Some machine characterizations of classes close to $\Delta_0^{\mathbb{N}}$

A thesis submitted to the University of Manchester for the degree of Doctor of Philosophy in the Faculty of Science.

1986

William G. Handley

Department of Mathematics

ABSTRACT

In [Bel'tyukov] a type of machine (the SRM) is defined and certain classes of recursively defined functions are characterized as complexity classes of this machine. One of the results in that paper is a characterization of $\Delta_0^{\mathbb{N}}$ (the class of sets of numbers which are defined by arithmetic formulae containing only bounded quantification). This thesis begins by going over the proof of this again.

A family of closure properties of classes of sets of numbers is then defined. Each closure property is associated with a set X with binary operation (closure under counting modulo X). Relations between these closure properties are considered and are shown to be linked with connections between the associated sets with binary operation. Classes at least as large as $\Delta_0^{\mathbb{N}}$ are defined using the new properties ($X\Delta_0^{\mathbb{N}}$) and certain of these classes are shown to be complexity classes of Bel'tyukov's machines ($S_n\Delta_0^{\mathbb{N}} = \mathrm{Space}_*(\underline{n},\Box)$).

The Bel'tyukov machines are modified, giving a new family of machine types. Each type is associated with a finite set of numbers Q. Some others of the $X\Delta_0^{\mathbb{N}}$ classes are shown to be complexity classes of certain types of Q-machines ($\mathbb{Z}_{n!}\Delta_0^{\mathbb{N}} = \{1,n+1\}\text{-}\mathrm{Space}_*(\underline{1},\Box)$).

Classes already characterized are shown also to be complexity classes of the Q-machines.

DECLARATION

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institution of learning.

ATTRIBUTION

Except where indicated otherwise, results in this thesis are due to J.B.Paris and the author.

EDUCATION

I graduated from Manchester University in 1982 with a B.Sc.(Hons), class I in Mathematics.

From 1982 to 1985 I have been a research student at Manchester University under the supervision of J.B.Paris.

ACKNOWLEDGEMENTS

My thanks chiefly to my supervisor Jeff Paris. Also to the other staff and students at Manchester for their help.

I am grateful to the S.E.R.C. for financial support in the form of a Research Studentship (1982-5).

CONTENTS

Chapter I PRELIMINARIES

I.1 Definition: The first order language of arithmetic (LA) is defined to be the first order predicate calculus with the following non-logical symbols:

2-place relation symbols $=$ and $\leq$,

0-place function symbols 0 and 1,

2-place function symbols $+$ and $\cdot$.

I.2 Remark: We shall at all times be concerned with a particular LA-structure, namely $\mathbb{N}$: the set of natural numbers; the standard model of arithmetic, in which our symbols have the usual interpretations.

I.3 Definition: If a set of numbers A (our idea of a set of numbers will generally embrace the subsets of $\mathbb{N}^m$ for all $m \leq 1$) can be expressed as

$$\{(x_1,\dots,x_m) : \Phi\}$$

where Φ is an LA-formula whose free variables are drawn from $\{x_1,\dots,x_m\}$, then A is said to be defined by the formula Φ. (Strictly speaking we require that all free variables be chosen from $x_1,x_2,x_3,\dots$.)

Functions are defined by LA-terms in the obvious, corresponding way.

Notice that this means that each formula (term) defines an infinite number of sets (functions). However, for each arity, at most one set or function will be defined and if the arity is less than the largest free variable index in the formula or term then no set or function of that arity will be defined at all.

I.4 Definition: For Φ an LA-formula and τ an LA-term we have the following abbreviations.

$\forall w\ (w \leq \tau \rightarrow \Phi)$ is abbreviated by $\forall w \leq \tau\ \Phi$

and $\exists w\ (w \leq \tau \wedge \Phi)$ is abbreviated by $\exists w \leq \tau\ \Phi$.

Such formulae are said to be obtained from Φ by bounded universal and bounded existential quantification respectively.

I.5 Definition: Δ_0 is defined to be the largest class of LA-formulae whose members never contain $\forall$ or $\exists$ except as part of some bounded quantification.

I.6 Definition: $\Delta_0^{\mathbb{N}}$ is defined to be the class of sets of numbers that are defined by elements of Δ_0.

I.7 Definitions: We will mainly be concerned with sets of numbers and also with functions from numbers to numbers. There are obvious ways of connecting functions with sets and vice versa.

Firstly from $A \subseteq \mathbb{N}^m$ (some $m \geq 1$) we derive χ_A, the characteristic function of A. $\chi_A : \mathbb{N}^m \to \mathbb{N}$ and is such that

$$\chi_A(x_1,\dots,x_m) = \begin{cases} 0 & \text{if } (x_1,\dots,x_m) \in A \\ 1 & \text{otherwise.} \end{cases}$$

Secondly, for $f : \mathbb{N}^m \to \mathbb{N}$, we have graph(f), the graph of f. graph(f) $\subseteq \mathbb{N}^{m+1}$ and is such that

$$(x_1,\dots,x_{m+1}) \in \text{graph}(f) \quad \text{iff} \quad f(x_1,\dots,x_m) = x_{m+1}.$$

I.8 Definition: A function $f : \mathbb{N}^m \to \mathbb{N}$ is said to be a Δ_0-function if (i) graph(f) $\in \Delta_0^{\mathbb{N}}$

and (ii) there is a $p(x_1,\dots,x_m) \in \mathbb{N}[x_1,\dots,x_m]$ such that for all

$$x_1,\dots,x_m \in \mathbb{N},\ f(x_1,\dots,x_m) \leq p(x_1,\dots,x_m).$$

(Notice that the polynomials with natural number coefficients are, in fact, exactly the functions defined by LA-terms.)

I.9 Definition: A set $B \subseteq \mathbb{N}^m$ is said to be obtained from a set $A \subseteq \mathbb{N}^n$ by explicit transformation if

$$B = \{(x_1,\dots,x_m) : (u_1,\dots,u_n) \in A\}$$

where, for $1 \leq i \leq n$, $u_i \in \mathbb{N}$ or $u_i = x_j$ for some $1 \leq j \leq m$.

I.10 Definition: For given $m \geq 1$, the boolean operations on subsets of $\mathbb{N}^m$ are union, intersection and complementation in $\mathbb{N}^m$.

I.11 Definitions: We define the set operations bounded quantification (I) and bounded quantification (II) as follows:

If $A \subseteq \mathbb{N}^m$, then $B \subseteq \mathbb{N}^m$ is derived from A by bounded quantification (I) when

$B = \{(x_1,\dots,x_m) : Q\, y \leq x\, (y,x_2,\dots,x_m) \in A\}$ where $Q = \forall$ or $\exists$,

and by bounded quantification (II) when

$B = \{(x_1,\dots,x_m) : Q\, y \leq p(x_1,\dots,x_m)\, (y,x_2,\dots,x_m) \in A\}$

where $Q = \forall$ or $\exists$ and $p \in \mathbb{N}[x_1,\dots,x_m]$.

I.12 <u>Theorem</u>: $\Delta_0^{\mathbb{N}}$ is the smallest class $\mathcal{C}$ of sets of numbers such that

(i) $=, \leq$, graph(+), graph($\cdot$) $\in \mathcal{C}$

(ii) $\mathcal{C}$ is closed under explicit transformations

(iii) $\mathcal{C}$ is closed under boolean operations

(iv) $\mathcal{C}$ is closed under bounded quantification (II).

<u>Proof</u>: Easy.

I.13 <u>Definition</u>: [Bel'tyukov]

A <u>Stack Register Machine</u> (SRM) is as follows:

(A) For some $m \geq 0$, it has input registers (whose contents are labelled)

$x_1,\dots,x_m$.

(B) For some $k \geq 0$, it has stack registers (whose contents are labelled)

$t_0,\dots,t_k$.

(C) It has a work-register (containing)

r.

(D) It has a program $L_1 L_2 \dots L_p L_{p+1}$, where L_{p+1} is <u>halt</u> and for $1 \leq j \leq p$, L_j is one of three types of instruction:

<u>Type (i)</u>: $t_i := t_i + 1$; (for some $0 \leq i \leq k$)

<u>Type (ii)</u>: $r := z$; (for some $z \in \{t_0,\dots,t_k,x_1,\dots,x_m\}$)

<u>Type (iii)</u>: If $z_1 * z_2 = z_3$ then L_ξ else L_η; (where $*$ is $+$ or $\cdot$, $1 \leq \xi,\eta \leq p+1$ and $z_1,z_2,z_3 \in \{r,t_0,\dots,t_k,x_1,\dots,x_m\}$).

A type (i) instruction affecting t_i has the subsidiary effect that $t_j := 0$ for all $j < i$. Furthermore, for each $0 \leq i \leq k$, only one type (i) instruction affecting t_i is allowed in a given program.

I.14 Definition: A function $f : \mathbb{N}^m \to \mathbb{N}$ is said to be computed by an SRM M with m input registers when, for all $x_1, \ldots, x_m \in \mathbb{N}$, M is such that if we start M's program off (at instruction L_1) with values $x_1, \ldots, x_m$ in its input registers and zero in all its stack registers and in its work-register then

(i) M will halt eventually (i.e. reach instruction L_{p+1})

and (ii) when M halts, the top stack register t_k will have the value $f(x_1, \ldots, x_m)$.

I.15 Definition: For $u,v : \mathbb{N} \to \mathbb{N}$, non-decreasing and such that $u(1), v(1) \geq 1$, Space(u,v) is defined to be the class of functions as follows.

A function f is in Space(u,v) just if there is an SRM M which computes f and satisties this condition: that there are $j, \ell \geq 1$ such that, for all $x_1, \ldots, x_m \in \mathbb{N}$, if M starts off at L_1 with $r = t_0 = \ldots$ $\ldots = t_k = 0$ and values $x_1, \ldots, x_m$ in the input registers then at all times during M's computation

$$r < u^{<j>}(\max(x_1, \ldots, x_m) + 1)$$

and for all $0 \leq i \leq k$

$$t_i < v^{<\ell>}(\max(x_1, \ldots, x_m) + 1)$$

where $u^{<j>}, v^{<\ell>}$ are the j^{th} and ℓ^{th} iterates of u and v respectively.

Chapter II Space($\underline{1}$,□) and $\Delta_0^{\mathbb{N}}$

The main result of this chapter (theorem II.13) connects the class $\Delta_0^{\mathbb{N}}$ (or rather the Δ_0-functions) with a certain complexity class - Space($\underline{1}$,□) - of SRM-computable functions.

This result is one of the equivalences in theorem 6 of [Bel'tyukov]. It is, in the notation of that paper, that

$$\text{Rd} = \text{Srm}^{-}(\text{Space}, p_2^{<i>}) \quad \text{for any } i.$$

The proof, and that is to say the whole of this chapter, is taken from the proof of theorem 1 of [Bel'tyukov], particularly the part

$$\text{Srm}(\text{Space}, f^{<i>}) \subseteq \mathcal{E}f.$$

The $\text{SRM}_{\mathfrak{F}}$ introduced here is very similar to the Srm_f defined there (they are not however the same). Srm_f's do in fact turn up in lemma II.8, though they're not identified as such. $\text{SRM}_{\mathfrak{F}}$'s are handy tools, once lemma II.3 has been proved, for showing that given functions are in Space(u,v). This approach leads to lemma II.7: all Δ_0-functions are in Space($\underline{1}$,□).

The rest of the chapter simply follows Bel'tyukov's proof. This provides lemma II.12, the converse to lemma II.7.

The main result merely combines lemmas II.7 and II.12.

II.1 <u>Definition</u>: Suppose $\mathfrak{F}$ is a class of functions. We define the $\text{SRM}_{\mathfrak{F}}$, a variant of the ordinary SRM (definition I.14), as follows:

An $\text{SRM}_{\mathfrak{F}}$ still has registers $r, t_0, \ldots, t_k, x_1, \ldots, x_m$ and its program is still of the form $L_1 L_2 \ldots L_p L_{p+1}$, where L_{p+1} is <u>halt</u>, but $L_1, \ldots, L_p$ now take one of the following forms:

<u>Type (i)</u>: $t_i := t_i + 1;$

<u>Type (i)'</u>: $t_i := f(r, t_{i+1}, \ldots, t_k, x_1, \ldots, x_m);$

<u>Type (ii)'</u>: $r := f(r, t_0, \ldots, t_k, x_1, \ldots, x_m);$

<u>Type (iii)'</u>: If $f_1(r, t_0, \ldots, t_k, x_1, \ldots, x_m) = f_2(r, t_0, \ldots, t_k, x_1, \ldots, x_m)$ then L_ξ else L_η;

where f, f_1, f_2 must be functions in $\mathfrak{F}$. For each $0 \leq i \leq k$ at most one type (i)/(i)' instruction affecting t_i appears in the program and such an instruction sets $t_j := 0$ for all $j < i$ even if the value t_i does not change.

The initial configuration of such a machine has $r = t_0 = \ldots = t_k = 0$ and the inputs to the machine in $x_1, \ldots, x_m$. The first instruction executed is L_1.

The output will generally be the value in a specified stack upon halting - usually the top stack (t_k) or the bottom stack (t_0).

Am $SRM_{\mathfrak{F}}$ computes a given function if it always halts with the value of the function (given the inputs as arguments) in the top stack.

Notice that we can change a machine with output in the bottom stack into one that gives the same output in its top stack (and indeed vice versa). Moreover this new machine will run in the same space bounds (defined below) as the first.

So to prove a function is in a given space complexity class it is enough to show that there is an $SRM_{\mathfrak{F}}$ which halts with the value of the function in the bottom stack.

We define space complexity classes exactly as for ordinary SRM's (definition I.15).

II.2 <u>Definition</u>: If $u,v : \mathbb{N} \rightarrow \mathbb{N}$ are non-decreasing with $u(1), v(1) \geq 1$, then $g : \mathbb{N}^m \rightarrow \mathbb{N}$ is in <u>$\text{Space}_{\mathfrak{F}}(u,v)$</u> if there is an $SRM_{\mathfrak{F}}$ which computes g and throughout any run of this machine:

$$r < u^{<j>}(\max(x_1, \ldots, x_m) + 1)$$
$$t_i < v^{<\ell>}(\max(x_1, \ldots, x_m) + 1)$$

for all $0 \leq i \leq k$, for some $j, \ell > 0$.

II.3 <u>Lemma</u>: Suppose that $u,v : \mathbb{N} \rightarrow \mathbb{N}$ are non-decreasing with $u(1), v(1) \geq 1$ and such that

$$v(x) \geq x \quad \forall x \in \mathbb{N}$$

$$\forall i,j > 0 \quad \exists k > 0 \qquad u^{<i>}(x), u^{<j>}(x) \leq u^{<k>}(x) \quad \forall x \in \mathbb{N}$$
$$\forall i,j > 0 \quad \exists k > 0 \qquad u^{<i>}(v^{<j>}(x)) \leq u^{<k>}(x) \quad \forall x \in \mathbb{N}$$
$$\forall i > 0 \quad \exists j > 0 \qquad u^{<i>}(x) \leq v^{<j>}(x) \quad \forall x \in \mathbb{N}$$

Then

$$\mathrm{Space}_{\mathrm{Space}(u,v)}(u,v) = \mathrm{Space}(u,v).$$

Proof:

$\subseteq$: Suppose we have an $SRM_{Space(u,v)}$ M_1 which halts on all inputs and runs in bounds $(u^{<j>}, v^{<\ell>})$ - i.e. the work-register is bounded by $u^{<j>}(\max(x_1,\ldots,x_m) + 1)$ and the stack registers are all bounded by $v^{<\ell>}(\max(x_1,\ldots,x_m) + 1)$. We show that we can pick any (i)'/(ii)'/(iii)' line of M_1's program and replace this line with a block of ordinary SRM instructions to create a new program. In addition to M_1's registers this new program may refer to new stack registers which we add to M_1's registers to create appropriate machinery. This new machine will simulate M_1 in that the net effect of the new block on the original registers will be the same as the effect of the line it replaces. The new machine will be bounded by $(u^{<j'>}, v^{<\ell'>})$, for some $j', \ell' > 0$ and so the (u,v) space bounds are preserved. Finally, for each stack register of the new machine, there will be, in the new program, one type (i) instruction referring to it or one type (i)' instruction or none at all. So it will be an $SRM_{Space(u,v)}$.

We can carry on and replace each (i)'/(ii)'/(iii)' line of M_1's program with a block of ordinary SRM instructions, adding at the same time any new stack registers that are needed, and end up with an ordinary SRM which computes the same function as M_1 and runs in (u,v) bounds (the top stack of the final machine will originally have been the top stack of M_1).

Suppose then that M_1 is an $SRM_{Space(u,v)}$ with registers $r, t_0, \ldots$ $\ldots, t_k, x_1, \ldots, x_m$ and bounds $(u^{<j>}, v^{<\ell>})$. And let's say that the type (i)'/(ii)'/(iii)' instruction we wish to replace is L_ω.

<u>Case (i)'</u>: L_ω is $t_i := f(r, t_{i+1}, \ldots, t_k, x_1, \ldots, x_m);$.

$f \in \mathrm{Space}(u,v)$, so we know that there is an SRM M_2 computing f and running in bounds $(u^{<j'>}, v^{<\ell'>})$. We can assume (though this is not the usual labelling) that M_2 has input registers $w_{s+1}, t_{i+1}, \ldots, t_k, x_1, \ldots$ $\ldots, x_m$; stack registers $w_0, \ldots, w_s$; and work-register r. M_2 halts with $f(w_{s+1}, t_{i+1}, \ldots, t_k, x_1, \ldots, x_m)$ in w_s.

In constructing the new machine M_3, we first add some stack registers. These are $w_0, \ldots, w_s, w_{s+1}, w_{s+2}, t_{i+1}, t_{i+2}$. The ordering on the stacks (which is what determines when they are set to zero) is the one that puts w's lower than t's and otherwise works by suffix.

The replacement block is written out on the next page. We omit instruction labels when possible - which is not often.

```
$L_\omega$      If $t_{i+1} + w_{s+2} = w_{s+2}$ then $\alpha_0$ else $\delta$;
$\alpha_0$      If $t_0 + w_{s+2} = w_{s+2}$ then $\alpha_1$ else $\gamma$;
$\alpha_1$      If $t_1 + w_{s+2} = w_{s+2}$ then $\alpha_2$ else $\gamma$;
.........................................
$\alpha_i$      If $t_i + w_{s+2} = w_{s+2}$ then $\beta_0$ else $\gamma$;       Ensure that $t_0,\ldots,t_i,w_0,\ldots,w_{s+1}$
$\beta_0$       If $w_0 + w_{s+2} = w_{s+2}$ then $\beta_1$ else $\gamma$;       are set to zero.
.........................................                                       ($w_{s+1}$ is always zero.)
$\beta_{s+1}$   If $w_{s+1} + w_{s+2} = w_{s+2}$ then $\varepsilon$ else $\gamma$;
$\gamma$        $t_{i+1} := t_{i+1} + 1$;
                If $w_{s+2} + w_{s+2} = w_{s+2}$ then $\varepsilon$ else $\varepsilon$;
$\delta$        $t_{i+1} := t_{i+1} + 1$;
$\varepsilon$   If $w_{s+1} + w_{s+2} = r$ then $\eta$ else $\zeta$;
$\zeta$         $w_{s+1} := w_{s+1} + 1$;                                        Set $w_{s+1} := r$.
                If $w_{s+2} + w_{s+2} = w_{s+2}$ then $\varepsilon$ else $\varepsilon$;
$\eta$          $r := w_{s+2}$;                                                  Sets r to zero.
                || Here $M_2$'s program (less the final ||                       Puts the value
                || halt instruction $L_\nu$) is listed.  ||                      $f(w_{s+1},t_{i+1},\ldots,t_k,$
                || We assume the instruction labels      ||                           $x_1,\ldots,x_m)$
                || are different from any others in     ||                      into $w_s$.
                || the rest of the new program.          ||
$L_\nu$         $r := w_{s+1}$;                              Sets r back to its original value.
$\theta$        If $t_i + w_{s+2} = w_s$ then $L_{\omega+1}$ else $\iota$;           Sets $t_i := f(r,$
$\iota$         $t_i := t_i + 1$;                                                $t_{i+1},\ldots,t_k,x_1,\ldots,x_m)$.
                If $w_{s+2} + w_{s+2} = w_{s+2}$ then $\varepsilon$ else $\varepsilon$;
```

What is the effect of this block on the registers? Well first of all w_{s+2} is always zero since there is no type (i)/(i)' instruction to change it. It's best to consider $t_{i+\frac{1}{4}}$ and $t_{i+\frac{3}{4}}$ next.

During an execution of the block at most one of $t_{i+\frac{1}{4}}$, $t_{i+\frac{3}{4}}$ will be increased, and only once at that. This is because (without going outside the block) there is no way back to line γ or line δ from the later parts of the block and no way from γ to δ or vice versa.

Second thing to notice is that during the first execution of the block after an assignment to $t_{i+1}, \ldots, t_{k-1}$ or t_k, or after M_3 starts, if either of $t_{i+\frac{1}{4}}, t_{i+\frac{3}{4}}$ is increased it will be $t_{i+\frac{1}{4}}$. For in these circumstances $t_{i+\frac{3}{4}} = 0$ and so we bypass δ.

When we reach ε we will have

$$w_0 = w_1 = \ldots = w_s = w_{s+1} = w_{s+2} = t_0 = \ldots = t_i = 0,$$

for the only circumstance in which neither $t_{i+\frac{1}{4}}$ nor $t_{i+\frac{3}{4}}$ is increased (thus setting all these other stacks to zero) is that they are all already zero.

We can immediately see (since the block from ε onwards can do nothing to alter this) that the overall effect of the block on $t_0, \ldots, t_{i-1}$ is to set them to zero, while $t_{i+1}, \ldots, t_k$ are unaffected. This is of course what we want.

As to t_i: it rises to $f(r, t_{i+1}, \ldots, t_k, x_1, \ldots, x_m)$ in increments of one, starting from zero. Notice that whenever we reach ε, we have

$$w_0 = \ldots = w_{s+1} = 0,$$

for we will just have left the section ending in δ or will just have executed ι. Thus ε, ζ won't cause a loop and M_2's program will run properly with $f(w_{s+1}, t_{i+1}, \ldots, t_k, x_1, \ldots, x_m)$ ending up in w_s.

r is unchanged by the block as a whole for the last instruction to affect it is always L_ν. (w_{s+1} is unaffected by runs of M_2).

This block, then, does what we want. Does M_3 satisfy the right bounds though?

$t_0, \ldots, t_k$ satisfy the same bound, $v^{<\ell>}(\max(x_1, \ldots, x_m) + 1)$, as before. w_{s+2} is always zero.

w_{s+1} has the same bound as the work-register had in runs of M_1. This bound is $u^{<j>}(\max(x_1, \ldots, x_m) + 1)$. But then, by our conditions on u,v, there is an $\ell_1 > 0$ such that $v^{<\ell_1>}(\max(x_1, \ldots, x_m) + 1)$ is also a bound.

$w_0, \ldots, w_s$ are bounded by

$$\begin{aligned} & v^{<\ell'>}(\max(w_{s+1}, t_{i+1}, \ldots, t_k, x_1, \ldots, x_m) + 1) \\ \leq\ & v^{<\ell'>}(v^{<\max(\ell,\ell_1)>}(\max(x_1, \ldots, x_m) + 1)) \\ =\ & v^{<\ell_2>}(\max(x_1, \ldots, x_m) + 1) \qquad \text{for some } \ell_2 > 0. \end{aligned}$$

Meanwhile for the work-register we know that outside the execution of the listing of M_2's program r is bounded by $u^{<j>}(\max(x_1, \ldots, x_m) + 1)$.

During the "run" of M_2 r is bounded by

$$\begin{aligned} & u^{<j'>}(\max(w_{s+1}, t_{i+1}, \ldots, t_k, x_1, \ldots, x_m) + 1) \\ \leq\ & u^{<j'>}(v^{<\max(\ell,\ell_1)>}(\max(x_1, \ldots, x_m) + 1) \\ \leq\ & u^{<j_1>}(\max(x_1, \ldots, x_m) + 1) \qquad \text{for some } j_1 > 0. \end{aligned}$$

Overall, then, r will be bounded by

$$u^{<j_2>}(\max(x_1, \ldots, x_m) + 1) \qquad \text{for some } j_2 > 0.$$

This leaves $t_{i+\frac{1}{4}}$ and $t_{i+\frac{3}{4}}$ to consider. $t_{i+\frac{3}{4}}$ is clearly bounded by 2 (since its maximum value is 1). It will certainly be bounded by $v(\max(x_1, \ldots, x_m) + 1)$ except in one circumstance: that this bound itself has value 1. But in this case $w_0, \ldots, w_{s+1}, t_0, \ldots, t_i$, whose bounds we have already established, will have to be zero throughout M_3's run and so $t_{i+\frac{3}{4}}$ won't ever be increased and will thus still be bounded by

$$v(\max(x_1, \ldots, x_m) + 1).$$

As to $t_{i+\frac{1}{4}}$: go back for a moment to M_1 and assume that at two different times in a run L_ω is executed on the same values of $r, t_{i+1}, \ldots$

$\ldots, t_k$. On both occasions M will be in the same configuration following execution of L_ω. This implies a loop and therefore it never happens, for M_1 halts on all inputs. As far as M_3 is concerned this means that between successive encounters with the new block at least one of $r, t_{i+1}, \ldots$ $\ldots, t_k$ must change.

If one of the $t_{i+1}, \ldots, t_k$ changes then the second encounter will not change $t_{i+\frac{1}{4}}$ from the value zero which it thereby acquires, for, as we saw earlier, if one of $t_{i+\frac{1}{4}}, t_{i+\frac{3}{4}}$ rises in these circumstances it will be $t_{i+\frac{3}{4}}$.

On the other hand, as we have already seen, r is bounded by $v^{<\ell_1>}(\max(x_1, \ldots, x_m) + 1)$ outside executions of the block. Or putting it another way r cannot take more than $v^{<\ell_1>}(\max(x_1, \ldots, x_m) + 1)$ different values without a change in $t_{i+1}, \ldots, t_k$. So (since M_1 always halts implies M_3 always halts) the block cannot be executed more than $v^{<\ell_1>}(\max(x_1, \ldots, x_m) + 1)$ times in succession without a change in $t_{i+1}, \ldots$ $\ldots, t_k$. Now for at least one of these executions $t_{i+\frac{1}{4}}$ will not be increased (for there must first be an execution which sets $t_{i+\frac{3}{4}} := 1$). Thus $t_{i+\frac{1}{4}}$ cannot rise more than $v^{<\ell_1>}(\max(x_1, \ldots, x_m) + 1) - 1$ times without being set back to zero. That is to say $t_{i+\frac{1}{4}}$ is bounded by $v^{<\ell_1>}(\max(x_1, \ldots, x_m) + 1)$.

We have now shown that M_3 has bounds $(u^{<j_2>}, v^{<\ell_2>})$.

Cases (ii)'/(iii)': The replacement blocks for the other two instruction types are similar.

$\subseteq$: The second part of the proof is trivial since to calculate $f \in \text{Space}(u,v)$ we simply use the $\text{SRM}_{\text{Space}(u,v)}$ which has stack t_0, inputs $x_1, \ldots, x_m$ and program

```
1.  $t_0 := f(x_1, \ldots, x_m)$;
2.  halt.
```

Since $f \in \text{Space}(u,v)$ it must itself be bounded by $v^{<\ell>}$ for some

$\ell > 0$, and so this machine runs in the right bounds.

End of proof of lemma II.3.

II.4 <u>Proposition</u>: For u,v satisfying the conditions of lemma II.3 $\mathrm{Space}(u,v)$ is closed under composition.

<u>Proof</u>: Suppose $f(x_1,\dots,x_n)$, $g_1(x_1,\dots,x_m)$, ..., $g_n(x_1,\dots,x_m)$ are all in Space(u,v). Then $f(g_1(x_1,\dots,x_m),\dots,g_n(x_1,\dots,x_m))$ is in $\mathrm{Space}_{\mathrm{Space}(u,v)}(u,v)$ and is the bottom stack output of the following machine:

The machine has registers $t_0,t_1,\dots,t_n,x_1,\dots,x_m$ and program

```
1.      $t_n := g_n(x_1,\dots,x_m)$;
2.      $t_{n-1} := g_{n-1}(x_1,\dots,x_m)$;
..................................
n.      $t_1 := g_n(x_1,\dots,x_m)$;
n+1.    $t_0 := f(t_1,\dots,t_n)$;
n+2.    halt.
```

Because $f, g_1, \dots, g_n \in \mathrm{Space}(u,v)$ they must be bounded by iterates of v. That is there are $j_0,\dots,j_n$ such that

$$f(x_1,\dots,x_m) < v^{<j_0>}(\max(x_1,\dots,x_m)+1) \quad \forall x_1,\dots,x_m \in \mathbb{N}$$

$$g_i(x_1,\dots,x_m) < v^{<j_i>}(\max(x_1,\dots,x_m)+1) \quad \forall x_1,\dots,x_m \in \mathbb{N},$$

for all $1 \leq i \leq n$.

And now if $j_{max} = \max(j_1,\dots,j_n)$ our machine will be bounded by $(\underline{1},v^{<j_0+j_{max}>})$ thus proving that $f(g_1,\dots,g_n) \in \mathrm{Space}_{\mathrm{Space}(u,v)}(u,v)$.

