } & \exists \xi, \zeta < \alpha \, (\alpha = c_\xi < c_\zeta \wedge \xi < \zeta) \\ \text{or } & \exists \xi, \zeta, \eta < \alpha \, (\alpha = \dot{G}(c_\xi, c_\zeta, c_\eta) \wedge \eta = G(\xi, \zeta)) \\ \text{or } & \exists \xi < \alpha \, (\alpha = \dot{R}(c_\xi) \wedge \nu = \xi \wedge \chi = 1) \\ \text{or } & \exists \varphi < \alpha \, (\alpha = \neg\varphi \wedge \nu = \varphi \wedge \chi = 0) \\ \text{or } & \exists \varphi, \psi < \alpha \, (\alpha = (\varphi \vee \psi) \wedge (\nu = \varphi \vee \nu = \psi) \wedge \chi = 1) \\ \text{or } & \exists n < \omega \, \exists \xi < \alpha \exists \varphi < \alpha \, (\alpha = \exists v_n < c_\xi \; \varphi \wedge \exists \zeta < \xi \; \nu = \\ & \varphi \frac{c_\zeta}{v_n} \wedge \chi = 1)\big). \end{aligned}$$

1. Well-ordering axiom:
   $\forall\alpha,\beta,\gamma(\neg\alpha<\alpha\wedge(\alpha<\beta\wedge\beta<\gamma\rightarrow\alpha<\gamma)\wedge$
   $(\alpha<\beta\vee\alpha=\beta\vee\beta<\alpha))\wedge$
   $\forall a(\exists\alpha(\alpha\in a)\rightarrow\exists\alpha(\alpha\in a\wedge\forall\beta(\beta<\alpha\rightarrow\neg\beta\in a)));$

2. Axiom of infinity (existence of a limit ordinal):
   $\exists\alpha(\exists\beta(\beta<\alpha)\wedge\forall\beta(\beta<\alpha\rightarrow\exists\gamma(\beta<\gamma\wedge\gamma<\alpha)));$

3. Axiom of extensionality: $\forall a,b(\forall\alpha(\alpha\in a\leftrightarrow\alpha\in b)\rightarrow a=b);$

4. Initial segment axiom: $\forall\alpha\exists a\forall\beta(\beta<\alpha\leftrightarrow\beta\in a);$

5. Boundedness axiom: $\forall a\exists\alpha\forall\beta(\beta\in a\rightarrow\beta<\alpha);$

6. Pairing axiom (Gödel Pairing Function):
   $\forall\alpha,\beta,\gamma\,(g(\beta,\gamma)\leq\alpha\leftrightarrow\ \forall\delta,\epsilon((\delta,\epsilon)<^*(\beta,\gamma)\ \rightarrow g(\delta,\epsilon)<\alpha)).$
   Here $(\alpha,\beta)<^*(\gamma,\delta)$ stands for
   $\exists\eta,\theta(\eta=\max(\alpha,\beta)\wedge\theta=\max(\gamma,\delta)\wedge(\eta<\theta\vee$
   $(\eta=\theta\wedge\alpha<\gamma)\vee(\eta=\theta\wedge\alpha=\gamma\wedge\beta<\delta))),$
   where $\gamma=\max(\alpha,\beta)$ abbreviates $(\alpha>\beta\wedge\gamma=\alpha)\vee(\alpha\leq\beta\wedge\gamma=\beta);$

7. $g$ is onto: $\forall\alpha\exists\beta,\gamma\,(\alpha=g(\beta,\gamma));$

8. Axiom schema of separation: For all $L_{SO}$-formulae $\phi(\alpha,P_1,\ldots,P_n)$ postulate:
   $\forall P_1,\ldots,P_n\forall a\exists b\forall\alpha\,(\alpha\in b\leftrightarrow\alpha\in a\wedge\phi(\alpha,P_1,\ldots,P_n));$

9. Axiom schema of replacement: For all $L_{SO}$-formulae $\phi(\alpha,\beta,P_1,\ldots,P_n)$ postulate:
   $\forall P_1,\ldots,P_n\,(\forall\xi,\zeta_1,\zeta_2(\phi(\xi,\zeta_1,P_1,\ldots,P_n)\wedge\phi(\xi,\zeta_2,P_1,\ldots,P_n)\rightarrow\zeta_1=\zeta_2)\rightarrow$
   $\forall a\exists b\forall\zeta\,(\zeta\in b\leftrightarrow\exists\xi\in a\,\phi(\xi,\zeta,P_1,\ldots,P_n)));$

10. Powerset axiom:
   $\forall a\exists b(\forall z(\exists\alpha(\alpha\in z)\wedge\forall\alpha(\alpha\in z\rightarrow\alpha\in a)\rightarrow\exists^{=1}\xi\forall\beta(\beta\in z\leftrightarrow g(\beta,\xi)\in b))).$

**Theorem 2.** *A set $x$ of ordinals is ordinal computable if and only if it is an element of the constructible universe $L$.*

**Proof.** Let $x\subseteq\mathrm{Ord}$ be ordinal computable by the program $P$ from the ordinals $\delta_1,\ldots,\delta_{n-1}$, so that for every $\alpha\in\mathrm{Ord}$:

$$P\colon(\alpha,\delta_1,\ldots,\delta_{n-1},0,0,\ldots)\mapsto\chi_x(\alpha).$$

By the simple nature of the computation procedure the same computation can be carried out inside the inner model $L$, so that for every $\alpha\in\mathrm{Ord}$:

$$(L,\in)\vDash P\colon(\alpha,\delta_1,\ldots,\delta_{n-1},0,0,\ldots)\mapsto\chi_x(\alpha).$$

Hence $\chi_x\in L$ and $x\in L$.

Conversely consider $x\in L$. Since $(\mathrm{Ord},\mathcal{S},<,=,\in,G)$ is a model of the theory SO there is an inner model $M$ of set theory such that

$$\mathcal{S}=\{z\subseteq\mathrm{Ord}\,|\,z\in M\}.$$

Since $L$ is the $\subseteq$-smallest inner model, $L\subseteq M$. Hence $x\in M$ and $x\in\mathcal{S}$. Let $x=T(\mu,\alpha)$. By the computability of the truth predicate, $x$ is ordinal register computable from the parameters $\mu$ and $\alpha$. □