Strongly Bounded Turing Reducibilities
and Computably Enumerable Sets

Klaus Ambos-Spies
Ruprecht-Karls-Universität Heidelberg
Institut für Informatik

Lecture Notes
Sommersemester 2011

Preliminary Incomplete Version
Version 2 (26 April 2011)
Not for General Distribution
Preface

In this course we survey some recent work on the strongly bounded Turing reducibilities on the computably enumerable sets. Bounded Turing reducibilities are obtained from classical Turing reducibility by imposing upper bounds on the use functions (i.e., on the size of the oracle queries) of the reductions. The most popular bounded Turing reducibility which has been intensively studied in the past decades is bounded Turing (bT) reducibility - also called weak truth-table (wtt) reducibility - where the use function is bounded by a computable function.

Quite recently some reducibilities based on more strict bounds have been introduced, called computable Lipschitz reducibility and identity bounded Turing reducibility. Here the size of the queries is bounded by the input up to an additive constant or by the input itself: A set $A$ is computable-Lipschitz or cl-reducible to a set $B$ if $A$ is Turing reducible to $B$ by a Turing functional $\Phi$ where the use function $\varphi$ of $\Phi$ is bounded by the identity function up to an additive constant, i.e., $\varphi(n) \leq n + O(1)$. The special case of a cl-reduction where the use function is bounded by the identity function (i.e., where the additive constant is 0) is called an identity bounded Turing reduction (ibT-reduction). In the following we will refer to these two reducibilities as the strongly bounded Turing (sbT) reducibilities.

Computable Lipschitz reducibility (also called strong weak truth-table (sw) oder linear reducibility) was introduced by Downey, Hirschfeldt and LaForte in 2001 [DHL01, DHL04] in the context of some investigations in algorithmic randomness. Note that, for a set $A$ which is cl-reducible to a set $B$, the finite initial segment $A \upharpoonright n$ of $A$ can be computed from the corresponding initial segment $B \upharpoonright n$ of $B$ with the help of a constant number of additional bits. So, in particular, the Kolmogorov complexity of $A \upharpoonright n$ is bounded by the Kolmogorov complexity of $B \upharpoonright n$ up to an additive constant. Moreover, Downey, Hirschfeldt and LaForte have shown that, on the computably enumerable (c.e.) sets, cl-reducibility coincides with Solovay reducibility which may be viewed as a relative measure of the speed by which a real number can be effectively approximated by rational numbers. Identity bounded Turing reducibility was introduced by Soare in 2004 [So04] in the context of some applications of computability theory to some problems in differential geometry.

In the past years the partial orderings of the degrees induced by the strongly bounded Turing reducibilities have been extensively studied where the focus was led on the c.e. degrees (i.e., the degrees of the c.e. sets) and the left-computable degrees (i.e., the degrees of the reals which can be effectively approximated from below). In this course we will present the most interesting results on the c.e. strongly bounded Turing degrees. The proofs of these results combine some standard tech-
niques from computability theory with some specific methods and concepts which are quite typical for the strongly bounded Turing reducibilities.

The course assumes familiarity with the basic concepts of computability theory. Having seen some examples of the priority method will be helpful though we will shortly review the variants of this technique to be needed.

The course consists of fifteen lectures (corresponding to the 14 chapters of these notes where the first two chapters will be presented in 3 lectures). The outline of the course is as follows.

In Chapter 1 we summarize the basic concepts and facts from computability to be needed while in Chapter 2 we review the finite-injury method. In particular we prove the Friedberg-Muchnik Theorem which asserts that there are two incomparable c.e. Turing degrees. Then we discuss the permitting technique and outline the proof of the Sacks Splitting Theorem which asserts that any nonzero c.e. T-degree can be split into two lesser degrees. Chapter 3 gives a short survey on the different types of strong reducibilities. In particular we compare the nonadaptive reducibilities of truth-table type with the bounded and strongly bounded Turing reducibilities.

Then, in Chapter 4, we start our analysis of the strongly bounded Turing reducibilities. In particular, we discuss the role played by cl-reducibility in the theory of algorithmic randomness, and we explain the strong relation between ibT-reductions on the c.e. sets and the permitting technique.

Chapter 5 focusses on (bounded and computable) shifts which prove to be a very useful tool for the study of the (c.e.) degrees under the sbT-reducibilities. For instance, by some simple shift arguments, we show that, for \( r = \text{cl}, \text{ibT}, \) the partial ordering \( (\mathbb{R}_r, \leq) \) of the c.e. \( r \)-degrees has neither nonzero minimal elements nor maximal elements. So, in particular, there are no \( r \)-complete c.e. sets for the strongly bounded Turing reducibilities \( r = \text{cl}, \text{ibT} \). We also show that the bounded shifts induce automorphisms of the partial ordering \( (\mathbb{R}_{\text{ibT}}, \leq) \) of the c.e. ibT-degrees whence this structure is not rigid. Moreover, we contrast this observation by showing that no nontrivial computable shift induces an automorphism of \( (\mathbb{R}_{\text{cl}}, \leq) \). So the question whether the partial ordering of the c.e. cl-degrees is rigid remains open.

In Chapter 6 we study joins (least upper bounds) and meets (greatest lower bounds) in the partial orderings of the c.e. sbT-degrees. While, for most of the classical reducibilities \( r \), the partial ordering \( (\mathbb{R}_r, \leq) \) of the c.e. \( r \)-degrees is an upper semi-lattice (but not a lower-semi lattice), the partial ordering \( (\mathbb{R}_r, \leq) \) of the c.e. \( r \)-degrees is neither an upper semi-lattice nor a lower semi-lattice. Here we also discuss the question in what cases representatives of joins (or meets) in
one reducibility give representatives of joins (or meets) in some other reducibility. The observations made in this direction will be very useful for transferring results from one reducibility to another as shown in Chapter 7. There we prove some fundamental results on the theories of the c.e. sbT-degrees. First, by some simple observations on computable bounded shifts, we show that the elementary theory \( \text{Th}(\mathbb{R}_{\text{ibT}}, \leq) \) realizes infinitely many 2-types hence is not \( \omega \)-categorical (i.e., possesses countable nonstandard models). Then we show how some transfer lemmas (based on observations on joins and meets in the preceding chapter) can be used to transfer noncategoricity and undecidability proofs for the c.e. bounded Turing degrees to the c.e. strongly bounded Turing degrees thereby showing that, for \( r = \text{cl}, \text{ibT} \), the theory \( \text{Th}(\mathbb{R}_r, \leq) \) realizes infinitely many 1-types and is undecidable.

Chapter 8 deals with maximal pairs in the c.e. sbT-degrees. Here a pair \((a, b)\) of c.e. \( r \)-degrees is maximal if there is no c.e. \( r \)-degree \( c \) such that \( a, b \leq c \). By showing the existence of maximal pairs in the c.e. sbT-degrees, we obtain an alternative proof for the failure of the existence of complete c.e. sets under the strongly bounded Turing reducibilities. Moreover, we summarize some results from the literature on the distribution of maximal pairs.