And so, by lemma II.3, $f(g_1,\dots,g_n) \in \mathrm{Space}(u,v)$.

II.5 <u>Definition</u>: For $u,v : \mathbb{N} \to \mathbb{N}$, non-decreasing with $u(1),v(1) \geq 1$, $\mathrm{Space}_*(u,v)$ is defined to be the class of sets whose characteristic functions are in Space(u,v).

II.6 <u>Proposition</u>: $\Delta_0^{\mathbb{N}} \subseteq \mathrm{Space}_*(\underline{1},\square)$ where $\square \equiv (x+1)^2$ and $\underline{1}(x) = x$ for all $x \in \mathbb{N}$. That is to say the characteristic function of any $\Delta_0^{\mathbb{N}}$ set is computable by a workregisterless SRM with polynomially bounded stack registers.

Proof: All the machines in this proof will have output in the bottom stack.

(i) Functions defined by LA-terms are in Space($\underline{1}$,$\Box$).

LA-terms are built up from variables and the constants 0 and 1 using addition and multiplication.

Projection: For any $m \geq i \geq 1$, the function defined by x_i is in Space($\underline{1}$,$\Box$). It is computed by the machine with stacks t_0,t_1, inputs $x_1,\ldots,x_m$ and program

```
1.      If $t_0 + t_1 = x_i$ then 4. else 2.;
2.      $t_0 := t_0 + 1$;
3.      If $t_1 + t_1 = t_1$ then 1. else 1.;
4.      halt.
```

The bounds are obviously satisfied.

Constants: For any $m \geq 0$, the function defined by 0 is computed by the machine with stack t_0, inputs $x_1,\ldots,x_m$ and program

```
1.      halt.
```

For any $m \geq 0$, the function defined by 1 is computed by the machine with stack t_0, inputs $x_1,\ldots,x_m$ and program

```
1.      $t_0 := t_0 + 1$;
2.      halt.
```

Again the bounds are satisfied in both these cases.

To conclude part (i) of this proof it suffices now, because of proposition II.4 (whose conditions $\underline{1}$,$\Box$ satisfy), to show that addition and multiplication are in Space($\underline{1}$,$\Box$).

Addition/multiplication: The machine has stack t_0, inputs x_1,x_2 and program

```
1.      If $x_1 * x_2 = t_0$ then 4. else 2.;
2.      $t_0 := t_0 + 1$;
3.      If $t_0 + t_0 = t_0$ then 1. else 1.;
```

```
4.      halt.
```

(where $*$ is $+$ or $\cdot$ as appropriate). The bounds are satisfied.

Although it is not necessary for the current proof, we remark here that:

(ii) $\mathrm{Space}_\star(\underline{1},\Box)$ is closed under explicit transformations. This is because all elements of $\mathbb{N}$ are represented by LA-terms (either 0 or $1 + 1 + \ldots + 1$) and $\mathrm{Space}(\underline{1},\Box)$ contains the projection functions and is closed under composition.

(iii) Sets defined by basic LA-formulae are in $\mathrm{Space}_\star(\underline{1},\Box)$.

Given (i) all we require to show here is that the characteristic functions of equality and inequality are in $\mathrm{Space}(\underline{1},\Box)$.

For equality we have the SRM with stacks t_0, t_1, inputs x_1, x_2 and program

```
1.      If $x_1 + t_1 = x_2$ then 3. else 2.;
2.      $t_0 := t_0 + 1$;
3.      halt.
```

For inequality (i.e. $x_1 \leqslant x_2$) we have the SRM with stacks t_0, t_1, inputs x_1, x_2 and program

```
1.      If $t_1 + t_0 = x_1$ then 6. else 2.;
2.      If $t_1 + t_0 = x_2$ then 5. else 3.;
3.      $t_1 := t_1 + 1$;
4.      If $t_0 + t_0 = t_0$ then 1. else 1.;
5.      $t_0 := t_0 + 1$;
6.      halt.
```

Clearly the bounds are satisfied in both cases.

Again, though it is not needed at present, we remark as a special case of (iii) that:

(iv) $\mathrm{Space}_\star(\underline{1},\Box)$ contains $=$, $\leqslant$, graph$(+)$ and graph$(\cdot)$.

(v) $\mathrm{Space}_\star(\underline{1},\Box)$ is closed under boolean operations.

For this it suffices to know that $f, g \in \mathrm{Space}(\underline{1}, \Box)$ where

$$f(x) = \begin{cases} 1 & \text{if } x = 0 \\ 0 & \text{otherwise.} \end{cases} \qquad \text{(This corresponds to negation.)}$$

and

$$g(x_1, x_2) = \begin{cases} 0 & \text{if } x_1 = x_2 = 0 \\ 1 & \text{otherwise.} \end{cases} \qquad \text{(Conjunction.)}$$

This is easily shown.

(vi) Finally we show that $\mathrm{Space}_*(\underline{1}, \Box)$ is closed under bounded quantification (II).

Since we have negation it suffices to show closure under bounded existential quantification.

Suppose then that $f(x_1, \ldots, x_m) \in \mathrm{Space}(\underline{1}, \Box)$ and $\mathrm{Range}(f) \subseteq \{0,1\}$ and let $\tau(x_1, \ldots, x_m)$ be an LA-term. We wish to show that the characteristic function of

$$\exists w \leq \tau(x_1, \ldots, x_m) \big(f(x_1, \ldots, x_{i-1}, w, x_{i+1}, \ldots, x_m) = 0 \big)$$

is in $\mathrm{Space}(\underline{1}, \Box)$.

We have already seen in (i) that $\tau(x_1, \ldots, x_m) \in \mathrm{Space}(\underline{1}, \Box)$ and we have the following $\mathrm{SRM}_{\mathrm{Space}(\underline{1}, \Box)}$ to calculate the desired function. The stacks are t_0, t_1 and the inputs $x_1, \ldots, x_m$. The program is

```
1.    If $f(x_1,\ldots,x_{i-1},t_1,x_{i+1},\ldots,x_m) = 0$ then 6. else 2.;
2.    If $t_1 = \tau(x_1,\ldots,x_m)$ then 5. else 3.;
3.    $t_1 := t_1 + 1$;
4.    If $0 = 0$ then 1. else 1.;
5.    $t_0 := t_0 + 1$;
6.    halt.
```

The stacks are bounded by $\max(2, \tau(x_1, \ldots, x_m) + 1)$

$$\leq \Box^{<j>}(\max(x_1, \ldots, x_m) + 1) \qquad \text{for some } j > 0.$$

The work-register is not used. Therefore the function is in $\mathrm{Space}(\underline{1}, \Box)$ by lemma II.3.

End of proof of proposition II.6.

II.7 Lemma: All Δ_0-functions are in Space($\underline{1}$,$\Box$).

Proof: Suppose then that the function $f(x_1,\dots,x_m)$ has $\Delta_0^{\mathbb{N}}$ graph A and is polynomially bounded in its arguments. From proposition II.6 we know that the characteristic function of A, χ_A, is in Space($\underline{1}$,$\Box$) (as is 0). So we can compute f using the following $\mathrm{SRM}_{\mathrm{Space}(\underline{1},\Box)}$: the stack is t_0, the inputs are $x_1,\dots,x_m$ and the program is

```
1.    If $\chi_A(x_1,\dots,x_m,t_0) = 0$ then 4. else 2.;
2.    $t_0 := t_0 + 1$;
3.    If $0 = 0$ then 1. else 1.;
4.    halt.
```

The machine always halts because it just increases t_0 until $t_0 = f(x_1,\dots,x_m)$. f is bounded by $\Box^{<j>}$ for some j and thus t_0 is bounded by $\Box^{<j>}(\max(x_1,\dots,x_m) + 1)$. Therefore $f \in$ Space($\underline{1}$,$\Box$) by lemma II.3.

End of proof of lemma II.7.

We now move towards proving the converse of lemma II.7.

II.8 Proposition:

If M is an SRM with work-register r, stack registers $t_0,\dots,t_k$ and input registers $x_1,\dots,x_m$ and M halts on all inputs

then M's program can be replaced by an equivalent one of the form

```
L:  (If $(r,\vec{t},\vec{x}) \in A_0$ then $(r := f_0(r,\vec{t},\vec{x});\ t_0 := t_0 + 1;\ L)$;
     If $(r,\vec{t},\vec{x}) \in A_1$ then $(r := f_1(r,\vec{t},\vec{x});\ t_1 := t_1 + 1;\ L)$;
     ..........................................................
     If $(r,\vec{t},\vec{x}) \in A_k$ then $(r := f_k(r,\vec{t},\vec{x});\ t_k := t_k + 1;\ L)$;
     $r := f_{k+1}(r,\vec{t},\vec{x})$)
```

where $A_0,\dots,A_k \in \Delta_0^{\mathbb{N}}$ and $f_0,\dots,f_{k+1}$ are Δ_0-functions while $\vec{t}$ and $\vec{x}$ abbreviate $t_0,\dots,t_k$ and $x_1,\dots,x_m$ respectively. This new program has exactly the same effects upon the registers as the old one.

Proof: We observe of M that, at any time , its future actions are entirely determined by the values $r,\vec{t},\vec{x}$ and the (number of the) instruction about to be executed (the configuration). So e.g. at the beginning M's whole run is determined by the values $0,\vec{0},\vec{x}$ and the fact that L_1 is about to be executed (the initial configuration).

Such decisions as M makes depend upon the basic LA-formulae embodied within its type (iii) instructions.

Now in the absence of type (i) executions only the value of r and the number of the instruction about to be executed can change. r only changes by application of a type (ii) instruction assigning to it one of the values $\vec{t},\vec{x}$. Therefore, since r can only assume this finite number of values, the actions of M up to the next type (i) execution (or up to the execution of L_{p+1}) are always determined by the number of the instruction about to be executed and the truth or falsity of a finite number of basic formulae in the variables $r,\vec{t},\vec{x}$ (where r is the current value in the work-register). It is similarly determined which of the values $r,\vec{t},\vec{x}$ will be in the work-register just before this next type (i) execution (or halt). That value is therefore a Δ_0-function of $r,\vec{t},\vec{x}$.

Observe finally that if a type (i) instruction has just been executed then we know which configuration we are in from the lowest non-zero stack register. For this must have been the last to have been increased and there is a unique type (i) instruction which does this.

End of proof of proposition II.8.

Proposition II.8 has introduced a new type of machine with registers like an SRM but a different kind of program. Lemmas II.10 and II.11 will reveal more about these machines. First however we have a fact to be used in the proof of lemma II.10.

II.9 Fact: If $A \subseteq \mathbb{N}^n$ is in $\Delta_0^{\mathbb{N}}$ and $f_1(\vec{x}),\ldots,f_n(\vec{x})$ are Δ_0-functions then

$$\{(\vec{x}) : (f_1(\vec{x}),\dots,f_n(\vec{x})) \in A\} \in \Delta_0^{\mathbb{N}} .$$

<u>Proof</u>: For $1 \leq q \leq n$ (by the definition of a Δ_0-function) there is $\Theta_q \in \Delta_0$ and an LA-term τ_q such that Θ_q defines graph(f_q) and τ_q defines the polynomial bound for f_q. But now let $T \equiv \tau_1 + \dots + \tau_n$ so T is an LA-term and define $\Psi \in \Delta_0$ by

$$\Psi \equiv \exists\, w_1,\dots,w_n \leq T(\Theta_1 \wedge \dots \wedge \Theta_n \wedge \Phi) \qquad \text{where } \Phi \text{ defines } A.$$

Clearly Ψ will define the new set as required.

II.10 <u>Lemma</u>: (The "looping lemma")

<u>If</u> M is a machine with registers $t_0,\dots,t_k,x_1,\dots,x_m$, where $k \geq 1$, and program

```
L:  (If  $(\vec{t},\vec{x}) \in A_0$  then  $(t_0 := t_0 + 1;\ L)$;
     If  $(\vec{t},\vec{x}) \in A_1$  then  $(t_1 := t_1 + 1;\ L)$;
     .......................................
     If  $(\vec{t},\vec{x}) \in A_k$  then  $(t_k := t_k + 1;\ L)$)
```

where $A_0,\dots,A_k \in \Delta_0^{\mathbb{N}}$

And during any run of M the registers $\vec{t}$ are bounded by $p(\vec{x}) \in \mathbb{N}[\vec{x}]$

<u>Then</u> there is a machine M', with registers $t_1,\dots,t_k,\vec{x}$ and program

```
L':  (If  $(t_1,\dots,t_k,\vec{x}) \in A'_1$  then  $(t_1 := t_1 + 1;\ L')$;
      If  $(t_1,\dots,t_k,\vec{x}) \in A'_2$  then  $(t_2 := t_2 + 1;\ L')$;
      ................................................
      If  $(t_1,\dots,t_k,\vec{x}) \in A'_k$  then  $(t_k := t_k + 1;\ L')$)
```

$(A'_0,\dots,A'_k \in \Delta_0^{\mathbb{N}})$, which simulates M in that if, at some point in a run of M, $t_1,\dots,t_k$ (and $\vec{x}$) have the same values as at some point in a run of M' then the next register out of $t_1,\dots,t_k$ that M increases is also the next register increased by M' (or if M halts without changing $t_1,\dots,t_k$ then M' just halts). Thus, if we ignore the increases in t_0 that M can execute but M' cannot, M and M' are equivalent machines. In particular the final value of

t_k (i.e. the function computed) will be the same in both cases and furthermore M' is bounded by $p(\vec{x})$.

Proof: (i.e. derivation of the sets $A'_1,\ldots,A'_k$)

Let f be defined by $f(t_1,\ldots,t_k,\vec{x}) =$

$$\begin{cases} \text{the least number } y \leq p(\vec{x}) \text{ such that } (y,t_1,\ldots,t_k,\vec{x}) \notin A_0 \\ 0 \quad \text{if no such number exists.} \end{cases}$$

f is clearly a Δ_0-function. And f has this meaning. Suppose M has just increased one of $t_1,\ldots,t_k$ (or has just started). This will be followed by a (possibly empty) sequence of increases in t_0. The first of these increases is from $t_0 = 0$ to $t_0 = 1$ for t_0 must have just been set to zero. The last increase will be to the least y such that $(y,t_1,\ldots,t_k,\vec{x}) \notin A_0$ and when $(\vec{t},\vec{x}) \notin A_0$ either M is about to halt or $(\vec{t},\vec{x})$ is in at least one of $A_1,\ldots,A_k$. This least y must be less than $p(\vec{x})$ for otherwise M would increase t_0 past its bound. There may however be values of $\vec{t},\vec{x}$, not occurring in any actual run of M, for which the bound does not hold. It is for this reason that f was defined as it was. The next of $t_1,\ldots,t_k$ to be increased is thus determined by which of $A_1,\ldots,A_k$ the tuple $(f(t_1,\ldots,t_k,\vec{x}),t_1,\ldots,t_k,\vec{x})$ lies in. But now, by fact II.9, we can define, for $1 \leq j \leq k$,

$$A'_j = \{(x_1,\ldots,x_{m+k}) : (f(x_1,\ldots,x_{m+k}),x_1,\ldots,x_{m+k}) \in A_j\}.$$

(The projection functions are of course Δ_0-functions.)

End of proof of lemma II.10.

Lemma II.10 provides the inductive step in the proof of proposition II.11.

II.11 Proposition: All machines satisfying the conditions in lemma II.10 calculate Δ_0-functions.

Proof: By induction on the number (k+1) of stack registers.

If $k = 0$: The program must be of the form

$$L: \quad [\text{If} \quad (t_0,\vec{x}) \in A_0 \quad \text{then} \quad [t_0 := t_0 + 1;\ L]].$$

The function calculated is

$$\mu\, y \le p(\vec{x})\ ((y,\vec{x}) \notin A_0)$$

and this is clearly a Δ_0-function.

If true for k: Suppose that M has registers $t_0,\dots,t_{k+1},x_1,\dots,x_m$. Then by lemma II.10 there is a machine M', satisfying the required conditions and with $k + 1$ stack registers $t_0,\dots,t_{k+1}$, which calculates the same function as M. By the inductive hypothesis M' calculates a Δ_0-function and therefore so does M.

End of proof of proposition II.11.

II.12 Lemma: All the functions in $\mathrm{Space}(\underline{1},\Box)$ are Δ_0-functions.

Proof: By lemma II.8 (we omit all mention of r for it is always zero) we may replace any SRM running in bounds $(\underline{1},\Box)$ ($\Box$-bounds $\equiv$ polynomial bounds) by an equivalent machine which satisfies the conditions of lemma II.10. And then by proposition II.11 the function calculated must be a Δ_0-function.

End of proof of lemma II.8.

II.13 Theorem: [Bel'tyukov]

$\mathrm{Space}(\underline{1},\Box)$ is precisely the class of Δ_0-functions.

Proof: Lemma II.7 and lemma II.12.

End of proof of theorem II.13.

The ensuing chapters are mostly devoted to extending this result but the basic methods used will not really change from the ones used in this chapter.

Chapter III COUNTING

In the previous chapter we were mostly concerned with the class $\Delta_0^{\mathbb{N}}$ (and the closely related class of the Δ_0-functions). This class of sets of numbers underlies most of the classes under consideration here; nearly all the others we are interested in contain it and possess its closure properties.

This chapter introduces a new type of closure property, instances of which can be used in defining set-classes, usually by combining one or more of them with the $\Delta_0^{\mathbb{N}}$ closure properties.

Whether or not these new definitions actually give rise to new classes seems hard to show. But settling this question positively, in other words showing that certain of these classes are larger than $\Delta_0^{\mathbb{N}}$ (or indeed showing that any two of them are different from each other), would show separation of some existing classes from $\Delta_0^{\mathbb{N}}$.

Summation

Start with a set $A \subseteq \mathbb{N}^m$ (some $m \geqslant 1$). We have a characteristic function $\chi_{A'}$ such that

$$\chi_{A'}(x_1,\ldots,x_m) = \begin{cases} 1 & \text{if } (x_1,\ldots,x_m) \in A \\ 0 & \text{otherwise.} \end{cases}$$

(Note: this is really the characteristic function of $\mathbb{N}^m \setminus A$.)

We can use this function to define others:

For $1 \leqslant i \leqslant m$

$$\sigma_A^i(x_1,\ldots,x_m) = \sum_{y=0}^{x_i-1} \chi_{A'}(x_1,\ldots,x_{i-1},y,x_{i+1},\ldots,x_m)$$

Alternatively

$$\sigma_A^i(x_1,\ldots,x_m) = |A \cap \{x_1\}\times\ldots\times\{x_{i-1}\}\times\underset{\sim}{x}_i\times\{x_{i+1}\}\times\ldots\times\{x_m\}|.$$

(Recall $\underset{\sim}{x}_i = \{0,1,\ldots,x_i-1\}$.)

If A is recursive so will σ_A^i be. Moving down below the classes of recursive sets and functions we can ask, for a given class of sets $\mathcal{C}$: If

$A \in \mathcal{C}$ is there a limit to how complicated σ^i_A can be?

In fact, since we start with a set (i.e. A), it seems more convenient to consider graph(σ^i_A) rather than σ^i_A itself. This leads to:

III.1 <u>Definition</u>: A class of sets $\mathcal{C}$ is <u>closed under counting</u> if, for all $m \in \mathbb{N}$, for all $1 \leq i \leq m$, for all $A \subseteq \mathbb{N}^m$, $A \in \mathcal{C}$ implies

graph(σ^i_A) $\in \mathcal{C}$.

Now it seems that $\mathcal{C}$ has to be pretty weak before we can hope to find an A (and an i) such that $A \in \mathcal{C}$ and graph(σ^i_A) $\notin \mathcal{C}$. For instance

$A \in$ linear space implies graph(σ^i_A) $\in$ linear space

and $A \in \mathcal{E}^0_*$ implies graph(σ^i_A) $\in \mathcal{E}^0_*$.

However it is not known whether the same can be said for $\Delta^{\mathbb{N}}_0$. Thus closure under counting or the lack of it presents itself as a possible means of separating $\Delta^{\mathbb{N}}_0$ from these two larger classes. (linear space is the class of sets accepted by Turing machines running in linear space bounds and $\mathcal{E}^0_*$ is the Grzegorczyk class of that name. Both these classes contain $\Delta^{\mathbb{N}}_0$; they can, for instance, be expressed as complexity classes of Bel'tyukov SRM's, which are clearly larger than Space($\underline{1}$,0) [Bel'tyukov].)

<u>Counting modulo k</u>

From the function σ^i_A we can derive other, weaker notions of counting which may also prove to be means of showing this same separation (if it exists).

In the first place we can consider, instead of σ^i_A, σ^i_A mod k for given k (recall x mod k is the number j such that $0 \leq j \leq k-1$ and $x \equiv j$ mod k).

III.2 <u>Definition</u>: We say a class $\mathcal{C}$ is <u>closed under counting modulo k</u> if for all $m \in \mathbb{N}$, for all $1 \leq i \leq m$, for all $A \subseteq \mathbb{N}^m$,

$A \in \mathcal{C}$ implies graph(σ^i_A mod k) $\in \mathcal{C}$.

Graph(σ^i_A mod k) is of course Δ_0-derivable from graph(σ^i_A). (That is to say: the latter can be derived from the former using =, $\leq$, graph(+),

graph(·), explicit transformations, boolean operations and bounded quantification (II).) So, in any class containing $\Delta_0^{\mathbb{N}}$ and closed under the Δ_0 set operations, closure under counting implies closure under counting modulo k, for all $k \in \mathbb{N}$. Therefore linear space and $\mathcal{E}_*^0$ are closed under counting modulo k for all values of k.

Here we can note that in [Paris-Wilkie] it is shown that there are two classes very similar to $\Delta_0^{\mathbb{N}}$ and such that

(i) The first class is not enlarged by closing under counting modulo k for any $k \in \mathbb{N}$ (this is closure under the whole family of these new closure properties).

(ii) The second class is enlarged by closing under counting modulo 2 (just one of this family of closure properties, and the simplest).

Because of the close similarity of these two classes to $\Delta_0^{\mathbb{N}}$ (they are obtained from it by relativization) these results suggest it will be hard to prove a result analogous to either one when we go on to consider $\Delta_0^{\mathbb{N}}$.

To date it is not known that $\Delta_0^{\mathbb{N}}$ is closed under counting modulo k for any $k \in \mathbb{N}$. Nor is it known that $\Delta_0^{\mathbb{N}}$ is not closed under counting modulo k for any $k \in \mathbb{N}$.

So this idea has not yet produced separation of $\Delta_0^{\mathbb{N}}$ from linear space and $\mathcal{E}_*^0$, although it may eventually do so.

Nonetheless there are some interesting machine characterizations of classes, close to $\Delta_0^{\mathbb{N}}$ and defined with the aid of closure under various kinds of counting. Not all these kinds of counting have been introduced yet and so we continue with some more definitions.

Counting modulo X

Let's return to the notion of characteristic function and extend it. Suppose that X is a finite set on which is defined a binary operation. Then we consider functions from numbers to X.

Any $\langle chi \rangle : \mathbb{N}^m \to X$ will be a kind of characteristic function for it will define a partition of $\mathbb{N}^m$ into $|X|$ subsets.

Again we can "sum" this function $\langle chi \rangle$ to get new functions

$$\mathrm{Sigma}^i_{\langle chi \rangle} \quad \text{for each} \quad 1 \leq i \leq m.$$

$\mathrm{Sigma}^i_{\langle chi \rangle}$ is defined to be

$$\begin{aligned}
&\langle chi \rangle(x_1,\ldots,x_{i-1},x_i-1,x_{i+1},\ldots,x_m)\dotplus\\
&\quad\dotplus\bigl(\langle chi \rangle(x_1,\ldots,x_{i-1},x_i-2,x_{i+1},\ldots,x_m)\dotplus\\
&\qquad\dotplus\bigl(\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
&\qquad\quad\dotplus\bigl(\langle chi \rangle(x_1,\ldots,x_{i-1},1,x_{i+1},\ldots,x_m)\dotplus\\
&\qquad\qquad\dotplus\langle chi \rangle(x_1,\ldots,x_{i-1},0,x_{i+1},\ldots,x_m)\bigr)\ldots\bigr)\bigr)
\end{aligned}$$

where $\dotplus$ is the binary operation on X. If this operation is associative we need not worry about bracketing. This defines $\mathrm{Sigma}^i_{\langle chi \rangle}$ unless $x_i = 0$. For given X we shall simply define $\mathrm{Sigma}^i_{\langle chi \rangle}(x_1,\ldots,x_{i-1},0,x_{i+1},\ldots,x_m)$ to be some arbitrary element of X. If X possesses a right identity, i.e. an element e such that $x \dotplus e = x$ for all $x \in X$, then it is convenient to select e as this arbitrary element and we will do this where possible.

Now we want to use this "summation" of elements of a set - which is not necessarily a set of numbers - to define a closure property applicable to classes of sets of numbers.

As you will see if you refer to the eventual definition of this closure property (definition III.7), it seems sensible to require a certain strength of a class $\mathcal{C}$ before we can think of whether or not it is closed under counting modulo X.

The conditions on $\mathcal{C}$ are that it include $=$ and that it be closed under explicit transformations and boolean operations.

The useful properties that this condition ensures are proved in the next two propositions.

III.3 <u>Proposition</u>: If $\mathcal{C}$ includes $=$ and is closed under explicit transformations and boolean operations then

$\mathcal{C}$ includes all finite sets of (tuples of) numbers.

Proof: Let $A \subseteq \mathbb{N}^m$ (we do require that all tuples in a set be of the same arity). If A is finite we can list A as

$$\{(n_1^1,\ldots,n_m^1),(n_1^2,\ldots,n_m^2),\ldots,(n_1^{|X|},\ldots,n_m^{|X|})\}$$

for appropriate $n_j^q \in \mathbb{N}$, $1 \le q \le |X|$, $1 \le j \le m$.

For $1 \le q \le |X|$, $1 \le j \le m$ define

$$Y_j^q = \{(x_1,\ldots,x_m) : x_j = n_j^q\}.$$

$Y_j^q \in \mathcal{C}$ since it is defined from $=$ using an explicit transformation.

Now define $Y^q = \bigcap_{1 \le j \le m} Y_j^q$ for $1 \le q \le |X|$. These sets are in $\mathcal{C}$ because they are derived from the Y_j^q using repeated boolean operations.

To finish we have

$$A = \bigcup_{1 \le q \le |X|} Y^q$$

and so $A \in \mathcal{C}$ by closure under boolean operations.

III.4 Proposition: Let $\mathcal{C}$ include $=$ and be closed under explicit transformations and boolean operations.

Let f be a function whose domain is a finite set (of q-tuples) of numbers.

Let $g_1,\ldots,g_q : \mathbb{N}^m \to \mathbb{N}$ be functions taking values in the domain of f and let $\mathrm{graph}(g_1),\ldots, \mathrm{graph}(g_q) \in \mathcal{C}$.

Then

$$\mathrm{graph}(f(g_1,\ldots,g_q)) \in \mathcal{C}.$$

Proof: For $1 \le i \le q$ define $Y^i \in \mathcal{C}$ to be

$\{(x_1,\ldots,x_m,x_{m+1},x_{m+2},\ldots,x_{m+q+1}) : (x_1,\ldots,x_m,x_{m+i+1}) \in \mathrm{graph}(g_i)\}$.

Then define $Z \in \mathcal{C}$ to be

$\{(x_1,\ldots,x_{m+q+1}) : (x_{m+2},\ldots,x_{m+q+1},x_{m+1}) \in \mathrm{graph}(f)\}$. ($\mathrm{graph}(f)$ must be finite and so in $\mathcal{C}$ by proposition III.3.)

Next is $U \in \mathcal{C}$ which is given by

$$U = Z \cap \bigcap_{1 \le i \le q} Y^i.$$

And for all $n_1,\dots,n_q \in \mathbb{N}$, $V(n_1,\dots,n_q) \in \mathcal{C}$ where
$V(n_1,\dots,n_q) = \{(x_1,\dots,x_m,x_{m+1}) : (x_1,\dots,x_m,x_{m+1},n_1,\dots,n_q) \in U\}$.

Finally

$$\mathrm{graph}(f(g_1,\dots,g_q)) = \bigcup_{(n_1,\dots,n_q)\ \mathrm{dom}(f)} V(n_1,\dots,n_q)$$

and so is in $\mathcal{C}$, since dom(f) is finite.

This proves III.4. We could have abbreviated the process above by

$(x_1,\dots,x_m,x_{m+1}) \in \mathrm{graph}(f(g_1,\dots,g_q))$ if and only if

$$\bigvee_{(n_1,\dots,n_q)\ \mathrm{dom}(f)} \{(x_1,\dots,x_m,n_1)\in\mathrm{graph}(g_1) \wedge \dots \wedge (x_1,\dots,x_m,n_q)\in\mathrm{graph}(g_q) \wedge (n_1,\dots,n_q,x_{m+1})\in\mathrm{graph}(f)\}.$$

Propositions III.3 and III.4 provide a background for the following idea of defining, as a set of tuples of numbers, the graph of a function, which, like <chi>, has numbers as arguments but takes values in some finite set which may not always be a subset of $\mathbb{N}$.

The idea is to code this finite set by a set of numbers.

III.5 Definition: If <chi> : $\mathbb{N}^m \to X$, then for ¢ : $X \rightarrowtail \mathbb{N}$ (i.e. ¢ is one-one)

graph$_¢$(<chi>) $=_{\mathrm{def.}}$ graph(φ•<chi>).

