Erdös-Rado without choice

October 27, 2004

ABSTRACT

A version of the Erdös-Rado theorem on partitions of the unordered n-tuples from uncountable sets is proved, without using the axiom of choice. The case $n = 1$ is just the Sierpinski-Hartogs' result that $\aleph(\alpha) \leq 2^{2^{2^\alpha}}$.

Arithmetic notations

An aleph is a cardinal of a wellordered set. $\aleph_\alpha$ is the αth aleph, and in this usage α is of course an *ordinal*. $\aleph(\alpha)$ is the least aleph $\not\leq \alpha$, and in this usage α is a *cardinal* not an ordinal, and the Hebrew letter is being used to denote Hartogs' aleph function. When α is itself an aleph we often write 'α^+' for '$\aleph(\alpha)$'. By abuse of notation we will often use a notation denoting an aleph, such as '$\aleph(\alpha)$' or '$\aleph_\kappa$' to denote also the corresponding initial ordinal. Finally $\beth_0(\alpha) =: \alpha$; $\beth_{n+1}(\alpha) =: 2^{\beth_n(\alpha)}$.

Combinatorial notations

$[X]^n$ is the set of unordered n-tuples from X. "$\alpha \rightarrow (\beta)^\gamma_\delta$" means: take a set A of size α, partition the unordered γ-tuples of it into δ bits. Then there is a subset $B \subseteq A$ of size β such that all the unordered γ-tuples from it are in the same piece of the partition. Here γ will always be in $\mathbb{N}$, and α, β and δ will be infinite cardinals.

Points of a tree are **nodes**, and they have **children**; the set of children of a node is a **litter**.

The result

The proof will follow closely a standard procedure for proving Ramsey's theorem that $(\forall nm)(\omega \rightarrow (\omega)^n_m)$. We start by considering the case $n = 2$ and proceed to larger n by induction. We start by proving without AC the special "binary" case, first shown in [1].

REMARK 1. $\aleph(2^{2^{2^\kappa}}) \rightarrow (\aleph(\kappa))^2_2$.

Proof:

Let $\langle K, <_K \rangle$ be a wellordering of length $\aleph(2^{2^{2^\kappa}})$, and Δ a two-colouring of $[K]^2$ making every unordered pair red or blue. We will find a set monochromatic for Δ.

The idea is to remove ordered pairs from $<$ to obtain a wellfounded tree with **The Nice Path Property**: for all a, if b and c lie on the same branch as a and beyond a, then $\{a, b\}$ and $\{a, c\}$ are the same colour. We will delete ordered pairs by an iterative procedure which I now spell out.

We consider the elements of K in $<_K$ order. At each stage we have in hand an element a and a wellfounded strict partial ordering $<_a$—which is initialised to $<_K$. We use a to discard some ordered pairs.

We reach stage a equipped with the strict partial order $\bigcap_{b<a} <_b$, which we abbreviate to $<<_a$. We will weed out some ordered pairs from $<<_a$ to obtain $<_a$. We consider the members of $\{x : x >>_a a\}$. To ensure that the new, stricter, order we end up with has the nice path property we must ensure that whenever b and c in $\{x : x >>_a a\}$ are joined to a by edges of different ccolours then the new order believes they are incomparable. Accordingly if $b <<_a c$ we delete the ordered pair $\langle b, c \rangle$ from $<<_a$. (And if $c <<_a b$ we delete the ordered pair $\langle c, b \rangle$ from $<<_a$)

This teases apart the points $>>_a a$ into two *rays*, consisting of those elements joined to a by a red edge and those joined to a by a blue edge; points in different rays will have no common upper bound according to $<_a$.

After κ steps we have performed this for every $a \in K$ and the set of those ordered pairs that remain—let us call it $<'$—is a wellfounded partial order with the nice path property. We had better check this.

We need to check that $<'$ is a wellfounded tree with the nice path property. It is wellfounded because any subset of a graph of a wellfounded relation is itself a graph of a wellfounded relation. The nice path property is obvious by construction.

The two clauses that need checking are transitivity and the property that incomparable elements have no common upper bound.

(i) transitivity. It will fail to be transitive if there is $b <' c$ and $c <' d$ but $b \not<' d$. In this case we know $c <_K d$. But since $b <' c$, we know that for any $a <_K b$, $\{a, b\}$ and $\{a, c\}$ are the same colour. Similarly since $c <' d$, we know that for any $a <_K c$, $\{a, c\}$ and $\{a, d\}$ are the same colour. But then, for any $a <_K b$, $\{a, b\}$ and $\{a, d\}$ are the same colour.

(ii) the incomparable element condition. If we "separate" b and c as above then we cannot keep d above both b and c because $\{a, d\}$ must be the same colour as both $\{a, c\}$ and $\{a, b\}$ which is impossible.

So $\langle K, <' \rangle$ is a tree, and a *binary* tree at that, because there are only two colours.

