and a readable exposition of r.e. sets, and as such it is still read by students today. (See the following review for examples of its influence.)

The 1946 paper, *A variant of a recursively unsolvable problem*, proves the unsolvability of what is now called the Post correspondence problem. In his 1947 paper, *Recursive unsolvability of a problem of Thue*, Post shows the unsolvability of the word problem for semigroups.

Near the end of Post's life, his paper with Kleene was published, *The upper semi-lattice of degrees of recursive unsolvability*. Another classic, this paper deals with the theory of Turing degrees. It describes what can be done in that theory without the later method of priority arguments. Post's contribution to this joint paper was done in the late 1940's. Post eventually sent his work to Kleene, who reworked the paper and greatly strengthened the results. It was in 1956, after Post's death, that Friedberg (XXIII 225) and Muchnik (XXII 218) solved "Post's problem," showing the existence of intermediate r.e. degrees.

Post's papers are published in this volume in the order in which they were written. With the exception of *Absolutely unsolvable problems and relatively undecidable propositions—account of an anticipation*, the papers are reproduced photographically from the original publications. Even the monograph, ***The two-valued iterative systems of mathematical logic***, originally printed from a typescript, is here reproduced photographically.

The editor and the publisher have performed a service by making these early contributions to symbolic logic conveniently available. Less convenient is the fact that to find the date and place of publication for an abstract, the reader must refer to the page of copyright permissions. H. B. ENDERTON

ROBERT I. SOARE. *Automorphisms of the lattice of recursively enumerable sets. Part I: maximal sets.* ***Annals of mathematics***, ser. 2 vol. 100 (1974), pp. 80–120.

MANUEL LERMAN and ROBERT I. SOARE. *d-Simple sets, small sets, and degree classes.* ***Pacific journal of mathematics***, vol. 87 (1980), pp. 135–155.

PETER CHOLAK. ***Automorphisms of the lattice of recursively enumerable sets.*** Memoirs of the American Mathematical Society, no. 541. American Mathematical Society, Providence 1995, viii + 151 pp.

LEO HARRINGTON and ROBERT I. SOARE. *The $\Delta^0_3$-automorphism method and noninvariant classes of degrees.* ***Journal of the American Mathematical Society***, vol. 9 (1996), pp. 617–666.

**Historical origins.** Computability (or recursion) theory grew from our efforts to understand the *algorithmic* content of mathematics. One of the great achievements of the twentieth century is the development of a precise formulation of the notion of a computable function via the Church–Turing Thesis. In his very influential 1944 paper (X 18), Post articulated some of the fundamental notions at the heart of most undecidability proofs. He observed that these proofs worked by coding some "non-computability" into the theory at hand, thereby arguing that the relevant structures could emulate computation. One key concept was that of effective enumeration which leads to the notion of a computably (recursively) enumerable set. A computably enumerable set is a subset of $\mathbb{N}$ that is ($\emptyset$ or) the range of a computable (total) function. The intuitive idea is that if $f$ is computable, then I can "effectively list" (not necessarily in order) the range of $f$ as $\{f(0), f(1), \dots\}$. Think of consequences of a computably enumerable set of axioms for a formal system. The other key concept discovered by Turing (IV 128) and further developed in Post's paper was that of a *reducibility*. Here one says that $A \leqq B$ if one can compute $A$ *given* $B$ as read-only memory. The ordering $\leqq$ can vary according to the access mechanism given. For instance we say that $A$ is m-reducible to $B$ if there is a computable function $f$ such that $x \in A$ iff $f(x) \in B$. More generally, one says that $A$ is Turing reducible to $B$ ($A \leqq_T B$) if one can compute "$x \in A$?" from a finite number of calls to $B$. Reducibilities calibrate sets (and hence coded decision problems) into equivalence classes that measure "equicomputability." These classes are called *degrees*. Post observed that the structure of classical undecidability proofs worked by demonstrating that the various theories faithfully emulated some algorithmically unsolvable problem such as the halting problem. More formally, let $\{W_e : e \in \mathbb{N}\}$ be a computable enumeration of the computably enumerable sets. Then let $K_0 = \{\langle x, y\rangle : x \in W_y\}$. (Here $\langle \cdot, \cdot \rangle$ denotes a computable bijection taking $\mathbb{N} \times \mathbb{N}$ to $\mathbb{N}$.) $K_0$ codes the "halting problem" which asks if the $y$th algorithm halts on input $x$. A simple diagonal argument shows that $K_0$ is not computable. Post noted that the process of, say, arithmetization provides a reduction that codes $K_0$ into the relevant undecidable theory. In this way the formalism of the undecidability proofs of the time was removed, and the key ideas were revealed.

**Post's problem and degrees.** It is natural to seek to understand the nature of effective enumeration and computation by studying properties of computably enumerable sets and their degrees. Clearly the computably enumerable sets form a lattice $\mathscr{E}$. Their degrees form an upper semilattice traditionally denoted by $\mathscr{R}$. Ever since the ground-breaking paper of Post, there has been a persistent intuition that structural properties of computably enumerable sets have reflections in their degrees. One incarnation of this idea is that definability in $\mathscr{E}$ is closely linked with information content as measured by degree classes in $\mathscr{R}$. For instance, the complemented members of $\mathscr{E}$ are exactly the members of $\mathbf{0}$, the degree of the computable sets. Actually, it is more natural to analyse $\mathscr{E}^*$, the lattice formed by factoring $\mathscr{E}$ by the congruence $=^*$, where $A =^* B$ means that there is a finite set $F$ such that $A \cup F = B \cup F$. Analysing $\mathscr{E}^*$ is natural since $A =^* B$ means that from the point of view of computability theory $A$ and $B$ are identical, and furthermore $=^*$ is definable in $\mathscr{E}$ ($F$ is finite iff every computably enumerable subset of $F$ is complemented).