The point is now that

III.6 Proposition: For all finite sets X, for all $m \in \mathbb{N}$, for all <chi> : $\mathbb{N}^m \to \mathbb{N}$, for all $\phi_1,\phi_2 : X \rightarrowtail \mathbb{N}$, for all classes $\mathcal{C}$ containing = and closed under explicit transformations and boolean operations:

graph$_{\phi_1}$(<chi>) iff graph$_{\phi_2}$(<chi>) .

Proof:

There will be a function $f : \mathrm{Range}(\phi_1) \to \mathbb{N}$ such that

$f•¢_1 = \phi_2$.

And so graph$_{¢_1}$(<chi>) = graph($¢_1$ • <chi>) $\in \mathcal{C}$ implies, by proposition III.4, since f has a finite domain, that

graph(f•ϕ_1 • <chi>) $\in \mathcal{C}$, and this is just

$$\text{graph}(\phi_2 \bullet \text{<chi>}) = \text{graph}_{\phi_2}(\text{<chi>}).$$

By symmetry we also have

$$\text{graph}_{\phi_2}(\text{<chi>}) \in \mathcal{C} \quad \text{implies} \quad \text{graph}_{\phi_1}(\text{<chi>}) \in \mathcal{C}.$$

Proposition III.6 tells us that we can forget about specific codings, for if $\text{graph}_{\phi}(\text{<chi>}) \in \mathcal{C}$ for any one suitable function ϕ, then the same will be true for any other such suitable function. In the main therefore it is possible to drop the ϕ part and speak of graph(<chi>) tout court.

And now we can define our closure property.

III.7 Definition: For X a finite set with binary operation, a class of sets $\mathcal{C}$ which includes = and is closed under explicit transformations and boolean operations is said to be closed under counting modulo X if, for all $\text{<chi>} : \mathbb{N}^m \to X$,

$$\text{graph(<chi>)} \in \mathcal{C} \quad \text{implies} \quad \text{graph}(\text{Sigma}^i_{\text{<chi>}}) \in \mathcal{C}.$$

III.8 Example: X is the group $\mathbb{Z}_k = \{\bar{0}, \bar{1}, \dots, \overline{(k-1)}\}$ under addition $(k \geq 1)$. If $\mathcal{C}$ includes = and is closed under explicit transformations and boolean operations then,

$\mathcal{C}$ is closed under counting modulo $\mathbb{Z}_k$

if and only if

$\mathcal{C}$ is closed under counting modulo (the number) k.

Proof: (i) Closure under counting modulo $\mathbb{Z}_k$ implies closure under counting modulo k.

Recall definition III.2 and suppose $A \in \mathcal{C}$. Define $\text{<chi>} : \mathbb{N}^m \to \mathbb{Z}_k$ by

$$\text{<chi>}(x_1, \dots, x_m) = \begin{cases} \bar{0} & \text{if } (x_1, \dots, x_m) \notin A \\ \bar{1} & \text{otherwise.} \end{cases}$$

Clearly $A \in \mathcal{C}$ implies $\text{graph(<chi>)} \in \mathcal{C}$.

And then closure under counting modulo $\mathbb{Z}_k$ implies that, for all $1 \leq i \leq m$, $\text{graph}(\text{Sigma}^i_{\text{<chi>}}) \in \mathcal{C}$.

But it is easily seen that $\text{Sigma}^i_{\text{<chi>}}(x_1, \dots, x_m)$ is the equivalence class containing $\sigma^{i}_{A}(x_1,\ldots,x_m)$ and hence

$$\mathrm{graph}(\sigma^{i}_{A} \bmod k) \in \mathcal{C}.$$

(Bear in mind our convention that $\mathrm{Sigma}^{i}_{\text{<chi>}}(x_1,\ldots,x_{i-1},0,x_{i+1},\ldots,x_m)=\bar{0}$.)

(ii) Counting modulo k implies counting modulo $\mathbb{Z}_k$.

Suppose that $\text{<chi>} : \mathbb{N}^m \to \mathbb{Z}_k$ and that $\mathrm{graph}(\text{<chi>}) \in \mathcal{C}$. We want to show that $\mathrm{graph}(\mathrm{Sigma}^{i}_{\text{<chi>}}) \in \mathcal{C}$ for all $1 \leq i \leq m$. By the properties of $\mathcal{C}$ we have

$$A_1,\ldots,A_{(k-1)} \in \mathcal{C}$$

where $(x_1,\ldots,x_m) \in A_j$ iff $\text{<chi>}(x_1,\ldots,x_m) = \bar{\ell}$ for some $j \leq \ell \leq k-1$.

If $\sigma^{i}_{1},\ldots,\sigma^{i}_{(k-1)}$ are the summation functions derived from $A_1,\ldots$ $\ldots,A_{(k-1)}$ for given i then $\mathrm{graph}(\sigma^{i}_{1} \bmod k),\ldots,\mathrm{graph}(\sigma^{i}_{(k-1)} \bmod k) \in \mathcal{C}$ and furthermore $\mathrm{Sigma}^{i}_{\text{<chi>}}(x_1,\ldots,x_m)$ is the equivalence class containing

$$((\sigma^{i}_{1}(x_1,\ldots,x_m) \bmod k) + \ldots + (\sigma^{i}_{(k-1)}(x_1,\ldots,x_m) \bmod k)).$$

Therefore since there are only finitely many possible values of $\sigma^{i}(x_1,\ldots,x_m) \bmod k, \ldots, \sigma^{i}_{(k-1)}(x_1,\ldots,x_m) \bmod k,$

$$\mathrm{graph}(\mathrm{Sigma}^{i}_{\text{<chi>}}) \in \mathcal{C} \text{ by proposition III.4.}$$

Because we have example III.8 we shall, from now on, usually talk about closure under counting modulo $\mathbb{Z}_k$ rather than modulo k.

III.9 Theorem: If $\mathcal{C}$ is a class including = and closed under explicit transformations and boolean operations. And if X and Y are finite sets with binary operations. Then:

(i) $X \cong Y$ implies that

$\mathcal{C}$ is closed under counting modulo X

if and only if

$\mathcal{C}$ is closed under counting modulo Y.

(ii) $Y \subseteq X$ (so the operation on Y is just the operation on X restricted to elements of Y) implies that

if $\mathcal{C}$ is closed under counting modulo X

then $\mathcal{C}$ is closed under counting modulo Y.

(iii) $\mathcal{C}$ is closed under counting modulo X and $\mathcal{C}$ is closed under counting modulo Y

if and only if

$\mathcal{C}$ is closed under counting modulo X × Y.

Proof:

(i) If $\theta : X \cong Y$ and $\text{<chi>} : \mathbb{N}^m \to Y$, then $\phi = \theta^{-1}\circ\text{<chi>} : \mathbb{N}^m \to X$ and $\text{Sigma}^i_{\text{<chi>}}(x_1,\dots,x_m) = \theta\circ\text{Sigma}^i_{\phi}(x_1,\dots,x_m)$ for all $1 \le i \le m$, for all $(x_1,\dots,x_m) \in \mathbb{N}^m$. And so by proposition III.4, since θ (for any coding functions for X and Y) has a finite domain, closure under counting modulo X implies closure under counting modulo Y. The result follows by symmetry.

(ii) This part is equally trivial since a function $\text{<chi>}: \mathbb{N}^m \to Y$ may also be thought of as a function $\text{<chi'>} : \mathbb{N}^m \to X$ and

$$\text{Sigma}^i_{\text{<chi>}} = \text{Sigma}^i_{\text{<chi'>}} \quad \text{for all } 1 \le i \le m.$$

(iii) This part has slightly more substance.

If $\pi_1 : X\times Y \to X$, $\pi_2 : X\times Y \to Y$ are the projection functions then they have finite domains.

And now if $\text{<chi>} : \mathbb{N}^m \to X\times Y$ is such that $\text{graph}(\text{<chi>}) \in \mathcal{C}$, then by proposition III.4

$$\text{graph}(\pi_1\circ\text{<chi>}),\ \text{graph}(\pi_2\circ\text{<chi>}) \in \mathcal{C}.$$

Also (suppressing the variables $x_1,\dots,x_{i-1},x_{i+1},\dots,x_m$)

$$\text{Sigma}^i_{\text{<chi>}}(x_i) = (\text{Sigma}^i_{\pi_1\circ\text{<chi>}}(x_i),\text{Sigma}^i_{\pi_2\circ\text{<chi>}}(x_i))$$

for all $1 \le i \le m$, for all $(x_1,\dots,x_m) \in \mathbb{N}^m$.

Thus by proposition III.4, closure under counting modulo X and modulo Y implies closure under counting modulo X × Y.

For the implication in the other direction, suppose $\mathcal{C}$ is closed under counting modulo X × Y and pick some element $y \in Y$.

If we define $f : X \to X\times Y$ by $f(x) = (x,y)$ then by proposition III.3,

$\text{<chi>} : \mathbb{N}^m \to X$ and $\text{graph}(\text{<chi>}) \in \mathcal{C}$ together imply that

$$\text{graph}(f\bullet\text{<chi>}) \in \mathcal{C}$$

and therefore

$$\text{graph}(\text{Sigma}^{i}_{f\bullet\text{<chi>}}) \in \mathcal{C} \quad \text{for all} \quad 1 \leq i \leq m.$$

Thus, again using proposition III.3,

$$\text{graph}(\text{Sigma}^{i}_{\text{<chi>}}) \in \mathcal{C}$$

because $\text{Sigma}^{i}_{\text{<chi>}} = \pi_{1}\bullet\text{Sigma}^{i}_{f\bullet\text{<chi>}}$.

This shows counting modulo X. Counting modulo Y is obtained likewise.

Theorem III.9 enables us to see other closure properties of a class which we know to be closed under counting modulo some set X (with binary operation) by looking at X as a set.

If we restrict ourselves to the case that X is a group, as we do for most of the rest of this chapter, we can obtain stronger results of this sort.

III.10 <u>Theorem</u>: Let $\mathcal{C}$ be a class of sets of numbers including = and closed under explicit transformations and boolean operations.

Let H and J be finite groups with $J \triangleleft H$.

Then:

$\mathcal{C}$ is closed both under counting modulo J and under counting modulo H/J

if and only if

$\mathcal{C}$ is closed under counting modulo H.

<u>Proof</u>:

(i) Assume closure under counting modulos J and H/J.

We want to show that for all $\text{<chi>} : \mathbb{N}^{m} \to H$, $\text{graph}(\text{<chi>}) \in \mathcal{C}$ implies that $\text{graph}(\text{Sigma}^{i}_{\text{<chi>}}) \in \mathcal{C}$ for all $1 \leq i \leq m$.

Now, suppressing the variables $x_{1},\ldots,x_{i-1},x_{i+1},\ldots,x_{m}$, we can write $\text{Sigma}^{i}_{\text{<chi>}}(x_{i})$ as $\text{<chi>}(x_{i}-1)\cdot\text{<chi>}(x_{i}-2)\cdot\ldots\cdot\text{<chi>}(1)\cdot\text{<chi>}(0)$, where $\cdot$ is the group operation on H. So we're involved in multiplying a sequence of elements of H. If we're dealing with the multiplication of say three elements a_{1}, a_{2}, a_{3} of H then there is no problem for $\text{graph}(a_{1}\cdot a_{2}\cdot a_{3})$

is a finite set and we can apply proposition III.3.

$\text{Sigma}^{i}_{<\text{chi}>}$ does not of course reduce to multiplication of any fixed finite number of elements of H. It requires multiplication of x_i such elements. But we do know that we can perform these arbitrarily long multiplications for sequences of elements of J and of H/J. And here's how we use this ability to do the same for H itself.

Notice first of all that it is possible, in $\mathcal{C}$, to tell which element of H/J (i.e. which coset of J) $\text{Sigma}^{i}_{<\text{chi}>}(x_i)$ lies in. For we can define $\chi : \mathbb{N}^m \to H/J$ by $\chi(x_1,\ldots,x_m) = <\text{chi}>(x_1,\ldots,x_m)\cdot J$ (the coset containing $<\text{chi}>(x_i)$) and by proposition III.4 graph(<chi>)$\in \mathcal{C}$ implies graph(χ)$\in \mathcal{C}$. By closure under counting modulo H/J graph(Sigma^{i}_{χ})$\in \mathcal{C}$ and clearly

$$\text{Sigma}^{i}_{<\text{chi}>}(x_i) \in \text{Sigma}^{i}_{\chi}(x_i).$$

So $\mathcal{C}$ can tell us which element of H/J $\text{Sigma}^{i}_{<\text{chi}>}(x_i)$ lies in. The problem comes in narrowing $\text{Sigma}^{i}_{\chi}(x_i)$ to the single element $\text{Sigma}^{i}_{<\text{chi}>}(x_i)$. We achieve this as follows.

First we choose a fixed element r_x from each coset x of J. (It is convenient that the element chosen from J itself is $\underline{\text{id}}_H$.) Every element of H can be expressed as the product of an element of $\{r_x\}_{x\in H/J}$ and an element of J. In fact for all $h \in H$ there is a unique $j \in J$ such that $h = r_{h\cdot J}\cdot j$. Thus, given <chi>, there is a function $j : \mathbb{N}^m \to J$ such that

$$<\text{chi}>(x_i) = r_{\chi(x_i)} \cdot j(x_i) \qquad \text{for all } x_1,\ldots,x_m \ \ \mathbb{N}.$$

graph(j)$\in \mathcal{C}$ because j is the composition of <chi> with a function whose domain is finite.

Now we can rewrite $\text{Sigma}^{i}_{<\text{chi}>}(x_i)$ as

$$r_{\chi(x_i-1)}\cdot j(x_i-1)\cdot r_{\chi(x_i-2)}\cdot j(x_i-2)\cdot\ldots\cdot r_{\chi(1)}\cdot j(1)\cdot r_{\chi(0)}\cdot j(0).$$

Our aim is to work on this product until we make it into something we know we can calculate in $\mathcal{C}$ - i.e. something involving finite products of elements of H and arbitrarily long products of elements of J and of

H/J. To do this we use the fact that:

For all $x,y \in H/J$, for all $j \in J$, there is a $j' \in J$ such that

$$r_x \cdot j \cdot r_y = r_{xy} \cdot j'.$$

This is because $r_x \cdot j \cdot r_y \in xy \cdot J$.

In fact we can think of this as defining a function

$$\zeta : (H/J) \times J \times (H/J) \to J$$

where for $x,y \in H/J$, $j \in J$ $\quad r_x \cdot j \cdot r_y = r_{xy} \cdot \zeta(x,j,y)$.

Using ζ we can, very loosely speaking, jump elements of J over elements of $\{r_x\}_{x \in H/J}$. For instance take a product of three elements of H. As we have already seen, the graph of such a product must in any case be in $\mathcal{C}$ but the idea we use now can be extended to arbitrarily long products. Starting with our three elements of H in the form $r_{x(t)} \cdot j(t)$ for $t = 0,1,2$, we have

$$\begin{aligned}
& r_{x(2)} \cdot j(2) \cdot r_{x(1)} \cdot j(1) \cdot r_{x(0)} \cdot j(0) \\
= {} & r_{x(2)} \cdot j(2) \cdot r_{x(1)x(0)} \cdot j'(1) \cdot j(0) \quad \text{for } j'(1) = \zeta(x(1),j(1),x(0)) \\
= {} & r_{x(2)x(1)x(0)} \cdot j'(2) \cdot j'(1) \cdot j(0) \\
& \qquad\qquad \text{for } j'(2) = \zeta(x(2),j(2),(x(1)x(0)))
\end{aligned}$$

and we have reduced the problem to multiplying a product of elements of J by an element of H easily obtained from a product of elements of H/J.

We can now see that in general

$$\begin{aligned}
& r_{\chi(x_i-1)} \cdot j(x_i-1) \cdot r_{\chi(x_i-2)} \cdot j(x_i-2) \cdot \ldots \cdot r_{\chi(1)} \cdot j(1) \cdot r_{\chi(0)} \cdot j(0) \\
= {} & r_{\chi(x_i-1)\chi(x_i-2)\ldots\chi(1)\chi(0)} \cdot j'(x_i-1) \cdot j'(x_i-2) \cdot \ldots \cdot j'(1) \cdot j'(0)
\end{aligned}$$

where for all $x_i \in \mathbb{N}$ (,for all $x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_m \in \mathbb{N}$)

$$j'(x_i) = \zeta(\chi(x_i),j(x_i),\mathrm{Sigma}^i_\chi(x_i)).$$

(Notice that $j'(0) = j(0)$ because $\mathrm{Sigma}^i_\chi(0) = J$ by our convention.) $\mathrm{graph}(j') \in \mathcal{C}$ by proposition III.4 because ζ has a finite domain.

And so, since

$$\mathrm{Sigma}^i_{<chi>}(x_i) = r_{\mathrm{Sigma}^i_\chi(x_i)} \cdot \mathrm{Sigma}^i_{j'}(x_i),$$

$$\mathrm{graph}(\mathrm{Sigma}^i_{<chi>}) \in \mathcal{C} \quad \text{(also by proposition III.4).}$$

(ii) The implication in the other direction is very much simpler.

Assume closure under counting modulo H.

That we have closure under counting modulo J is a direct application of theorem III.9 (ii).

Suppose then that <chi> : $\mathbb{N}^m \to H/J$ and graph(<chi>)$\in \mathcal{C}$. Define $\chi: \mathbb{N}^m \to H$ by

$$\chi(x_1,\dots,x_m) = r_{\text{<chi>}(x_1,\dots,x_m)}$$

where $\{r_x\}_{x\in H/J}$ is as in part (i).

graph(χ)$\in\mathcal{C}$ and if $\mathcal{C}$ is closed under counting modulo H then

$$\text{graph}(\text{Sigma}_\chi^i)\in\mathcal{C} \text{ for all } 1 \leq i \leq m.$$

But then since

$$\text{Sigma}_\chi^i(x_i) \in \text{Sigma}_{\text{<chi>}}^i(x_i)$$

it is easy to see that

$$\text{graph}(\text{Sigma}_{\text{<chi>}}^i)\in\mathcal{C} \text{ for all } 1 \leq i \leq m.$$

End of proof of theorem III.10.

Some consequences of theorem III.10

Recall that a group is simple if it has no proper normal subgroups.

Recall that $H_1,\dots,H_k$ is a composition series for G if

$H_1 = \{\underline{id}_G\}$, $H_k = G$, $H_1 \triangleleft H_2 \triangleleft \dots \triangleleft H_{k-1} \triangleleft H_k$

and for all $1 \leq i \leq k-1$ H_{i+1}/H_i is simple.

Every finite group has a composition series (since it must have a maximal normal subgroup and this will produce a factor group which is simple).

Therefore

III.11 Corollary: If $\mathcal{C}$ includes = and is closed under explicit transformations and boolean operations then closure under counting modulo G, G a finite group, is equivalent to closure under counting modulo each of some (finite collection) of finite simple groups. (Notice that counting modulo $\{\underline{id}\}$ is always possible.)

Recall also that G is solvable if it has a composition series such that, for $1 \leq i \leq k-1$, H_{i+1}/H_i is abelian (and simple).

Now the finite abelian simple groups are (up to isomorphism) just the $\mathbb{Z}_p$, p a prime.

Thus

III.12 Corollary: If $\mathcal{C}$ includes $=$ and is closed under explicit transformations and boolean operations then closure under counting modulo G, for G a finite solvable group, is equivalent to closure under counting modulo $\mathbb{Z}_m$, m square-free.

Proof: Clearly, by theorems III.9 and III.10, counting modulo G is equivalent to counting modulo each of some collection of finite abelian simple groups. That is: modulo $\mathbb{Z}_{p_1}, \ldots, \mathbb{Z}_{p_n}$ for some primes $p_1, \ldots, p_n$, which we may assume to be all different.

But then it can be shown that

$$\mathbb{Z}_{p_1 p_2 \cdots p_n} \cong \mathbb{Z}_{p_1} \times \mathbb{Z}_{p_2} \times \ldots \times \mathbb{Z}_{p_n}$$

and so, since $p_1 p_2 \ldots p_n$ is square-free, the corollary follows.

$X\Delta_0^{\mathbb{N}}$

Let us now use counting to define new - or rather possibly new - classes of sets based on $\Delta_0^{\mathbb{N}}$.

If X is a set with a binary operation then

III.13 Definition: $X\Delta_0^{\mathbb{N}}$ is defined to be the smallest class $\mathcal{C}$ of sets of numbers such that

(i) $\mathcal{C}$ includes $=$, $\leq$, graph$(+)$, graph$(\cdot)$

(ii) $\mathcal{C}$ is closed under explicit transformations

(iii) $\mathcal{C}$ is closed under boolean operations

(iv) $\mathcal{C}$ is closed under bounded quantification (II)

(v) $\mathcal{C}$ is closed under counting modulo X.

Because $X\Delta_0^{\mathbb{N}}$ always satisfies the conditions of theorems III.9 and III.10 we have the following:

III.14 <u>Examples</u>:

(i) $\mathbb{Z}_4\Delta_0^{\mathbb{N}} = \mathbb{Z}_2\Delta_0^{\mathbb{N}}$.

This is because

$$\{\bar{0}\} \triangleleft \{\bar{0},\bar{2}\} \triangleleft \mathbb{Z}_4 \cong \{\bar{0},\bar{1},\bar{2},\bar{3}\}$$

and $\{\bar{0},\bar{2}\} \cong \mathbb{Z}_4/\{\bar{0},\bar{2}\} \cong \mathbb{Z}_2$.

(ii) $S_3\Delta_0^{\mathbb{N}} = \mathbb{Z}_6\Delta_0^{\mathbb{N}}$

since $\{\underline{id}\} \triangleleft A_3 \triangleleft S_3$

and $A_3 \cong \mathbb{Z}_3$, $S_3/A_3 \cong \mathbb{Z}_2$.

(iii) $S_4\Delta_0^{\mathbb{N}} = \mathbb{Z}_6\Delta_0^{\mathbb{N}}$

since

$\{\underline{id}\} \triangleleft \{\underline{id},(12)(34)\} \triangleleft \{\underline{id},(12)(34),(13)(24),(14)(23)\} \triangleleft A_4 \triangleleft S_4$

and $\{\underline{id},(12)(34)\} \cong \mathbb{Z}_2$,

$\{\underline{id},(12)(34),(13)(24),(14)\ 23)\}/\{\underline{id},(12)(34)\} \cong \mathbb{Z}_2$,

$A_4/\{\underline{id},(12)(34),(13)(24),(14)(23)\} \cong \mathbb{Z}_3$,

and $S_4/A_4 \cong \mathbb{Z}_2$.

In the chapters following we use these examples to obtain some perhaps rather surprising results for space complexity classes of SRM's and similar machines.

Chapter IV   Counting modulo ${}^{n}n$ and modulo S_n

Having brought out certain equivalences by concerning ourselves with counting modulo some group, we turn our attention to the monoid (see footnote) ${}^{n}n$ of functions with domain and codomain $\underset{\sim}{n} = \{0,1,\ldots,n-1\}$ under the binary operation of composition.

The following theorem connects counting modulo ${}^{n}n$ and counting modulo S_n.

IV.1 Theorem: Suppose $\mathcal{C}$ is a class of sets of numbers which includes $=$, $<$ and graph(suc) and is closed under explicit transformations, boolean operations and bounded quantification (I). Then, for $n \geq 1$:

$\mathcal{C}$ is closed under counting modulo ${}^{n}n$

if and only if

$\mathcal{C}$ is closed under counting modulo S_n.

Proof:

(i) Closure under counting modulo ${}^{n}n$ implies closure under counting modulo S_n.

This is trivial by theorem III.9 for S_n is a submonoid of ${}^{n}n$.

(ii) Closure under counting modulo S_n implies closure under counting modulo ${}^{n}n$.

This part proceeds by induction on n. The inductive step requires the following lemmas IV.2 and IV.3:

IV.2 Lemma: If $\mathcal{C}$ includes $=$ and graph(suc) and is closed under explicit transformations, boolean operations and bounded quantification (I), then, for $n \geq 1$:

Closure under counting modulo ${}^{n}n$ implies closure under counting modulo $({}^{n+1}n{+}1 \setminus S_{n+1})$.

Footnote: A monoid is a semigroup with identity.
A semigroup is a set with a binary associative operation.

(N.B. We can count modulo $\left({}^{n+1}n+1 \setminus S_{n+1}\right)$ because this set of functions is closed with respect to composition.)

Proof: Suppose $f : (\underset{\sim}{n+1}) \to (\underset{\sim}{n+1})$ but $f \notin S_{n+1}$. Then f cannot be one-one because the only one-one functions in ${}^{n+1}n+1$ are the permutations.

It follows that there are functions $g_1 : (\underset{\sim}{n+1}) \to \underset{\sim}{n}$ and $g_2 : \underset{\sim}{n} \to (\underset{\sim}{n+1})$ such that

$$g_2 \circ g_1 \circ f = f,$$

for since f is not one-one there is some $w < n$ such that

$$w \notin \text{Range}(f)$$

and we can define g_1 by

$$g_1(z) = \begin{cases} z & \text{if } z \leq w \\ z-1 & \text{if } w < z \leq n, \end{cases}$$

while g_2 can be

$$g_2(z) = \begin{cases} z & \text{if } z < w \\ z+1 & \text{if } w \leq z < n. \end{cases}$$

These functions depend upon our choice of w of course, but since there are only a finite number of functions f to consider, there are functions α and β with finite domains where

$$\alpha : \left({}^{n+1}n+1 \setminus S_{n+1}\right) \to {}^{n+1}n$$

and $$\beta : \left({}^{n+1}n+1 \setminus S_{n+1}\right) \to {}^{n}n+1$$

and for all $f \in \left({}^{n+1}n+1 \setminus S_{n+1}\right)$,

$$\beta(f) \circ \alpha(f) \circ f = f.$$

Now if, rather than a single function, we have a sequence

$$f_0, f_1, f_2, \ldots$$

of elements of $\left({}^{n+1}n+1 \setminus S_{n+1}\right)$, then for $y \in \mathbb{N}$ we can calculate

$$f_{y-1} \circ f_{y-2} \circ \ldots \circ f_1 \circ f_0$$

as follows:

For $z \geq 1$ define g_z by

$$g_z = \alpha(f_z) \circ f_z \circ \beta(f_{z-1}).$$

Then

$$
\begin{aligned}
& g_{y-1}\circ\dots\bullet g_1 \\
= {} & \alpha(f_{y-1})\bullet f_{y-1}\bullet\beta(f_{y-2})\circ\alpha(f_{y-2})\bullet f_{y-2}\bullet\dots\bullet\alpha(f_1)\bullet f_1\circ\beta(f_0) \\
= {} & \alpha(f_{y-1})\bullet f_{y-1}\circ f_{y-2}\circ\dots\circ f_1\bullet\beta(f_0)
\end{aligned}
$$

and so

$$
\begin{aligned}
& \beta(f_{y-1})\circ g_{y-1}\bullet\dots\bullet g_1\circ\alpha(f_0)\ f_0 \\
= {} & f_{y-1}\bullet\dots\bullet f_1\circ f_0 .
\end{aligned}
$$

But $g_1, g_2, g_3, \dots$ is only a sequence of elements of ${}^{n}n$. So composing sequences of elements of ${}^{n}n$ enables us to compose sequences of elements of $({}^{n+1}n{+}1 \setminus S_{n+1})$.

The foregoing presents the idea of the proof proper which proceeds as follows:

Suppose then that $\mathcal{C}$ is a class including $=$, graph(suc) and closed under explicit transformations, boolean operations, bounded quantification (I) and counting modulo ${}^{n}n$.

Suppose also that we have a function $\text{<chi>} : \mathbb{N}^{m} \to ({}^{n+1}n{+}1 \setminus S_{n+1})$ and that $\text{graph(<chi>)} \in \mathcal{C}$.

If we suppress the variables $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_m$ we can write
$\text{Sigma}^{i}_{\text{<chi>}}(x_i) = \text{<chi>}(x_i-1)\bullet\text{<chi>}(x_i-2)\bullet\dots\circ\text{<chi>}(1)\circ\text{<chi>}(0)$
and we want to show that

$$\text{graph}(\text{Sigma}^{i}_{\text{<chi>}}) \in \mathcal{C}.$$

For this we define three other functions γ, δ_i and χ_i.
$\gamma : \mathbb{N}^{m} \to {}^{n+1}n$ and, suppressing $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_m$,

$$\gamma(x_i) =_{\text{def.}} \alpha(\text{<chi>}(x_i)).$$

$\text{graph}(\gamma) \in \mathcal{C}$ by proposition III.4.

If c is some arbitrary element of ${}^{n}n{+}1$ then, for $1 \leq i \leq m$,
$\delta_i : \mathbb{N}^{m} \to {}^{n}n{+}1$ is defined by
$\delta_i(x_i) = y$ iff $(\exists z \leq x_i (z+1 = x_i \wedge \beta(\text{<chi>}(z)) = y)) \vee (x_i = 0 \wedge y = c)$.
As we saw likewise in the proof of lemma III.4 this formula tells us that

$\mathrm{graph}(\delta_i) \in \mathcal{C}$, for it is built up from $\mathrm{graph}(\beta\bullet\text{<chi>})$, $=$ and $\mathrm{graph}(\mathrm{suc})$, all of which are in $\mathcal{C}$, by steps corresponding to closure properties of $\mathcal{C}$. For $x_i \geq 1$,

$$\delta_i(x_i) = \beta(\text{<chi>}(x_i-1)).$$

Finally $\chi_i : \mathbb{N}^m \to {}^n n$ is defined, for $1 \leq i \leq m$, by

$$\chi_i(x_i) = \begin{cases} \gamma(x_i)\bullet\text{<chi>}(x_i)\bullet\delta_i(x_i) & \text{if } x_i \geq 1 \\ \underline{id} & \text{if } x_i = 0. \end{cases}$$

Clearly $\mathrm{graph}(\chi_i) \in \mathcal{C}$.