In Chapter 9 we review the minimal pair method which is an example of an infinite-injury priority argument, and we survey the central results on sublattices of the c.e. Turing and bounded Turing degrees. Then, in Chapter 10, we apply variants of the minimal pair method in order to study meets and sublattices of the c.e. sbT-degrees. In particular we show that the 5-element nonmodular lattice \( N_5 \) can be embedded into the partial orderings \((\mathbb{R}_r, \leq)\) of the c.e. \( r \)-degrees (for \( r = \text{ibT} \) and \( r = \text{cl} \)) whence these partial lattices are not distributive. Moreover, we show that any degree in these orderings is branching (i.e., any degree is the meet of an incomparable pair of degrees). We conclude that the finite lattices which can be embedded into the c.e. strongly bounded Turing degrees and the finite lattices which can be embedded into the c.e. Turing degrees are not the same (though in both cases a complete characterization of the embeddable lattices is still not known).

In Chapter 11 we compare the theories of the c.e. ibT-degrees and the c.e. cl-degrees. In the previous chapters we have given a number of elementary differences between the partial orderings of the c.e. strongly bounded Turing degrees on the one side and the partial orderings of the c.e. degrees under the classical reducibilities on the other side. But so far we have not yet met any elementary difference between the two types of strongly bounded Turing degrees, i.e., between the partial ordering of the c.e. cl-degrees \( (\mathbb{R}_{cl}, \leq) \) and the partial ordering of the c.e. ibT-degrees \( (\mathbb{R}_{ibT}, \leq) \). Here we show that these partial orderings are not elementarily equivalent by studying some cupping properties.
One of the most fundamental classical results on the partial ordering of the c.e. Turing degrees, Sacks’ density theorem, easily carries over to the c.e. bounded Turing degrees. For the strongly bounded Turing degrees of c.e. sets, however, density fails. In Chapter 12 we sketch the proof of this result for $r = \text{ib}T$.

In Chapter 13 we discuss the question whether the meet of c.e. $r$-degrees $a$ and $b$ in the partial ordering $(R_r, \leq)$ of the c.e. $r$-degrees coincides with the meet of these degrees in the larger partial ordering $(D_r, \leq)$ of all $r$-degrees. Lachlan has shown that this is the case for Turing reducibility but, as we will show, this is in general not true for the strongly bounded Turing reducibilities though we obtain a partial positive result here too. Moreover, we discuss the corresponding question for joins.

In the final Chapter 14 we shortly discuss some further results not treated in the previous sections and state some open problems.

NOTE. These lecture notes are in a preliminary form. They are intended for the use of the participants of the course. They are not for general distribution.
6 Joins and meets in the c.e. sbT-degrees
   6.1 Pairs without joins in the c.e. bt-degrees ............................. 64
   6.2 Comparing joins and meets in the ibT- and cl-degrees ............... 64
   6.3 Comparing joins and meets in the cl- and bT-degrees ................. 66
   6.4 Pairs without meets in the c.e. bt-degrees .............................. 68

7 The theories of the c.e. sbT-degrees: noncategoricity and undecidability
   7.1 Minimal pairs, embedding distributive lattices, and nonbounding
degrees ...................................................................................... 70
   7.2 The theories of the partial orderings of the c.e. sbT-degrees are not
ω-categorical .............................................................................. 74
   7.3 The theories of the partial orderings of the c.e. sbT-degrees are
undecidable .................................................................................. 75

8 Maximal pairs in the c.e. sbT-degrees
   8.1 Existence of maximal pairs ..................................................... 83
   8.2 Variations of the maximal pair construction ............................ 86
   8.3 Maximal pairs and array noncomputability ......................... 88

9 C.e. Turing degrees and the priority method II
   9.1 A minimal pair in the c.e. Turing degrees ................................. 91
   9.2 Some applications of the minimal pair technique .................. 98

10 Nondistributivity of the c.e. sbT-degrees
   10.1 The nonmodular 5-element lattice $N_5$ is embeddable into the c.e.
 sbT-degrees. ........................................................................... 102
   10.2 Embeddings vs. 0-preserving embeddings ............................ 105
   10.3 Branching degrees ............................................................. 107
   10.4 Embeddability in the c.e. T-degrees vs. embeddability in the c.e.
 sbT-degrees ........................................................................... 110

11 Comparing the theories of the c.e. ibT-degrees and the c.e. cl-degrees 113
   11.1 Cupping and noncupping in the c.e. ibT-degrees .................... 114
   11.2 Cupping and noncupping in the c.e. cl-degrees ..................... 116

12 Nondensity of the strongly bounded c.e. degrees .......................... 121

13 Joins and meets: local vs. global structure ............................... 123
   13.1 Meets of c.e. degrees in the c.e. degrees and in the degrees in general 123
   13.2 Joins of c.e. degrees in the c.e. degrees and in the degrees in general 128
14 Further results and open problems

A Partial orderings and lattices

A.1 Partial orderings
A.2 Suborderings, embeddings, isomorphisms
A.3 Automorphisms
A.4 Joins and meets
A.5 Semi-lattices and lattices
A.6 Lattice embeddings and representations of distributive lattices
A.7 First order logic and partial orderings
In this chapter we shortly review some of the basic concepts and facts from computability to be needed later. For a more complete account see any of the standard text books, e.g., [Co04, DH10, Od89, Ro67, So87]. Most of our notation will be standard. Unexplained notation - as well as references to the origins of the concepts and results discussed here - can be found in the cited text books.

We will deal with sets of natural numbers, in the following just called sets. The set of all natural numbers \( \{0, 1, 2, \ldots \} \) is denoted by \( \mathbb{N} \) or \( \omega \). Numbers are denoted by lower case letters (\( n, x, \) etc.) while capital letters denote sets.

For numbers \( k \) and \( j \) and a set \( A \) we let \( kA + j = \{kx + j : x \in A\} \). So, in particular, \( 2\mathbb{N} \) and \( 2\mathbb{N} + 1 \) are the sets of even and odd numbers respectively. We use the symbol \( \dot{\cup} \) to denote disjoint unions. I.e., \( A = A_0 \dot{\cup} A_1 \) if \( A = A_0 \cup A_1 \) and \( A_0 \cap A_1 = \emptyset \).

Sometimes we identify a set \( A \) with its characteristic sequence or its characteristic function \( c_A \), i.e., we write \( A(x) = 1 \) if \( c_A(x) = 1 \) if \( x \in A \) and \( A(x) = 0 \) if \( c_A(x) = 0 \) if \( x \notin A \). The initial segment of \( A \) of length \( n \) is denoted by \( A \upharpoonright n \), i.e., \( A \upharpoonright n = A(0)A(1)A(2)\ldots A(n-1) = \{x \in A : x < n\} \).