We note that $K_0$ is the most complicated computably enumerable set, since if we consider any computably enumerable set $W_z$, then $x \in W_z$ iff $\langle x, z\rangle \in K_0$. Hence every computably enumerable set is m-reducible to $K_0$. For this reason, if we let $\mathbf{0}'$ denote the degree of $K_0$, we call $\mathbf{0}'$ (m-) *complete*. It was noted by Post and others that all computably enumerable sets known to them were either computable or complete. Post asked if there were any incomplete computably enumerable degrees: computably enumerable degrees aside from $\mathbf{0}$ and $\mathbf{0}'$, or was it the case that computably enumerable problems were either computable or the "halting problem in disguise"?

Post even suggested a programme to answer this question. Since complemented members of $\mathscr{E}$ are computable, but $K_0$ had many infinite computably enumerable sets disjoint from it, perhaps a very "non-complemented" member of $\mathscr{E}$ would turn out to be incomplete. Towards this goal, Post defined a computably enumerable set $A$ to be *simple* if $\mathbb{N} - A$ was infinite and for all infinite computably enumerable $W$, $W \cap A \neq \emptyset$. Post constructed a simple set and then proved that a simple set $A$ is not of the same m-degree as $\mathbf{0}'$. Thus, for the finer classification of m-reducibility, structural properties of $\mathscr{E}$ suffice to construct incomplete degrees. Simple sets can be Turing-complete, however, and simple sets are not enough for the realization of Post's programme.

**The priority method.** As it turned out, Post's problem was eventually solved by Friedberg (XXIII 225) and Muchnik (XXII 218) not by a structural analysis of $\mathscr{R}$, but by using a sort of effective forcing argument called the "priority method," which built on earlier ideas of Kleene and Post (XXI 407). This method has been developed and refined and is now the mainstay of classical computability theory. The method has also found applications in computer science, descriptive set theory, combinatorial group theory, and the like.

The basic idea of this method is the following. One is trying to construct a computably enumerable set $A$ using some effective step-by-step process. One has to achieve certain goals $G_0, G_1, \ldots$, so that eventually all goals are satisfied. A very primitive analogy would be Cantor's diagonalization in which our goal is to construct a real $A$ that differs from the list of given reals. So the $e$th goal is to diagonalize the $e$-real on the list. The trouble is that in an effective construction, one has only partial information given by the finite approximations to the hypotheses of the goals. For instance, a typical goal might say if a certain computably enumerable set is infinite, then do $\mathscr{P}$, else do $\mathscr{Q}$. In thinking about this as some sort of game against the universe, one must ensure that the step-by-step construction is sufficiently generic that the goals can all be met. This can be especially difficult since the goals are often in *conflict* and hence some internal *priority ordering* of the requirements is used to allow all to be met.

The two features, (i) reasoning with partial and asynchronous information, and (ii) internal conflicts of the goals, tend to make priority arguments difficult, delicate, and combinatorial. In modern versions, various combinatorial devices are employed to organize the construction. These include "infinite pinball machines" and "strategy trees" (more on this below).

Following the introduction of the priority method in the theory of degrees, work blossomed through the efforts of Sacks, Yates, Shoenfield, Lachlan, and others. In parallel to direct investigations of the computably enumerable degrees, there was also a long line of development into $\mathscr{E}$. For instance, Lachlan demonstrated that the two-quantifier theory of $\mathscr{E}$ is decidable. Earlier, Myhill observed that if Post's original programme were to succeed then *maximal sets* should be incomplete. He defined a computably enumerable set $M$ to be *maximal* if $M$ was a co-atom in $\mathscr{E}^*$. (That is, $\mathbb{N} - M$ is infinite and for all computably enumerable $W$, if $W \supseteq M$ then either $W =^* M$ or $W =^* \mathbb{N}$.) Maximal sets were first constructed by Friedberg.

**Maximal sets and invariant degree classes.** Yates demonstrated that maximal sets could be complete, however. In fact, Tennenbaum suggested that perhaps *all* maximal sets were complete. Tennenbaum's suggestion was incorrect since maximal sets can be incomplete as was observed by Sacks. Nevertheless, Tennenbaum's idea proved remarkably fruitful. In a beautiful paper, D. A. Martin (XXXII 528(2)) showed that maximal sets have *high* degrees and conversely all high computably enumerable degrees contain maximal sets. A high degree is defined as follows. For any set $A$ we can define the jump $A'$ of $A$ as $\{\langle x, y\rangle : x \in W_y^A\}$ where $W_y^A$ denotes the $y$th set computably enumerable *given $A$ as read-only memory*. (Note that $K_0$ is $\emptyset'$.) The jump of $A$ is simply the halting problem *relative to $A$*. A set $A$ is called *low* if $A'$ has degree $\mathbf{0}'$ and is called *high* if $A'$ has degree $\mathbf{0}''$, the degree of the halting problem relativized to $K_0$. Martin's theorems demonstrate that despite the fact that maximal sets may not necessarily be complete, they do have high information content in the sense that they have the same jump as the halting problem. Martin had discovered the first *invariant class* in $\mathscr{R}$ in the sense that the high degrees are precisely those realized by the (definable class of) maximal sets. Of course, when one begins to consider definability in $\mathscr{E}$, one is naturally lead to study the automorphism groups of $\mathscr{E}$ and $\mathscr{E}^*$. Hence, for instance, we call a class $\mathscr{C}$ of computably enumerable degrees *invariant* if there is a collection $\mathscr{C}'$ of computably enumerable sets closed under automorphisms of $\mathscr{E}$ such that $\mathscr{C} = \{\deg(A) : A \in \mathscr{C}'\}$.