Now for $x_i \geq 1$,

$$\chi_i(x_i) = \alpha(\text{<chi>}(x_i))\circ\text{<chi>}(x_i)\bullet\beta(\text{<chi>}(x_i-1))$$

and thus, as above with $f_0, f_1, \ldots$ and $g_1, g_2, \ldots$,

$$\begin{aligned} \mathrm{Sigma}^i_{\text{<chi>}}(x_i) &= \text{<chi>}(x_i-1)\circ\ldots\bullet\text{<chi>}(0) \\ &= \beta(\text{<chi>}(x_i-1))\bullet\chi_i(x_i-1)\bullet\ldots\bullet\chi_i(1)\bullet\alpha(\text{<chi>}(0))\circ\text{<chi>}(0) \\ &= \beta(\text{<chi>}(x_i-1))\bullet\chi_i(x_i-1)\bullet\ldots\bullet\chi_i(1)\bullet\chi_i(0)\circ\alpha(\text{<chi>}(0))\bullet\text{<chi>}(0) \\ &= \beta(\text{<chi>}(x_i-1))\circ\mathrm{Sigma}^i_{\chi_i}(x_i)\circ\alpha(\text{<chi>}(0))\bullet\text{<chi>}(0). \end{aligned}$$

$\chi_i : \mathbb{N}^m \to {}^n n$ and $\mathrm{graph}(\chi_i) \in \mathcal{C}$ together imply that

$$\mathrm{graph}(\mathrm{Sigma}^i_{\chi_i}) \in \mathcal{C},$$

and therefore, since we are composing functions drawn from finite sets,

$$\mathrm{graph}(\mathrm{Sigma}^i_{\text{<chi>}}) \in \mathcal{C}.$$

End of proof of lemma IV.2.

IV.3 <u>Lemma</u>: Given $=$, $<$, explicit transformations, boolean operations and bounded quantification (I), for $n \geq 2$,

Closure under counting modulo S_n and closure under counting modulo $({}^n n \setminus S_n)$ together imply

Closure under counting modulo ${}^n n$.

<u>Proof</u>: This lemma might at first seem trivially true, but being able to compose sequences of elements of S_n and sequences of elements of $({}^n n \setminus S_n)$ does not imply directly that we can compose mixed sequences of elements of ${}^n n$.

Suppose therefore that we have a class $\mathcal{C}$, which includes $=$ and $\leq$ and is closed under explicit transformations, boolean operations, bounded quantification (I), counting modulo S_n and counting modulo $({}^n n \setminus S_n)$.

Suppose also that <chi> : $\mathbb{N}^m \to {}^n n$ and graph(<chi>)$\in \mathcal{C}$. Suppressing the variables $x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_m$, we can write

$$\mathrm{Sigma}^i_{\text{<chi>}}(x_i) = \text{<chi>}(x_i-1)\bullet\text{<chi>}(x_i-2)\bullet\ldots\bullet\text{<chi>}(1)\bullet\text{<chi>}(0)$$

and we want to show that graph$(\mathrm{Sigma}^i_{\text{<chi>}})\in \mathcal{C}$.

In general the sequence

<chi>(0), <chi>(1), <chi>(2), ...

will consist of blocks of elements of S_n alternating with blocks of elements of $({}^n n \setminus S_n)$.

The idea of the proof is to turn each block of elements of S_n into a block of elements of $({}^n n \setminus S_n)$.

To do this we use the fact that if $h \in ({}^n n \setminus S_n)$ and $j \in S_n$ then:

(a) $j \bullet h \in ({}^n n \setminus S_n)$.

This is because composition of a one-one function with a function that isn't must always produce a function that is not one-one.

(b) There is a $j' \in ({}^n n \setminus S_n)$ such that

$$j' \circ h = j \circ h.$$

Indeed we can define a function $\zeta : (S_n \times ({}^n n \setminus S_n)) \to ({}^n n \setminus S_n)$ such that, for all $j \in S_n$, for all $h \in ({}^n n \setminus S_n)$,

$$\zeta(j,h) \bullet h = j \bullet h.$$

ζ has finite domain. As does composition of two elements of ${}^n n$.

If we go on to consider a block $j_1,\ldots,j_k$ of elements of S_n composed with a single element h of $({}^n n \setminus S_n)$, we see, by (a), that for all $1 \leq i \leq k$,

$$j_{i-1}\bullet\ldots\bullet j_1\bullet h \in ({}^n n \setminus S_n)$$

and so if, for $1 \leq i \leq k$,

$$j_i' = \zeta(j_i,((j_{i-1}\bullet\ldots\bullet j_1)\bullet h)$$

then

$$j_i' \circ j_{i-1} \circ \ldots \circ j_1 \circ h = j_i \circ j_{i-1} \circ \ldots \circ j_1 \circ h,$$

which will also mean that

$$j_k' \circ \ldots \circ j_1' \circ h = j_k \circ \ldots \circ j_1 \circ h.$$

Thus do we replace a block of elements of S_n by a block of elements of $({}^n n \setminus S_n)$.

The important point to make about the derivation of j_i' is that it does not depend on $j_1', \ldots, j_{i-1}'$ but only on j_i, $(j_{i-1} \circ \ldots \circ j_1)$ and h. Furthermore $(j_{i-1} \circ \ldots \circ j_1)$ is a composition of elements of S_n.

We proceed more formally and systematically as follows:

For all $x_i \in \mathbb{N}$ the sequence

$$<\mathrm{chi}>(0),\ <\mathrm{chi}>(1),\ \ldots,\ <\mathrm{chi}>(x_i - 1)$$

contains a (possibly empty) cofinal block of elements of S_n.

We alter this block using two functions: f_i and g_i.

Define $f_i : \mathbb{N}^m \to \mathbb{N}$ by

$$f_i(x_1, \ldots, x_m) = \begin{cases} \text{the largest } y < x_i \text{ such that } <\mathrm{chi}>(y) \notin S_n \\ x_i \quad \text{if } <\mathrm{chi}>(z) \in S_n \text{ for all } z < x_i. \end{cases}$$

Then $\mathrm{graph}(f_i) \in \mathcal{C}$ for $f_i(x_1, \ldots, x_m) = y$ if and only if

$$\left(y < x_i \wedge <\mathrm{chi}>(y) \notin S_n \wedge \forall z < x_i (y < z \to <\mathrm{chi}>(z) \in S_n)\right)$$
$$\vee \left(y = x_i \wedge \forall z < x_i (<\mathrm{chi}>(z) \in S_n)\right)$$

(Note: Since S_n is finite, $\{z : <\mathrm{chi}>(z) \in S_n\} \in \mathcal{C}$.)

Observe that $f_i(x_1, \ldots, x_m) \leq x_i$.

Now define $g_i : \mathbb{N}^{m+1} \to S_n$ by $g_i(x_1, \ldots, x_m, x_{m+1}) =$

$$\begin{cases} <\mathrm{chi}>(x_1, \ldots, x_{i-1}, x_{m+1}, x_{i+1}, \ldots, x_m) & \text{if } f_i(x_1, \ldots, x_m) < x_{m+1} < x_i \\ \underline{\mathrm{id}} & \text{otherwise.} \end{cases}$$

Clearly $\mathrm{graph}(g_i) \in \mathcal{C}$. ($f_i(x_1, \ldots, x_m) < x_{m+1} < x_i$ if and only if $\forall z < x_i (z \geq x_{m+1} \to <\mathrm{chi}>(z) \in S_n)$.)

Now, suppressing variables other than x_i and x_{m+1},

$$g_i(x_i, 0),\ g_i(x_i, 1),\ g_i(x_i, 2),\ \ldots$$

$$= \underbrace{\underline{id},\underline{id},\dots,\underline{id}}_{f_i(x_i)+1\ \underline{id}\text{'s}},\underbrace{\text{<chi>}(f_i(x_i)+1),\text{<chi>}(f_i(x_i)+2),\dots,\text{<chi>}(x_i-1)}_{\text{Block of elements of } S_n},\underline{id},\underline{id},\dots$$

And $\text{Sigma}^{m+1}_{g_i}(x_i,x_i)$

$$= \text{<chi>}(x_i-1)\bullet\dots\bullet\text{<chi>}(f_i(x_i)+2)\bullet\text{<chi>}(f_i(x_i)+1)$$

which is the product of the block of elements of S_n cofinal with

$$\text{<chi>}(0),\text{<chi>}(1),\dots,\text{<chi>}(x_i-1)$$

unless <chi>$(z) \in S_n$ for all $z < x_i$, in which case we just get $\underline{id}$.

Also $g_i : \mathbb{N}^{m+1} \to S_n$ and $\text{graph}(g_i) \in \mathfrak{C}$ together imply that

$$\text{graph}(\text{Sigma}^{m+1}_{g_i}) \in \mathfrak{C}.$$

The two functions g_i and f_i are used to obtain a further function $\chi_i : \mathbb{N}^m \to {}^{n}n$.

Suppose that <chi>$(x_i) \in S_n$ and $f_i(x_i) \neq x_i$ (i.e. there is a $y < x_i$ such that <chi>$(y) \notin S_n$).

If we define $\chi_i(x_i)$ to be

$$\zeta(\text{<chi>}(x_i),(\text{Sigma}^{m+1}_{g_i}(x_i,x_i)\bullet\text{<chi>}(f_i(x_i))))$$

then as with the j's and h above

$$\chi_i(x_i)\bullet\chi_i(x_i-1)\bullet\dots\bullet\chi_i(f_i(x_i)+1)\circ\text{<chi>}(f_i(x_i))$$
$$= \text{<chi>}(x_i)\circ\text{<chi>}(x_i-1)\circ\dots\circ\text{<chi>}(f_i(x_i)+1)\circ\text{<chi>}(f_i(x_i)).$$

If we further define that, for x_i such that <chi>$(x_i) \notin S_n$ or such that <chi>$(y) \in S_n$ for all $y < x_i$,

$$\chi_i(x_i) = \text{<chi>}(x_i),$$

we can see that

$$\text{Sigma}^{i}_{\chi_i} = \text{Sigma}^{i}_{\text{<chi>}}.$$

Now χ_i is very nearly a function taking values only in $({}^{n}n \setminus S_n)$. There is an initial segment of values of x_i (those such that <chi>$(y) \in S_n$ for all $y \leq x_i$) for which $\chi_i(x_i) \in S_n$, but for all other values of x_i, $\chi_i(x_i) \in ({}^{n}n \setminus S_n)$.

Before moving on to elimination of this initial segment, we had better be satisfied that $\text{graph}(\chi_i) \in \mathfrak{C}$.

We know that ζ and $\bullet$ (composition of elements of ${}^{n}n$) have finite domains and that graph($\text{Sigma}^{m+1}_{g_i}$), graph(f) and $\{(x_1,\dots,x_m)$: <chi>(x_i) $S_n\}$ are in $\mathcal{C}$.

Furthermore $f_i(x_i) \leq x_i$ for all $x_i \in \mathbb{N}$.

And so $\chi_i(x_i) = y$ if and only if

$\Big(\big($ <chi>(x_i) $\notin S_n \vee \forall z < x_i($<chi>(z) $\in S_n)\big) \wedge \big($<chi>($x_i$) $= y\big)\Big)$

$\vee \Big(\big(($<chi>(x_i)$\in S_n \wedge \exists z < x_i($<chi>(z) $\notin S_n)\big)$

$\wedge \exists z \leq x_i(f_i(x_i) = z \wedge \zeta($<chi>($x_i$),($\text{Sigma}^{m+1}_{g_i}(x_i, x_i) \circ$<chi>(z))) $= y)\Big)$.

Thus, by the closure properties of $\mathcal{C}$, graph(χ_i)$\in \mathcal{C}$.

O.K. then. How can we get rid of this initial segment of elements of S_n in the sequence

$\chi_i(0),\ \chi_i(1),\ \chi_i(2),\ \dots$?

Well observe first of all that if <chi>(0) $\in ({}^{n}n \setminus S_n)$, there is no such initial segment and the problem goes away.

We cannot in general change <chi>(0) into an element of $({}^{n}n \setminus S_n)$ and still preserve $\text{Sigma}^{i}_{\text{<chi>}}$.

But what we can do is this:

Select some strict subset A of $\underset{\sim}{n} = \{0,1,\dots,n-1\}$. Then for any $j \in {}^{n}n$ there is an $h \in ({}^{n}n \setminus S_n)$ such that

$j \restriction A = h \restriction A$.

(If $j \in S_n$ we take some $a \in \underset{\sim}{n} \setminus A$ and define h to be exactly as j except that h(a) = j(b) some $b \in A$.)

So we can change <chi>(0) into an element <chi$_i$>(0) of $({}^{n}n \setminus S_n)$ which has the same effect when applied to A.

If for $x_i \neq 0$ we define <chi$_i$>(x_i) = <chi>(x_i), then we have a function <chi$_i$> : $\mathbb{N}^m \to {}^{n}n$ with graph(<chi$_i$>)$\in \mathcal{C}$ and for all $x_1,\dots$ $\dots,x_{i-1},x_{i+1},\dots,x_m$ <chi$_i$>(0) $\in ({}^{n}n \setminus S_n)$.

Furthermore, for all $x_1,\dots,x_m$ $\mathbb{N}$,

$\text{Sigma}^{i}_{\text{<chi}_i\text{>}}(x_1,\dots,x_m) \restriction A = \text{Sigma}^{i}_{\text{<chi>}}(x_1,\dots,x_m) \restriction A$.

So if χ_i^A is derived from $<\text{chi}_i>$ as χ_i was from $<\text{chi}$, we have, for all $x_1,\dots,x_m \in \mathbb{N}$,

(i) $\text{Sigma}^{i}_{\chi_i^A}(x_1,\dots,x_m) \upharpoonright A = \text{Sigma}^{i}_{<\text{chi}>}(x_1,\dots,x_m) \upharpoonright A$

(ii) $\text{graph}(\text{Sigma}^{i}_{\chi_i^A}) \in \mathcal{C}$. (Because $\chi_i^A : \mathbb{N}^m \to ({}^n n \setminus S_n)$.)

Select also some $B \subsetneq \underline{n}$ such that $A \cup B = \underline{n}$. We can derive χ_i^B such that $\text{graph}(\text{Sigma}^{i}_{\chi_i^B}) \in \mathcal{C}$ and for all $x_1,\dots,x_m \in \mathbb{N}$

$$\text{Sigma}^{i}_{\chi_i^B}(x_1,\dots,x_m) \upharpoonright B = \text{Sigma}^{i}_{<\text{chi}>}(x_1,\dots,x_m) \upharpoonright B.$$

Finally we have $\text{Sigma}^{i}_{<\text{chi}>}(x_1,\dots,x_m) = y$ if and only if $(\text{Sigma}^{i}_{\chi_i^A}(x_1,\dots,x_m) \upharpoonright A = y \upharpoonright A) \wedge (\text{Sigma}^{i}_{\chi_i^B}(x_1,\dots,x_m) \upharpoonright B = y \upharpoonright B)$.

Therefore, since ${}^n n$, A and B are finite,

$$\text{graph}(\text{Sigma}^{i}_{<\text{chi}>}) \in \mathcal{C}.$$

End of proof of lemma IV.3.

We can now conclude our proof that, under the given conditions, closure under counting modulo S_n implies closure under counting modulo ${}^n n$.

The proof is by induction on n.

<u>n = 1</u>: $S_1 = {}^1 1$ and the proposition is trivially true.

<u>If true for n</u>: Then, since S_n is isomorphic to a subgroup of S_{n+1}, closure under counting modulo S_{n+1} implies closure under counting modulo S_n and therefore, by the inductive hypothesis, closure under counting modulo ${}^n n$.

But now, by lemma IV.2, we have closure under counting modulo $({}^{n+1}n{+}1 \setminus S_{n+1})$.

And this implies, by lemma IV.3, that there is closure under counting modulo ${}^{n+1}n{+}1$ since we have started by assuming closure under counting modulo S_{n+1}.

End of proof of theorem IV.1.

Theorem IV.1 has the direct corollary that:

IV.4 <u>Corollary</u>:

$$S_n\Delta_0^{\mathbb{N}} = {}^n n\Delta_0^{\mathbb{N}} .$$

Chapter V Space($\underline{n}$,$\square$) and counting modulo S_n

This chapter extends the results of chapter II connecting $\Delta_0^{\mathbb{N}}$ and Space($\underline{1}$,$\square$) to all other values of n.

Recall the definition of a Δ_0-function and define

V.1 Definition: The $S_n\Delta_0$-functions are the functions with graph in $S_n\Delta_0^{\mathbb{N}}$ and values bounded by a polynomial in the arguments.

The ${}^{n}n\Delta_0$-functions are defined similarly.

As another simple corollary to theorem IV.1 we have:

V.2 Corollary: The class of $S_n\Delta_0$-functions is identical with the class of ${}^{n}n\Delta_0$-functions.

And we can prove a result analogous to fact II.9:

V.3 Facts:

(i) If $A \subseteq \mathbb{N}^q$, $A \in S_n\Delta_0^{\mathbb{N}}$ and $f_1(x_1,\ldots,x_m),\ldots,f_q(x_1,\ldots,x_m)$ are $S_n\Delta_0$-functions

then $\{(x_1,\ldots,x_m) : (f_1(x_1,\ldots,x_m),\ldots,f_q(x_1,\ldots,x_m)) \in A\} \in S_n\Delta_0^{\mathbb{N}}$.

(ii) The $S_n\Delta_0$-functions are closed under composition.

Proof: Similar to the proof of fact II.9.

Now we can go on to link these classes of functions with the space complexity classes of SRM's.

V.4 Theorem:

Space($\underline{n}$,$\square$) is exactly the class of $S_n\Delta_0$-functions.

Proof: The proof is very much on the lines of chapter II, which proved the case $n = 1$.

(i) All $S_n\Delta_0$-functions are in Space($\underline{n}$,$\square$).

We restrict our attention at first to functions taking values in $\{0,1\}$.

Recall definition II.5 defining $\text{Space}_*(u,v)$ as the class of sets whose characteristic functions are in Space(u,v).

Observing that $\underline{n},\Box$ satisfy the conditions of lemma II.3 and applying the arguments used in the proof of proposition II.6 we can show that $\mathrm{Space}_*(\underline{n},\Box)$ includes $=$, $\leq$, graph(+) and graph(·) and is closed under explicit transformations, boolean operations and bounded quantification (II). In order to show that

$$S_n\Delta_0^{\mathbb{N}} \subseteq \mathrm{Space}_*(\underline{n},\Box)$$

it merely remains to show that $\mathrm{Space}_*(\underline{n},\Box)$ is closed under counting modulo S_n.

Suppose then that we have $\langle chi\rangle : \mathbb{N}^m \to S_n$ and that

$$\mathrm{graph}(\langle chi\rangle) \in \mathrm{Space}_*(\underline{n},\Box)$$

where ¢, the function encoding S_n takes S_n to the set $\{a_\pi\}_{\pi \in S_n}$ of natural numbers.

We show first of all that for all $1 \leq i \leq m$ and $j,k < n$ the set $X^i_{j,k} = \{(x_1,\dots,x_m) : (\mathrm{Sigma}^i_{\langle chi\rangle}(x_1,\dots,x_m))(j) = k\} \in \mathrm{Space}_*(\underline{n},\Box)$. (Remember $\mathrm{Sigma}^i_{\langle chi\rangle}(x_1,\dots,x_m) : \underset{\sim}{n} \to \underset{\sim}{n}$.)

We know that χ, the characteristic function of graph(<chi>) is in $\mathrm{Space}(\underline{n},\Box)$ and so we have the following $\mathrm{SRM}_{\mathrm{Space}(\underline{n},\Box)}$:

It has registers $r, t_0, t_1, x_1, \dots, x_m$ and program

```
begin   $r := j$;
  ℓ.    If $\chi(x_1,\dots,x_{i-1},t_0,x_{i+1},\dots,x_m,a_{\pi_1}) = 0$ then $r := \bar{\pi}_1(r)$;
        If $\chi(x_1,\dots,x_{i-1},t_0,x_{i+1},\dots,x_m,a_{\pi_2}) = 0$ then $r := \bar{\pi}_2(r)$;
        ..............................................................
        If $\chi(x_1,\dots,x_{i-1},t_0,x_{i+1},\dots,x_m,a_{\pi_{n!}}) = 0$ then $r := \bar{\pi}_{n!}(r)$;
        $t_0 := t_0 + 1$;
        If $t_0 \neq x_i$ then goto ℓ.;
        If $r \neq k$ then $t_1 := t_1 + 1$;    halt
```

where $\{\pi_1,\dots,\pi_{n!}\} = S_n$ and for all $1 \leq s \leq n!$ $\bar{\pi}_s$ is a function in $\mathrm{Space}(\underline{n},\Box)$ such that $\bar{\pi}_s \upharpoonright \underset{\sim}{n} = \pi_s$ (such functions can easily be found). The constant functions 0,j,k are also easily shown to be in $\mathrm{Space}(\underline{n},\Box)$.

(Note: This is not strictly the form of program required for an $SRM_{\mathfrak{Z}}$ but it may be changed into the correct form with ease.)

If we suppress $x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_m$ as variables of <chi> then we see that this machine repeatedly sets

$$r := (\text{<chi>}(t_0))(r)$$

as t_0 rises from 0 to (x_i-1). r has starting value j. It thus calculates

$$(\text{Sigma}^{i}_{\text{<chi>}}(x_i))(j).$$

This value is then compared with k and t_1 is set accordingly.

The machine quite clearly runs in bounds $(\underline{n},\square)$.

Thus by lemma II.3, $X^{i}_{j,k} \in \text{Space}_{\star}(\underline{n},\square)$.

And now, for each $\pi \in S_n$, the set

$$X^{i}_{\pi} = \bigcap_{j=0}^{n-1} X^{i}_{j,\pi(j)}$$

is in $\text{Space}(\underline{n},\square)$ by the closure under boolean operations which we have already established. Now $(x_1,\ldots,x_m) \in X^{i}_{\pi}$ if and only if

$$\text{Sigma}^{i}_{\text{<chi>}}(x_1,\ldots,x_m) = \pi$$

and so, as there are only finitely many elements π of S_n,

$$\text{graph}(\text{Sigma}^{i}_{\text{<chi>}}) \in \text{Space}_{\star}(\underline{n},\square).$$

So we have all the right closure properties to show that:

$$S_n\Delta_0^{\mathbb{N}} \subseteq \text{Space}_{\star}(\underline{n},\square).$$

The arguments of lemma II.7 apply just as well to $\text{Space}(\underline{n},\square)$ and show, in conclusion, that

All $S_n\Delta_0$-functions are in $\text{Space}(\underline{n},\square)$.

(ii) All elements of $\text{Space}(\underline{n},\square)$ are $S_n\Delta_0$-functions.

We begin with an extension of lemma II.10.

V.5 <u>Lemma</u>:

<u>If</u>

M is a machine such that:

M has registers $r,t_0,\ldots,t_k,x_1,\ldots,x_m$ where $k \geq 1$.

M has program L =

```
($If\ (r,\vec{t},\vec{x}) \in Y_0\ then\ (r := \psi_0(r,\vec{t},\vec{x});\ t_0 := t_0 + 1;\ L)$;
 $If\ (r,\vec{t},\vec{x}) \in Y_1\ then\ (r := \psi_1(r,\vec{t},\vec{x});\ t_1 := t_1 + 1;\ L)$;
 ..........................................................
 $If\ (r,\vec{t},\vec{x}) \in Y_k\ then\ (r := \psi_k(r,\vec{t},\vec{x});\ t_k := t_k + 1;\ L)$;
 $r := \psi_{k+1}(r,\vec{t},\vec{x})$)
```

where $Y_0,\ldots,Y_k \in S_n\Delta_0^{\mathbb{N}}$ and $\psi_0,\ldots,\psi_{k+1}$ are $S_n\Delta_0$-functions. (And $\vec{t}$ stands for $t_0,\ldots,t_k$ while $\vec{x}$ stands for $x_1,\ldots,x_m$.)

There is $p(\vec{x}) \in \mathbb{N}[\vec{x}]$ such that if M starts with inputs $\vec{x}$ then throughout M's run

$$t_0,\ \ldots,\ t_k < p(\vec{x}).$$

Also, throughout any run,

$$r < n.$$

<u>then</u>

There is a machine M' with registers $r,t_1,\ldots,t_k,x_1,\ldots,x_m$ and program L' =

```
($If\ (r,t_1,\ldots,t_k,\vec{x}) \in Y'_1\ then\ (r:=\psi'_1(r,t_1,\ldots,t_k,\vec{x});t_1:=t_1+1;L)$;
 $If\ (r,t_1,\ldots,t_k,\vec{x}) \in Y'_2\ then\ (r:=\psi'_2(r,t_1,\ldots,t_k,\vec{x});t_2:=t_2+1;L)$;
 ..............................................................
 $If\ (r,t_1,\ldots,t_k,\vec{x}) \in Y'_k\ then\ (r:=\psi'_k(r,t_1,\ldots,t_k,\vec{x});t_k:=t_k+1;L)$;
 $r := \psi_{k+1}(r,\ t_1,\ldots,t_k,\vec{x})$)
```

where $Y'_1,\ldots,Y'_k \in S_n\Delta_0^{\mathbb{N}}$ and $\psi'_1,\ldots,\psi'_{k+1}$ are $S_n\Delta_0$-functions, such that M' is equivalent to M in the following sense.

For a given input $x_1,\ldots,x_m$, the (finite) sequence of increases in $t_1,\ldots,t_k$ executed by M' and the sequence of increases in these registers executed by M will be one and the same. M of course can also execute increases in t_0 but these are ignored in this comparison.

Furthermore if we take an element of this sequence, which is an execution performed by both machines at some point in their respective runs, then at the time of this execution the value of r will be the same in both machines. Also upon halting the value of r will be the same for both.

Proof: (i.e. derivation of $Y'_1,\dots,Y'_k,\psi'_1,\dots,\psi'_{k+1}$)

As in the proof of lemma II.10 we first find a function f (an $S_n\Delta_0$-function) such that, for all $r,t_1,\dots,t_k,x_1,\dots,x_m$ occurring in a computation, $f(r,t_1,\dots,t_k,x_1,\dots,x_m)$ is the least value of t_0 such that $(r,t_0,\dots,t_k,x_1,\dots,x_m) \notin Y_0$.

However this is not so straightforward as before for we really mean "the least value of t_0 reached by repeated executions of

$$r := \psi_0(r,t_0,\dots,t_k,x_1,\dots,x_m);\ t_0 := t_0 + 1;$$

starting with $t_0 = 0$"

and so we must take the changes in r into account as well.

Observe firstly that because, in computations, r is bounded by n we can, in practice, think of ψ_0 as defining a function

$$g : \mathbb{N}^{m+k+1} \to {}^n n$$

where for $0 \leq i < n$

$$(g(t_0,\dots,t_k,x_1,\dots,x_m))(i) = \begin{cases} \psi_0(i,t_0,\dots,t_k,x_1,\dots,x_m) & \text{if this is} < n \\ 0 & \text{otherwise.} \end{cases}$$

and it is easy to see that, because ψ_0 is an $S_n\Delta_0$-function, g has an $S_n\Delta_0^{\mathbb{N}}$ graph.

Suppress the variables $t_1,\dots,t_k,x_1,\dots,x_m$ for the time being.

For all $t_0 \in \mathbb{N}$, $g(t_0) : \underset{\sim}{n} \to \underset{\sim}{n}$ and tells us the effect of

$$r := \psi_0(r,t_0).$$

Therefore, for all $y \in \mathbb{N}$,

$\mathrm{Sigma}^1_g(y) = g(y-1)\ g(y-2)\ \dots\ g(1)\ g(0)$

tells us the effect on r of executing

$$r := \psi_0(r,t_0);\ t_0 := t_0 + 1;$$

y times starting with $t_0 = 0$, whatever the value of r we start with may be, as long as $\psi_0(r,t_0) < n$ for all $t_0 < y$.

If $r = w$ when we start, then the value in the work-register as t_0 is increased to y will be

$$(\mathrm{Sigma}_g^1(y))(w).$$

Thus if we start from $r = w$ and $t_0 = 0$, the lowest value of t_0 for which $(r,t_0) \notin Y_0$ will be the least y such that

$$((\mathrm{Sigma}_g^1(y))(w),y) \notin Y_0.$$

That is to say we can define f by

$f(r,t_1,\dots,t_k,x_1,\dots,x_m) = y$ if and only if

$$\Big(\big(((\mathrm{Sigma}_g^1(y))(r),y) \notin Y_0 \wedge \forall z < y\ (((\mathrm{Sigma}_g^1(z))(r),z) \in Y_0)\big)$$
$$\wedge \ \big(y < p(x_1,\dots,x_m)\big)\Big)$$
$$\vee \big(\forall z < p(x_1,\dots,x_m)\ (((\mathrm{Sigma}_g^1(z))(r),z) \in Y_0 \wedge y = 0)\big).$$

Remember that the stacks of M, in particular t_0, are bounded by $p(x_1,\dots,x_m)$. Also we may assume that $\mathrm{Sigma}_g^1(0) = \underline{id}$.

