Definability, Invariance, and Automorphisms of the Computably Enumerable Sets

Rachel Epstein

Further information on the poster presented at the 2010 AWM Workshop at the Joint Mathematical Meetings

1. Computability Theory

The study of computability theory began with the search for a formal definition of the intuitively computable functions and sets. For example, the set of even numbers is a computable set because we can determine whether or not a number is even. A set is not computable if no algorithm or computer program could tell you whether or not a number is in the set. Kleene came up with six schemes defining the class of computable functions. Turing defined these same functions using machines with a single infinite tape that can be read and changed one cell at a time, which we call *Turing machines*.

Turing also showed how his machines could be used to compute functions and sets relative to some non-computable set. The concept of relative computability allows us to study and classify the noncomputable functions and sets. We say a set A is *Turing reducible to* B and write $A \leq_{\mathrm{T}} B$ if A can be computed from an algorithm that can ask whether or not any element is in B. The sets A and B are *Turing equivalent*, $A \equiv_{\mathrm{T}} B$, if $A \leq_{\mathrm{T}} B$ and $B \leq_{\mathrm{T}} A$. The degree $\mathbf{a}$ of A is $\mathbf{a} = \{B \mid A \equiv_{\mathrm{T}} B\}$. $\mathbf{0}$ is the degree of the computable sets.

Computability theory is concerned mainly with sets and functions that are not computable. One particularly important class of sets is the class of computably enumerable sets. A *computably enumerable (c.e.) set* A is a set where there is an effective algorithm that can list the elements of A in some order. The c.e. sets can be equivalently defined as the domains of partial computable functions, ranges of computable functions, or Σ^0_1 sets. Note that all computable sets are c.e. We say a degree is c.e. if it contains a c.e. set. An important c.e. degree is the degree $\mathbf{0}'$ of the halting set, which is the set that tells you which algorithms halt on which input. Every c.e. set is Turing reducible to $\mathbf{0}'$. When a set has degree $\mathbf{0}'$, we say it is *complete*. The c.e. degrees have been studied extensively since Post first asked in 1944 whether there is a c.e. degree strictly between $\mathbf{0}$ and $\mathbf{0}'$, but there are still many interesting open questions involving the c.e. sets and degrees.

There are also many applications of c.e. sets to other areas of mathematics. The word problem in algebra is one example. The solution to Hilbert's 10th problem also uses c.e. sets. Perhaps the most well-known use of c.e. sets was Gödel's Incompleteness Theorem. While Gödel did not use the words "computably enumerable," his proof heavily relies on the concept of c.e. sets. More recently, Nabutovsky and Weinberger have found applications of c.e. sets in differential geometry.

The primary way of classifying degrees is by their jumps. The *jump* A' of a set A is the degree of the halting set relative to A. We call a c.e. set A *high* if $\mathbf{0}'' \leq_{\mathrm{T}} \mathbf{A}'$ and *low* if $A' \leq_{\mathrm{T}} \mathbf{0}'$. The high sets have information content close to that of $\mathbf{0}'$ and the low sets have low information content. They can be thought of as nearly computable. Another important class of sets is the low_2 sets, where a set A is low_2 if $A'' \leq_{\mathrm{T}} \mathbf{0}''$. The low_2 sets have slightly more information content than the low sets. We can generalize these definitions so that the low_n sets have more information content than the low_{n-1} sets, and the high_n sets have less information content than the high_{n-1} sets. Since the jumps of two sets in the same degree also have the same degree, these properties are well-defined for degrees. We often abbreviate these classes by L_n and H_n when referring to the low_n and high_n degrees. We call these *jump classes* of degrees.

2. Definability and invariance

Definability is one of the most fundamental themes in computability theory, and indeed in all of logic. My research involves studying which jump classes of degrees are definable in the structure of the c.e. sets. This work led to the solution to a 40-year-old problem about the definability of degree classes. We can examine the c.e. sets by looking at the structure $\mathcal{E}$ of the c.e. sets under inclusion. We say a class of sets $\mathcal{S}$ is definable in $\mathcal{E}$ if we can describe $\mathcal{S}$ in the language of set inclusion. A class of degrees $\mathbf{D}$ is definable in $\mathcal{E}$ if there is a class of sets $\mathcal{S}$ definable in $\mathcal{E}$ such that $\mathbf{D} = \{\deg(W) \mid W \in \mathcal{S}\}$. The question of which classes of degrees are definable has been the topic of much research.

Definable jump classes. It is natural to ask which jump classes of degrees are definable in the structure $\mathcal{E}$. Cholak, Harrington, Lachlan, Millar, Shoenfield, and Soare have all contributed to the results over the past 40 years. The only jump class for which the definability question remained open was the nonlow degrees. I have shown that the nonlow degrees are not definable in $\mathcal{E}$, using a two-phase method that should be applicable to other open problems in the area. My work completed the problem of determining which jump classes are definable in $\mathcal{E}$.