**Automorphisms and the Soare paper.** Early work showed that both of the automorphism groups $\mathrm{Aut}(\mathscr{E})$ and $\mathrm{Aut}(\mathscr{E}^*)$ are large since each has $2^{\aleph_0}$ automorphisms. Martin used a priority construction to show that a certain construction of Post (hypersimplicity) was not invariant under automorphisms of $\mathscr{E}$.

The original programme of Post was to find a "thinness" property of the complement of a computably enumerable set that would guarantee incompleteness. The final blow to Post's original programme was the marvellous 1974 paper under review of Soare, who shows that maximal sets form an orbit in $\mathrm{Aut}(\mathscr{E})$. In particular, no "extra" property together with maximality could guarantee incompleteness. This paper has been very influential. It introduces an intricate priority "pinball" method to construct complicated automorphisms of $\mathscr{E}$. The reader might gauge some idea of the complexity of the method by considering that while the automorphism (and the permutation of $\mathbb{N}$ inducing it) might be as complicated as, say, $\mathbf{0}''$ and hence highly non-computable, the result will take computably enumerable sets consistently to computably enumerable sets. I should remark that Soare also demonstrates that this complexity is necessary since there are maximal sets not automorphic with each other via "effective" automorphisms. Furthermore, to this day there are very few orbits of $\mathrm{Aut}(\mathscr{E})$ known.

**Automorphisms effective on skeletons.** I will try briefly to describe the method, called the "automorphism machinery." The $e$-state of $x \in \mathbb{N}$ is the set $\{y \leqq e : x \in W_y\}$. Note that $e$-states code the intersections needed for automorphisms. They can be also defined relative to any given array of sets $\{B_y : y \in \mathbb{N}\}$ in place of $\{W_y : y \in \mathbb{N}\}$. In particular, we often replace $\{W_y : y \in \mathbb{N}\}$ by some *skeleton* of computably enumerable sets $\{U_e : e \in \mathbb{N}\}$ where we guarantee that for all $e$ there is an $x$ such that $W_e =^* U_x$. This is because, as Soare observes, to get an automorphism of $\mathscr{E}^*$, and hence $\mathscr{E}$, it is enough to "match" states relative to some skeleton. More precisely, the basic goal of the automorphism machinery is to take two copies of $\mathbb{N}$, denoted by $\mathbb{N}$ and $\widehat{\mathbb{N}}$. We take two skeletons $\{U_e : e \in \mathbb{N}\}$ (on the $\mathbb{N}$ side), and $\{V_e : e \in \mathbb{N}\}$ (on the $\widehat{\mathbb{N}}$ side). The idea is that we construct sets $\{\widehat{U}_e : e \in \mathbb{N}\}$ (on the $\widehat{\mathbb{N}}$ side) and $\{\widehat{V}_e : e \in \mathbb{N}\}$ (on the $\mathbb{N}$ side), and we will send $U_e \mapsto \widehat{U}_e$ and $V_e \mapsto \widehat{V}_e$. On the $\mathbb{N}$ side we measure $e$-states relative to the array $U_0, \widehat{V}_0, U_1, \widehat{V}_1, \ldots$ and on the $\widehat{\mathbb{N}}$ side we measure $e$-states relative to the array $\widehat{U}_0, V_0, \widehat{U}_1, V_1, \ldots$. To make things clear, we put a hat on states to indicate that they are measured on the $\widehat{\mathbb{N}}$ side. To make this mapping an automorphism, by a back-and-forth argument, it is enough to match states. That is, it is enough to ensure that for all $e$, and $e$-states $\sigma$, we meet the requirement:

$$(\mathrm{R}_\sigma) \qquad (\exists^\infty x)[x \text{ has state } \sigma] \text{ iff } (\exists^\infty \hat{x})[\hat{x} \text{ has state } \hat{\sigma}].$$

Now from the last paper under review, one nice model for this is the Harrington–Soare device of viewing this as a two person game between two players RED and BLUE. RED plays the skeletons $\{U_e : e \in \mathbb{N}\}$ and $\{V_e : e \in \mathbb{N}\}$, while in response BLUE plays the corresponding hatted sets.

**Automorphisms of maximal sets.** Now imagine that we are given two maximal sets $M$ and $\widehat{M}$. We are attempting to send $M \mapsto \widehat{M}$. Naturally in the above scheme, we let $U_0 = M$ and $V_0 = \widehat{M}$. $M$ and $\widehat{M}$ divide the universe into two parts each, $\overline{M} = \mathbb{N} - M$ and $M$, and similarly for $\widehat{M}$ (that is, the part disjoint from $M$ and the part inside). In some sense, they can be treated separately. An $e$-state will be formed by the part intersecting $\overline{M}$ and the part intersecting $M$. We are trying to construct the automorphism *first* as an isomorphism on the $\overline{M} \mapsto \overline{\widehat{M}}$ part, and *then* extend it to a complete automorphism on the $M \mapsto \widehat{M}$ part.