Now $\mathrm{graph}(g) \in S_n\Delta_0^{\mathbb{N}}$ implies $\mathrm{graph}(\mathrm{Sigma}_g^1) \in S_n\Delta_0^{\mathbb{N}}$ because $S_n\Delta_0^{\mathbb{N}}$ is the same thing as ${}^n{}_n\Delta_0^{\mathbb{N}}$ by corollary IV.4. Therefore, since the values of Sigma_g^1 are drawn from a finite set with $n!$ elements, $\{(r,t_0,\dots,t_k,x_1,\dots,x_m) : ((\mathrm{Sigma}_g^1(t_0))(r),t_0) \in Y_0\} \in S_n\Delta_0^{\mathbb{N}}$.

Thus, by the closure properties of $S_n\Delta_0^{\mathbb{N}}$,

$$\mathrm{graph}(f) \in S_n\Delta_0^{\mathbb{N}}.$$

And so, because f is bounded by $p(x_1,\dots,x_m)$, f is an $S_n\Delta_0$-function.

To conclude we can, by facts V.3, find an $S_n\Delta_0^{\mathbb{N}}$ set Y_j', for $1 \leq j \leq k$, such that $(r,t_1,\dots,t_k,x_1,\dots,x_m) \in Y_j'$ if and only if $((\mathrm{Sigma}_g^1(f(r,t_1,\dots,t_k,\vec{x})))(r),f(r,t_1,\dots,t_k,\vec{x}),t_1,\dots,t_k,\vec{x}) \in Y_j$ and Y_j' will be the set required for M'.

Meanwhile, resuming suppression of $t_1,\dots,t_k,x_1,\dots,x_m$, define

$$\psi_j' = \psi_j((\mathrm{Sigma}_g^1(f(r)))(r),f(r))$$

and ψ_j', for $1 \leq j \leq k+1$, will be the $S_n\Delta_0$-functions we want.

End of proof of lemma V.5.

Using lemma V.5 we can go on to prove the counterpart of proposition II.11, that is that:

All machines with work-register r bounded by n, stack registers $t_0,\dots,t_k$ bounded by $p(x_1,\dots,x_m)$, where $x_1,\dots,x_m$ are the input registers, and program which is of the form given in lemma V.6 and has $S_n\Delta_0$ conditions and assignments, compute $S_n\Delta_0$-functions.

Finally we know, by lemma II.8, that for any SRM running in bounds $(\underline{n},\Box)$ there is an equivalent machine of the type required for lemma V.6. And so we conclude that all functions in $\mathrm{Space}(\underline{n},\Box)$ are $S_n\Delta_0$-functions. End of proof of theorem V.4.

Also, because (as is easily shown) all $S_n\Delta_0$-functions taking values in $\{0,1\}$ are characteristic functions of $S_n\Delta_0^{\mathbb{N}}$ sets, we have the simple

V.6 <u>Corollary</u>:

$$\mathrm{Space}_*(\underline{n},\Box) = S_n\Delta_0^{\mathbb{N}}.$$

Finally, because of examples III.14 (ii) and (iii), we have

V.7 <u>Corollary</u>:

$$\mathrm{Space}(\underline{4},\Box) = \mathrm{Space}(\underline{3},\Box).$$

This is somewhat surprising since we have no reason to suppose e.g. that $\mathrm{Space}(\underline{2},\Box) = \mathrm{Space}(\underline{3},\Box)$ or that $\mathrm{Space}(\underline{4},\Box) = \mathrm{Space}(\underline{5},\Box)$.

End of proof of lemma V.5.

Using lemma V.5 we can go on to prove the counterpart of proposition II.11, that is that:

All machines with work-register r bounded by n, stack registers $t_0,\dots,t_k$ bounded by $p(x_1,\dots,x_m)$, where $x_1,\dots,x_m$ are the input registers, and program which is of the form given in lemma V.6 and has $S_n\Delta_0$ conditions and assignments, compute $S_n\Delta_0$-functions.

Finally we know, by lemma II.8, that for any SRM running in bounds $(\underline{n},\Box)$ there is an equivalent machine of the type required for lemma V.6. And so we conclude that all functions in $\mathrm{Space}(\underline{n},\Box)$ are $S_n\Delta_0$-functions. End of proof of theorem V.4.

Also, because (as is easily shown) all $S_n\Delta_0$-functions taking values in $\{0,1\}$ are characteristic functions of $S_n\Delta_0^{\mathbb{N}}$ sets, we have the simple

V.6 <u>Corollary</u>:

$$\mathrm{Space}_*(\underline{n},\Box) = S_n\Delta_0^{\mathbb{N}}.$$

Finally, because of examples III.14 (ii) and (iii), we have

V.7 <u>Corollary</u>:

$$\mathrm{Space}(\underline{4},\Box) = \mathrm{Space}(\underline{3},\Box).$$

This is somewhat surprising since we have no reason to suppose e.g. that $\mathrm{Space}(\underline{2},\Box) = \mathrm{Space}(\underline{3},\Box)$ or that $\mathrm{Space}(\underline{4},\Box) = \mathrm{Space}(\underline{5},\Box)$.

Chapter VI MORE NATURAL IDEAS OF COUNTING

In chapter V we have succeeded in linking certain, quite natural complexity classes of Bel'tyukov stack register machines with certain classes of sets of numbers. But we could now be forgiven for asking ourselves how natural these latter classes are. There are two aspects to this.

Firstly the notation $X\Delta_0^{\mathbb{N}}$ which we have used does appear to have some meaning to it. There does seem to be a connection between X and $X\Delta_0^{\mathbb{N}}$. The relationship between two classes $X\Delta_0^{\mathbb{N}}$ and $Y\Delta_0^{\mathbb{N}}$ seems to depend closely on the relationship between X and Y (Theorems III.9 and III.10). This is particularly true where X and Y are groups: all inclusions $G_1\Delta_0^{\mathbb{N}} \subseteq G_2\Delta_0^{\mathbb{N}}$ (G_1, G_2 groups) which are at present known can be reduced to applications of theorems III.9 and III.10. That we have a family of classes which appears to tie in so closely with existing mathematical objects suggests that the members of this family do indeed possess some mathematical substance.

On the other hand, the crucial notion - that of "closure under counting modulo X" (definition III.7) - is perhaps rather strained. The terminology suggests closure under some set operation and indeed clearly that operation must be the derivation of graph(Sigma$^{i}_{<chi>}$) from graph(<chi>). However the connection between these two sets, considered simply as sets rather than in terms of the functions whose graphs they are, is so oblique that it wasn't deemed worthwhile to define "counting modulo X" in its own right. Furthermore the operation is not one that can be applied to all sets of numbers. It only works for graphs of particular kinds of function. (We could extend the operations so as to work for all sets but this would be highly artificial.) Finally for "counting modulo X" to be well defined we would need to tie ourselves down to a particular function ς with which to code X. One of the more important aspects of "closure under counting modulo X" was that the actual coding used was irrelevant.

Now it may be that for all appropriate sets X (or perhaps at least for all groups) there is a set operation which applies naturally and simply to all sets and is such that the closure of $\Delta_0^{\mathbb{N}}$ with respect to it brings us back to $X\Delta_0^{\mathbb{N}}$. But until we know more we may consider the most interesting classes $X\Delta_0^{\mathbb{N}}$ to be those for which such an operation is already known.

This is in fact the case for $X = \mathbb{Z}_n$ $(n \geq 1)$.

Counting modulo $\mathbb{Z}_n$

VI.1 *Definition*: Let $A \subseteq \mathbb{N}$ and suppose that

$$A = \{a_0, a_1, a_2, \ldots\} \quad \text{where} \quad a_0 < a_1 < a_2 < \ldots .$$

Then we define $A^{(n)}$ by

$$A^{(n)} = \{a_0, a_n, a_{2n}, \ldots\}$$

i.e. $A^{(n)}$ contains every n^{th} element of A.

Before generalising VI.1 we make

VI.2 *Definition*: For any $m \geq 1$, $B \subseteq \mathbb{N}^m$, $1 \leq i \leq m$, $x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_m \in \mathbb{N}$, $B^i(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_m)$ is defined to be the set $\{x : (x_1, \ldots, x_{i-1}, x, x_{i+1}, \ldots, x_m) \in B\}$.

We combine definitions VI.1 and VI.2 to produce:

VI.3 *Definition*: For $n,m \geq 1$, $A \subseteq \mathbb{N}^m$, $1 \leq i \leq m$, $A^{i,(n)} \subseteq \mathbb{N}^m$ is defined to be the set such that for all $x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_m \in \mathbb{N}$

$$(A^{i,(n)})^i(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_m) = (A^i(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_m))^{(n)}.$$

Certainly VI.1 gives a natural set operation and I believe that VI.3 provides the obvious generalisation of it.

VI.4 *Definition*: A class $\mathcal{C}$ of sets of numbers is said to be *closed under *-counting modulo n* if for all $m \geq 1$, for all $A \subseteq \mathbb{N}^m$, for all $1 \leq i \leq m$,

$$A \in \mathcal{C} \quad \text{implies} \quad A^{i,(n)} \in \mathcal{C}.$$

We can't call this "closure under counting modulo n" for we have used up that terminology already in definition III.2. However we now go on to show that in most cases the two ideas are the same and therefore also, by example III.8, the same as "closure under counting modulo $\mathbb{Z}_n$".

VI.5 Theorem: If

$\mathcal{C}$ is a class of sets of numbers and $\mathcal{C}$ is closed under explicit transformations and boolean operations and contains $=$

then, for $n \geq 1$,

$\mathcal{C}$ is closed under *-counting modulo n (definition VI.4)

if and only if

$\mathcal{C}$ is closed under counting modulo n (definition III.2).

Proof:

(i) Counting modulo n implies *-counting modulo n.

Suppose $\mathcal{C}$ is closed under counting modulo n. Let $A \subseteq \mathbb{N}^m$ for some $m \geq 1$. Then for all $1 \leq i \leq m$, $A^{i,(n)} \in \mathcal{C}$ for if σ^i_A is the summation function defined near the beginning of chapter III, it is easy to see that

$$(x_1,\dots,x_m) \in A^{i,(n)} \quad \text{iff} \quad \sigma^i_A(x_1,\dots,x_m) \equiv 0 \bmod n,$$

that is to say

$$A^{i,(n)} = \{(x_1,\dots,x_m : (x_1,\dots,x_m,0) \in \mathrm{graph}(\sigma^i_A \bmod n)\} \quad .$$

(ii) *-counting modulo n implies counting modulo n.

First we prove two lemmas.

VI.6 Lemma: Let $\mathcal{C}$ be closed under explicit transformations and boolean operations and contain $=$. Let $m,n \geq 1$. Let $A \subseteq \mathbb{N}^m$. Then for all $0 \leq j \leq n$, $1 \leq i \leq m$,

$A \in \mathcal{C}$ implies there exists $B \in \mathcal{C}$ such that for all $x_1,\dots,x_{i-1}$, $x_{i+1},\dots,x_m \in \mathbb{N}$, for all $x_i \geq n-1$,

$$\sigma^i_B(x_1,\dots,x_m) \equiv 0 \bmod n \quad \text{iff} \quad \sigma^i_A(x_1,\dots,x_m) \equiv j \bmod n.$$

Proof:

The case $j = 0$ holds trivially.

For $j > 0$ the idea of the proof is best seen by first looking at the case that $m = 1$, i.e. A is a 1-ary set. Necessarily $i = 1$.

For $x_1 \geq n-1$, B is identical to A, that is

$$B \cap \{x_1 : x_1 \geq n-1\} = A \cap \{x_1 : x_1 \geq n-1\}.$$

The set $\{x_1 : x_1 \geq n-1\} \in \mathcal{C}$ because it is the complement of a finite set.

For $B \cap \{x_1 : x_1 < n-1\}$ we replace $A \cap \{x_1 : x_1 < n-1\}$ by a subset of $\{x_1 : x_1 < n-1\}$ of appropriate size.

In the current case $\sigma^1_A(n-1)$ is a constant since A is 1-ary and now $\{x_1 : x_1 < (\sigma^1_A(n-1) - j) \bmod n\} \in \mathcal{C}$ will be a set of the size required. (Recall that for $x \in \mathbb{Z}$, $x \bmod n$ is the number $0 \leq y \leq n$ such that $x \equiv y \bmod n$.)

If we now define

$$B = \{x_1 : x_1 < (\sigma^1_A(n-1) - j) \bmod n\} \cup \left(A \cap \{x_1 : x_1 \geq n-1\}\right) \in \mathcal{C}$$

then for all $x_1 \geq n-1$,

$$\sigma^1_B(x_1) \equiv \sigma^1_A(x_1) - j \text{ modulo } n,$$

and B is the set we want.

We can now move on to the general case $m \geq 1$.

As in the 1-ary case, from A we derive a set B which is identical for $x_i \geq n-1$ but such that

$$\sigma^i_B(x_1,\ldots,x_{i-1},n-1,x_{i+1},\ldots,x_m) \equiv \sigma^i_A(x_1,\ldots,x_{i-1},n-1,x_{i+1},\ldots,x_m) - j \bmod n.$$

In the general case we have the extra problem that we must be able to do this uniformly - i.e. for all (possible) values of $x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_m$ simultaneously.

First we define $Q^{m,i}_u$ by

$$Q^{m,i}_u = \{(x_1,\ldots,x_m) : x_i = 0 \vee x_i = 1 \vee \ldots \vee x_i = u-1\}.$$

$Q^{m,i}_u \in \mathcal{C}$ because it is derived from $=$ using explicit transformations and boolean operations.

If we now look at the set

$$A \cap (\mathbb{N}^m \setminus Q^{m,i}_{n-1}) \in \mathcal{C},$$

we see that we have isolated the part of A which we wish to leave unchanged, i.e. for which $x_i \geq n-1$.

Observe also that

$$c^{i}_{Q^{m}_{u}i}(x_1,\ldots,x_{i-1},n-1,x_{i+1},\ldots,x_m) = u.$$

Thus if we can replace

$$A \cap Q^{m,i}_{n-1}$$

by

$$Q^{m,i}_{s}$$

where $s = (\sigma^i_A(x_1,\ldots,x_{i-1},n-1,x_{i+1},\ldots,x_m) - j) \bmod n$, we will have what we need for the rest of B.

However we cannot construct B directly but must proceed by cases according to the value of $\sigma^i_A(x_1,\ldots,x_{i-1},n-1,x_{i+1},\ldots,x_m)$.

If for given $0 \leq y < n$ we can isolate the set

$$R^{A,i}_y = \{(x_1,\ldots,x_m) : \sigma^i_A(x_1,\ldots,x_{i-1},n-1,x_{i+1},\ldots,x_m) = y\}$$

then for these values of $x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_m$ such that

$$\sigma^i_A(x_1,\ldots,x_{i-1},n-1,x_{i+1},\ldots,x_m) = y,$$

the part of B we still need is

$$R^{A,i}_y \cap Q^{m,i}_{(y-j)\bmod n}.$$

We obtain $R^{A,i}_y$ as follows.

For $0 \leq w \leq n-2$ define $P^{A,i}_w$ by

$$P^{A,i}_w = \{(x_1,\ldots,x_m) : (x_1,\ldots,x_{i-1},w,x_{i+1},\ldots,x_m) \in A\}.$$

$P^{A,i}_w \in \mathcal{C}$ by closure under explicit transformations.

You will see that (as for $R^{A,i}_y$) membership of $P^{A,i}_w$ does not depend on the value of x_i; so $\exists x_i((x_1,\ldots,x_m) \in P^{A,i}_w)$ if and only if $\forall x_i((x_1,\ldots,x_m) \in P^{A,i}_w)$.

Now for each of the 2^{n-1} sets $Z \subseteq \{0,1,\ldots,n-1\}$ define

$$P^{A,i}_Z = \Big(\bigcap_{w \in Z} P^{A,i}_w\Big) \cap \Big(\bigcap_{\substack{w \leq n-2 \\ w \notin Z}} (\mathbb{N}^m \setminus P^{A,i}_w)\Big) \in \mathcal{C}.$$

Again $\exists x_i((x_1,\ldots,x_m) \in P^{A,i}_Z)$ iff $\forall x_i((x_1,\ldots,x_m) \in P^{A,i}_Z)$.

Furthermore for all $x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_m \in \mathbb{N}$

$$\exists!\ Z \subseteq \{0,1,\ldots,n-1\} \mid \forall x_i((x_1,\ldots,x_m) \in P^{A,i}_Z)\}$$

and $(x_1,\ldots,x_m) \in P^{A,i}_Z$ implies

$$\sigma^i_A(x_1,\ldots,x_{i-1},n-1,x_{i+1},\ldots,x_m) = |Z|.$$

It should be clear that we can now define

$$R^{A,i}_y = \bigcup_{|Z|=y} P^{A,i}_Z \qquad .$$

Finally

$$B = \{A \cap (\mathbb{N}^m \setminus Q^{m,i}_{n-1})\} \cup \{\bigcup_{0\leq y<n} (R^{A,i}_y \cap Q^{m,i}_{(y-j) \bmod n})\} \in \mathcal{C}.$$

End of proof of lemma VI.6.

VI.7 <u>Lemma</u>: For $n \geq 1$, if $\mathcal{C}$ is closed under explicit transformations, boolean operations and $*$-counting modulo n and contains $=$, then for all $m \geq 1$, $A \subseteq \mathbb{N}^m$, $1 \leq i \leq m$, $0 \leq j \leq n$

$$A \in \mathcal{C} \quad \text{implies} \quad \{(x_1,\ldots,x_m) : \sigma^i_A(x_1,\ldots,x_m) \equiv j \bmod n\} \in \mathcal{C}.$$

<u>Proof</u>:

Again the case $j = 0$ is trivial since

$$\{(x_1,\ldots,x_m) : \sigma^i_A(x_1,\ldots,x_m) \equiv 0 \bmod n\} = A^{i,(n)}.$$

For $j \geq 1$ we first derive the set B as in lemma VI.6 and then

$$\{(x_1,\ldots,x_m) : (x_i \geq n-1) \wedge (\sigma^i_A(x_1,\ldots,x_m) \equiv j \bmod n)\}$$

$$= \{(x_1,\ldots,x_m) : (x_i \geq n-1) \wedge (\sigma^i_B(x_1,\ldots,x_m) \equiv 0 \bmod n)\}$$

$$= B^{i,(n)} \cap (\mathbb{N}^m \setminus Q^{m,i}_{n-1})$$ where $Q^{m,i}_u$ is as in the proof of VI.6.

$$\in \mathcal{C}.$$

It remains to supply the part

$$\{(x_1,\ldots,x_m) : (x_i < n-1) \wedge (\sigma^i_A(x_1,\ldots,x_m) \equiv j \bmod n)\}$$

Now for each $Z \subseteq \{0,1,\ldots,n-2\}$ and $P^{A,i}_Z$ as in VI.6, $(x_1,\ldots,x_m) \in P^{A,i}_Z$ implies that for $w \leq n-2$,

$$\sigma^i_A(x_1,\ldots,x_{i-1},w,x_{i+1},\ldots,x_m) = |Z \cap \{0,1,\ldots,w-1\}|.$$

Therefore, if for $Z \subseteq \{0,1,\ldots,n-2\}$ we define

$$\bar{Z} = \{w \leq n-2 : |Z \cap \{0,1,\ldots,w-1\}| \equiv j \bmod n\} \subseteq \{0,1,\ldots,n-2\}$$

then we will have

$$\{(x_1,\ldots,x_m) : (x_i < n-1) \wedge (\sigma^i_A(x_1,\ldots,x_m) \equiv j \bmod n)\} = \bigcup_{Z \subseteq \{0,1,\ldots,n-2\}} (P^{A,i}_Z \cap Q^{m,i}_{\bar{Z}}).$$

End of proof of lemma VI.7.

We can now conclude our proof of theorem VI.5.

It remains to be shown that $A \in \mathcal{C}$ implies that graph(Sigma^i_A mod n) is in $\mathcal{C}$.

But if we now define

$$T^{A,j} = \{(x_1,\ldots,x_m,x_{m+1}) : (\sigma^i_A(x_1,\ldots,x_m) \equiv j \bmod n) \wedge x_{m+1} = j\}$$

then clearly $T^{A,j} \in \mathcal{C}$.

And finally

$$\text{graph}(\sigma^i_A \bmod n) = \bigcup_{0 \leq j < n} T^{A,j}.$$

End of proof of theorem VI.5.

VI.8 <u>Corollary</u>: <u>If</u>

$\mathcal{C}$ is a class of sets of numbers closed under explicit transformations and boolean operations and containing =

<u>then</u>, for all $n \geq 1$,

$\mathcal{C}$ is closed under *-counting modulo n

if and only if

$\mathcal{C}$ is closed under counting modulo $\mathbb{Z}_n$.

<u>Proof</u>: Example III.8 and theorem VI.5.

Chapter VII Q-MACHINES

Chapter VI has established that closure under counting modulo $\mathbb{Z}_n$ is, in most circumstances, the same as closure under a quite natural set operation. Now we ask: Is there a machine characterisation of $\mathbb{Z}_n \Delta_0^{\mathbb{N}}$? The answer is YES - well, at least sometimes. The proof of this forms the substance of chapters VII and VIII. In chapter VII we concentrate on the machines themselves.

To begin we must define our new machines. They are variants of the Bel'tyukov Stack Register Machines and were devised by Jeff Paris.

VII.1 Definition: Suppose $Q = \{a_1, a_2, \dots, a_s\} \subseteq \mathbb{N}$.

Q-SRM's are defined in exactly the same way as ordinary SRM's (definition I.13) except that type (i) instructions are replaced by instructions of the form:

$$\text{type } (\bar{i}) \left\{ \begin{array}{l} \text{If } \phi^i_1(r, t_0, \dots, t_k, x_1, \dots, x_m) \text{ then } t_i := t_i + a_1; \\ \text{If } \phi^i_2(r, t_0, \dots, t_k, x_1, \dots, x_m) \text{ then } t_i := t_i + a_2; \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ \text{If } \phi^i_s(r, t_0, \dots, t_k, x_1, \dots, x_m) \text{ then } t_i := t_i + a_s; \end{array} \right.$$

where $\phi^i_1, \phi^i_2, \dots, \phi^i_s$ are open LA-formulae defining s subsets of $\mathbb{N}^{k+m+2}$ which are mutually disjoint but whose union is the whole of $\mathbb{N}^{k+m+2}$.

So a Q-SRM has a work-register r, stack registers $t_0, \dots, t_k$ and input registers $x_1, \dots, x_m$. Programs are of the form $L_1 L_2 \dots L_p L_{p+1}$ where L_{p+1} is stop! and $L_1, L_2, \dots, L_p$ each take one of the forms $(\bar{i})$ (immediately above), (ii) or (iii) (definition I.13).

There is at most one type $(\bar{i})$ instruction for each stack register t_i and a type $(\bar{i})$ instruction increasing t_i also has the effect of setting $t_j := 0$ for all $j < i$.

The initial configuration has the work-register and the stack registers all zero and the values of the inputs in the input registers. The first instruction executed is L_1.

The notion of an m-ary function being calculated by a Q-SRM with m input registers is exactly as for an ordinary SRM (definition I.14). The value of the function always ends up in the top stack t_k.

VII.2 <u>Definition</u>: Again for $u,v : \mathbb{N} \to \mathbb{N}$, non-decreasing with $u(1),v(1) \geq 1$, we can define <u>Q-Space(u,v)</u> by:

$g : \mathbb{N}^m \to \mathbb{N}$ is in Q-Space(u,v) just if there is a Q-SRM which computes g such that throughout any run of the machine,

$$r < u^{<j>}(\max(x_1,\ldots,x_m) + 1) \qquad \text{for some } j > 0,$$

and

$$t_i < v^{<\ell>}(\max(x_1,\ldots\ldots,x_m) + 1) \qquad \text{for all } 0 \leq i \leq k, \text{ for some } \ell > 0.$$

VII.3 <u>Example</u>: Let $Q = \{1,2,3\}$. Consider the $\{1,2,3\}$-SRM with stack registers t_0,t_1,t_2,t_3, input registers x_1,x_2 and program

```
        ⎧ If $0 \neq 0$ then $t_3 := t_3 + 1$;
1.      ⎨ If $r \leq x_2$ then $t_3 := t_3 + 2$;
        ⎩ If $r > x_2$  then $t_3 := t_3 + 3$;
        ⎧ If $0 = 0$ then $t_2 := t_2 + 1$;
2.      ⎨ If $0 \neq 0$ then $t_2 := t_2 + 2$;
        ⎩ If $0 \neq 0$ then $t_2 := t_2 + 3$;
3.        If $r + t_1 = x_1$ then 8. else 4.;
        ⎧ If $0 = 0$ then $t_0 := t_0 + 1$;
4.      ⎨ If $0 \neq 0$ then $t_0 := t_0 + 2$;
        ⎩ If $0 \neq 0$ then $t_0 := t_0 + 3$;
5.        If $r + t_2 = t_0$ then 6. else 4.;
6.        $r := t_0$;
7.        If $t_1 + t_1 = t_1$ then 1. else 1.;
8.        Stop!
```

Here is what the machine does on input (2,1):

	r	t_0	t_1	t_2	t_3	x_1	x_2
Initial contents of stacks.	0	0	0	0	0	2	1
Apply 1.	0	0	0	0	2	2	1
Apply 2.	0	0	0	1	2	2	1
Apply 3. No effect on stacks. Goto 4.	0	0	0	1	2	2	1
Apply 4.	0	1	0	1	2	2	1
Apply 5. Goto 6.	0	1	0	1	2	2	1
Apply 6.	1	1	0	1	2	2	1
Apply 7. Goto 1.	1	1	0	1	2	2	1
Apply 1. ($t_0, t_1, t_2 := 0$)	1	0	0	0	4	2	1
Apply 2.	1	0	0	1	4	2	1
Apply 3. Goto 4.	1	0	0	1	4	2	1
Apply 4.	1	1	0	1	4	2	1
Apply 5. Goto 4.	1	1	0	1	4	2	1
Apply 4.	1	2	0	1	4	2	1
Apply 5. Goto 6.	1	2	0	1	4	2	1
Apply 6.	2	2	0	1	4	2	1
Apply 7. Goto 1.	2	2	0	1	4	2	1
Apply 1.	2	0	0	0	7	2	1
Apply 2.	2	0	0	1	7	2	1
Apply 3. Goto 8.	2	0	0	1	7	2	1
Stop!	2	0	0	1	7	2	1

The output of the machine, i.e. the final contents of stack register t_3 is 7.

In fact this machine calculates the function f where

$$f(x_1, x_2) = \begin{cases} 2x_1 + 2 & \text{if } x_1 \leq x_2 \\ 3x_1 - x_2 + 2 & \text{if } x_1 > x_2 . \end{cases}$$

(This is not a function of any particular significance.)

Furthermore, whatever the values of the imputs, this machine will always have

$$r \leq x_1 < (\max(x_1,x_2) + 1)$$

and $$t_0, t_1, t_2, t_3 < 3(\max(x_1,x_2) + 1)$$

which means that

$$f \in \{1,2,3\}\text{-Space}(\underline{id},\underline{lin})$$

where <u>id</u> is the identity function and <u>lin</u> : $x \to 2x + 1$.

We can also prove a theorem, very similar to lemma II.3, which simplifies the task of showing a given function to be in Q-Space(u,v).

First for $\mathfrak{F}$ some class of functions we make:

VII.4 Definition: A $\underline{\text{Q-SRM}}_{\mathfrak{F}}$ is a machine with work-register r, stack registers $t_0,\ldots,t_k$ and input registers $x_1,\ldots,x_m$. Its program is $L_1 L_2 \ldots L_p L_{p+1}$ where L_{p+1} is Stop! and $L_1,\ldots,L_p$ are each one of the following types:

Type ($\bar{\text{i}}$)':

$$\begin{cases} \text{If } f_1^i(r,\vec{t},\vec{x}) = g_1^i(r,\vec{t},\vec{x}) \text{ then } t_i := t_i + a_1; \\ \text{If } f_2^i(r,\vec{t},\vec{x}) = g_2^i(r,\vec{t},\vec{x}) \text{ then } t_i := t_i + a_2; \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \text{If } f_s^i(r,\vec{t},\vec{x}) = g_s^i(r,\vec{t},\vec{x}) \text{ then } t_i := t_i + a_s; \end{cases}$$

where $f_1^i,\ldots,f_s^i,g_1^i,\ldots,g_s^i \in \mathfrak{F}$ and the conditions so defined are at all times of application mutually exclusive and also exhaustive.

Type (i)':

$$t_i := f(r,t_{i+1},\ldots,t_k,x_1,\ldots,x_m);$$

where $f \in \mathfrak{F}$.

Type (ii)':

$$r := f(r,t_0,\ldots,t_k,x_1,\ldots,x_m);$$

where $f \in \mathfrak{F}$.

Type (iii)':

If $f(r,\vec{t},\vec{x}) = g(r,\vec{t},\vec{x})$ then L_j else L_ℓ;

where $f,g \in \mathcal{F}$, $1 \leq j,\ell \leq p+1$.

For each stack register t_i, $0 \leq i \leq k$, there is either one type $(\bar{i})'$ instruction or one type $(i)'$ instruction or none at all. Type $(\bar{i})'/(i)'$ instructions set $t_j := 0$ for all $j < i$ even if the value of t_i is not itself altered.