A number theoretic function \( f : \mathbb{N}^n \to \mathbb{N} \) will be simply called a \((n\text{-ary})\) function; and a partial \( n \text{-ary} \) function will be a function \( \varphi \) such that the domain \( \text{dom}(\varphi) \) of \( \varphi \) is contained in \( \mathbb{N}^n \). Usually we denote (total) functions by lower case Latin letters \( f, g, h \) while we denote partial functions by lower case Greek letters \( \varphi, \psi \) etc. We write \( \varphi(x) \downarrow \) if \( \varphi(x) \) is defined (in this case we also say that \( \varphi(x) \) converges) and \( \varphi(x) \uparrow \) (\( \varphi(x) \) diverges) otherwise. We write \( \varphi(x) = \psi(y) \) if either \( \varphi(x) \uparrow \) and \( \psi(y) \uparrow \) or \( \varphi(x) \) and \( \psi(y) \) are defined and agree; and we write \( \varphi = \psi \) if \( \varphi(x) = \psi(x) \) for all numbers \( x \), i.e., if the domains of \( \varphi \) and \( \psi \) agree, \( \text{dom}(\varphi) = \text{dom}(\psi) \), and \( \varphi(x) = \psi(x) \) for all \( x \) in the domain of \( \varphi \). For an \((n+1)\text{-ary}\) partial function \( \psi \) we let \( \psi_e \) be the \( e \)th branch of \( \psi \), \( \psi_e(x) = \psi(e, x) \) \((e \in \mathbb{N}, x \in \mathbb{N}^n)\).
1.1 Computability and computable enumerability

The central concepts of computability theory are computability and computable enumerability. Here a set \( A \) is *computable* (or *decidable*) if there is an algorithm (i.e., effective procedure) which on input \( x \) tells in finitely many steps whether \( x \) is an element of \( A \) or not. Similarly, a function \( f \) is *computable*, if there is an algorithm which on input \( x \) produces the value \( f(x) \) of \( f \) at \( x \) in finitely many steps. If a function \( \psi \) is partial then we say that \( \psi \) is *partial computable* if there is an algorithm which on an input \( x \) from the domain of \( \psi \) produces \( \psi(x) \) in finitely many steps and does not give an output if \( \psi \) is not defined at \( x \) (where, in the latter case, the algorithm might run forever, i.e., may not terminate). Finally, a set \( A \) is *computably enumerable (c.e.)* if there is an algorithm which outputs the elements of \( A \) (not necessarily in order of magnitude).

Note that a set is computable iff (i.e., if and only if) its characteristic function is computable. Moreover, any computable set is c.e. In fact, a set \( A \) is computable iff the set \( A \) and its complement \( \overline{A} \) are c.e.; and a set \( A \) is computable iff \( A \) can be computably enumerated in order (of magnitude).

**Lemma 1.1** *For a set \( A \) the following are equivalent.*

(i) \( A \) is computable.

(ii) The characteristic function \( c_A \) of \( A \) is computable.

(iii) \( A \) and \( \overline{A} \) are computably enumerable.

(iv) There is an algorithm which outputs the elements of \( A \) in order.

The equivalence \((i) \iff (iii)\) above is sometimes called *Complementation Lemma*. The computably enumerable sets can be characterized in various ways.

**Lemma 1.2 (Characterization Lemma)** *For a set \( A \subseteq \mathbb{N} \) the following are equivalent.*

(i) \( A \) is c.e.

(ii) \( A \) is the range of a partial computable function.

(iii) \( A \) is empty or \( A \) is the range of a (total) computable function.

(iv) \( A \) is finite or \( A \) is the range of a computable one-to-one function.

(v) \( A \) is the domain of a partial computable (0-1-valued) function.
(vi) $A$ is the projection of a computable set $B \subseteq \mathbb{N} \times \mathbb{N}$: $x \in A \iff \exists y \ [(x, y) \in B]$.

Lemma 1.2 can be easily extended to $n$-ary sets $A \subseteq \mathbb{N}^n$. In order to do so, for the parts (ii) - (iv) of the lemma, we have to define when a partial function of type $\mathbb{N}^n \to \mathbb{N}^m$ is computable: We say that such a partial function $\varphi$ is computable if there are partial computable functions $\varphi_i : \mathbb{N}^n \to \mathbb{N}$ ($i = 1, \ldots, m$) such that $\varphi(\vec{x}) = (\varphi_1(\vec{x}), \ldots, \varphi_m(\vec{x}))$. Alternatively, we can define computability of a function of type $\varphi : \mathbb{N}^n \to \mathbb{N}^m$ by computable codings of $m$-tuples. Since we will use such coding functions in the following quite frequently we introduce some notation first.

In the following $\langle x_1, \ldots, x_n \rangle$ will denote a computable bijection $\mathbb{N}^n \to \mathbb{N}$ (for arbitrary but fixed $n \geq 2$; the chosen $n$ will be determined by the context). Moreover, we let $(\cdot)_i$ denote the corresponding $i$th projection (i.e., $(\langle x_1, \ldots, x_n \rangle)_i = x_i$). (Note that these projections are computable.) Finally, we assume that the coding function $\langle x_1, \ldots, x_n \rangle$ is chosen so that it is monotone in all arguments and satisfies $\langle x_1, \ldots, x_n \rangle \geq \max \{x_1, \ldots, x_n\}$.

Now, using these coding functions, we may alternatively say that $\varphi : \mathbb{N}^n \to \mathbb{N}^m$ is partial computable if there is a partial computable function $\hat{\varphi} : \mathbb{N}^n \to \mathbb{N}$ such that $\varphi(\vec{x}) = ((\hat{\varphi}(\vec{x}))_1, \ldots, (\hat{\varphi}(\vec{x}))_m)$ (or, equivalently, $\hat{\varphi}(\vec{x}) = \langle \varphi(\vec{x}) \rangle$).

If a c.e. set $A$ is the range of a computable function $f$ then we call $f$ an enumeration function of $A$. In the following we will often enumerate a c.e. set $A$ by specifying larger and larger finite parts of $A$. In order to define this sort of enumeration more formally we need some notation first.

For a finite set $A = \{x_0, \ldots, x_n\}$, where $x_0 < x_1 < \cdots < x_n$, the number $y = 2^x_0 + 2^x_1 + \cdots + 2^x_n$ is called the canonical index of $A$ (and the canonical index of $\emptyset$ is defined to be 0). The finite set with canonical index $y$ is denoted by $D_y$. (Note that a finite set can be effectively reconstructed from its canonical index.)

A sequence $\{D_{f(i)}\}_{i \geq 0}$ where $f$ is computable is called a strong array of finite sets. Then a computable enumeration of a set $A$ is a strong array $\{A_s\}_{s \geq 0}$ of finite sets such that

$$A_0 \subseteq A_1 \subseteq A_2 \subseteq \ldots \text{ and } A = \bigcup_{s \geq 0} A_s.$$ 

**Lemma 1.3** A set $A$ is c.e. if and only if there is a computable enumeration $\{A_s\}_{s \geq 0}$ of $A$.

**EXERCISES**

**Exercise 1.4** (a) Show that the class of the computable sets is closed under complement, union and intersection. (I.e., for computable sets $A$ and $B$, $\overline{A}$, the sets $A \cup B$ and $A \cap B$ are computable again.)

(b) Show that the class of the c.e. sets is closed under union and intersection.
Exercise 1.5 Prove the Characterization Lemma (Lemma 1.2).

Exercise 1.6 Prove Lemma 1.3.

Exercise 1.7 Give a computable function \( f \) such that \( \{D_{f(n)}\}_{n \geq 0} \) is a computable enumeration of the set \( A \) where

(i) \( A = \emptyset \)

(ii) \( A = 2\mathbb{N} \)

(iii) \( A = \text{range}(g) \) where \( g : \mathbb{N} \to \mathbb{N} \) is computable.

1.2 Turing machines

The intuitive notions of computability and computable enumerability have been formalized in various ways. Here we shortly discuss the formalization by Turing machines.