We can think of the $U_i$'s and $V_i$'s as essentially $W_i$ and all are initially empty. As the construction proceeds RED must gradually enumerate more and more elements into these sets, thereby changing various states, causing us (BLUE) to enumerate in response elements into the corresponding hatted versions. We agree that since we will first look only at the complements, we will control the enumerations of the given sets so that if $x$ enters $M$ in some state $\sigma$ with $x$ not yet in $W_{\overline{z}}$ then we do not allow $x$ to enter $W_{\overline{z}}$ in this part of the construction. That is, elements can enter sets only while they live in the complements. Measuring states is difficult since it is a $\mathbf{0}'''$ question in general to see if a state has infinitely many members. (The terminology is that the state is "well resided.") In the case of a maximal set, our task is easier since on $\overline{M}$, $W$ basically looks like $\emptyset$ or it looks like $\overline{M}$, which is the relevant universe.

The basic idea is that when RED makes $U_e$, say, look more and more like the universe $\overline{M}$, then in response we (BLUE) make $\widehat{U}_e$ look more and more like $\overline{\widehat{M}}$. Trouble is caused by the fact that, in this case, $W_e \cap \overline{M} =^* \overline{M}$ rather than $W_e \cap \overline{M} = \overline{M}$. Thus, there will be a finite number of exceptions to concern us and these will need to be "guessed." Soare's idea is to use a $\mathbf{0}''$ priority construction first to replace the standard skeleton $\{W_e : e \in \mathbb{N}\}$ by a skeleton $\{U_e : e \in \mathbb{N}\}$ where the $U_e$ if infinite contain all of $\overline{M}$ and such that "good entry conditions," to be described below, are satisfied. The task of the partial automorphism on $\overline{M} \mapsto \overline{\widehat{M}}$ is then relatively simple. As $U_e$ looks more and more infinite, we put more and more (in order of magnitude) elements of $\overline{\widehat{M}}$ into $\widehat{U}_e$ in response. Naturally, should we do this infinitely often, then $U_e$ will contain $\overline{M}$, and the same will hold on the hatted side. If it happens finitely often, then both $U_e$ and $\widehat{U}_e$ are finite. The same story holds for the $V_e$ and $\widehat{V}_e$. Thus we achieve an isomorphism from $\overline{M}$ to $\overline{\widehat{M}}$.

The true complexity of the problem is revealed when one considers the second part of the construction, the $M \mapsto \widehat{M}$ part. It is true that we now need only to make an isomorphism from computably enumerable subsets of $M$ to those of $\widehat{M}$. Of course, for *any* computably enumerable sets $X$ and $Y$ the lattice of computably enumerable *sub*sets are isomorphic. But to achieve automorphisms, the isomorphisms on the complements and those on the subsets must be *compatible*. There is a radical difference, however, from the situation on the complements. When an element enters the complement it can be regarded as not being in any set. When an element $x$ enters the new universe of discourse, $M$, it might well already be in many sets due to *our* activity on the $\overline{M} \mapsto \overline{\widehat{M}}$ part. If, for instance, infinitely many elements enter $M$ in some state $\sigma$ yet only finitely many elements enter in the corresponding state $\widehat{\sigma}$, we are in big trouble. Furthermore, it is within *our* (BLUE's) power only to put elements into $\widehat{U}_e$ and $\widehat{V}_e$. If $\widehat{V}_n \in \sigma$, but for all $\tau$ on the $\widehat{\mathbb{N}}$ side for which $V_e$ is infinite, it is not within our (BLUE's) power to enumerate elements into $\widehat{U}_e$'s and $\widehat{V}_e$'s to lift the states $\sigma$ and $\tau$ to a common state (we say that $\sigma$ is *covered* if $\tau$ exists), then we must lose. We cannot hope to achieve the automorphism since it is already killed by *bad entry states*.

**A simple task fails.** The following example might serve to illustrate the problem. Suppose that we attempt to prove the false claim that all simple computably enumerable sets are automorphic. We are trying to send $A \mapsto \widehat{A}$. Now we consider a set $U$ on the $\mathbb{N}$ side. Suppose that many elements are entering $U_s - A_s$, at stage $s$. We need to decide what to do on the $\widehat{\mathbb{N}}$ side. In particular, should we put elements into $\widehat{U}_s - \widehat{A}_s$? The problem is that perhaps $U \subseteq A$. In that case, we are forced to ensure that $\widehat{U} \subseteq \widehat{A}$. But $\widehat{A}$ is not under our control. So our opponent's (RED's) strategy is to make $U - A$ infinite unless we (BLUE) do not put infinitely many elements into $\widehat{U} - \widehat{A}$. If, on the other hand, we (BLUE) respond making $\widehat{U} - \widehat{A}$ infinite, our opponent (RED) will ensure that $U \subseteq A$, despite the fact that $U_s - A_s$ is large infinitely often. Thus we have already lost on the $\overline{A}$ part. Next we might hope that if we had some certification that $U - A$ is infinite we might act. One example of this is the case where $A$ and $\widehat{A}$ are low sets. Roughly speaking, if a set $X$ is low, then for any computably enumerable $W$, if $W_s - X_s$ looks infinite infinitely often, then $W - X$ really is infinite and furthermore we have at our disposal an infinite collection of elements, an infinite subset of which we can guarantee are in $W - X$. Thus, lowness allows us, with some degree of confidence, to respond by putting elements into $\widehat{U}_s - \widehat{A}_s$ with a guarantee that if this process repeats infinitely often, then both states will be well resided. This lowness mechanism can be pushed through to demonstrate that if $X$ and $Y$ are low sets, then $\{W_e \cap \overline{X} : e \in \mathbb{N}\}$ is isomorphic to $\{W_e \cap \overline{Y} : e \in \mathbb{N}\}$ (and hence isomorphic to $\mathscr{E}^*$). This is a result of Soare which has been extended and generalized by Maass and others using weaker and weaker approximations to replace lowness.

