Prompt sets and automorphisms of $\mathcal{E}$:

Why building an effective automorphism can be easier than building a Δ^0_3 one

Rachel Epstein

Department of Mathematics
Harvard University

April 1, 2012
ASL North American Annual Meeting
University of Wisconsin - Madison

Definition (Myhill)

Let $\mathcal{E}$ be the structure of the c.e. sets under set inclusion. That is, the lattice $\mathcal{E} = \{\{W_e\}_{e\in\omega}, \cup, \cap, \omega, \emptyset\}$.

Unlike the structure of c.e. degrees $\mathcal{R}$, we know of many nontrivial automorphisms of $\mathcal{E}$.

The simplest automorphisms arise from computable permutations of ω.

Theorem (Kent)

There are $2^{\aleph_0}$ many automorphisms of $\mathcal{E}$.

Let $\mathcal{F}$ be the ideal of finite sets.

$\mathcal{F}$ is definable in $\mathcal{E}$ because the computable (complemented) sets are definable and $\mathcal{F}$ is the class of sets whose subsets are all computable.

Let $\mathcal{E}^* = \mathcal{E}/\mathcal{F}$.

Theorem (Soare)

Every automorphism of $\mathcal{E}^*$ is induced by a permutation of ω, so every automorhpism of $\mathcal{E}^*$ is induced by an automorphism of $\mathcal{E}$.

The automorphisms in Kent's theorem all induce the trivial automorphism of $\mathcal{E}^*$. However:

Theorem (Lachlan)

There are $2^{\aleph_0}$ many automorphisms of $\mathcal{E}^*$.

An automorphism Φ of $\mathcal{E}^*$ is *presented* by h if $(\forall n)(\Phi([W_n]) = [W_{h(n)}])$.

We say an automorphism of $\mathcal{E}^*$ is *effective* if it is presented by a computable h.

Theorem (Soare)

Let Φ be an automorphism of $\mathcal{E}^*$ presented by h. Then it is induced by a permutation of ω of degree at most $(h \oplus \emptyset')'$.

Traditionally, the term “effective automorphism of $\mathcal{E}$” actually refers to any automorphism that induces an effective automorphism of $\mathcal{E}^*$.

Effective Extension Theorem

Let $\mathcal{E}^*(S)$ be the structure of c.e. sets contained in S, up to finite difference.

Soare's Effective Extension Theorem (EET) (1974) says that if you effectively build an isomorphism from $\mathcal{E}^*(\overline{A})$ to $\mathcal{E}^*(\overline{B})$ satisfying certain properties, then you can extend this isomorphism to an effective automorphism of $\mathcal{E}^*$ taking A to B.

This theorem was actually originally used to build non-effective automorphisms.

Theorem (Soare)

Given any two maximal sets, there is an automorphism of $\mathcal{E}^*$ (and thus of $\mathcal{E}$) that takes one to the other, but there is not always an effective automorphism of $\mathcal{E}^*$ that does.

This was the first major theorem to use the EET, but it built non-effective automorphisms!

A *skeleton* of the c.e. sets is a listing of all c.e. sets up to finite difference.

The EET builds an effective automorphism of a skeleton $\{U_n\}$ of c.e. sets, but does not provide a computable function f such that $W_e =^* U_{f(e)}$.

Δ^0_3 automorphisms of $\mathcal{E}$

In the 1990's, Cholak and Harrington/Soare developed methods for building Δ^0_3 automorphisms.

These methods allowed for many new and interesting theorems.

Theorem (Cholak 1995, Harrington-Soare 1996)

Every noncomputable c.e. set is automorphic to a high set.

Corollary

All downward closed jump classes L_n, $\overline{H_n}$, $n \geq 1$, are noninvariant, and thus not definable.

Definition

A c.e. set A is *promptly simple* iff there exists a computable p s.t. $(\forall W_e)(\exists x)[x \in W_{e,\text{at}\,s} \cap A_{p(s)}]$.
A c.e. set A is *prompt* if it is the same degree as a promptly simple set, or if it "promptly permits": there exists a computable p s.t. $(\forall W_e)(\exists x)[x \in W_{e,\text{at}\,s} \implies A_{p(s)}\restriction x \neq A_s\restriction x]$.

There are prompt degrees in all jump classes.

Theorem (Harrington-Soare, 1996)

For all prompt sets A, there exists $B \equiv_{\mathrm{T}} \mathbf{0}'$ such that A is automorphic to B.

Note that this and the previous theorem move sets up in degree, using coding.

To move sets down in degree, we need restraint. Some examples of theorems:

Theorem (Soare)

If A has a semilow complement (i.e. $\{e : \overline{A} \cap W_e \neq \emptyset\} \leq_{\mathrm{T}} \emptyset'\}$) then $\mathcal{E}^*(\overline{A})$ is isomorphic to $\mathcal{E}^*$.