Since Turing machines operate on words we have to represent numbers by words where we use the unary representation \( 0^n \) of \( n \). Then a Turing machine computing a (partial) \( n \)-ary number theoretic function consists of a memory unit and a control unit as follows. The memory consists of a 2-sided infinite work tape divided into cells where each cell can house one letter from a given alphabet (tape alphabet). Almost all cells are empty, i.e., contain the blank symbol \( b \). The tape is accessed by a read-write-head which may scan one cell (work cell). The control of the machine is given by a program (transition function). At any time the machine is in some (program) state.

In one step the machine

(1) reads the symbol in the work cell,

(2) replaces it by a new symbol (which may be the old one),

(3) moves the head to the cell to the left (L), to the right (R) or does not move the cell (S), and

(4) updates the program state.
The action of a step is determined by the program and only depends on the current state and the current inscription of the work cell.

An instantaneous description (configuration) of the machine is determined by the current state, the position of the RW-head, and the inscription of the work tape. The machine stops if it reaches a configuration for which no transition is defined (stop configuration).

On input \( \vec{x} = (x_1, \ldots, x_n) \) the unary codes of the components of the input separated by blanks, \( 0^{x_1}b0^{x_2}b \ldots 0^{x_n} \), are written on the (otherwise empty) work tape just to the right of the work cell. If the machine stops then the input is accepted and the output is the length of the block of 0s next (right) to the work cell; otherwise the input is rejected (and no output is given).

Formally, a Turing machine \( M \) computing a (partial) \( n \)-ary number theoretic function is specified by a tuple \( M = (n, \Gamma, Q, \delta, q_0) \) where \( n \) is the arity of the computed function, \( \Gamma \supseteq \{0, b\} \) is the tape alphabet (where \( b \) is the blank symbol), \( Q \) is a finite set of states, \( \delta \) is a partial function of type \( \delta : Q \times \Gamma \rightarrow \Gamma \times \{L, R, S\} \times Q \) (the transition or program function), and \( q_0 \in Q \) is the initial state.

The (partial) function \( \Phi_M : \mathbb{N}^n \rightarrow \mathbb{N} \) computed by \( M \) is given as follows. If \( M \) stops on input \( \vec{x} \) then \( \Phi_M(\vec{x}) \) is the above described output determined by the stop configuration; and \( \Phi_M(\vec{x}) \uparrow \) if \( M \) does not stop.

For a more formal definition of \( \Phi_M \) we have to define configurations (instantaneous descriptions) \( \alpha \) of \( M \) (which are given by the current state, the current head position and the current tape inscription), the initial configuration \( \alpha(\vec{x}) \) of \( M \) on input \( \vec{x} \), the successor configuration \( s(\alpha) \) of a configuration \( \alpha \) determined by the transition function \( \delta \) (if defined; otherwise \( \alpha \) is a stop configuration). Now, if there is a number \( m \) such that \( s^m(\alpha(\vec{x})) \) is defined for all \( m' \leq m \) and \( s^m(\alpha(\vec{x})) \) is a stop configuration, then \( \alpha(\vec{x}), s(\alpha(\vec{x})), \ldots, s^m(\alpha(\vec{x})) \) is called the (converging) computation of \( M \) on input \( \vec{x} \) (of length \( m \)), and the value of \( \Phi_M(\vec{x}) \) is extracted from \( s^m(\alpha(\vec{x})) \). Otherwise \( \alpha(\vec{x}), s(\alpha(\vec{x})), s^2(\alpha(\vec{x})), \ldots \) is the (diverging or infinite) computation of \( M \) on input \( \vec{x} \) and \( \Phi_M(\vec{x}) \) is undefined.

We call \( M \) total if \( M \) stops on every input, i.e., if all computations of \( M \) are finite. Note that \( M \) is total iff \( \Phi_M \) is total.

We say that a (partial) function \( \varphi : \mathbb{N}^n \rightarrow \mathbb{N} \) is (partial) Turing computable if there is a Turing machine \( M \) which computes \( \varphi \), i.e., \( \varphi = \Phi_M \). Moreover, we say that a set \( A \) is Turing computable if its characteristic function is Turing computable and that a set \( A \) is Turing computably enumerable if \( A \) is the domain of a partial Turing computable function.

It is generally believed that Turing computability adequately formalizes computability, i.e., that a (partial) function is (partial) computable (in the intuitive
sense) iff it is (partial) Turing computable (Church-Turing Thesis). By the above mentioned relations between computability of a set and computability of its characteristic function and by the Characterization Lemma for c.e. sets, the Church-Turing Thesis also implies that a set is computable (computably enumerable) iff it is Turing computable (Turing computably enumerable).

In the following we adopt the Church-Turing Thesis and write computable (c.e.) in place of Turing computable (Turing c.e.).

EXERCISES

Exercise 1.8 Specify a Turing machine $M = (\Delta, \Sigma, Q, \delta, q_0)$ which computes the function $f(x_1, x_2) = x_1 + x_2$.

1.3 Relativized computability: Turing reducibility

By relativizing computability we can compare the degree of noncomputability of noncomputable sets and functions. Intuitively, a function $f$ is computable in a function $g$ ($f \leq_{\text{eff}} g$) if there is an algorithm $A$ computing $f$ using some hypothetical algorithm for computing $g$ as a subroutine. I.e., in the computation of $f(n)$ the algorithm $A$ may use an oracle which answers questions about the values of $g$ on any given arguments $m$. This can be formalized by the concept of an oracle machine where usually the model is limited to 0-1-valued oracle functions $g$, i.e., to oracle sets.

An oracle machine $M$ is a Turing machine with an additional read-only 1-sided oracle tape where, when working with oracle set $A$, the characteristic sequence $A(0)A(1)A(2) \ldots$ of $A$ is initially written on the oracle tape. Intuitively, in one step the machine $M$ first reads the symbol $a$ in the work cell of the work tape and the bit $i$ in the scanned cell of the oracle tape; then, depending on $a$, $i$ and its current state $q$, $M$ updates its state and the inscription of the work cell and moves the heads on the work and oracle tapes. So, here the program function $\delta$ in addition depends on the bit scanned by the R-head of the oracle tape, and $\delta$ has to record the moves of the head on the oracle tape:

$$\delta : Q \times \Gamma \times \{0, 1\} \rightarrow \Gamma \times \{L, R, S\} \times \{L, R, S\} \times Q$$
Correspondingly, the (partial) function computed by an oracle machine depends on the oracle set $A$. So an oracle machine $M$ for computing unary functions computes a functional

$$\Phi_M : \text{POWER}(\mathbb{N}) \times \mathbb{N} \to \mathbb{N},$$

called a Turing functional. Instead of $\Phi_M(A, x)$ we usually write $\Phi^A_M(x)$, i.e., let $\Phi^A_M : \mathbb{N} \to \mathbb{N}$ denote the partial function computed by $M$ with oracle $A$.

Just as in case of Turing machines, the formal semantics of an oracle machine $M$ are defined in terms of configurations and computations. Note that now a configuration in addition codes the position of the reading head of the oracle tape. Moreover, any configuration $\alpha$ of $M$ now may have two successor configurations, $s_i(\alpha)$ ($i = 0, 1$), depending on the value $i$ on the scanned cell on the oracle tape (which, of course, depends on the chosen oracle set). So, in case of an oracle machine $M$, for any input $x$ we obtain a configuration tree $CT_M(x)$, called the computation tree of $M$ on input $x$, where each oracle set $A$ selects a path of $CT_M(x)$, the computation of $M$ on input $x$ relative to $A$.