Returning to our story, our next guess would be that all low simple sets are automorphic. This too is false. The false reasoning is the following. Lowness means that we cannot lose on the complements. Also simplicity of $A$ and $\widehat{A}$ leads us to hope that we will not lose on the entering elements of $A$. Hence if $U - A$ is infinite, we will respond by putting infinitely many elements into $\widehat{U}_s - \widehat{A}_s$, with guarantees that they are infinitely many are in $\overline{A}$. Of course, infinitely many elements will enter $\widehat{A}_s$, but since $A$ is simple, also infinitely many matching elements enter $A$ from $U$.

The obstacle to building an automorphism occurs with just one further set. First suppose that on the $\widehat{\mathbb{N}}$ side, there is a set $V$. Of those elements we (BLUE) put in $\widehat{U}_s - \widehat{A}_s$, RED ensures that half of them also enter $V$. Back on the $A$ side, we (BLUE) need to decide whether elements should be put into $\widehat{V}$ or not. Suppose that we respond by putting infinitely many elements into $\widehat{V}$. It might be that infinitely many of the elements on the $\widehat{\mathbb{N}}$ side that enter $\widehat{A}$ from $\widehat{U}$ are *not* in $V$, yet on the $\mathbb{N}$ side, *all* elements that enter $A$ in $\widehat{V}$ *are also* in $U$ since RED first puts them into $U$ and then into $A$ (from $\widehat{V}$). It is not within *our* (BLUE's) power to put elements into $V$ since $V$ is under the opponent's (RED's) control. By the same token, if we do not respond by putting elements in response into $\widehat{V}$, then RED causes us to lose on $\overline{A}$.

**$\mathscr{E}$-definable properties and non-automorphic sets.** In fact, it is possible to turn this obstacle (to the making of automorphisms) around in order to construct non-automorphic low simple sets. Lerman and Soare, in their 1980 paper under review, define a computably enumerable set $A$ to be *$d$-simple* if for all computably enumerable $X$ there exists a computably enumerable $Y \subseteq X$ such that $X \cap \overline{A} = Y \cap \overline{A}$, and such that for all computably enumerable $Z$, $|Z - X| = \infty$ implies $(Z - Y) \cap A \neq \emptyset$. (Clearly, $d$-simplicity is $\mathscr{E}$-definable.) As well as proving a number of results about the relationship of $d$-simple sets with $\mathscr{R}$, using "small" sets, Lerman and Soare observe that low simple sets can be either $d$-simple or non-$d$-simple and hence are not always automorphic. Note that the definition reflects the obstruction above. RED first plays elements into $Z_s - X_s$. Now whether or not $A$ is $d$-simple, if this happens infinitely often then $A$ must gain some of these elements. In the non-$d$-simple case, infinitely many elements from $Z_s - X_s$ will "promptly" enter $A$, *before* they enter $X$. In the $d$-simple case, if an element from $Z_s - X_s$ is to enter $A$, RED first enumerates it into $X_t$, forcing BLUE to put it into $Y_u - A_u$ for some $u \geqq t$ (or else lose on $X \cap \overline{A} = Y \cap \overline{A}$, because RED will not play again until BLUE responds), and only then allowing the relevant element to enter $A$. Thus the enumeration of $A$ is "tardy" in the non-$d$-simple case.

The Lerman–Soare result was the pioneering example of a *non-automorphism* result obtained by first analysing the manner by which the automorphism machinery failed and then finding an elementary lattice-theoretical property of $\mathscr{E}$ that reflected this dynamical property. The feature of such failure is that the property was a reflection of the dynamic relative speeds of enumeration of the sets, a fact that is very strongly borne out by the recent investigations of, particularly, Harrington and Soare. I should mention that finding elementary properties reflecting failures of the machinery is very difficult indeed. There have been a number of (increasingly complex) successes in this area due to Maass, Harrington, Downey, Stob and others. The most spectacular was the Harrington–Soare solution to Post's programme (*Post's program and incomplete recursively enumerable sets*, ***Proceedings of the National Academy of Sciences of the United States of America***, vol. 88 (1991), pp. 10242–10246) where the authors found an elementary property $Q(A)$ that guaranteed Turing incompleteness. Further $\mathscr{E}$-definable properties have been studied recently by Harrington and Soare as a device for limiting automorphisms.

**Promptly simple sets.** Returning to our simple set construction we have discovered a blockage to making low simple $A$ automorphic to $\widehat{A}$, caused by the relative speed of enumerations of the two sets. Maass (in this JOURNAL, vol. 47 (1982), pp. 809–823) introduced the idea of a *promptly simple* set for which, roughly speaking, we had the following at our disposal: If infinitely often we had some decision as to whether to raise states, etc., we could have infinitely many elements that would enter the relevant set "promptly" in the low state and hence not go "out of covering." In this way, we would avoid the problem described above. Maass demonstrated that this is enough for us to be able to show that any two low promptly simple sets are automorphic.