VII.5 Definition: For appropriate functions u and v, we define $\text{Q-Space}_{\mathcal{F}}(u,v)$ in the obvious way following definitions I.14, II.2 and VII.2.

And now, following lemma II.3, we have

VII.6 Theorem: Let Q be a finite set of numbers and u,v be non-decreasing functions from numbers to numbers with $u(1),v(1) \geq 1$.

If

$v(x) \geq x$ for all $x \in \mathbb{N}$

$\forall i,j > 0, \exists k > 0,$ $u^{<i>}(x),u^{<j>}(x) \leq u^{<k>}(x)$ for all $x \in \mathbb{N}$

$\forall i,j > 0, \exists k > 0,$ $u^{<i>}(v^{<j>}(x)) \leq u^{<k>}(x)$ for all $x \in \mathbb{N}$

$\forall i > 0, \exists k > 0,$ $u^{<i>}(x) \leq v^{<k>}(x)$ for all $x \in \mathbb{N}$

$1 \in Q$

then

$\text{Q-Space}_{\text{Q-Space}(u,v)}(u,v) = \text{Q-Space}(u,v)$.

Proof: Similar to that of lemma II.3. Showing inclusion from right to left is trivial; the hard part is to show the opposite inclusion.

First observe that $1 \in Q$ implies that Q-SRM's are at least as powerful as ordinary SRM's for type (ii)/(iii) instructions are the same in both cases and we can change a type (i) instruction

$t_i := t_i + 1;$

into the type $(\bar{i})$ instruction

$$\left\{\begin{array}{l} \text{If } \ 0 = 0 \ \text{ then } \ t_i := t_i + 1; \\ \text{If } \ 0 \neq 0 \ \text{ then } \ t_i := t_i + a_2; \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \text{If } \ 0 \neq 0 \ \text{ then } \ t_i := t_i + a_s; \end{array}\right.$$

(without loss of generality $a_1 = 1$.)

Therefore Space(u,v) $\subseteq$ Q-Space(u,v).

Now instruction types (i)'/(ii)'/(iii)' are the same for both $\text{SRM}_{\mathfrak{F}}$'s and Q-$\text{SRM}_{\mathfrak{F}}$'s. It should now be easy to see that we can use the method of lemma II.3, which translates a single $\text{SRM}_{\text{Space}(u,v)}$ instruction of type (i)'/(ii)'/(iii)' into a block of SRM instructions and produces a program running in the desired space bounds, equally well in this proof. So we can translate a single Q-$\text{SRM}_{\text{Q-Space}(u,v)}$ instruction of type (i)'/(ii)'/(iii)' (which is the same thing as an $\text{SRM}_{\text{Q-Space}(u,v)}$ instruction of type (i)'/(ii)'/(iii)') into a block of Q-SRM instructions and produce a program for a machine running in the desired space bounds.

This leaves the type (ī)' instructions to deal with. However, a type (ī)' instruction can be simulated by type (i)/(ī)/(iii)' instructions. For if a machine has the instruction

$$\left\{\begin{array}{l} \text{If } \ f_1^i = g_1^i \ \text{ then } \ t_i := t_i + a_1; \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \text{If } \ f_s^i = g_s^i \ \text{ then } \ t_i := t_i + a_s; \end{array}\right.$$

then, assuming first that $0 \notin Q$, we can add new stack registers

$$t_{-(s-1)}, \ldots, t_{-2}, t_{-1}$$

and replace our line by the block at the top of the next page.

```
1.      If $f_1^i = g_1^i$ then 2. else 3.;
2.      $t_{-1} := t_{-1} + 1$;
3.      If $f_2^i = g_2^i$ then 4. else 5.;
4.      $t_{-2} := t_{-2} + 1$;
        ................................
2s-3.   If $f_{s-1}^i = g_{s-1}^i$ then 2s-2. else 2s-1.;
2s-2.   $t_{-(s-1)} := t_{-(s-1)} + 1$;
        { If $t_{-1} \neq 0$ then $t_i := t_i + a_1$;
        { If $(t_{-1} = 0) \wedge (t_{-2} \neq 0)$ then $t_i := t_i + a_2$;
2s-1.   { ...............................................
        {
        { If $(t_{-1} = 0) \wedge \ldots \wedge (t_{-(s-1)} = 0)$ then $t_i := t_i + a_s$;
```

The effect of all this will be as required and:

(i) As we have seen lines 2.,4.,...,2s-2. are equivalent to type $(\bar{\text{i}})$ Q-SRM instructions.

(ii) Since 2s-1. sets $t_{-(s-1)},\ldots,t_{-1} := 0$, whenever we encounter the block afresh $t_{-(s-1)} = \ldots = t_{-1} = 0$. Therefore these registers never hold a value greater than 1. Now since the block only comes into play if the old machine would definitely have been about to execute the line that has been replaced, and this means that t_i must become non-zero. Therefore since a non-zero t_i does so, these new stack registers must satisfy whatever space bounds are in force.

Thus we have the right kind of machine running in the right space bounds.

If $0 \in Q$ then there is in fact no problem for we can assume without loss of generality that $a_s = 0$. It is easy to see that when the "$t_i := t_i + a_s$" comes to be executed then all the new stack registers are still 0, so there is no trouble in satisfying any bounds.

End of proof of theorem VII.6.

Notice that having $0 \in Q$ makes very little difference at all and therefore the Q's that we encounter will not include 0. The various theorems would still hold if we added it in.

VII.7 Corollary: For appropriate $Q \ni 1$,

$$\text{Q-Space}_{\text{Q-Space}(\underline{1},\square)}(\underline{1},\square) = \text{Q-Space}(\underline{1},\square)$$

where $\underline{1} : x \mapsto 1$ and $\square : x \mapsto 2x+1$ as before.

Proof: $(\underline{1},\square)$ satisfy the conditions of theorem VII.6.

Chapter VIII $\{1,n+1\}$-Space$(\underline{1},\square)$ and counting modulo $\mathbb{Z}_{n!}$

In this chapter we give a machine characterization of $\mathbb{Z}_{n}\Delta_0^{\mathbb{N}}$ for certain numbers n, namely

$1!,\ 2!,\ 3!,\ \ldots\ .$

Our main result (theorem VIII.16) is that the class of $\mathbb{Z}_{n!}\Delta_0$-functions is precisely $\{1,n+1\}$-Space$(\underline{1},\square)$.

This means however that $\mathbb{Z}_{n}\Delta_0^{\mathbb{N}}$ (and the $\mathbb{Z}_{n}\Delta_0$-functions) are characterized for other values of n as well. For by theorem III.10

$$\mathbb{Z}_{n!}\Delta_0^{\mathbb{N}} = \mathbb{Z}_{m}\Delta_0^{\mathbb{N}}$$

for any m having the same prime factors as $n!$. In other words we characterize $\mathbb{Z}_{m}\Delta_0^{\mathbb{N}}$ for any m such that

$(\ p$ prime and $p|m)$ implies $(q|m$ for all primes $q \leq p)$.

VIII.1 <u>Definition</u>: For all $n \in \mathbb{N}$, the <u>$\mathbb{Z}_{n}\Delta_0$-functions</u> are the functions with graph in $\mathbb{Z}_{n}\Delta_0^{\mathbb{N}}$ and with values bounded by some polynomial in the arguments.

VIII.2 <u>Definition</u>: For Q a finite subset of $\mathbb{N}$ and non-decreasing functions $u,v : \mathbb{N} \to \mathbb{N}$ with $u(1),v(1) \geq 1$,

<u>Q-Space$_*$(u,v)</u> is defined to be the class of sets whose characteristic functions lie in Q-Space(u,v).

VIII.3 <u>Facts</u>: For all $n \in \mathbb{N}$,

(i) If $A \subseteq \mathbb{N}^q$ some $q \in \mathbb{N}$, $A \in \mathbb{Z}_{n}\Delta_0^{\mathbb{N}}$, and $f_1,f_2,\ldots,f_q : \mathbb{N}^m \to \mathbb{N}$ are $\mathbb{Z}_{n}\Delta_0$-functions

then

$$\{(x_1,\ldots,x_m) : (f_1(x_1,\ldots,x_m),\ldots,f_q(x_1,\ldots,x_m)) \in A\} \in \mathbb{Z}_{n}\Delta_0^{\mathbb{N}}.$$

(ii) The $\mathbb{Z}_{n}\Delta_0$-functions are closed under composition.

<u>Proof</u>: Very easy.

Now we present a result analogous to the first part of theorem V.4.

VIII.4 Proposition: For all $n \in \mathbb{N}$,

$$\mathbb{Z}_{n!},\Delta_0^{\mathbb{N}} \subseteq \{1,n+1\}\text{-Space}_*(\underline{1},\square).$$

Proof: Given corollary VII.7 it is easy to show, in the manner of propositions II.4 and II.6, that $\{1,n+1\}$-Space$(\underline{1},\square)$ is closed under composition and that $\{1,n+1\}$-Space$_*(\underline{1},\square)$ includes $=$, $\leq$, graph(+) and graph($\cdot$) and is closed under explicit transformations, boolean operations and bounded quantification (II).

It remains to show that $\{1,n+1\}$-Space$_*(\underline{1},\square)$ is closed under counting modulo $\mathbb{Z}_{n!}$.

By theorem III.10 this is equivalent to closure under counting modulo all of $\mathbb{Z}_1$, $\mathbb{Z}_2$, ..., $\mathbb{Z}_n$.

This in turn, as we have seen in theorem VI.5, is equivalent to closure under *-counting modulo all of $2,3,\ldots,n$.

We show first closure under *-counting modulo n. The method used can be extended, as we shall see later, to *-counting modulos $2,\ldots,n-1$.

VIII.5 Lemma:

$\{1,n+1\}$-Space$_*(\underline{1},\square)$ is closed under *-counting modulo n.

Proof: Suppose that a set $A \subseteq \mathbb{N}^m$ is in $\{1,n+1\}$-Space$_*(\underline{1},\square)$. It is best to think first how we might approach the problem of calculating $A^{i,(n)}$ using more powerful machines than Q-SRM's. The obvious way is this: we keep two running totals. The first (T_1) simply counts upwards one at a time. The second (T_2) counts up until it reaches the value $n-1$; at the next time T_2 would ordinarily have increased, it falls back to zero and starts again. Given inputs $x_1,\ldots,x_m$ we work in the following fashion: each time we increase T_1, we increase T_2 just if

$$(x_1,\ldots,x_{i-1},T_1,x_{i+1},\ldots,x_m) \in A.$$

We stop at the point where T_1 is increased to x_i (so before we have a chance to increase T_2 on that step): we accept if

$$(x_1,\ldots,x_m) \in A \wedge T_2 = 0,$$

otherwise we reject.

A machine that worked in this fashion would accept just if

$$(x_1,\ldots,x_m) \in A^{i,(n)},$$

for at the time T_1 is increased to a given value y it may easily be seen that T_2 stores

$$|\{z < y : (x_1,\ldots,x_{i-1},z,x_{i+1},\ldots,x_m) \in A\}| \text{ modulo } n.$$

Unfortunately, so far as Q-SRM's are concerned, it is not possible to store these totals as separate stacks. Either increasing T_1 would set T_2 to zero or vice versa.

But perhaps we can get round this difficulty by coding T_1 and T_2 into a single number. After all T_2 is always less than n. How about coding $<T_1,T_2>$ by

$$nT_1 + T_2?$$

Then maybe we can make a $\{1,n+1\}$-$\mathrm{SRM}_{\{1,n+1\}\text{-Space}(\underline{1},\square)}$ which calculates the characteristic function of $A^{i,(n)}$ by holding to the specification that:

if, at any stage of the computation, the top stack holds value T then

$$|\{z < [T/n] : (x_1,\ldots,x_{i-1},z,x_{i+1},\ldots,x_m) \in A\}| \equiv T \bmod n.$$

The machine accepts just in case $(x_1,\ldots,x_m) \in A$ and the top stack at some time holds the value nx_i.

A moment's thought will show that during a run of the machine we will want to be able to choose between increasing T by

$$1,\ n \text{ or } n+1$$

depending on whether $(x_1,\ldots,x_{i-1},T,x_{i+1},\ldots,x_m) \in A$ and on the value of $T \bmod n$.

We could obviously accomplish this with a $\{1,n,n+1\}$-SRM but with a $\{1,n+1\}$-SRM however it does not seem possible.

And so we abandon this particular coding but not the idea itself. It is possible to find a somewhat more complicated coding which can be made to work.

First there is a number $N > n$.

Secondly a set D of numbers all distinct modulo N.

Within that set D a subset $E = \{e_0,\ldots,e_{n-1}\}$ (it is usually convenient that $e_0 = 0$).

Lastly a function $q : (D\setminus E) \to \{1,n+1\}$.

These must all fit together in such a way that

(i) If we start from e_j $(0 \leq j \leq n-1)$ and to start with add 1. but from then onward add successively $q(e_j + 1)$, $q(e_j + 1 + q(e_j + 1))$ and so on, so that we hop from number to number by adding $q(x)$ when we reach $x \in D$, then the first number we reach outside D will be $N + e_j$.

(ii) If at the first step we add $n+1$ to e_j rather than 1, and then carry on as before according to q, then the first number we reach outside D is $N + e_{j+1}$ (or $N + e_0$ if $j = n - 1$).

<u>For instance</u>: Suppose $n = 5$. Let $N = 50$ and list D as

0	1	7	13	19	25	26	32	38	44
5	6	12	18	24	30	31	37	43	49
10	11	17	23	29	35	36	42	48	54
15	16	22	28	34	40	41	47	53	59
20	21	27	33	39	45	46	52	58	64.

Define $e_j = 5j$ for $0 \leq j \leq n-1$. So $E = \{0,5,10,15,20\}$.
q is the function that takes us along rows, e.g. $q(1) = 6$, $q(30) = 1$.
Also $q(44) = q(49) = q(54) = q(59) = q(64) = 6$.

Do N, D, E and q fit together in the right way?

Well, if we start from e.g. $e_2 = 10$, then adding 1 takes us to 11. $q(11) = 6$, which brings us to 17 and thence to 23, 29, 35, 36, 42, 48, 54 and finally to $60 \notin D$. $60 = 50 + 10 = N + e_2$ as desired.

If on the other hand we start from $e_2 = 10$ and make our first jump

6 rather than 1 then we will follow the sequence

$$16, 22, 28, 34, 40, 41, 47, 53, 59, 65.$$

$65 \notin D$ and $65 = 50 + 15 = N + e_3$ as required.

Note also the case for $e_4 = 20$ where if we make our first jump 6 we follow the sequence $26, 32, 38, 44, 50 = N + e_0$.

It isn't hard to see that N, D, E and q do indeed work in the way we specified.

What now will be the coding we derive from such a set up and how will it help us construct machines?

We begin by coding pairs of elements of $\mathbb{N}$ and of D. $\langle x,d\rangle$ is coded by $Nx + d$. This coding is well defined because all elements of D are distinct modulo N.

For $y \in \mathbb{N}$ we define y^D to be that $d \in D$ such that

$$y \equiv d \text{ modulo } N$$

and we define $a(y)$ to be

$$\begin{cases} \frac{1}{N}(y - y^D) & \text{if } y \geq y^D \\ 0 & \text{otherwise.} \end{cases}$$

So y codes $\langle a(y), y^D\rangle$ for all but a few values of y such that $y < y^D$.

Fix $x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_m$ for the moment. Imagine now that we start from $y = 0$, and move through the natural numbers by adding $q(y)$ to y if $y^D \in D\setminus E$, and if $y^D \in E$ adding 1 if $(x_1,\ldots,x_{i-1},a(y),x_{i+1},\ldots$ $\ldots,x_m) \notin A$ or $n+1$ if $(x\ ,\ldots,x_{i-}\ ,a(y),x_{i+}\ ,\ldots,x_m) \in A$.

Given the collective properties of N, D, E and q it should be clear that if, for given $w \in \mathbb{N}$ and $0 \leq j \leq n-1$, we ever reach

$$wN + e_j$$

then the least number z encountered such that $a(z) = w + 1$ is

$$(w + 1)N + e_j \qquad \text{if } (x_1,\ldots,x_{i-1},w,x_{i+1},\ldots,x_m) \notin A$$

and $$(w + 1)N + e_{(j+\ )\text{mod } n} \qquad \text{if } (x_1,\ldots,x_{i-1},w,x_{i+1},\ldots,x_m) \in A.$$

This fact provides us in turn with an inductive proof that if we start from $y = 0 = 0 \cdot N + e_0$ and proceed as specified, then for all $w \in \mathbb{N}$, if z is the first number encountered such that $\bar{z} = w$ then in fact $z = \langle w, e_s \rangle$ where $s = \sigma^i_A(x_1, \ldots, x_{i-1}, w, x_{i+1}, \ldots, x_m) \bmod n$.

$w \in A^{i,(n)}$ just if this first number encountered is $wN = wN + e_0$ and $(x_1, \ldots, x_{i-1}, w, x_{i+1}, \ldots, x_m) \in A$.

Let's now use this to construct a machine.

Since D is a finite set, y^D and $a(y)$ are both Δ_0-functions and therefore, by theorem II.13, both in Space$(\underline{1}, \Box) = \{1\}$-Space$(\underline{1}, \Box)$, which is clearly a subset of $\{1,n+1\}$-Space$(\underline{1}, \Box)$. Also, because $\{1,n+1\}$-Space$(\underline{1}, \Box)$ is closed under composition, the set $\bar{A}$ defined by

$$\bar{A} = \{(x_1, \ldots, x_m) : (x_1, \ldots, x_{i-1}, a(x_i), x_{i+1}, \ldots, x_m) \in A\}$$

is in $\{1,n+1\}$-Space$_*(\underline{1}, \Box)$. Also the set $S \subseteq D\ E$ defined by

$$d \in S \quad \text{iff} \quad q(d) = n+1$$

is a finite set and therefore in $\{1,n+1\}$-Space$_*(\underline{1}, \Box)$.

Consider now the following set:

$$K = \{(x_1, \ldots, x_m) : \left[(x_1, \ldots, x_m) \in \bar{A} \wedge \bigvee_{d \in E} (d \equiv x_i \bmod N)\right] \vee \left[\bigvee_{d \in S} (d \equiv x_i \bmod N)\right]\}.$$

K $\{1,n+1\}$-Space$_*(\underline{1}, \Box)$ for it is obtained from $\bar{A}$ and the equivalence relation modulo N (which is in $\Delta_0^{\mathbb{N}}$) by explicit transformations and boolean operations. So the characteristic function (χ_K) of this set is in $\{1,n+1\}$-Space$(\underline{1}, \Box)$. As are the constant functions $0, 1$ and N, multiplication and χ_A, the characteristic function of A.

The following is therefore an SRM$_{\{1,n+1\}\text{-Space}(\underline{1}, \Box)}$. It has input registers $x_1, \ldots, x_m$ and stack registers t_0, t_1. The program is:

```
1.    If $a(t_0) = x_i$ then 4. else 4.;
2.    { If $\chi_K(x_1,\ldots,x_{i-1},t_0,x_{i+1},\ldots,x_m) = 1$ then $t_0 := t_0 + 1$;
      { If $\chi_K(x_1,\ldots,x_{i-1},t_0,x_{i+1},\ldots,x_m) = 0$ then $t_0 := t_0 + n+1$;
3.    If $0 = 0$ then 1. else 1.;
4.    If $t_0 = N\cdot x_i$ then 5. else 6.;
5.    If $\chi_A(x_1,\ldots,x_m) = 0$ then 7. else 6.;
6.    $t_1 := 1$;
7.    Stop!
```

This machine also happens to compute the characteristic function of $A^{i,(n)}$.

First observe that if $t^D \in D \setminus E$ then line 2. will always give $t_0 := t_0 + q(t^D)$; for we add 1 if $(x_1,\ldots, x_{i-1},t_0,x_{i+1},\ldots,x_m) \notin K$, i.e. if $t^D \notin S$ and so $q(t^D) = 1$; and we add $n+1$ if $t^D \in S$, i.e. $q(t^D) = n+1$.

If $t^D \in E$ then we add 1 if $(x_1,\ldots,x_{i-1},a(t_0),x_{i+1},\ldots,x_m) \notin A$
and $n+1$ if $(x_1,\ldots,x_{i-1},a(t_0),x_{i+1},\ldots,x_m) \in A$.

Therefore if we keep repeating line 2. we will find, as we saw above, that $(x_1,\ldots,x_m) \in A^{i,(n)}$ just if the first value of t_0 encountered such that $a(t_0) = x_i$ is in fact $N\cdot x_i$ and $(x_1,\ldots,x_m) \in A$. This is precisely the circumstance in which our machine accepts (i.e. produces output $t_1 = 0$).

Thus $A^{i,(n)} \in \{1,n+1\}$-Space$_*(\underline{1},\mathcal{D})$ by corollary VII.7.

However before we declare this proof at an end it remains of course to show that we can always find suitable number N, suitable sets D and E and suitable function q. We have shown this is possible for $n = 5$, we must now extend this. We proceed in exactly the same way for any $n \geq 1$.

First for given n define $a_i = (i-1)(n+1) + 1$ for $1 \leq i \leq n-1$ and define $a_0 = 0$. So $a_0, a_1, \ldots, a_{n-1}$ is the same as

$$0,\ 1,\ n+2,\ 2n+3,\ 3n+4,\ \ldots,\ (n-2)(n+1)+1.$$

$a_0,\ldots,a_{n-1}$ are all distinct modulo n, in fact a_i falls into the equivalence class $\bar{i}$.

Define now the row of numbers $r(x) = r_0(x),\ldots,r_{n-1}(x)$ by

$$r_i(x) \quad = \quad a_i + n\cdot x.$$

Then, for all $x \in \mathbb{N}$, the set of numbers given by

$$r(0) \cup r(1) \cup \ldots \cup r(x)$$

has $n\cdot(x+1)$ elements and each of these elements is distinct modulo $n\cdot(x+1)$. This is because, for $0 \leq i,j \leq n-1$, $0 \leq y,z \leq x$,

$$r_i(y) \equiv r_j(z) \bmod n\cdot(x+1)$$

implies $r_i(y) \equiv r_j(z) \bmod n$

implies $a_i = r_i(0) \equiv r_j(0) = a_j \bmod n$

implies $i = j$

implies $n\cdot y \equiv n\cdot z \bmod n\cdot(x+1)$

implies $y \equiv z \bmod (x+1)$

implies $y = z$ since $0 \leq y,z \leq x$.

For each n we define our set D to be

$$r(0) \cup r(1) \cup \ldots \cup r(2n-1)$$

and so $N = 2n^2 > n$ for all $n \geq 1$.

$E = \{e_0,\ldots,e_{n-1}\}$ is defined by $e_i = n\cdot i$.

Finally observe that for all $0 \leq i \leq n-2$

$$a_{i+1} - a_i \in \{1,n+1\},$$

and that for all $j \in \mathbb{N}$

$$\begin{aligned} r_0(n+j) = n\cdot(n+j) &= ((n-2)(n+1) + 1 + n\cdot j) + (n+1) \\ &= r_{n-1}(j) + (n+1). \end{aligned}$$

Also $r_0(2n+j) = 2n^2 + n\cdot j = N + n\cdot j$, so $r_0(2n+j) = N + e_j$.

We can therefore define q to be that function which carries $r_i(x)$ to $r_{i+1}(x)$ for all $0 \leq x \leq 2n-1$, for all $0 \leq i \leq n-2$ - that is to say $q(r_i(x)) = r_{i+1}(x) - r_i(x) \in \{1,n+1\}$.

For $0 \leq x \leq 2n-1$

$$q(r_{n-1}(x)) = n+1$$

which will take $r_{n-1}(x)$ to $r_0(n + x)$.

It is left to the reader to convince themselves that N, D, E and q will work together in the desired way.

End of proof of lemma VIII.5.

VIII.6 Lemma:

$\{1,n+1\}$-$\text{Space}_*(\underline{1},\Box)$ is closed under $*$-counting modulo m for all $1 \leq m < n$.

Proof: An examination of the proof of lemma VIII.5 should convince the reader that in order to show closure for a given m it suffices to find a number $N' > n$, a set $D' \subseteq \mathbb{N}$ such that $|D'| = N'$ and all the elements of D' are distinct modulo N', a subset $E' = \{e'_0,\ldots,e'_{m-1}\}$ of D' and a function $q' : (D' \setminus E') \to \{1,n+1\}$, which four work together in such a way that

(i) If, for $0 \leq j \leq m-1$, we start from e_j and first add 1 and thereafter add $q'(x)$ whenever we reach $x \in D'$ then the first number reached outside D' is $N' + e_j$.

(ii) If we do the same but add $n+1$ at the beginning instead of 1, we eventually reach $N' + e_{(j+1) \bmod m}$.

Well in fact given N, D, E and q which work for n it is not too difficult to derive N', D', E', q' working for m.

For $j \in \mathbb{N}$ define

$$D_j = \{x : (x - jN) \in D\}.$$

(In this proof N, D, E and q will be as in the proof of lemma VIII.5.)

For $m < n$ define

$$N' = (n-m+1)N$$

$$D' = D_0 \cup D_1 \cup \ldots \cup D_{n-m}$$

and E' by $e'_0 = e_0, \ldots, e'_m = e_m$.

Finally define q' as follows:

On $D_0 = D$ it is exactly the same as q.

For D_j, $1 \le j \le n-m$,

$$q'(d) = \begin{cases} q(d - jN) & \text{if } (d - jN) \in D \setminus E \\ 1 & \text{if } (d - jN) \in E \setminus \{e_{m+j-1}\} \\ n+1 & \text{if } (d - jN) = e_{m+j-1}. \end{cases}$$

And now

(i) Consider what happens if we start at $e_j' = e_j$ $(0 \le j \le m-1)$, add 1 and then move through D' according to q'.

On D_0 q' is the same as q, so the first number we reach outside D_0 will be $N + e_j$. Since $j \le m-1$, $q'(N+e_j) = 1$ and we now move through D_1 exactly as we did through D_0, eventually reaching $2N + e_j$. Proceeding in this way we eventually reach $(n-m+1)N + e_j = N' + e_j'$ as desired.

(ii) So what happens when we add $n+1$ rather than 1 when we start from $e_j' = e_j$? Take the case $0 \le j \le m-2$ first. Again, because q' is the same as q for D_0 we eventually reach $N + e_{j+1}$ and now as we have already seen this will lead in the end to $N' + e_{j+1} = N' + e_{j+1}'$ as desired.

This leaves the case $j = m - 1$ to deal with. This time after leaving D_0 we arrive at $N + e_m$.

Now then $N + e_m - N = e_m = e_{m+1-1}$ and so

$$q'(N + e_m) = n+1.$$

Thereafter in D_1, q' behaves like q behaves on D. Therefore we will eventually reach $2N + e_{m+1}$. Once again

$$q'(2N + e_{m+1}) = n+1$$

and the process continues through $D_2, \ldots, D_{n-m}$ eventually leading to $(n-m+1)N + e_{(m+n-m) \bmod n} = (n-m+1)N + e_0 = N' + e_0'$ as desired.

End of proof of lemma VIII.6.

Lemmas VIII.5 and VIII.6 are all that remained to be shown to prove proposition VII.4.

End of proof of proposition VII.4.

Proposition VII.4 corresponds to the first part of theorem V.4. Now we have results akin to those in, or used in, the second part of that theorem.

We first consider some submonoids of ${}^n n$ and derive some results like those of chapter IV.

VIII.7 Definition: R_n is defined to be the subset of ${}^n n$ comprising those functions $f : n \to \underset{\sim}{n}$ such that for all $0 \leq i \leq n$

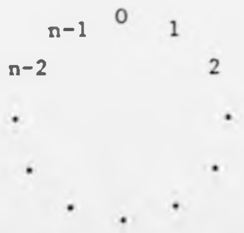
$$f(i+1) \equiv f(i) + 1 \text{ modulo } n$$

and $$f(0) \equiv f(n-1) + 1 \text{ modulo } n.$$

We can think of such functions as rotations of an n-gon, so in fact $R_n \cong \mathbb{Z}_n$.

VIII.8 Definition: T_n is defined to be the smallest subnonoid of ${}^n n$ containing the non-decreasing functions (i.e. functions $f : \underset{\sim}{n} \to \underset{\sim}{n}$ such that for all $0 \leq i,j \leq n$, $i \leq j$ implies $f(i) \leq f(j)$) and R_n.

The functions in T_n can be thought of as cyclically increasing. That is: if we start from $f(0)$ and move to $f(1)$, $f(2)$ etc. by moving clockwise around the circle



then although we may in time come back to $f(0)$ we never go beyond it for a second time.

For instance if $n = 5$ then the function f such that

$$f(0) = 3,\ f(1) = 4,\ f(2) = 4,\ f(3) = 0,\ f(4) = 3$$

is in T_5.

We now have a result very similar to lemma IV.3 but with T_n rather than ^nn and R_n instead of S_n.

VIII.9 <u>Lemma</u>: Given $=$, $<$, explicit transformations, boolean operations and bounded quantification (I), for $n \geq 2$ closure under counting modulo R_n and closure under counting modulo $(T_n \setminus R_n)$ together imply closure under counting modulo T_n.

<u>Proof</u>: After observing that $(T_n \setminus R_n)$ is indeed closed under composition and that R_n is precisely the submonoid of one-one functions of T_n, the proof of lemma IV.3 carries over directly to the present theorem.

This is followed, not surprisingly, by an analogue of lemma IV.2.