We say that $M$ is total, if for all inputs $x$ and all oracles $A$, the computation of $M$ on input $x$ relative to $A$ is finite. Note that $M$ is total iff $\Phi_M$ is total, i.e., if, for any oracle $A$, the function $\Phi^A_M$ is total. Moreover, by König’s Lemma, $M$ is total iff the computation trees $CT_M(x)$ are total for all numbers $x$.

We say that a (partial) function $\psi$ is Turing computable in a set $A$ if there is an oracle machine $M$ which computes $\psi$ with oracle set $A$ (i.e., $\psi = \Phi^A_M$) and we say that a set $B$ is Turing computable in $A$ if the characteristic function of $B$ is Turing computable in $A$. For sets $A$ and $B$ such that $A$ is computable in $B$ we also say that $A$ is Turing (T-) reducible to $B$ and write $A \leq_T B$.

The Relativized Church-Turing Thesis asserts that relative computability in the intuitive sense and relative Turing computability coincide. We adopt this thesis here and write relative computability in place of relative Turing computability. (Instead of computable in $A$ we also shortly say $A$-computable.) Note that the Relativized Church-Turing Thesis generalizes the plain Church-Turing Thesis since a set (or function) is (Turing) computable iff it is (Turing) computable in the empty set.

**EXERCISES**

**Exercise 1.9 (Reduction Lemma)** Show that for any sets $A$ and $B$ the following holds:

$$A \leq_T B \text{ & } B \text{ computable } \Rightarrow A \text{ computable}$$

$$A \leq_T B \text{ & } A \text{ noncomputable } \Rightarrow B \text{ noncomputable}$$
Exercise 1.10  (a) For any set $A \subseteq \mathbb{N}$ let 
- $\overline{A} = \mathbb{N} \setminus A$ be the complement of $A$
- $A_k = \{ x : \{ x, \ldots, x + (k - 1) \} \cap A \neq \emptyset \} \ (k \geq 1)$
- $A_{\omega} = \{ x : \{ x, \ldots, 2x \} \cap A \neq \emptyset \}$

Show that $\overline{A}$, $A_k$, $A_{\omega} \leq T A$.

(b) Let $B$ and $C$ be disjoint c.e. sets. Show that $B \leq_T B \cup C$.

Exercise 1.11  Show that Turing reducibility is invariant under finite variations. I.e., for any sets $A, B$ and $\hat{A}, \hat{B}$ 

$$A \leq_T B & \hat{A} =^* A & \hat{B} =^* B \Rightarrow \hat{A} \leq_T \hat{B}$$

holds. (Here $X =^* Y$ abbreviates that the symmetric difference $(X \setminus Y) \cup (Y \setminus X)$ of $X$ and $Y$ is finite. I.e., $X =^* Y$ iff $X(x) = Y(x)$ for almost all $x$.)

1.4 Turing degrees

Note that Turing reducibility is reflexive (i.e., $A \leq_T A$) and transitive (i.e., $A \leq_T B$ and $B \leq_T C$ imply $A \leq_T C$). So, if we call sets $A$ and $B$ Turing equivalent ($A =_T B$) if $A \leq_T B$ and $B \leq_T A$, then Turing equivalence is an equivalence relation. The equivalence classes are called Turing (T-) degrees (or degrees of unsolvability). We let 

$$deg_T(A) = \{ B : B =_T A \}$$

denote the T-degree of $A$. Moreover, we denote degrees by boldface lower case letters ($\mathbf{a}, \ldots$) and let $\mathbf{D}_T$ denote the class of all T-degrees. Note that $\mathbf{D}_T$ is partially ordered by $\leq$ where 

$$deg_T(A) \leq deg_T(B) \Leftrightarrow A \leq_T B. \quad (1.1)$$

Since a set $A$ is computable iff $A$ is T-reducible to all sets $B$, the partial ordering $(\mathbf{D}_T, \leq)$ has a least element, namely the degree 

$$\mathbf{0} = \{ A : A \text{ computable} \}$$

of the computable sets. Moreover, $(\mathbf{D}_T, \leq)$ is an upper semi-lattice (u.s.l.), i.e., for any two degrees $\mathbf{a}$ and $\mathbf{b}$ the join (least upper bound) $\mathbf{a} \vee \mathbf{b}$ is defined where $\mathbf{a} \vee \mathbf{b}$ is represented by the effective disjoint union of any representatives of $\mathbf{a}$ and $\mathbf{b}$. 

Lemma 1.12 (Turing Join Lemma) Let $A$ and $B$ be any sets and let $A \oplus B = 2A \cup 2B + 1$. Then
\[
\deg_T(A \oplus B) = \deg_T(A) \lor \deg_T(B).
\]

EXERCISES

Exercise 1.13 (a) Show that $\leq_T$ is reflexive (i.e., $A \leq_T A$) and transitive (i.e., $A \leq_T B$ and $B \leq_T C$ imply $A \leq_T C$) and that $=_{T}$ is reflexive, transitive and symmetric (i.e., $A =_{T} B$ implies $B =_{T} A$).

(b) Show that the partial ordering on the Turing degrees given by (1.1) is well defined (i.e., does not depend on the choice of the representatives of the degrees).

Exercise 1.14 Show that a set $A$ is computable iff $A$ is $T$-reducible to all sets $B$. Conclude that the partial ordering $(D_T, \leq)$ has a least element, namely the degree $0 = \{A : A$ computable$\}$ of the computable sets.

Exercise 1.15 Prove the Turing Join Lemma (Lemma 1.12).

Exercise 1.16 (a) Show that for any set $A$ there are countably infinitely many sets $B$ such that $B \leq_T A$ and countably infinitely many sets $B$ such that $B =_{T} A$.

(b) Show that there are continuum many Turing degrees.

(c) Show that for any Turing degree $a$ there are at most countably infinitely many Turing degrees $b$ such that $b \leq a$.

(d) Show that there are no maximal Turing degrees. I.e., for any Turing degree $a$ there is a Turing degree $b$ such that $a < b$. (Hint: Conclude from (b) and (c) that there is a degree $c \not\leq a$ and apply the Turing Join Lemma.)

(e) Show that any countable class of Turing degrees is bounded. I.e., for any class $\{a_n : n \geq 0\}$ of Turing degrees $a_n$ there is a Turing degree $a$ such that $a_n \leq a$. (Hint: Consider the degree of $A = \{(n, x) : x \in A_n\}$ for sets $A_n \in a_n$.)

(f) Show that for any Turing degree $a$ there are uncountably many Turing degrees $b$ such that $a \leq b$.

1.5 Universal machines and universal functions

Turing machines or oracle machines computing $n$-ary number theoretic functions can be normed by letting $\Gamma = \{0, 1, b\}$ be the tape alphabet and by letting $Q$ be
a finite initial segment of \( \mathbb{N} \) (where \( q_0 = 0 \)). This allows to effectively code the specification of such a machine \( M \) by a number \( \text{code}(M) \), called the index of \( M \), and to define a universal (Turing or oracle) machine \( U_n \) which on input \( \langle \text{code}(M), \vec{x} \rangle \) simulates \( M \) on input \( \vec{x} \).