In any case, in summary, for the automorphism machinery to succeed, it is *necessary* to ensure that *for all entry states $\sigma$ (and dually for $\widehat{\sigma}$), if infinitely many elements enter $M$ in state $\sigma$, then there is some covering entry state $\widehat{\tau}$ ($\tau$, respectively).*

In the construction of the skeleton $U_e$, for the maximal set case, Soare shows that the necessary covering entry condition above can be satisfied. (This is the "order-preserving enumeration theorem.") The second and most famous part of his paper is to show also that satisfying the covering entry condition is *sufficient*. The *extension lemma* says that if you have two computably enumerable universes where the entry covering condition is satisfied, then it is possible to enumerate elements into the sets $\widehat{U}_e$ and $\widehat{V}_e$ so as to make an isomorphism. While the proof of this lemma was greatly simplified by Maass, Soare, and Stob, it remains strangely complex. But the extension lemma seems inevitable and occurs in most automorphism results, at least so far.

Further results on the automorphism group of $\mathcal{E}$ followed. As mentioned earlier, Soare observes that low sets ought to resemble computable sets, and uses the machinery to prove that if $A$ is low then the lattice of supersets of $A$ is isomorphic to $\mathcal{E}$. Maass later weakened the hypothesis to "semilow$_{1.5}$." The methodology has also been used by Hammond for the sets computably enumerable relative to some $A$. Other uses of the original machinery include work of Downey and Stob, Stob and Maass, and Stob.

**The present works on $\Delta_3^0$ automorphisms.** This all brings us to the last two works under review. There were a number of central open questions that had resisted all attempts at solution. They included the basic question discussed earlier of whether all non-computable computably enumerable sets were automorphic to complete sets (the general form of Post's programme), and whether every lattice of supersets can be found in each high computably enumerable degree. In the early 1980's, in unpublished work, Harrington had developed an alternative methodology for constructing automorphisms of $\mathcal{E}$. In Harrington's formulation, one constructs a tree of computably enumerable sets in place of the skeletons given before. For instance, in our maximal set example, we would still send $M$ to $\widehat{M}$, but at the next level of the tree, we might have infinitely many possible versions of $U_1$ and $\widehat{U}_1$, and $V_1$ and $\widehat{V}_1$. In our case at hand, these could correspond to different guesses as to the number of elements not in $W_1 \cap \overline{M}$, perhaps plus other relevant information. For instance, the nodes would also contain all the states of that length that were "well visited" in the sense that they had infinitely many elements flowing through them.

The fundamental idea is to then treat automorphisms like other tree arguments, so that the relevant automorphism is read off from the leftmost path visited infinitely often. (See Soare's book, ***Recursively enumerable sets and degrees***, LV 356, Ch. XIV.) Put crudely, the original methodology constructs an "effective" automorphism on some skeleton where $U_e$ always goes to *a fixed* $\widehat{U}_e$, whereas the new methodology constructs many possible automorphisms that are effective in the path. But you can figure out *which* $\widehat{U}_\alpha$ we send $U_\alpha$ to only from the knowledge of the true path of the construction.

"Theoretically" at least, the method gives no additional power, since Cholak observes that any $\Delta_3^0$ automorphism is effective on some skeleton. Here is an easy proof. Suppose that $W_e \mapsto W_{f(e)}$ is a $\Delta_3^0$ automorphism. We can regard the function $e \mapsto f(e)$ as being given by an approximation $f(e,s)$ where $f(e) = \mu j(\exists^\infty s)[f(e,s) = j]$. For a fixed $e$, we can define an array of pairs of sets, exactly one pair being infinite, as follows. Define $U_{\langle e,j,t\rangle,s} = W_{e,s}$ and $V_{\langle e,j,t\rangle,s} = W_{j,s}$ to hold if $f(e,s) = j$ and $t$ is least such that for all $u$ with $t \leq u \leq s$, $f(e,u) \geq j$. It is clear that for all $e$ there is a unique $j,t$ such that $U_{\langle e,j,t\rangle} = \bigcup_s U_{\langle e,j,t\rangle,s} = W_e$ and $V_{\langle e,j,t\rangle} = \bigcup V_{\langle e,j,t\rangle,s} = W_j = W_{f(e)}$. Notice that for all $\langle j',t'\rangle \neq \langle j,t\rangle$, both $U_{\langle e,j,t\rangle}$ and $V_{\langle e,j,t\rangle}$ are finite. We can regard the $U_i$ and $V_i$'s as giving a skeleton. We then map $U_i \mapsto V_i$ for all $i$ and it is clear that this induces the same automorphism as does $f$ and the map $U_i \mapsto V_i$ is effective on the skeleton so constructed.

The observation above, however, belies the fact that the conceptual advantage of constructing the automorphism on a tree is huge. If one wishes to achieve certain goals one simply puts requirements as additional nodes on the tree in the standard way. One must then simply prove some sort of compatibility. Furthermore, it is very difficult to construct the relevant skeleton without some sort of tree construction. Finally, there is the hope that the tree methodology will give rise to automorphisms at higher levels, perhaps even proving the Slaman–Woodin conjecture that the set $\{\langle e,j\rangle : W_e \text{ is automorphic to } W_j\}$ is $\Sigma_1^1$-complete.