We want this tree to have a branch in it of length at least (size) $\aleph(\kappa)$. If all branches die within $\aleph(\kappa)$ steps then there are at most $2^{\aleph(\kappa)}$ points in K, so $\aleph(2^{2^{2^\kappa}}) \leq 2^{\aleph(\kappa)}$. This relies on a perfect binary tree of height α having precisely 2^α points. This is true as long as there is a choice function on the set of litters, and this is a nontrivial assumption. Fortunately this is true here: the litters are uniformly ordered because one child in the litter is joined to the parent by a blue edge and one by a red edge.

So if all branches die within $\aleph(\kappa)$ steps then $\aleph(\kappa) \leq^* 2^{2^\kappa}$ whence we would have $\aleph(2^{2^{2^\kappa}}) \leq 2^{\aleph(\kappa)} \leq 2^{2^{2^\kappa}}$, contradicting the definition of $\aleph(2^{2^{2^\kappa}})$.

So there is a branch of length $\aleph(\kappa)$. Every element in this branch can be thought of as a *red* point (if it is joined to all later points in that branch by a red edge) or as a *blue* point (if it is joined to all later points in that branch by a blue edge). So there are either $\aleph(\kappa)$ red points or $\aleph(\kappa)$ blue points, so one way or another we get a monochromatic set of size $\aleph(\kappa)$.

We proved slightly more than we will need or exploit. What proved was that the tree has a branch of length *greater* than $\aleph(\kappa)$ whereas all we use is that it has a branch of length *at least* $\aleph(\kappa)$.

Increasing the exponent

Next we tackle the "higher exponent" version, as seen in [2] (item (95) p 471). Let $\langle K, <_K \rangle$ be as before, and Δ a 2-colouring of $[K]^{n+1}$. We want to discard ordered pairs to be left with a tree ordering $<$ such that whenever $a_1 < a_2 < \ldots < a_n < b_1 < b_2$ then $\{a_1 \ldots a_n, b_1\}$ and $\{a_1 \ldots a_n, b_2\}$ are the same colour.

So, given an n-tuple $a_1 <_K a_2 <_K \ldots <_K a_n$ and $b_1 <_K b_2$ beyond a_n we can "separate" b_1 from b_2—as in the binary case—whenever $\{a_1 \ldots a_n, b_1\}$ and $\{a_1 \ldots a_n, b_2\}$ are coloured differently. We do this by deleting the ordered pair $\langle b_1, b_2 \rangle$ from $<_K$, and we can do this simultaneously for all $b_1 < b_2$ beyond a_n.

So the algorithm performs this procedure iteratively, once for each n-tuple $a_1 < a_2 < \ldots < a_n$, and considers these n-tuples in lexicographic order.

Now we must check as before that the set of ordered pairs that remain is a wellfounded tree with the nice path property.

The chief complication is with the branching number. To be able to run the cardinality argument as we did above we need to be able not merely to bound the size of the litters but to order all the litters uniformly. We need to be able to inject all litters simultaneously into some fixed set of size $2^{\aleph(\alpha)}$.

Each litter is indexed by colours, in the sense that for no node ν can two children have the same colour. 'Colour' here does not refer directly to Δ: two children c and c' of ν have different colours as long as there is an n-tuple $\langle a_1 \ldots a_{n-1}, \nu \rangle$ such that $\langle a_1 \ldots a_{n-1}, \nu, c \rangle$ and $\langle a_1 \ldots a_{n-1}, \nu, c' \rangle$ are coloured differently by Δ. How many colours does that make? Clearly, 2-to-the-power-of the number of such n-tuples. But this tells us how to label the colours uniformly. Fix a wellordering $\langle W, \leq_w \rangle$ of length $\aleph(\kappa)$. think of the colour of c as the set of n-tuples $\vec{i}$ from W such that Δ colours $\langle a_{i_1} \ldots a_{i_n}, c \rangle$ blue. This works uniformly for all litters, and means that we can argue that the tree is one where every node has at most $2^{\aleph(\kappa)}$ children, and is of height $\aleph(\kappa)$ and has therefore at most $(2^{\aleph(\kappa)})^{\aleph(\kappa)} = 2^{\aleph(\kappa)}$ points, which is impossible as before.

Any path through this tree has the property that for every n-tuple $\langle a_1 < \ldots < a_n \rangle$ on it, all tuples $\langle a_1 \ldots a_n, c \rangle$ with $c > a_n$ are coloured the same by Δ. Thus we have a two-colouring of n-tuples from a set K of size $\aleph(\kappa)$, and we have obtained this from a two-colouring of the $n+1$-tuples from an $\aleph(2^{2^{2^\kappa}})$-sized superset of K

We have already established the base case, and these manipulations are precisely what is needed for the induction step in the proof by induction on n that

THEOREM 1. $\aleph(\beth_{3n}(\kappa)) \to (\aleph(\beth_{n-1}(\kappa)))^n_2$

1. P. Erdös. Some set-theoretical properties of graphs. Revista Universidad Nacional de Tucuman, Série A vol 3 (1942) pp 363-367.

2. P. Erdös and R. Rado. A partition calculus in Set theory. Bull Am Maths Soc 1956 pp 427-498.