Theorem (Wald)

If A has semilow complement and C is promptly simple, then there exists $B \leq_{\mathrm{T}} C$ such that A is automorphic to B.

We can sometimes move degrees down even if we don't have semilow complement:

Theorem (Epstein)

*There is a non-low degree **d** such that all c.e. sets in **d** are automorphic to a low set.*

In general:

- Promptness allows us to move sets up in degree.
- Having a semilow complement allows us to move sets down in degree.

Question

Can we combine these?

Theorem (Maass)

If A and B both have semilow complements and are promptly simple, then A is automorphic to B.

This gave rise to the following:

Theorem (Wald)

If A has semilow complement and is promptly simple, and C is prompt, then there is a $B \equiv_T C$ s.t. A is automorphic to B.

Conjecture

If A has semilow complement and is prompt, and C is prompt, then there is a $B \equiv_T C$ s.t. A is automorphic to B.

Maass' theorem is not true if you replace “promptly simple” with “prompt”, so the same proof as Wald's theorem won't work.

We want to build B so that:

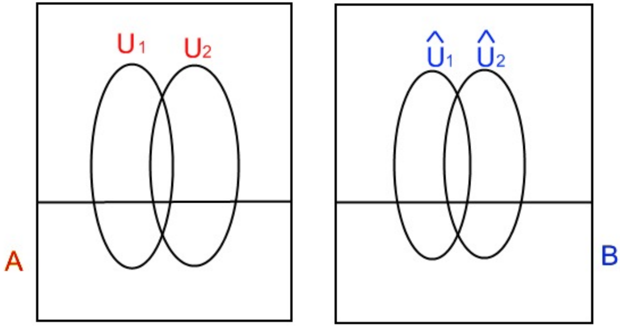
- Using promptness of A, we can code C into B
- Using that A has a semilow complement, we can keep the degree of B down
- Using promptness of C, we can compute B from C (permitting)

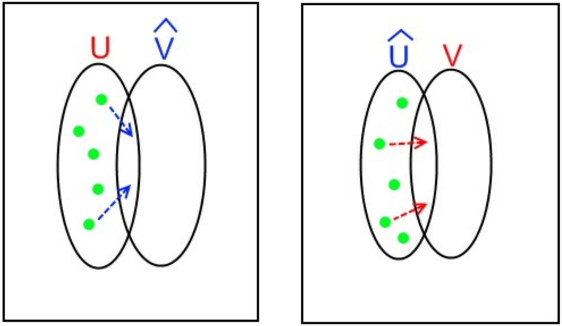
Building an automorphism (extreme over-simplification)

Red = Things we are given

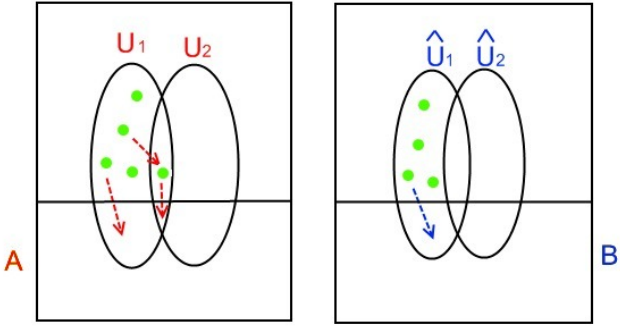
Blue = Things we build

- Given an enumeration $\{U_n\}_{n\in\omega}$ of the c.e. sets, where $U_0 = A$.
- Build an enumeration $\{\widehat{U_n}\}_{n\in\omega}$ of the c.e. sets. Let $B = \widehat{U_0}$.
- We build $\widehat{U_n}$ so that $\Theta : U_n \mapsto \widehat{U_n}$ is an automorphism.


U_1
U_2
A
$\hat{U}_1$
$\hat{U}_2$
B


U
$\hat{V}$
$\hat{U}$
V

How prompt allows coding



What goes wrong?

To keep $B \leq_{\mathrm{T}} C$, C must know whenever we put in a coding marker into B.

This seems fine, because we are coding C itself into B.

But: in the Δ_3^0 machinery, we build the automorphism on a tree.

When our approximation to the true path moves left, we must dump in coding markers to alert us that this is not where the coding will be done.

But for C to compute this extra stuff, C would need to compute the true path, and it can't.

What about using the effective automorphism machinery instead?

Recall that the effective automorphism machinery doesn't necessarily build only effective automorphisms.

The prompt coding method would have to be adapted to effective autos (not easy), but the problems of the Δ^0_3 case would not apply.

Theorem (Cholak, Downey, Stob)

Every promptly simple set is effectively automorphic to a complete set.

This method doesn't work for prompt sets, but it gives hope.

Despite its flexibility, the Δ^0_3-automorphism method is (in a way) *more* restrictive than the effective automorphism method!

Previously, people mostly only looked at the effective machinery if they specifically wanted an effective automorphism.

It may in fact help us solve some unsolved automorphism problems.

Thanks for listening!