Each of the last two works under review gives an exposition of the $\Delta_3^0$ method, and uses it to solve major questions. The method regards elements as "balls" that begin at the top (root) of the tree. An element will then roll down some path according to the flow of information ($\Delta_3^0$) at the nodes. The nodes process the balls into various sets to achieve the desired state matching and other requirements of the construction. The reader should note that, while the set $U_\alpha$ occurs only at some level of the tree (where we consider $\alpha$-states), activity with respect to $U_\alpha$ will occur for all nodes below the node $\delta$ processing $\alpha$-states. After all, if $\tau$ extends $\alpha$ then a $\tau$-state is also an $\alpha$-state. Basically, the action at an automorphism node $\delta$ processing $\alpha$ states will be to ensure that matching occurs on the $\alpha$ side and conversely. It must ensure as well that if we discover that $\alpha$ is not well resided because almost all elements leave (because all leave below $\delta$) then $\delta$ must also ensure that this situation is mirrored on the hatted side (and conversely).

All of this is $\Delta_3^0$ information at $\delta$. Of course, the construction has at its disposal only finite *approximations* to the real information. Hence an element $x$ might get to some node $\sigma$ extending a false outcome of $\delta$, based on wrong information. When we later discover that this information is wrong, we will let $\delta$ pull the ball $x$ back up to $\delta$ for reprocessing. We should note, however, that we may well have performed a large amount of enumeration into (e.g.) sets $\widehat{U}_\beta$ for $\beta$ *above* $\alpha$ for the sake of (e.g.) $\sigma$ which is false. We must ensure that this is not fatal. We can restart the sets $U$ and $V$ living below $\delta$ but not those above $\delta$. The problems here are quite subtle, and it is worth mentioning that this area is littered with false claims, fortunately none of which have been published.

**The heart of the $\Delta_3^0$ automorphism method.** I will illustrate the basic principles of the $\Delta_3^0$ automorphism method in an extremely trivial setting. Suppose that a computably enumerable set $U$ is played by the opponent (RED) and we (BLUE) must play a set $\widehat{U}$ that is similar in the sense that $U$ is infinite iff $\widehat{U}$ is infinite and likewise for their complements, but so that $\widehat{U}$ can have other properties such as being complete or simple. In a vastly oversimplified version of the *effective* automorphism method it is easy to build such a $\widehat{U}$ by a variant of the following technique. Let $\overline{U_s} = \{x_0^s < x_1^s < \cdots < x_n^s < \cdots\}$ and similarly for the hatted side. Now whenever a new element appears in $U$ we enumerate a new element in $\widehat{U}$, and whenever $x_n^s \neq x_n^{s+1}$ we put $\hat{x}_m^s$ into $\widehat{U}$ for some particular $m$ so that if $\lim_s x_n^s = \infty$ then $\lim_s \hat{x}_m^s = \infty$ also.

In the $\Delta_3^0$ version there are exactly three atomic diagrams for $U$: the first, $D_a$, asserts that $\overline{U} =^* \emptyset$; the second, $D_b$, asserts that $U$ and $\overline{U}$ are *both* infinite; and the third, $D_c$, asserts that $U =^* \emptyset$. The essence of the method is to put the three nodes $\{a, b, c\}$ in order $a <_T b <_T c$ on a tree $T$, whose *true path* $f$ is defined by the true status of $U$ with respect to these three possibilities, which we denote by $\langle i \rangle \subset f$, for $i \in \{a, b, c\}$. Next we want to to find a computable approximation $\{f_s\}_{s \in \omega}$ so that $\liminf_s f_s = f$ (as in LV 356, Ch. XIV) and allow the strategy for $D_a$, for example, to act exactly when $\langle a \rangle \subseteq f_s$, namely when control of the construction passes to node $\langle a \rangle$ on $T$. The three strategies each build a distinct set $\widehat{U}_i$, $i \in \{a, b, c\}$, only one of which will be the final $\widehat{U}$, namely that $\widehat{U}_i$ for $\langle i \rangle \subseteq f$. The main advantage over the effective method is that each strategy has only one task to carry out and can thus be more easily combined with other non-automorphism strategies.

The main problem with achieving $\liminf_s f_s = f$ in the present situation is that if we could achieve it, then determining whether $\langle i \rangle \subseteq f$ would be a combination of $\Pi_2$ and $\Sigma_2$ questions, contradicting the fact that the question whether $\overline{U}$ is coinfinite is $\Sigma_3$-complete. Therefore, we replace the tree above by its iteration, $T = \{a, b, c\}^{<\omega}$. For this tree we *can* find a computable approximation $\{f_s\}_{s \in \omega}$ such that the following hold: $\overline{U}$ is finite iff $f$ goes through at least one $a$ (and hence cofinitely many); $U$ is infinite iff $f$ goes through at least one $a$ or $b$ (hence cofinitely many); $U$ is finite iff $f$ goes through all $c$'s. We still build three candidates $\{\widehat{U}_i\}_{i \in \{a,b,c\}}$ each emanating from a level-one node $\langle i \rangle \in T$ but in our new tree the diagram $D_b$ now no longer asserts that $\overline{U}$ is infinite, but merely that $U$ is infinite and that the cardinality of $\overline{U}$ is not determined. Similarly, $D_c$ makes no commitments about the cardinalities of *either* $U$ or $\overline{U}$. Nevertheless, the diagram $D_a$ asserts that $U$ is cofinite as before. The set $\widehat{U}_i$ is no longer enumerated by node $\langle i \rangle$ alone but rather by all the nodes $\alpha \in T$, $\alpha \supseteq \langle i \rangle$. In particular, if $f \upharpoonright n$ changes from the alternative $b$ to $a$ as $n$ increases, and empties out the state $\sigma_0 = \mathbb{N} - \widehat{U_i}$, then it is likely to be done by a very long node $\alpha = f \upharpoonright k$.