The primary method of showing whether a class is definable is to find a definition for the class. However, to show that a class is not definable in $\mathcal{E}$, we show that it is noninvariant. We say a class of sets $\mathcal{S}$ is *invariant* if every automorphism of $\mathcal{E}$ takes $\mathcal{S}$ to itself. A class of c.e. degrees $\mathbf{D}$ is *invariant* if there is an invariant class of sets $\mathcal{S}$ such that $\mathbf{D} = \{\deg(W) \mid W \in \mathcal{S}\}$. Every definable class is invariant because automorphisms must preserve the structure of $\mathcal{E}$. To show that a class $\mathbf{D}$ is noninvariant, we must show there is some degree $\mathbf{d} \in \mathbf{D}$ such that for all sets $D \in \mathbf{d}$, there is an automorphism of $\mathcal{E}$ taking D outside of $\mathbf{D}$.

For the downward closed jump classes, the problem of definability in $\mathcal{E}$ has been solved. The class of computable sets, L_0, is definable. Harrington and Soare [1996b] showed that all other downward closed jump classes are noninvariant, and thus not definable.

For the upward closed jump classes, Martin [1966] showed that the high degrees are definable, and Lachlan [1968] and Shoenfield [1976] showed that $\overline{L_2}$ is definable. Cholak and Harrington [2002] solved the problem for all but $\overline{L_1}$ by proving that for $n \geq 2$, H_n and $\overline{L_n}$ are definable. The only remaining jump class was $\overline{L_1}$.

$\overline{L_1}$ is not definable in $\mathcal{E}$.

Conjecture 1 (Harrington-Soare (1996a))**.** *$\overline{L_1}$ is noninvariant.*

We prove this conjecture by the following theorem.

Theorem 2.1 (Epstein (ip2))**.** *There exists a nonlow set D such that for all c.e. $A \leq_{\mathrm{T}} D$, there exists a low set B such that A is taken to B by an automorphism of $\mathcal{E}$.*

Corollary 2.2 (Epstein (ip2)). *$\overline{L_1}$ is the only upward closed jump class that is not definable.*

Harrington and Soare believed that their automorphism method in [1996b] would be enough to prove their conjecture. Indeed, other researchers have used this method for other important results. However, they soon realized that their method would not work here, due to the need to incorporate restraint into the automorphism machinery. Therefore, the Harrington-Soare conjecture remained open until [Epstein, ip2].

The new automorphism method. My $\overline{L_1}$ theorem is the only automorphism theorem that involves sending a set A *down* to a low set B. Cholak [1995] and Harrington-Soare [1996b] showed that for every noncomputable set A, there is an automorphism of $\mathcal{E}$ taking A to a high set B. In addition, Harrington-Soare [1996b] showed that all prompt sets are automorphic to complete sets. Both of these theorems took the given sets *up* to high sets and complete sets. They did this using coding methods. No restraint was involved in these theorems. In contrast, my theorem involves a complicated restraint mechanism to ensure that every $A \leq_{\mathrm{T}} D$ is automorphic to a low set B. I developed new machinery that works within $\overline{A}$ and $\overline{B}$ in order to handle the problem of restraint while building a partial automorphism of $\mathcal{E}$. In addition, I produced a new way to extend partial automorphisms to automorphisms of $\mathcal{E}$. These two techniques together prove the theorem.

References

[1] [Cholak, 1995]
P. Cholak. Automorphisms of the lattice of recursively enumerable sets, *Mem. Amer. Math. Soc.*, **113** (1995), (541) viii+151.

[2] [Cholak-Harrington, 2002]
P. Cholak and L. A. Harrington. On the definability of the double jump in the computably enumerable sets, *J. Math. Logic*, **2**(2) (2002), pp. 261–296.

[3] [Epstein, ip]
R. Epstein, Invariance and automorphisms of the computably enumerable sets, in preparation.

[4] [Harrington-Soare, 1996a]
L. Harrington and R. I. Soare, Definability, automorphisms, and dynamic properties of computably enumerable sets, *Bull. Symbolic Logic*, **2** (1996), 199–213.

[5] [Harrington-Soare, 1996b]
L. Harrington and R. I. Soare, The Δ^0_3 automorphism method and noninvariant classes of degrees, *Jour. Amer. Math. Soc.*, **9** (1996), 617–666.

[6] [Lachlan, 1968]
A. H. Lachlan, Degrees of recursively enumerable sets which have no maximal superset, *J. Symbolic Logic* **33** (1968), 431–443.

[7] [Martin, 1966]
D. A. Martin, Classes of recursively enumerable sets and degrees of unsolvability, *Z. Math. Logik Grundlag. Math.* **12** (1966), 295–310.

[8] [Shoenfield, 1976]
J. R. Shoenfield, Degrees of classes of r.e. sets, *J. Symbolic Logic* **41** (1976), 695–696.

Department of Mathematics
University of Chicago
Chicago, Illinois 60637-1546
E-mail: rachel@math.uchicago.edu
World Wide Web: http://www.math.uchicago.edu/∼rachel