In case of Turing machines this gives the existence of \( n \)-universal functions for the class of the partial computable functions. Namely, for \( \varphi^{(n)} = \Phi_{U_n} \) where \( U_n \) is a universal Turing machine (for the \( n \)-ary number theoretic functions), the following holds.

**Theorem 1.17 (Enumeration Theorem (for partial computable functions))** The \((n + 1)\)-ary partial function \( \varphi^{(n)} \) is partial computable and, for any \( n \)-ary partial computable function \( \psi \), there is a number \( e \) (called a (\( \varphi \)-index of \( \psi \)) such that \( \psi = \varphi^{(n)}_e \) (i.e., \( \psi(x) = \varphi^{(n)}_e(e, \vec{x}) \) for all \( \vec{x} \in \mathbb{N}^n \)).

In the following, where we usually consider unary functions, we write \( \varphi \) in place of \( \varphi^{(1)} \) and, sometimes, we write \( \{e\} \) in place of \( \varphi_e \).

Since a set is c.e. iff it is the domain of a partial computable function, the Enumeration Theorem also gives the existence of universal c.e. sets. (We state this only for \( n = 1 \).)

**Corollary 1.18 (Enumeration Theorem (for c.e. sets))** Let \( W = \text{dom}(\varphi) \). Then \( W \) is computably enumerable and \( \{W_e : e \geq 0\} \) is an enumeration of the (unary) computably enumerable sets.

We call a sequence of (partial) computable function \( \{\psi_e\}_{e \geq 0} \) uniformly (partial) computable if the function \( \psi(e, x) = \psi_e(x) \) is partial computable too. Similarly, a sequence of sets \( \{A_e\}_{e \geq 0} \) is uniformly computable (or uniformly c.e.) if the set \( A = \{(e, x) : x \in A_e\} \) is computable (c.e.). So, by the Enumeration Theorems there is a uniformly partial computable sequence of all partial computable functions and a uniformly c.e. sequence of all c.e. sets. (But there is no uniformly computable sequence of all computable sets or all computable functions.)

The universal partial computable functions \( \varphi^{(n)} \) for different arities \( n \) are related to each other as follows.

**Theorem 1.19 (s\( n \)-Theorem)** For any \( n, m \geq 1 \) there are one-to-one total computable functions \( s^{(n)}_m : \mathbb{N}^m \rightarrow \mathbb{N} \) such that

\[
\forall e, x_1, \ldots, x_m, y_1, \ldots, y_n \left[ \Phi^{(n)}_{s^{(n)}_m(e, x_1, \ldots, x_m)}(y_1, \ldots, y_n) = \varphi^{(n + m)}_e(x_1, \ldots, x_m, y_1, \ldots, y_n) \right]
\]  
(1.2)

**Proof (idea).** Given \( e \) and \( x_1, \ldots, x_m \). First, from \( e \), compute the Turing machine \( M \) for computing an \((n + m)\)-ary function with index \( e \) (i.e., \( M \) computes \( \varphi^{(n + m)}_e \)).
Then, from $M$ and $x_1, \ldots, x_m$, define a Turing machine $M'$ for computing an $n$-ary function as follows: $M'$ first writes the (representations of the) numbers $x_1, \ldots, x_m$ to the left of the (representations of the) input numbers $y_1, \ldots, y_n$ and then simulates $M$ on the thus extended input. Note that $M'$ - hence an index $e'$ of $M'$ - can be effectively defined from $M$ (hence $e$) and $x_1, \ldots, x_m$. By construction, however, 

$$
\phi_e^{(n)}(y_1, \ldots, y_n) = \phi_{e'}^{(n+m)}(x_1, \ldots, x_m, y_1, \ldots, y_n)
$$

So, for $s^m_n(e, x_1, \ldots, x_m) = e'$, (1.2) holds and, by effectivity of the construction, $s^m_n$ is computable. Finally, obviously, $s^m_n$ is one-to-one.

**Corollary 1.20** For any partial computable function $\psi^{(2)}$ there is a total one-to-one computable function $f$ such that $\psi_e = \phi_f(e)$. Similarly, for any c.e. set $V^{(2)}$ there is a total one-to-one computable function $f$ such that $V_e = W_f(e)$.

Note that, by Corollary 1.20, for any sequence of uniformly partial computable functions $\{\psi_e\}_{e \geq 0}$ there is a 1-1 computable function $f$ computing the index of $\psi_e$, i.e., satisfying $\psi_e = \phi_f(e)$ (and, similarly, for any sequence of uniformly c.e. sets $\{V_e\}_{e \geq 0}$ there is a 1-1 computable function $f$ computing the c.e. index of $V_e$, i.e., satisfying $V_e = W_f(e)$).

By taking universal oracle machines $U_n$ we similarly get universal Turing functionals hence universal partial $\Lambda$-computable functions for any set $A$. (Again we state this only for $n = 1$.) Let $\Phi = \Phi_{U_1}$ where $U_1$ is a universal oracle machine (for the 1-ary number theoretic functions).

**Theorem 1.21 (Relativized Enumeration Theorem)** For any Turing functional $\Psi$ there exists a number $e$ (called a $(\Phi)$-index of $\Psi$) such that

$$\forall A \subseteq \mathbb{N} \forall x \in \mathbb{N} [\Psi^A(x) = \Phi^A(e, x)].$$

(So, in particular, for any set $A$, the (2-ary) partial $\Lambda$-computable function $\Phi^A$ is universal for the class of the (1-ary) partial $\Lambda$-computable functions.)

Note that Theorem 1.21 implies Theorem 1.17 (for $n = 1$) since the partial computable functions are just the partial $\emptyset$-computable functions. So in the following we may assume $\varphi = \Phi^\emptyset$. Also note that the $s^m_n$-Theorem relativizes, i.e., holds for Turing functionals in place of partial computable functions too.

**EXERCISES**

**Exercise 1.22** (a) Show that the class $\text{FIN}$ of the finite sets and the class $\text{FIN}_n$ of the sets with $n$ elements ($n \geq 0$) are uniformly computable.

(b) Show that the class $\text{CE} \geq n$ of the c.e. sets with at least $n$ elements is uniformly c.e. ($n \geq 0$).
Exercise 1.23 Show that for any sequence \( \{ \psi_e \}_{e \geq 0} \) of uniformly partial computable functions there is a total computable one-to-one function \( f \) such that \( \psi_e = \varphi_{f(e)} \) for all \( e \geq 0 \). (Hint: Apply the \( s_n^m \)-Theorem.)

Exercise 1.24 Formulate and prove the Relativized \( s_n^m \)-Theorem.

1.6 Undecidability of the halting problem and the jump operator

The Enumeration Theorem 1.17 easily implies that the halting problem for Turing machines is undecidable. More formally this can be expressed as follows.

**Theorem 1.25 (Uncomputability of the Halting Problem)** The halting set

\[ K = \{ \langle e, x \rangle : \varphi_e(x) \downarrow \} = \{ \langle e, x \rangle : x \in W_e \} \]

is computably enumerable but not computable.