By the requirement ($\mathrm{R}_\sigma$) it suffices to show that along the true path $f$, both sides $M$ and $\widehat{M}$ have the same sequence of diagrams $D_\alpha$, for all $\alpha \subset f$. In dealing with the full automorphism method there are two new difficulties: we must consider infinitely many sets in place of one $U$; and RED plays sets on *both* sides $M$ and $\widehat{M}$, so that both RED and BLUE will be emptying out states on both sides. These difficulties are more technical in nature than conceptual, however, and are solved in the above spirit.

**The Cholak monograph.** Cholak's memoir is based on his 1991 Wisconsin dissertation, and is the first published version of the $\Delta_3^0$ method. This work is indeed impressive for someone's dissertation. The two most notable results are (i) that each non-computable computably enumerable set is automorphic with a high set (a result independently obtained by Harrington and Soare), and (ii) that, if $L^*(X)$ denotes the lattice of supersets of $X$ modulo finite sets (i.e., $\{W_e \cap \overline{X} : e \in \mathbb{N}\}$), then for any computably enumerable set $X$ and any high computably enumerable degree **a** there is a computably enumerable set $A$ of degree **a** such that $L^*(A)$ is isomorphic to $L^*(X)$. Result (ii) answers a longstanding question of Soare. I remark that it is unknown whether one can find $A$ with $L^*(X)$ effectively isomorphic to $L^*(A)$. Cholak also proves a more technical result pushing Soare's result that if $X$ is low then $L^*(X)$ is isomorphic to $\mathscr{E}^*$. (Specifically, Cholak proves that if $A$ is "semilow$_2$" with the "outer splitting property" of Maass then $L^*(A)$ is isomorphic to $\mathscr{E}^*$.) Cholak's results are based on tree versions of the extension lemma, often combined with the tree version of high permitting developed by Slaman and Shore, and independently Weinstein.

**The Harrington–Soare paper.** The Harrington–Soare paper is a tightly written account of the $\Delta_3^0$ method. The method is presented first as a general lemma which roughly says that "if a construction obeys the following rules then the resulting sets are automorphic." The idea is to present the method as a "black box." Harrington and Soare then present some elegant results including a strong extension of the theorem (i) above which says that every nontrivial computably enumerable set can be sent into some high computably enumerable degree. They define a class of sets to be "codable" if they have a certain "slow" enumeration. Then Harrington and Soare are able to prove that for any codable set $Q$ and any computably enumerable set $A$, there is a $B$ automorphic to $A$ such that $Q \leqq_T B$. Since there are codable sets of high degree (in fact each set satisfying $Q(A)$ mentioned earlier in the context of the general solution to Post's programme is codable of high degree), it follows that every nontrivial computably enumerable set is automorphic to one of high degree. Harrington and Soare also provide a partial characterization of the class of computably enumerable sets automorphic to complete sets in terms of *almost prompt* sets. These are a natural extension of the notion of sets of "promptly simple" degree, and extend earlier results of Cholak, Downey, and Stob, as well as complementing ones of Downey and Stob (*Automorphisms of the lattice of recursively enumerable sets: orbits*, ***Advances in mathematics***, vol. 92 (1992), pp. 237–265). The conclusion that Harrington and Soare draw is that properties of Aut($\mathcal{E}$) seem to depend not only on information content but on *speed of enumeration*.

**General comments and conclusions.** The $\Delta^0_3$ method has already found several extensions and applications. Since both Cholak and Harrington and Soare present the $\Delta^0_3$ method, it is perhaps worth commenting on the presentation. Cholak tries very hard to motivate general ideas behind the automorphism machinery and in that I believe that he is significantly more successful than is the Harrington–Soare paper (and significantly more successful than in his dissertation). His account is, however, full of many, many definitions and notation and is quite difficult to read in detail. The Harrington–Soare paper does little to motivate the *intuition behind* constructing automorphisms of $\mathcal{E}$ in general, but does provide a very good intuition of the *mechanics* of the $\Delta^0_3$ construction at hand. The details are carefully presented and stated quite modularly. One can see how to modify the construction to obtain a variety of applications. (One example is the "refined coding theorem" which gives a general sufficient condition as to what sort of coding is allowable.) It is not clear to me that all applications can be obtained via the "black box" approach (e.g. Cholak's theorem on $L^*(X)$), but the Harrington–Soare account is sufficiently transparent and flexible that I believe it will be a mainstay of the area for many years.

In summary, I regard these two publications as some of the most significant work since Soare's original paper. The presentations, particularly in the Harrington–Soare paper, are good and the results deep. They should clearly be understood in the context of the definability results of Harrington and Soare, as in their 1991 paper previously cited. They certainly represent the first major technical innovation since the original machinery. They are clearly a must for workers in the area and for those looking towards studying automorphism groups of other related areas. ROD DOWNEY