VIII.10 <u>Lemma</u>: Given $=$, graph(suc), explicit transformations, boolean operations and bounded quantification (I), for $n \geq 1$ closure under counting modulo T_n implies closure under counting modulo $(T_{n+1} \setminus R_{n+1})$.

<u>Proof</u>: Again very much the same as for lemma IV.2. If we observe that the functions $g_1 : (\underset{\sim}{n+1}) \to \underset{\sim}{n}$ and $g_2 : \underset{\sim}{n} \to (\underset{\sim}{n+1})$ defined in that earlier proof are non-decreasing then all we need to know for that proof to carry over is that for any such g_1 and g_2 and any $f \in T_{n+1}$,

$$g_1 \bullet f \bullet g_2 \in T_n.$$

It suffices to know that this holds in two special cases:

(i) For non-decreasing functions f of $^{n+1}n+1$ (this is easy to see).

(ii) For any element f of R_{n+1} (and again a moment's consideration should convince the reader that this is so).

The proof of lemma IV.2 can now be applied with T_n substituted for ^nn and R_n for S_n throughout.

We can now prove that:

VIII.11 <u>Lemma</u>: Given $=$, $<$, graph(suc), explicit transformations, boolean operations and bounded quantification (I), for $n \geq 1$ closure under counting modulo $\mathbb{Z}_{n!}$ implies closure under counting modulo T_n.

<u>Proof</u>: By induction on n.

<u>$n = 1$</u> is trivial.

If true for n then closure under counting modulo $\mathbb{Z}_{(n+1)!}$ implies closure under counting modulo $\mathbb{Z}_{n!}$ and under counting modulo $\mathbb{Z}_{(n+1)}$. From the first we deduce, via the inductive hypothesis, closure under counting modulo T_n and from the second, since $R_{n+1} \equiv \mathbb{Z}_{n+1}$, closure under counting modulo R_{n+1}.

Then by lemma VIII.10 we deduce closure under counting modulo $(T_{n+1} \setminus R_{n+1})$ and from there, using lemma VIII.9 we deduce closure under counting modulo T_{n+1}. Q.E.D.

This is what we need to show:

VIII.12 Proposition:

All elements of $\{1,n+1\}$-Space$(\underline{1},\square)$ are $\mathbb{Z}_{n!}\Delta_0$-functions.

Proof: (Which is like that of lemma II.8.)

Step 1. If $f \in \{1,n+1\}$-Space$(\underline{1},\square)$ then there is a $\{1,n+1\}$-SRM, computing f whose work register is always zero and whose stack registers are strictly bounded by

$$p(\max(x_1,\dots,x_m) + 1)$$

where $p \in \mathbb{N}[x]$ and $x_1,\dots,x_m$ are the input values (f is an m-ary function).

In exactly the same way as with lemma II.8 we can find a machine with the same registers and still calculating f but with program L of the form

$$\begin{array}{l}
\Big(\text{If } (\vec{t},\vec{x}) \in \Theta_0 \text{ then } \Big(\left\{\begin{array}{l} \text{If } (\vec{t},\vec{x}) \in \Phi_0 \text{ then } t_0 := t_0 + 1 \\ \text{If } (\vec{t},\vec{x}) \notin \Phi_0 \text{ then } t_0 := t_0 + n+1 \end{array}\right\} ; L\Big); \\
\text{If } (\vec{t},\vec{x}) \in \Theta_1 \text{ then } \Big(\left\{\begin{array}{l} \text{If } (\vec{t},\vec{x}) \in \Phi_1 \text{ then } t_1 := t_1 + 1 \\ \text{If } (\vec{t},\vec{x}) \notin \Phi_1 \text{ then } t_1 := t_1 + n+1 \end{array}\right\} ; L\Big); \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
\text{If } (\vec{t},\vec{x}) \in \Theta_k \text{ then } \Big(\left\{\begin{array}{l} \text{If } (\vec{t},\vec{x}) \in \Phi_k \text{ then } t_k := t_k + 1 \\ \text{If } (\vec{t},\vec{x}) \notin \Phi_k \text{ then } t_k := t_k + n+1 \end{array}\right\} ; L\Big)\Big)
\end{array}$$

where $\Theta_0,\dots,\Theta_k \in \Delta_0^{\mathbb{N}}$ while $\Phi_0,\dots,\Phi_k \in \Delta_0^{\mathbb{N}}$ are the original conditions in the type $(\bar{i})$ instructions of the (1,n+1)-SRM computing f. (And $\vec{t}$ represents $t_0,\dots,t_k$ while $\vec{x}$ represents $x_1,\dots,x_m$.) There is effectively no work-register to worry about.

Step 2. (Which is like lemma V.6 or lemma II.10.)

Programs of the form just given, with $\Theta_0,\dots,\Theta_k,\Phi_0,\dots,\Phi_k \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$, can be replaced by similar programs which calculate the same function in the same bounds and still have $\mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ conditions but ones where t_0 is not mentioned.

Because setting $t_i := t_i + 1$ or $t_i + n+1$ means setting $t_j := 0$ for all $j < i$, this again comes down to showing that a function f_0 defined in a certain way is a $\mathbb{Z}_{n!}\Delta_0$-function. To define f_0 we first need a set $F_0 \subseteq \mathbb{N}^{k+m+1}$.

For given $t_1,\dots,t_k,x_1,\dots,x_m$

(i) $(0,t_1,\dots,t_k,x_1,\dots,x_m) \in F_0$.

(ii) $(y,t_1,\dots,t_k,x_1,\dots,x_m) \in F_0$ implies

$$\begin{cases} (y+1,t_1,\dots,t_k,x_1,\dots,x_m) \in F_0 & \text{if } (y,t_1,\dots,t_k,x_1,\dots,x_m) \in \Phi_0 \\ (y+n+1,t_1,\dots,t_k,x_1,\dots,x_m) \in F_0 & \text{otherwise.} \end{cases}$$

(iii) F_0 contains no elements other than those given by (i) and (ii).

Clearly F_0 is the set of values which appear in t_0 if we simply keep on repeating the operation

$$\left\{\begin{array}{lll} \text{If } (\vec{t},\vec{x}) \in \Phi_0 & \text{then} & t_0 := t_0 + 1 \\ \text{If } (\vec{t},\vec{x}) \notin \Phi_0 & \text{then} & t_0 := t_0 + n+1 \end{array}\right\}.$$

It can be shown that $\Phi_0 \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ implies $F_0 \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$. To see this we make:

VIII.13 Definition: For $\Phi \subseteq \mathbb{N}^{m+k+1}$ define

$$G_\Phi^{n+1} : \mathbb{N}^{m+k+1} \to {}^{n+1}n{+}1 \text{ by:}$$

If $q,t_1,\dots,t_k,x_1,\dots,x_m \in \mathbb{N}$ and $0 \le j \le n$ then

$$(G_\Phi^{n+1}(q,t_1,\dots,t_k,x_1,\dots,x_m))(j)$$

is the element of $\{0,1,\dots,n\}$ equivalent modulo $n+1$ to the first number greater than or equal to $(n+1)(q+1)$ reached by starting from $t_0 = (n+1)q + j$ and repeating

$$\left\{\begin{array}{l} \text{If } (\vec{t},\vec{x}) \in ¢ \text{ then } t_0 := t_0 + 1 \\ \text{If } (\vec{t},\vec{x}) \notin ¢ \text{ then } t_0 := t_0 + n+1 \end{array}\right\}.$$

<u>For instance</u>:

Suppose that $n = 5$ and that for some given $t_1,\dots,t_k,x_1,\dots,x_m$ (which we suppress),

$$18 \notin ¢,\ 19 \in ¢,\ 20 \notin ¢,\ 21 \notin ¢,\ 22 \in ¢,\ 23 \in ¢.$$

Then $G^{\epsilon}_{¢}(3)$ is as follows (from here on we won't usually bother to write in the sub- and superscripts):

For $(G(3))(0)$ we start at $3(n+1) + 0 = 18$. We must then add $n+1 = 6$ since $18 \notin ¢$. This brings us to 24. $24 \geq 6\cdot4$ and now $24 \equiv 0 \bmod 6$ implies $(G(3))(0) = 0$.

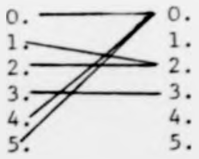
For $(G(3))(1)$ we start at 19. We then add 1 since $19 \in ¢$. This brings us to 20 to which we add 6 bringing us to $26 \geq 6\cdot4$. $26 \equiv 2 \bmod 6$ and so $(G(3))(1) = 2$. This also shows that $(G(3))(2) = 2$.

Continuing in this fashion we find that

$$(G(3))(3) = 3$$

and $$(G(3))(4) = (G(3))(5) = 0.$$

Diagrammatically $G(3)$ is the function



As we shall see later, not all elements of ${}^{n+1}n{+}1$ can be obtained in such a way. However for the moment let us connect $¢_0$, $G^{n+1}_{¢_0}$ and F_0.

(a) $¢_0 \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ implies $\text{graph}(G) \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$.

This is easy to see since (suppressing the variables $t_1,\dots,t_k,x_1,\dots$ $\dots,x_m$) membership of graph(G) can be deduced from the truth or falsity of the propositions

$$(n+1)q \in ¢_0,\ ((n+1)q+1) \in ¢_1, \dots,\ ((n+1)q+n) \in ¢_0 .$$

(b) $\mathrm{graph}(\mathrm{Sigma}^1_G) \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ implies $F_0 \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$.

This follows because (suppressing $t_1,\dots,t_k x_1,\dots,x_m$), for all $q \in \mathbb{N}$, for all $0 \leq j \leq n+1$,

$$(n+1)q + j \in F_0 \quad \text{implies} \quad (n+1)(q+1) + (G(q))(j) \in F_0,$$

or in other words, for all $t_0 \in \mathbb{N}$, $t_0 \in F_0$ implies

$$(n+1)([t_0/n+1] + 1) + (G([t_0/n+1]))(t_0 \bmod n+1) \in F_0 .$$

This means, since $0 \in F_0$, that

$$n+1 + (G(0))(0) \in F_0$$

and hence

$$2(n+1) + (G(1)\bullet G(0))(0) \in F_0,$$

and in fact for all $q \in \mathbb{N}$

$$q(n+1) + (G(q-1)\bullet\dots\bullet G(0))(0) \in F_0,$$

i.e.

$$q(n+1) + (\mathrm{Sigma}^1_G(q))(0) \in F_0 .$$

But now we can tell whether $t_0 \in F_0$ from the value of

$$x = [t_0/n+1](n+1) + (\mathrm{Sigma}^1_G([t_0/n+1]))(0)$$

and the truth or falsity of

$$x \in ¢_0,\ (x+1) \in ¢_0, \dots,\ (t_0-1) \in ¢_0 .$$

(If $t_0 = x$ then of course $t_0 \in F_0$ and if $t_0 < x$ then it should be clear that $t_0 \notin F_0$.)

Thus membership of F_0 is Δ_0-derivable from $\mathrm{graph}(\mathrm{Sigma}^1_G)$. Therefore $\mathrm{graph}(\mathrm{Sigma}^1_G) \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ implies $F_0 \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$.

We would now like to show that

$$\mathrm{graph}(G) \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}} \quad \text{implies} \quad \mathrm{graph}(\mathrm{Sigma}^1_G) \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}} .$$

To do this we consider the set Γ_n of all elements of ${}^{n+1}n{+}1$ that can ever be values of a function defined as in definition VIII.13. By our example earlier we know for instance that the function f defined by

$$f(0) = f(4) = f(5) = 0,\ f(1) = f(2) = 2,\ f(3) = 3$$

is an element of Γ_5.

If we can show, for all $n \in \mathbb{N}$, that closure under counting modulo $\mathbb{Z}_{n!}$ implies closure under counting modulo Γ_n then we will have proved that graph(G) $\in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ implies graph(Sigma^1_G) $\in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ as desired.

Now, for $n \geq 1$, Γ_n is always a strict subset of $^{n+1}n+1$. In particular the only one-one function in Γ_n is the identity function $\underline{id}$. Why is this? Well suppose, for some $0 \leq j \leq n$, that

$$((n+1)q + j, t_1, \dots, t_k, x_1, \dots, x_m) \in ¢_0 .$$

(i) If $j < n$ this means that (if the other registers hold $t_1, \dots, t_k, x_1, \dots, x_m$) t_0 would jump from $(n+1)q + j$ to $(n+1)q + j+1$, and then, since $j+1 \leq n$ we will have (suppressing $t_1, \dots, t_k, x_1, \dots, x_m$)

$$(G(q))(j) = (G(q))(j+1)$$

and so $G(q)$ is not one-one.

(ii) On the other hand if $j = n$ then t_0 jumps from $(n+1)q + n$ to $(n+1)(q+1)$. This forces $(n+1)q \in ¢_0$ for $(n+1)q \notin ¢_0$ implies that

$$(G(q))(0) = 0 = (G(q))(n).$$

But now since $0 < n$ having $(n+1)q \in ¢_0$ brings us back to case (i) and $G(q)$ can't have been one-one after all.

Now let $\Gamma^*_n = \{\gamma \in \Gamma_n : \gamma \text{ is not one-one}\} = \Gamma_n \setminus \{\underline{id}\}$.

Then we have a rather simpler version of lemmas IV.3 and VIII.9 which says that:

VIII.14 <u>Lemma</u>: Given $=,<$, explicit transformations, boolean operations and bounded quantification (I),

Closure under counting modulo Γ^*_n implies closure under counting modulo Γ_n.

<u>Proof</u>: Let $G : \mathbb{N}^m \to \Gamma_n$ and assume, for given i, that:

<u>Case one</u>: For all $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_m \in \mathbb{N}$

$$G(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_m) \neq \underline{id}.$$

Next observe that for all $\gamma \in \Gamma^*_n$ there is a $\bar{\gamma} \in \Gamma^*_n$ such that

$$\bar{\gamma}*\gamma = \gamma.$$

because $\gamma \in \Gamma^*_n$ implies there is a least $0 \leq j \leq n$ such that $j \notin \mathrm{Range}(\gamma)$. Now define $\bar{\gamma}$ to be the function $G(0)$ that would be produced by

$$0 \notin ¢,\ 1 \notin ¢,\ \ldots,\ j-1 \notin ¢,\ j \in ¢,\ j+1 \notin ¢,\ \ldots,\ n \notin ¢.$$

This gives a function that is the same as the identity function except that $\bar{\gamma}(j) \neq j$.

But now if $\gamma' = G(x_1,\ldots,x_{i-1},\mu,x_{i+1},\ldots,x_m)$ where μ is the greatest number less than x_i such that $G(x_1,\ldots,x_{i-1},\mu,x_{i+1},\ldots,x_m)$ is not $\underline{id}$ then we can define $G' : \mathbb{N}^m \to \Gamma^*_n$ by

$$G'(x_1,\ldots,x_m) = \begin{cases} G(x_1,\ldots,x_m) & \text{if } G(x_1,\ldots,x_m) \neq \underline{id} \\ \bar{\gamma}' & \text{otherwise} \end{cases}$$

and we will have

$$\mathrm{Sigma}^i_G = \mathrm{Sigma}^i_{G'}.$$

Clearly also graph(G') is derivable from graph(G) using explicit transformations, boolean operations and bounded quantification (I) and we have our result. Notice we don't need to assume any counting properties.

<u>Case two</u>: For some $x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_m \in \mathbb{N}$

$$G(x_1,\ldots,x_{i-1},0,x_{i+1},\ldots,x_m) = \underline{id}.$$

This case may be reduced to case one in the same way that we dealt with initial segments of elements of S_n in the proof of lemma IV.3.

End of proof of lemma VIII.14.

We also have:

VIII.15 <u>Lemma</u>: Given =, graph(suc), explicit transformations, boolean operations and bounded quantification (I), closure under counting modulo T_n implies closure under counting modulo Γ^*_n.

<u>Proof</u>: (Similar to the proofs of lemmas IV.2 and VIII.10.)

We define

$$\alpha : ({}^{n+1}n+1 \setminus S_n) \to {}^{n+1}n$$

$$\beta : ({}^{n+1}n+1 \setminus S_n) \to {}^{n}n+1$$

as in lemma IV.2, noting that $\Gamma^*_n \subseteq [{}^{n+1}n+1 \setminus S_n]$. Now, for all f in $[{}^{n+1}n+1 \setminus S_n]$, $a(f)$ and $s(f)$ are non-decreasing functions. If we can show that for all non-decreasing functions $g_1 : (n+1) \to n$ and $g_2 : n \to (n+1)$ and for all $\gamma \in \Gamma^*_n$,

$$g_1 \circ \gamma \circ g_2 \in T_n$$

then the proof of lemma IV.2 can be applied to prove the current lemma.

We prove this as follows:

Case one: γ is non-decreasing.

Then $g_1 \circ \gamma \circ g_2$ is non-decreasing and thus in T_n by definition VIII.8.

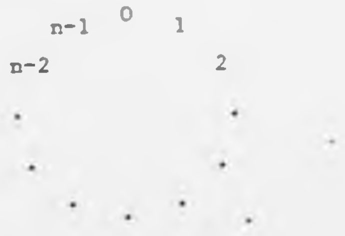
Case two: γ is not non-decreasing.

The only time this happens is when $((n+1)q + n) \in ¢$ giving rise to a function γ such that $\gamma(n) = 0$. In this case it will be seen that if μ is the least $0 \leq j \leq n$ such that, for all $j \leq \ell \leq n$ $((n+1)q + \ell) \in ¢$ then

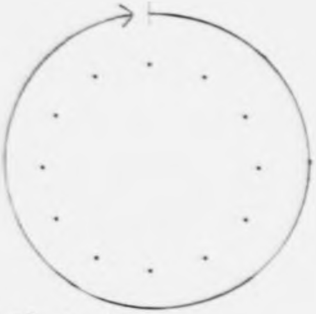
(i) $\gamma \restriction \{\ell : 0 \leq \ell < \mu\}$ is a non-decreasing function and

(ii) for $\mu \leq \ell \leq n$, $\gamma(\ell) = 0$.

Recall now the interpretation following definition VIII.8 and think what $g_1 \circ \gamma \circ g_2$ does to the circle

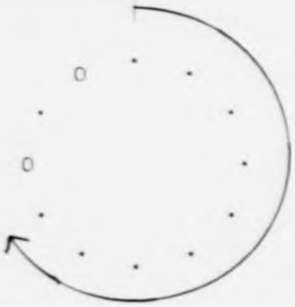


We see that when we apply g_2 (i.e. replace each value around the circle by g_2 applied to that value) it changes to:



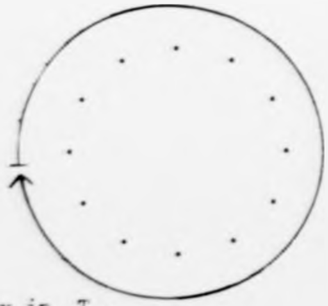
non-decreasing values from {0,...,n} along the arrow.

And thence on application of γ to



non-decreasing along the arrow, elsewhere zero.

And so ,applying F_{*}, we have



non-decreasing along the arrow.

This function is clearly in T_n.

End of proof of lemma VIII.15.

And now:

$$\begin{aligned} \Gamma_n\Delta_0^{\mathbb{N}} &\subseteq \Gamma^{*}_n\Delta_0^{\mathbb{N}} && \text{by lemma VIII.14} \\ &\subseteq T_n\Delta_0^{\mathbb{N}} && \text{by lemma VIII.15} \\ &\subseteq \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}} && \text{by lemma VIII.11.} \end{aligned}$$

Therefore graph(G) $\in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ implies graph(Sigma^1_G) $\in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$.

And thus $\phi_0 \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ implies $F_0 \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ as required.

Having demonstrated that $F_0 \in \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ we can now move on to define f_0.

Recalling that, for $t_1,\dots,t_k,x_1,\dots,x_m$ actually occurring in a computation, $t_0 < p(\max(x_1,\dots,x_m)+1)$, we define $f_0(t_1,\dots,t_k,x_1,\dots,x_m)$ to be

$$\mu t_0\ ([(t_0,\dots,t_k,x_1,\dots,x_m) \in F_0] \vee [t_0 = p(\max(x_1,\dots,x_m)+1)]).$$

f_0 is obviously a $\mathbb{Z}_{n!}\Delta_0$-function.

We can now proceed as usual, deleting

$$\text{"If } (\vec{t},\vec{x}) \in \Theta_0 \text{ then } \left(\left\{\begin{array}{l} \text{If } (\vec{t},\vec{x}) \in \Phi_0 \text{ then } t_0 := t_0 + 1 \\ \text{If } (\vec{t},\vec{x}) \notin \Phi_0 \text{ then } t_0 := t_0 + n+1 \end{array}\right\} ; L\right);\text{"}$$

and everywhere else substituting

$$f_0(t_1,\dots,t_k,x_1,\dots,x_m)$$

for t_0, producing, by facts VIII.3, new $\mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$ conditions

$$\Theta'_1,\dots,\Theta'_k,\Phi'_1,\dots,\Phi'_k.$$

Step 3. (Which is like proposition II.11.)

Finally we build on step 2. to prove, by induction on k, that for all machines with such programs and bounds the function computed is a $\mathbb{Z}_{n!}\Delta_0$-function.

End of proof of proposition VIII.12.

Now we have the main result of the chapter.

VIII.16 Theorem: For all $n \geq 1$

$$\{1,n+1\}\text{-Space}(\underline{1},\square) \text{ is exactly the class of } \mathbb{Z}_{n!}\Delta_0\text{-functions.}$$

Proof: Proposition VIII.12 and the obvious corollary of proposition VIII.4.

VIII.17 Corollary: For all $n \geq m \geq 1$

$$\{1,m+1\}\text{-Space}(\underline{1},\square) \subseteq \{1,n+1\}\text{-Space}(\underline{1},\square).$$

Proof: Clearly $\mathbb{Z}_{m!}\Delta_0^{\mathbb{N}} \subseteq \mathbb{Z}_{n!}\Delta_0^{\mathbb{N}}$.

Corollary VIII.18 does not obviously follow from our original definition of a Q-SRM.

VIII.19 Corollary:

$$\{1,2\}\text{-Space}(\underline{1},\square) = \{1\}\text{-Space}(\underline{1},\square) = \text{the } \Delta_0\text{-functions.}$$

Corollary VIII.19 is surprising when we consider that there is at present no reason to suppose that having $Q = \{1,3\}$ does not enlarge the class of functions computed (from when $Q = \{1\}$). But for $Q = \{1,2\}$ we just get back to the Δ_2-functions.

Chapter IX $\{1,2,n+1\}$-Space$(\underline{1},\square)$ and counting modulo S_n

To conclude our dealings with Q-SRM's we show that as well as characterizing classes for which a complexity characterization had not previously been provided, they can also be used for classes we have already dealt with. Namely the $S_n\Delta_0^{\mathbb{N}}$ classes.

Our main result (IX.4) follows in much the same way as, or rather builds on the methods of, the results of chapter VIII.

We make use of corollary V.2 that the class of $S_n\Delta_0$-functions is the same as the class of ${}^n n\Delta_0$-functions.

IX.1 <u>Lemma</u>: For $n \geq 2$

$$S_n\Delta_0^{\mathbb{N}} \subseteq \{1,2,n+1\}\text{-Space}_*(\underline{1},\square).$$

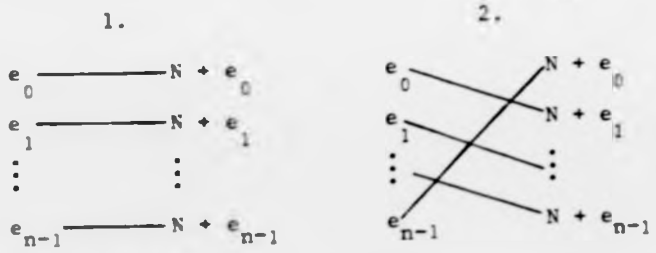
<u>Proof</u>: After all we have done before it is easy to see that

$$\{1,2,n+1\}\text{-Space}_*(\underline{1},\square)$$

contains $=$, $\leq$, graph$(+)$, graph$(\cdot)$ and is closed under explicit transformations, boolean operations and bounded quantification (II).

It remains to show that it is closed under counting modulo S_n. We have a somewhat circuitous route to follow before we achieve this.

Think back to the proof of lemma VIII.5 and our block of numbers D. We described two ways of moving through D, that is: two sets of paths through D, and these can be represented by the two diagrams:



As before define $D_i = \{x : (x - N\cdot i) \in D\}$. Think now of the value in stack t_0 of the machine in lemma VIII.5 and imagine the blocks

$$D_0, D_1, D_2, \ldots$$

stretching off to infinity.

As t_0 enters each new block the machine, in effect, decides, depending on $a(t_0)$'s being in A, whether we follow diagram 1. or diagram 2.

If $a(t_0) \notin A$ we follow diagram 1.

If $a(t_0) \in A$ we follow diagram 2.

(As before $a(t_0)$ is the number j such that $t_0 \in D_j$.)

The decision is effected by adding either 1 or $n+1$ to the value of t_0 when it happens that t_0 is congruent to an element of E. The rest of the time the amount to be added (i.e. $q(t_0^D)$) is determined purely by the value of t_0 mod N.

Clearly we can think of diagrams 1. and 2. as elements of ${}^{n}n$, that is: functions with domain and codomain $\underset{\sim}{n}$.

Diagram 1. is the identity function $\underline{id}$.

Diagram 2. is a particular rotation of an n-gon, call it $\underline{twist}$. So $\underline{twist}(0) = 1$, $\underline{twist}(1) = 2$, ..., $\underline{twist}(n-1) = 0$.

At the beginning of a calculation we start off at $t_0 = 0 = e_0$ and an alternative view is that as t_0 moves through D_0 it calculates $\underline{twist}(0)$ or $\underline{id}(0)$ according as $(x_1,\dots,x_{i-1},0,x_{i+1},\dots,x_m) \in A$ or not.

As t_0 moves through D_1 it again applies $\underline{twist}$ or $\underline{id}$ according as $(x_1,\dots,x_{i-1},1,x_{i+1},\dots,x_m) \in A$ or not.

And so on.

In fact if we define a function $F : \mathbb{N}^m \to {}^{n}n$ by

$$F(x_1,\dots,x_m) = \begin{cases} \underline{id} & \text{if } (x_1,\dots,x_m) \notin A \\ \underline{twist} & \text{if } (x_1,\dots,x_m) \in A, \end{cases}$$

then the effect of t_0 starting from $e_0 = 0$ and moving through

$$D_0,\dots,D_{(x_i-1)}$$

is (suppressing $x_1,\dots,x_{i-1},x_{i+1},\dots,x_m$) to calculate

$$\big(F(x_i-1)\big)\big(\big(F(x_i-2)\big)(\dots\big(F(1)\big)\big(\big(F(0)\big)(0)\big)\big)\big)$$

i.e. $$\big(\mathrm{Sigma}_F^i(x_1,\dots,x_m)\big)(0).$$

It should be clear that $\mathrm{Sigma}^{i}_{F}(x_1,\ldots,x_m) \in {}^{n}n$ must be a rotation and that therefore $\mathrm{Sigma}^{i}_{F}(x_1,\ldots,x_m)$ is determined by $\left(\mathrm{Sigma}^{i}_{F}(x_1,\ldots,x_m)\right)(0)$. In a sense then, the major task of the machine constructed in lemma VIII.5 is to calculate the "summation" of a particular function taking values in ${}^{n}n$.

Indeed we can start from any function $F : \mathbb{N}^m \to \{\underline{id},\underline{twist}\}$ such that $\mathrm{graph}(F) \in \{1,n+1\}\text{-Space}_{*}(\underline{1},\Box)$. By the properties of $\{1,n+1\}\text{-Space}_{*}(\underline{1},\Box)$ given at the beginning of lemma VIII.5 (properties that hold indeed for nearly any sensible complexity class) we have $A \in \{1,n+1\}\text{-Space}_{*}(\underline{1},\Box)$ where

$$(x_1,\ldots,x_m) \in A \quad \text{iff} \quad F(x_1,\ldots,x_m) = \underline{twist}.$$

The machine constructed in lemma VIII.5 around this particular set A can easily be modified so as to output $\mathrm{Sigma}^{i}_{F}(x_1,\ldots,x_m)$. (Recall that ${}^{n}n$ can be coded by a finite set of numbers.)

Now F is rather restricted above but we can extend our result. F need not take values only in $\{\underline{id},\underline{twist}\}$ but can be any function with domain $\mathbb{N}^m$ and codomain R_n (recall definition VIII.7: R_n is the group of rotations of an n-gon). This is hardly surprising since $R_n \cong \mathbb{Z}_n$ and closure under counting modulo $\mathbb{Z}_n$ is precisely what lemma VIII.5 proved, but let's see how the machines themselves can cope with this.