**Proof of Theorem 1.25.** \( K \) is c.e. since \( K \) is the domain of the partial computable function \( \psi(x) = \varphi((x)_1, (x)_2) \). In order to show that \( K \) is not computable, for a contradiction assume that \( K \) is computable. Then the function

\[ f(x) = \begin{cases} \varphi(x,x) + 1 & \text{if } \langle x,x \rangle \in K, \text{ i.e., } \varphi_e(x) \downarrow \\ 0 & \text{otherwise, i.e., if } \varphi_e(x) \uparrow. \end{cases} \]

is computable and differs from all branches \( \varphi_e \) of \( \varphi \). So \( \varphi \) is not universal for the class of the partial computable functions. Contradiction. \( \square \)

The above observations can be easily relativized. Let

\[ A' = K^A = \{ \langle e, x \rangle : \Phi^A_e(x) \downarrow \} \]

be the \( A \)-relativized halting set which is also called the *jump* of \( A \). (Note that \( K = K^0 = 0' \) by \( \varphi = \Phi^0 \).)

**Theorem 1.26** For any set \( A, A <_T A' \).
Note that the jump operation respects the ordering under Turing reducibility, i.e.,

\[ A \leq_T B \Rightarrow A' \leq_T B'. \quad (1.3) \]

So, in particular, the jump is T-degree invariant and we may define the jump \( a' \) of a degree \( a \) by letting \( a' = \text{deg}_T(A) \) for any set \( A \in a \).

The iterated jump is inductively defined by \( A^{(0)} = A \) and \( A^{(n+1)} = (A^{(n)})' \). Note that, for any set \( A \),

\[ A = A^{(0)} <_T A' = A^{(1)} <_T A'' = A^{(2)} <_T \ldots \]

A set \( A \) is called low \( n \) if \( A^{(n)} = \emptyset^{(n)} \) and \( A \) is low if \( A \) is low \( 1 \). Note that any low \( n \) set is also low \( n+1 \). The low \( 0 \) sets are just the computable sets but (as we will show in the next lecture) there are noncomputable c.e. low sets. Also note that any low \( 1 \) set is \( K \)-computable and \( K \) is not low \( n \) for any \( n \geq 0 \).

In the definition of the jump we may replace the halting set \( K \) by the diagonal halting problem \( \hat{K} = \{ e : \varphi_e(e) \downarrow \} \) (see Exercise 1.30 below). This observation is quite convenient and will be used by us below.

**EXERCISES**

**Exercise 1.27** Prove Theorem 1.26.

**Exercise 1.28** Prove (1.3).

**Exercise 1.29** Show that \( K \) is not low \( n \) (\( n \geq 0 \)).

**Exercise 1.30** The diagonal halting problem is defined by \( \hat{K} = \{ e : \varphi_e(e) \downarrow \} \) and, in relativized form, by \( \hat{K}^A = \{ e : \Phi^A_e(e) \downarrow \} \). Show that, for any set \( A \), \( K^A =_T \hat{K}^A \).

(So, in particular, \( K = K^0 =_T \hat{K}^0 = \hat{K} \).)

(Hint. It suffices to show that \( K =_T \hat{K} \) and then to relativize the proof. \( \hat{K} \leq_T K \) is straightforward. For a proof of \( K \leq_T \hat{K} \) apply the \( s^m \)-Theorem in order to show that there is a computable function \( f \) such that, for any \( e, x, y \), \( \Phi_{f(e,x)}(y) = \varphi_e(x) \).)
1.7 C.e. Turing degrees

In Section 1.4 we have already introduced Turing degrees. We now look at the c.e. Turing degrees. Here a degree \( a \) is called c.e. if it contains a c.e. set.\(^1\) In the following we let \((R_T, \leq)\) denote the partial ordering of the c.e. T-degrees (where \( \leq \) is the restriction of the partial ordering \( \leq \) defined on the set \( D_T \) of all Turing degrees).

Since the effective disjoint union of c.e. sets is c.e. again, it follows from the Turing Join Lemma that \((R_T, \leq)\) is an upper semi-lattice. Moreover, the degree 0 of the computable sets is the least degree in \((R_T, \leq)\). And, by Theorem 1.25 there is at least one more degree in \((R_T, \leq)\), namely the degree of the halting set \( K = \overline{0'} \) which in the following will be denoted by

\[ 0' = \text{deg}_T(K) = \text{deg}_T(0'). \]

In fact, \( 0' \) is the greatest element of \( R_T \) by the proposition below.

**Definition 1.31** A set \( A \) is called T-hard (for the class of c.e. sets) if all c.e. sets are T-reducible to \( A \); and \( A \) is T-complete if \( A \) is T-hard and \( A \) is c.e.

**Proposition 1.32** \( K \) is T-complete.

**Proof.** By Theorem 1.25, \( K \) is c.e. So it suffices to show that \( K \) is T-hard. I.e., given a c.e. set \( A \) it suffices to show that \( A \leq_T K \). Fix a c.e. index \( e \) of \( A \). Then, by definition of \( K \), \( x \in A \) if and only if \( \langle e, x \rangle \in K \) (for all \( x \geq 0 \)). Obviously, this implies \( A \leq_T K \). \( \square \)

Post’s question (1944) whether there are incomplete noncomputable c.e. sets, i.e., whether there is a c.e. T-degree \( a \) such that \( 0 < a < 0' \), became most influential for the further development of computability theory. Friedberg and, independently, Muchnik affirmatively answered this question in 1956 by inventing the priority method which became one of the most fundamental tools in computability theory. We will review Friedberg and Muchnik’s proof and some other examples of priority arguments in Section 2. Before we will do this, however, we have to make some technical observations on Turing reductions.

\(^1\)Note that, in general, in a c.e. degree \( a \) not all sets are c.e. In fact, the degree 0 of the computable sets is the only c.e. degree with this property. Namely, for a c.e. degree \( a \neq 0 \) and a c.e. set \( A \in a \), \( \overline{A} \not\in a \) but \( \overline{A} \) is not c.e.
1.8 Approximations and the use principle

Since the partial computable function $\varphi_e$ is defined by a Turing machine, we can approximate the value of $\varphi_e(x)$ by letting

$$\varphi_{e,s}(x) = \begin{cases} \varphi_e(x) & \text{if the computation of } \varphi_e(x) \text{ converges in } \leq s \text{ steps} \\ \uparrow & \text{otherwise.} \end{cases}$$

(Here we say that the computation of $\varphi_e(x)$ *converges* (in $s$ steps) if the Turing machine $M_e$ with index $e$ which is carrying out this computation stops (in $s$ steps), i.e., if the computation of $M_e$ on input $x$ (defined in terms of configurations) is finite (has length $s$); and we say that the computation *diverges* if the machine does not stop, i.e., if the computation of $M_e$ on input $x$ is infinite.)

Note that

$$\lim_{s \to \infty} \varphi_{e,s}(x) = \varphi_e(x)$$

(i.e., $\varphi_{e,s}(x) = \varphi_e(x)$ for all sufficiently large $s$). Moreover, the sets

$$\{ \langle e,x,s \rangle : \varphi_{e,s}(x) \downarrow \} \text{ and } \{ \langle e,x,y,s \rangle : \varphi_{e,s}(x) = y \}$$

are computable.