Observe that any element of R_n is equal to the composition of exactly n-1 elements of $\{\underline{id},\underline{twist}\}$. That is:

$$R_n = \{\underline{id}^{n-1}, \underline{id}^{n-2}\circ\underline{twist}, \ldots, \underline{id}\circ\underline{twist}^{n-2}, \underline{twist}^{n-1}\}.$$

Now define

$$\bar{D} =_{\text{def.}} D_0 \cup D_1 \cup \ldots \cup D_{n-2}.$$

and $$\bar{D}_y =_{\text{def.}} \{x : (x - y(n-1)N) \in \bar{D}\}.$$

And we have two Δ_0-functions a and b defined by

$$a(x) = y \quad \text{iff} \quad x \in D_y \quad (= 0 \text{ if no such } y \text{ exists})$$

$$b(x) = y \quad \text{iff} \quad x \in \bar{D}_y \quad (= 0 \text{ if no such } y \text{ exists}).$$

For $G : \mathbb{N}^r \to R_n$ we can calculate $\lceil \mathrm{Sigma}^i_G(x_1,\ldots,x_m)\rceil(0)$ if we can find a machine with stack t_0 whose values move through blocks

$\bar{D}_1, \bar{D}_2, \bar{D}_3, \ldots$

as follows:

For $x_1,\ldots,x_m \in \mathbb{N}$, $G(x_1,\ldots,x_m) = \underline{id}^j\ \underline{twist}^{n-1-j}$ for some $j \leq n-1$.

The correct path through $\bar{D}_{x_i}$ is then produced by $n-1-j$ <u>twist</u>'s in blocks $D_X, D_{X+1},\ldots,D_{X+n-2-j}$ (where $X = (n-1)x_i$), followed by j <u>id</u>'s in blocks $D_{X+n-1-j},\ldots,D_{X+n-2}$.

Where there is a difference between the value that has to be added to t_0 in order to produce an <u>id</u> and that which produces a <u>twist</u>, the decision as to which to add rests in the first place on the value of $b(t_0)$, or rather of $G(x_1,\ldots,x_{i-1},b(t_0),x_{i+1},\ldots,x_m)$. This will tell us the general strategy for $\bar{D}_{b(t_0)}$, i.e. how many <u>id</u>'s and how many <u>twist</u>'s. When we know what $\bar{D}_{b(t_0)}$ should be, $a(t_0)$, or rather $a(t_0) \bmod n-1$ will tell us whether we should currently be following an <u>id</u> path or a <u>twist</u> path. In other cases the choice of the amount to be added to t_0 depends purely on $t_0 \bmod N$.

If $\mathrm{graph}(G) \in \{1,n+1\}\text{-Space}_*(\underline{1},\square)$ this decision - which is a choice between adding 1 to the value of t_0 or adding $n+1$ - can be expressed as a type $(\bar{i})'$ instruction of an $\mathrm{SRM}_{\{1,n+1\}\text{-Space}(\underline{1},\square)}$. In this way we can define a machine with a register t_0 which, as it increases, (almost) codes the pair

$$<b(t_0), \lceil \mathrm{Sigma}^i_G(x_1,\ldots,x_{i-1},b(t_0),x_{i+1},\ldots,x_m)\rceil(0)>.$$

From here an $\mathrm{SRM}_{\{1,n+1\}\text{-Space}(\underline{1},\square)}$ which calculates the characteristic function of $\mathrm{graph}(\mathrm{Sigma}^i_G)$ is easily derivable.

We have just shown that $\{1,n+1\}\text{-Space}_*(\underline{1},\square)$ is closed under counting modulo R_n, i.e. modulo $\mathbb{Z}_n$. So we have an alternative proof of lemma VIII.5.

We can now go on to generalise this method, but first there is just one point to make.

In the case above it happens that $\mathrm{Sigma}^{i}_{G}(x_1,\dots,x_m)$ is immediately determined by $(\mathrm{Sigma}^{i}_{G}(x_1,\dots,x_m))(0)$. Even had this not been the case it should however be clear that if the method works to calculate $(\mathrm{Sigma}^{i}_{G}(x_1,\dots,x_m))(0)$ for a given function $G : \mathbb{N}^m \to {}^n n$, then it will also work to calculate

$$(\mathrm{Sigma}^{i}_{G}(x_1,\dots,x_m))(j) \qquad \text{for all} \quad 0 \leq j \leq n-1.$$

Rather than starting from $t_0 = e_0 = 0$ we can arrange our $(\bar{i})'$ instruction so as to set $t_0 = e_j$ before it does anything else (this is easy to do) and the resultant machine will calculate $(\mathrm{Sigma}^{i}_{G}(x_1,\dots,x_m))(j)$.

If we have machines calculating $(\mathrm{Sigma}^{i}_{G}(x_1,\dots,x_m))(j)$ for all $0 \leq j \leq n-1$ then we will usually be able to construct one calculating the characteristic function of $\mathrm{graph}(\mathrm{Sigma}^{i}_{G})$.

Having made that point, I will now say that my aim is to convince the reader that once those sets of paths, those elements of ${}^n n$, $\underline{id}$ and $\underline{twist}$ had been found, the problem was as good as solved. $\{1,n+1\}\text{-Space}_{*}(\underline{1},\square)$ was closed under counting modulo R_n just because R_n is the submonoid of ${}^n n$ consisting of products (compositions) of a fixed length (n-1) of these two functions. (This is the same as saying that R_n is the submonoid generated by $\underline{twist}$.) This closure did not depend on any further properties of R_n, such as the fact that for $G \in R_n$, $G(0)$ determines G.

Indeed so confident am I of all this being obvious to the reader that I will now state a theorem without any further proof.

We generalise the idea of N, D, E and q a little first.

Let Q be a finite set of numbers.

Let D be a set of numbers distinct modulo N, some $N > n$, and let $E = \{e_0,\dots,e_{n-1}\} \subseteq D$.

Let $\bar{q} : D \to Q$ be such that, for all $0 \leq j \leq n-1$, if we start at e_j

and jump according to $\bar{q}$ we will first move outside D in jumping to $N + e_\ell$ for some $0 \leq \ell \leq n-1$. Such a $\bar{q}$ corresponds to an element of $^n n$. $\bar{q}$ corresponds to $f \in {}^n n$ if the path from e_j leads to $N + e_{f(j)}$

IX.2 *Theorem*: For $Q \ni 1$ and N, D, E as described:

If $\bar{q}_1, \ldots, \bar{q}_s : D \to Q$ correspond to
$f_1, \ldots, f_s \in {}^n n$ and $\underline{id} \in \{f_1, \ldots, f_s\}$

then $Q\text{-Space}_*(\underline{1}, \square)$ is closed under counting modulo $\langle f_1, \ldots, f_s \rangle$ (the submomoid of $^n n$ generated by $f_1, \ldots, f_s$).

Example:

Returning to $Q = \{1, n+1\}$ and D as in VIII.5, this means that if we could find a third appropriate set of paths through D then we could perhaps extend the closure properties of $\{1, n+1\}\text{-Space}_*(\underline{1}, \square)$.

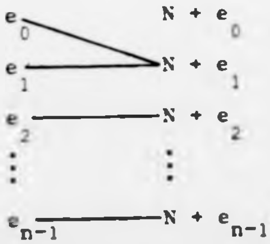
For instance, there is the following rule - call it *merge*. *merge* is defined by:

For $d \in D \setminus E$, *merge* follows the path given by our original q.

For $e_1, \ldots, e_{n-1}$, *merge* decrees a jump of size 1 (the same as $\underline{id}$).

For e_0 *merge* decrees a jump of size $n+1$ (like *twist*).

The *merge* diagram is



Now the submonoid of $^n n$ generated by $\underline{id}$, *twist* and *merge* is in fact T_n (definition VIII.8). Thus $\{1, n+1\}\text{-Space}_*(\underline{1}, \square)$ is closed under counting modulo T_n.

Also, because under the usual conditions counting modulo T_n implies counting modulo T_{n-1} (simulate *twist* on $n-1$ elements by *merge*•*twist* on n, *merge* by *twist*•*merge*•*twist*$^{-1}$ and $\underline{id}$ by *merge*), it is not hard to show, by induction on n, that under the usual conditions closure under counting modulo T_n implies closure under counting modulo $\mathbb{Z}_n$.

This proves lemma VIII.6 for the second time.

Nothing new yet then; but wait a bit. If we allow ourselves jumps of size 2 as well as 1 and $n+1$ we can derive a fourth function *switch*.

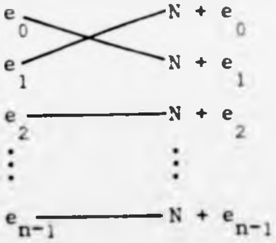
For $d \in D \setminus E$, *switch* follows the path given by q.

For $e_2, \ldots, e_{n-1}$, *switch* follows the path of *id* or *merge*.

For e_0, *switch* decrees a jump of size $n+1$.

For e_1, *switch* decrees a jump of size 2.

This means that *switch* takes e_0 to $e_0 + n+1 = e_1 + 1$ according to our definition of E and from here q proceeds to $N + e_1$. Meanwhile e_1 goes to $e_1 + 2 = e_0 + n+2 = e_0+1 + n+1 = e_0+1 + q(e_0+1)$, so q will now bring us to $N + e_0$. The diagram for *switch* is:



It is not hard to see that *id*, *twist* and *switch* together generate S_n. For, using the usual notation for permutations, we can write *twist* as $(12\ldots n)$ and *switch* as (12). *id* of course is the identity. And now it can be shown that (12) and $(12\ldots n)$ together generate the whole of S_n. It is well-known that S_n is generated by transpositions and that therefore $(12), (13), \ldots, (1n)$ generate S_n since, for general i,j,

$$(ij) = (1i)(1j)(1i).$$

But for all $1 \leq i \leq n-1$

$$(i\ (i+1)) = (12\ldots n)^{i-1}(12)(12\ldots n)^{n-i+1}$$

and so for $2 \leq j \leq n$

$$(1j) = (12)(23)\ldots((j-1)\ j)\ldots(23)(12)$$

is expressible as a product of (12) and (12...n) and these two therefore generate all of S_n.

This means by theorem IX.2 that $\{1,2,n+1\}$-Space$_*(\underline{1},\square)$ is closed under counting modulo S_n.

Observe too that id, twist, merge and switch together generate all of ${}^n n$.

End of proof of lemma IX.1.

Our next result is that:

IX.3 Lemma: For all $n \geq 2$,

all elements of $\{1,2,n+1\}$-Space$(\underline{1},\square)$ are ${}^n n\Delta_0$-functions.

Proof: (This is much the same as the proof of proposition VIII.12.)

Again at step 1. we show we can calculate our given function using a machine whose program is really just a single line which is repeated over and over again.

At step 3. we prove that all such machines calculate ${}^n n\Delta_0$-functions.

Step 3. uses step 2. which depends on the fact that for a function $G : \mathbb{N}^{m+k+1} \to {}^{n+1}n+1$ defined in a particular way, graph$(\mathrm{Sigma}^i_G) \in {}^n n\Delta_0^{\mathbb{N}}$.

In the current case G will be defined by:

For $q,t_1,\ldots,t_k,x_1,\ldots,x_m \in \mathbb{N}$, $0 \leq j \leq k$,

$\big(G(q,t_1,\ldots,t_k,x_1,\ldots,x_m)\big)(j)$ is the element of $\{0,1,\ldots,n-1\}$ which is congruent modulo $n+1$ to the least number $\geq (n+1)(q+1)$ reached by starting from $t_0 = (n+1)q + j$ and repeating

$$\left\{\begin{array}{l}\text{If } \Phi^1_0(t_0,\ldots,t_k,x_1,\ldots,x_m) \text{ then } t_0 := t_0 + 1;\\ \text{If } \Phi^2_0(t_0,\ldots,t_k,x_1,\ldots,x_m) \text{ then } t_0 := t_0 + 2;\\ \text{If } \Phi^3_0(t_0,\ldots,t_k,x_1,\ldots,x_m) \text{ then } t_0 := t_0 + n+1;\end{array}\right.$$

where Φ^1_0, Φ^2_0, Φ^3_0 are open LA-formulae defining subsets of $\mathbb{N}^{m+k+1}$ which are disjoint and exhaustive.

Call the set of values (the values of G are elements of ${}^{n+1}n+1$) definable in this way L_n.

If we can show that for all $G : \mathbb{N}^{m+k+1} \to L_n$, $\mathrm{graph}(G) \in {}^n n\Delta_0^{\mathbb{N}}$ implies

$$\mathrm{graph}(\mathrm{Sigma}_G^1) \in {}^n n\Delta_0^{\mathbb{N}}$$

then the proof of proposition VII.12 carries over more or less directly to the present case.

It suffices therefore to show that, under the usual conditions, closure under counting modulo ${}^n n$ implies closure under counting modulo L_n.

Observe then that (suppressing the variables $t_1,\dots,t_k,x_1,\dots,x_m$) the only one-one function $G(q)$ derivable in the way given, i.e. the only one-one element of L_n, is the identity function. This arises when for all $0 \leq j \leq k$,

$$\phi_0^3((n+1)q + j,t_1,\dots,t_k,x_1,\dots,x_m),$$

in which case t_0 jumps directly from $(n+1)q + j$ to $(n+1)(q+1) + j$. In no other circumstances is $G(q)$ one-one, for suppose:

<u>Case one</u>: For some $0 \leq j \leq n$ $\phi_0^1((n+1)q + j)$.

(i) If $j \leq n-1$, t_0 would jump from $(n+1)q + j$ to $(n+1)q + j+1$ and since $j+1 \leq n$ this means that $\big(G(q)\big)(j) = \big(G(q)\big)(j+1)$.

(ii) If $j = n$, we have $\big(G(q)\big)(n) = 0$. Therefore to keep $G(q)$ one-one we need $\neg\phi_0^3((n+1)q + 0)$. In this case $\big(G(q)\big)(0) = \big(G(q)\big)(1)$ or $\big(G(q)\big)(2)$ (recall $n \geq 2$ so $n+1 \geq 3$), for one of ϕ_0^1 and ϕ_0^2 must hold, and so $G(q)$ is not one-one after all.

<u>Case two</u>: For some $0 \leq j \leq n$ $\phi_0^2((n+1)q + j)$.

(i) If $j \leq n-2$ then $\big(G(q)\big)(j) = \big(G(q)\big)(j+2)$.

(ii) If $j = n-1$ then $\big(G(q)\big)(n-1) = 0$ and so we require $\neg\phi_0^3((n+1)q + 0)$, which implies as in case one (ii) that $G(q)$ is not one-one.

(iii) If $j = n$ then $\big(G(q)\big)(n) = 1$. So we require $\neg\phi_0^3((n+1)q + 1)$. But $\phi_0^1((n+1)q + 1)$ implies that $\big(G(q)\big)(1) = \big(G(q)\big)(2)$. And $\phi_0^2((n+1)q + 1)$ implies that $\big(G(q)\big)(1) = \big(G(q)\big)(3)$ if $n > 2$ and if

$n = 2$, it brings us back to the case two (ii), which has already been dealt with.

So, then, the only one-one function in L_n is $\underline{id}$.

The proof of lemma VIII.14 can now be pretty directly applied to show that closure under counting modulo $(L_n \setminus \{\underline{id}\})$ implies closure under counting modulo L_n.

The proof of lemma IV.3 can be applied to show that closure under counting modulo ${}^n n$ implies closure under counting modulo $(L_n \setminus \{\underline{id}\})$.

And so we prove the soundness of step 2. and conclude the proof of lemma IX.3.

IX.4 <u>Theorem</u>: For all $n \geq 2$,

$\{1,2,n+1\}$-Space$(\underline{1},\square)$ is exactly the class of $S_n\Delta_0$-functions.

<u>Proof</u>:

Lemma IX.3, theorem IV.1 and the obvious corollary to lemma IX.1.

IX.5 <u>Corollary</u>: For all $n \geq 2$,

$\{1,2,n+1\}$-Space$(\underline{1},\square)$ = $\{1\}$-Space$(\underline{n},\square)$.

<u>Proof</u>:

Theorem IX.4 and theorem V.4.

IX.6 <u>Corollary</u>: For all $n \geq 2$,

$S_n\Delta_0^{\mathbb{N}}$ = $\{1,2,n+1\}$-Space$_\star(\underline{1},\square)$.

<u>Proof</u>:

Trivial.

IX.7 <u>Theorem</u>: For all $n \geq 2$,

$\{1,2,3,\ldots,n,n+1\}$-Space$(\underline{1},\square)$ is exactly the $S_n\Delta_0$-functions.

<u>Proof</u>:

Clearly $\{1,2,n+1\}$-Space$(\underline{1},\square)$ $\subseteq$ $\{1,2,\ldots,n+1\}$-Space$(\underline{1},\square)$ and this thus contains the class of $S_n\Delta_0$-functions by theorem IX.4.

The proof that all elements of $\{1,\ldots,n+1\}$-Space$(\underline{1},\square)$ are $S_n\Delta_0$-functions (i.e. ${}^n n\Delta_0$-functions) is much the same as for lemma IX.3. It hinges on the fact that if $G(q,t_1,\dots,t_k,x_1,\dots,x_m)$ is defined from

$$\left\{\begin{array}{l}
\text{If } \phi^1_0(t_0,\dots,t_k,x_1,\dots,x_m) \text{ then } t_0 := t_0 + 1; \\
\text{If } \phi^2_0(t_0,\dots,t_k,x_1,\dots,x_m) \text{ then } t_0 := t_0 + 2; \\
\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots \\
\text{If } \phi^{n+1}_0(t_0,\dots,t_k,x_1,\dots,x_m) \text{ then } t_0 := t_0 + n+1;
\end{array}\right.$$

as we have defined G functions in the past, then once again the only possible one-one value of G is in fact $\underline{id}$. This is not too hard to show.

End of proof of theorem IX.7.

IX.8 Theorem: For all $n \geq 3$, for all $1 < j < n$,

$$\{1,j,n+1\}\text{-Space}(\underline{1},\Box) = \{1,2,n+1\}\text{-Space}(\underline{1},\Box)$$

(= the $S_n\Delta_0$-functions).

Proof:

Clearly $\{1,j,n+1\}\text{-Space}(\underline{1},\Box) \subseteq \{1,2,n+1\}\text{-Space}(\underline{1},\Box)$ by theorems IX.4 and IX.7.

For inclusion the other way we use theorem IX.2.

Let N, D, E and q be as in the proof of lemma VIII.5.

Define $\underline{rot}_j$ by:

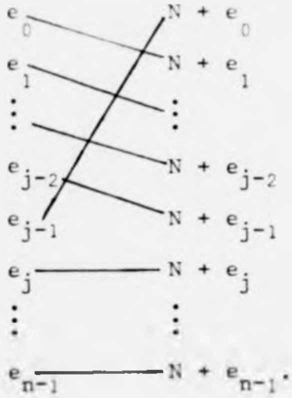
For $d \in D \setminus \{e_0,\dots,e_{j-1}\}$, $\underline{rot}_j$ has the same jumps as $\underline{id}$.

For $e_0,\dots,e_{j-2}$, $\underline{rot}_j$ decrees a jump of $n+1$ (like $\underline{twist}$).

For e_{j-1}, $\underline{rot}_j$ decrees a jump of size 1.

Now
$$\begin{aligned}
e_{j-1} + j &= n(j-1) + j \\
&= (n+1)(j-1) + 1 \\
&= r_j(0)
\end{aligned}$$

(recalling our definition of D as $r(0) \cup \dots \cup r(2n-1)$) and from $r_j(0)$, $\underline{rot}_j$ (or $\underline{id}$) will lead to $N + e_0$. Therefore the diagram for $\underline{rot}_j$ will be:



So in S_n terms $\underline{rot}_j$ is the permutation $(12...j)$.

Showing closure under counting modulo S_n requires some elementary facts about S_n or rather about A_n the alternating subgroup.

It is well-known (see e.g. [Cohn], vol. 1, section 3.5, theorem 4) that for all $n \geq 3$, A_n is generated by the permutations

$$(123), (124), \ldots, (12n).$$

From this the following three lemmas (also presumably well-known results) are easily proved.

IX.9 Lemma: For all $n \geq 3$, A_n is a subgroup of the group generated by

$$(123) \text{ and } (12...n).$$

Proof: Let $I = \langle(123),(12...n)\rangle$. Then for all $n \geq 3$, for all $4 \leq j \leq n$,

$$(12j) \in I.$$

For given n, this is shown by induction on j.

If $\underline{j = 4}$ (and so $n \geq 4$) then

$$(234) = (12...n)(123)(12...n)^{-1} \in I$$

and so $(124) = (234)^{-1}(123)^{-1}(234) \in I$.

If $\underline{(12i) \in I \text{ for all } 3 \leq i < j > 4}$ then by the theorem in [Cohn], $A_{j-1} \subseteq I$, and so, as $1, 2, j-1, j-2$ are distinct, $(1\ (j-2))(2\ (j-1))$ is in I because it is an even permutation. Also

$$((j-2)\ (j-1)\ j) = (12...n)^{j-1}(123)(12...n)^{n-j+1} \in I$$

and so

$$(2j) = (1\ (j-2))(2\ (j-1))((j-2)\ (j-1)\ j)(1\ (j-2))(2\ (j-1)) \in I$$

(because $j > 4$ implies $1, 2, j-2, j-1, j$ are distinct numbers).

IX.10 Lemma: For all $n \geq 5$, $n-2 > j \geq 3$,

$$(12j) \in \langle (12\ (j+2)), (12\ldots n) \rangle .$$

Proof: Let $J = \langle (12\ (j+2)), (12\ldots n) \rangle$.

Then $(23\ (j+3)) = (12\ldots n)(12\ (j+2))(12\ldots n)^{-1} \in J$ since $j+3 \leq n$.

And thus $(3\ (j+3)\ (j+2)) = (12\ (j+2))(23\ (j+3))(12\ (j+2))^{-1} \in J$

since $j+2 > 3$.

And so $(2\ (j+2)\ (j+1)) = (12\ldots n)^{-1}(3\ (j+3)\ (j+2))(12\ldots n) \in J$

and $(1\ (j+1)\ j) = (12\ldots n)^{-1}(2\ (j+2)\ (j+1))(12\ldots n) \in J$.

Giving $(12j) = (2\ (j+2)\ (j+1))(1\ (j+1)\ j)(2\ (j+2)\ (j+1))^{-1} \in J$.

IX.11 Lemma: For all $n \geq 4$, for all $3 \leq j < n$, the group

$$K = \langle (12\ldots j), (12\ldots n) \rangle$$

contains A_n.

Proof: If $j = n-1$ then

$$(12\ldots(n-1))(12\ldots n)^{-1} = (1n) \in K$$

and so $(12) = (12\ldots n)(1n)(12\ldots n)^{-1} \in K$.

Thus $K = S_n$ by our remarks in the proof of lemma IX.1.

If $j < n-1$

then $(23\ldots(j+1)) = (12\ldots n)(12\ldots j)(12\ldots n)^{-1} \in K$.

But then $(12\ (j+1)) = (12\ldots j)(23\ldots(j+1))^{-1} \in K$

and $(1\ (j+1)\ j) = (12\ldots(j+1))^{-1}(12\ldots j) \in K$

giving $(12j) = (1\ (j+1)\ j)(12\ (j+1)) \in K$.

If j is even then $(12\ (j+1)) \in K$ implies, by lemma IX.10, that $(123) \in K$.

If j is odd then $(12j) \in K$ implies that $(123) \in K$.

Thus, by lemma IX.9, $A_n \subseteq K$.

End of proof of lemma IX.11.

To return to the main proof (that of theorem IX.8):

We know we have $\underline{rot}_j = (12...j)$, and since $\{1,n+1\} \subseteq \{1,j,n+1\}$, we also have $\underline{id}$ and $\underline{twist} = (12...n)$.

Therefore since $j < n$, by theorem IX.2 and lemma IX.11,

$$\{1,j,n+1\}\text{-Space}_*(\underline{1},\square) \text{ is closed under counting modulo } A_n.$$

Clearly we also have = and closure under explicit transformations and boolean operations.

And because $\{1,n+1\} \subseteq \{1,j,n+1\}$ we have closure under counting modulo $\mathbb{Z}_{n!}$ which implies closure under counting modulo $\mathbb{Z}_2$.

But now

$$A_n \triangleleft S_n \quad \text{and} \quad S_n/A_n \cong \mathbb{Z}_2$$

and so, by theorem III.10 (i), we must have closure under counting modulo S_n.

And now it must be clear that

$$S_n\Delta_0^{\mathbb{N}} \subseteq \{1,j,n+1\}\text{-Space}_*(\underline{1},\square)$$

and that therefore all $S_n\Delta_0$-functions are in $\{1,j,n+1\}\text{-Space}(\underline{1},\square)$.

End of proof of theorem IX.8.

IX.12 <u>Theorem</u>: For all $n \geq 2$,

$$\{1,n,n+1\}\text{-Space}(\underline{1},\square) \text{ is exactly the class of } \mathbb{Z}_{n!}\Delta_0\text{-functions.}$$

<u>Proof</u>:

Since $\{1,n+1\} \subseteq \{1,n,n+1\}$ all $\mathbb{Z}_{n!}\Delta_0$-functions are in $\{1,n,n+1\}\text{-Space}(\underline{1},\square)$.

Showing inclusion the other way is much the same as the proof of lemma IX.3 or of proposition VIII.12.

The crucial step is step 2., where we consider the nature of a certain submonoid of $^{n+1}n+1$, whose elements are in turn values of certain other functions (the G functions).

If for function $q : (\underline{n+1}) \to \{1,n,n+1\}$ we define $H(q) : (\underline{n+1}) \to (\underline{n+1})$ by:

For $0 \leq j \leq n$,

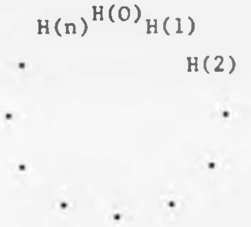
$\{H(q)\}(j)$ is the element of $(\underset{\sim}{n}+1) = \{0,1,\dots,n\}$ congruent modulo $n+1$ to the first number $> n$ reached by starting from j and making a jump of size $q(x)$ if we meet x.

The set of functions we are currently interested in - call it M_n - can be defined by

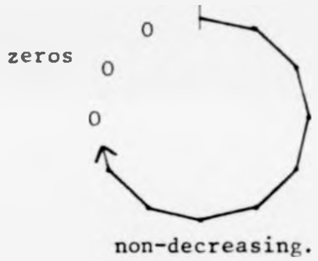
$$M_n = \{H(q) : q \in {}^{(\underset{\sim}{n}+1)}\{1,n,n+1\}\}.$$

As we remarked more generally in the proof of theorem IX.7, the only one-one function in M_n is $\underline{id}$.

It also turns out (and isn't hard to see) that, arranging the values of $H \in M_n$ around a circle as we did before,



will generally look like:



This means that if $g_1 : (\underset{\sim}{n}+1) \to \underset{\sim}{n}$ and $g_2 : \underset{\sim}{n} \to (\underset{\sim}{n}+1)$ are non-decreasing then

$$g_1 \bullet H \bullet g_2 \in T_n.$$

We can now use the arguments of proposition VIII.12 to show that all elements of $\{1,n,n+1\}$-Space$(\underline{1},\square)$ are $\mathbb{Z}_{n!}, \Delta_0$-functions.

End of proof of theorem IX.12.

Given theorem IX.8, theorem IX.12 seems somewhat anomalous. If, as may be the case, $\mathbb{Z}_{n!}\Delta_0^{\mathbb{N}} \neq S_n\Delta_0^{\mathbb{N}}$ for $n \geq 5$ then, for $n \geq 5$, $\{1,j,n+1\}$-Space$(\underline{1},\square)$ is the same for $2 \leq j \leq n-1$, but smaller for $j = n$. It is difficult to pick out anything that makes $\{1,n,n+1\}$ different, although if we add 0 to Q (recall this makes no difference to the functions in Q-Space(u,v)) then $\{0,1,n,n+1\}$ possesses a symmetry lacking in $\{0,1,j,n+1\}$ for $2 \leq j \leq n-1$.

Notice also that the proof of IX.2 most definitely does not carry over for other values of j. If, for instance, we take $n = 5$, $j = 2$ we can define $q : \underset{\sim}{6} \rightarrow \{1,2,6\}$ by $q(0) = 6$, $q(1) = 2$, $q(2) = 6$, $q(3) = 6$, $q(4) = 6$ and $q(5) = 6$.

If, now, $H : \underset{\sim}{6} \rightarrow \underset{\sim}{6}$ is derived from q as $H(q)$ was derived from q in IX.12 then

$$H(0) = 0,\ H(1) = 3,\ H(2) = 2,\ H(3) = 3,\ H(4) = 4,\ H(5) = 5$$

and the non-zero part of the values is not non-decreasing.

We conclude this chapter with the observation that we have now characterised, using our $X\Delta_0^{\mathbb{N}}$ classes, Q-Space$(\underline{1},\square)$ for all sets Q such that $1 \in Q$. For any such sets Q not directly referred to above can easily be shown to produce complexity classes equal to those produced by Q's which have been dealt with.

REFERENCES

[Bel'tyukov] "A machine description and the hierarchy of initial
Grzegorczyk classes" by A. P. Bel'tyukov
Journal of Soviet Mathematics vol. 20 (1982) pp. 2280-9.

[Cohn] "Algebra" (vol. 1) by P. M. Cohn
John Wiley and Sons 1974.

[Paris-Wilkie] "Counting problems in bounded arithmetic"
by Jeffrey Paris and A. Wilkie
Methods in Mathematical Logic (Proceedings of the 6th
Latin American syposium on mathematical logic, held in
Caracas, Venezuela 1983)
Lecture Notes in Mathematics 1130
Springer-Verlag 1985.

Works consulted but not directly referred to in the text.

"Introduction to automata theory, languages, and computation"
by J. E. Hopcroft and J. D. Ullman
Addison-Wesley 1979.

"Theory of formal systems" by Raymond M. Smullyan
(Annals of Mathematics Studies number 47)
Princeton University Press 1961.