The Turing functionals $\Phi_e$ can be approximated in a similar way by letting

$$\Phi^A_{e,s}(x) = \begin{cases} \Phi^A_e(x) & \text{if the computation of } \Phi^A_e(x) \text{ (with oracle } A) \\ \text{converges in } \leq s \text{ steps} \\ \uparrow & \text{otherwise.} \end{cases}$$

Then, again,

$$\lim_{s \to \infty} \Phi^A_{e,s}(x) = \Phi^A_e(x)$$

and, for any (oracle) set $A$, the sets

$$\{ \langle e,x,s \rangle : \Phi^A_{e,s}(x) \downarrow \} \text{ and } \{ \langle e,x,y,s \rangle : \Phi^A_{e,s}(x) = y \}$$

are $A$-computable.

For working with Turing functionals the Use Principle which we look at next will be crucial.

The *use function* $\varphi^A(x)$ of a Turing functional $\Phi^A(x)$ with oracle $A$ is defined by

$$\varphi^A(x) = \begin{cases} y & \text{if } \Phi^A(x) \downarrow \text{ and } y \text{ is the greatest oracle query in the computation} \\ 0 & \text{if } \Phi^A(x) \downarrow \text{ and there are no oracle queries in the computation} \\ \uparrow & \text{otherwise} \end{cases}$$
and we let $\varphi^A_e(x)$ and $\varphi^A_{e,s}(x)$ be the use functions of the functionals $\Phi^A_e(x)$ and $\Phi^A_{e,s}(x)$, respectively.\footnote{Our definition of the use function is slightly nonstandard. Usually the use is defined as the least number $y$ such that all numbers used in the computation are less than $y$. Here we let the use be the least number $y$ such that all numbers used in the computation are less than or equal to $y$. Our modification of the use will simplify the definition of the strongly bounded Turing reducibilities.}

Note that for a converging computation $\Phi^A(x)$ the computation (hence the value) only depends on the finite initial segment $A \upharpoonright \varphi^A(x) + 1$ of the oracle $A$. This simple but crucial observation is commonly called the Use Principle.

**Lemma 1.33 (Use Principle)**

$$\Phi^A(x) \downarrow \& A \upharpoonright \varphi^A(x) + 1 = B \upharpoonright \varphi^A(x) + 1 \Rightarrow \Phi^B(x) = \Phi^A(x) \& \varphi^B(x) = \varphi^A(x)$$

and

$$\Phi^A_e(x) \downarrow \& A \upharpoonright \varphi^A(x) + 1 = B \upharpoonright \varphi^A(x) + 1 \Rightarrow \Phi^B_e(x) = \Phi^A_e(x) \& \varphi^B_e(x) = \varphi^A_e(x)$$

For a c.e. oracle $A$ and a computable enumeration $\{A_s\}_{s \geq 0}$ of $A$ the Use Principle implies the following

**Lemma 1.34 (Use Lemma)** Let $\{A_s\}_{s \geq 0}$ be a computable enumeration of the c.e. set $A$. Then the following hold.

1. For any numbers $e, s, x$,

$$\Phi^A_{e,s}(x) \downarrow \& A \upharpoonright \varphi^A_{e,s}(x) + 1 = A \upharpoonright \varphi^A_{e,s}(x) + 1$$

$$\Downarrow$$

$$\Phi^A_e(x) = \Phi^A_{e,s}(x) \& \varphi^A_e(x) = \varphi^A_{e,s}(x).$$

2. For any numbers $e, x$,

$$\Phi^A_e(x) \downarrow \Rightarrow \exists s_0 \forall s \geq s_0 [\Phi^A_{e,s}(x) = \Phi^A_e(x) \& \varphi^A_{e,s}(x) = \varphi^A_e(x)].$$

**Proof.** The first claim is immediate by the Use Principle. For a proof of the second claim assume that $\Phi^A_e(x) \downarrow$. Let $u = \varphi^A_e(x)$ and let $t$ be a stage such that $\Phi^A_{e,t}(x) \downarrow$. Finally, pick $s_0 \geq t$ such that $A_{s_0} \upharpoonright u + 1 = A \upharpoonright u + 1$ (such an $s_0$ exists since $\{A_s\}_{s \geq 0}$ is a monotone approximation of $A$). Then $\Phi^A_{e,s}(x) = \Phi^A_e(x)$ and $\varphi^A_{e,s}(x) = \varphi^A_e(x)$ for all $s \geq s_0$ by (the second formulation of) the Use Principle. \qed
defined in \( \leq s \) steps then we can guarantee that \( \Phi^A_e(x) = \Phi^A_{e,s}(x) \) by ensuring that no number \( \leq \varphi^A_{e,s}(x) \) enters \( A \) after stage \( s \). Conversely, if \( \Phi^A_e(x) = y \) for some number \( y \) then for all sufficiently large \( s \), \( \Phi^A_{e,s}(x) = y \) too.

The following convention will be useful.

**CONVENTION**

\[
\Phi^A_{e,s}(x) = y \Rightarrow e, x, y, \varphi^A_{e,s}(x) < s. \tag{1.4}
\]

(Since, for \( \Phi^A_{e,s}(x) = y \), \( s \) is an upper bound on the number of steps done by the machine to compute \( \Phi^A_e(x) \), the inequality \( \varphi^A_{e,s}(x) < s \) is immediate since asking an oracle query \( z \) requires \( z + 1 \) steps. The inequalities \( e, x, y < s \) can be achieved by slowing down the machine \( M_e \), e.g., by requiring that \( M_e \) reads its index and input (in unary) first before it starts the actual computation and that, after completing the computation, \( M_e \) reads its output \( y \) before stopping.)

We conclude this section by showing that the above observation on \( A \)-computable approximations of the Turing functionals \( \Phi^A \) holds uniformly. In order to make this more precise we have to introduce some notation.

For a finite binary string \( \sigma \) and a Turing functional \( \Phi \) we let

\[
\Phi^\sigma(x) = \begin{cases} 
\Phi^{S_\sigma}(x) & \text{if } \Phi^{S_\sigma}(x) \downarrow \text{ and } \varphi^{S_\sigma}(x) < |\sigma| \\ 
\uparrow & \text{otherwise}
\end{cases}
\]

where \( |\sigma| \) is the length of \( \sigma \) and \( S_\sigma = \{ x < |\sigma| : \sigma(x) = 1 \} \).

Note that, for an oracle \( A \) and an initial segment \( \sigma = A \upharpoonright y + 1 \) of the characteristic sequence of \( A \), \( \Phi^\sigma(x) = \Phi^{A\upharpoonright y+1}(x) \) is defined (and equal to \( \Phi^A(x) \)) if and only if \( \Phi^A(x) \) is defined and all oracle queries are \( \leq y \). So (by the use principle)

\[
\lim_{y \to \infty} \Phi^{A\upharpoonright y+1}(x) = \Phi^A(x).
\]

(NB In the following whenever we use an initial segment \( A \upharpoonright y + 1 \) as an oracle then this initial segment is always considered to be a string, i.e., an initial segment of the characteristic sequence of \( A \).) Now, using the above notation on strings as oracles, we obtain the uniform effective approximability of Turing functionals by the following theorem.

**Theorem 1.35 (Master Enumeration Theorem)** The sets

\[
\{ \langle e, \sigma, x, s \rangle : \Phi^\sigma_{e,s}(x) \downarrow \} \text{ and } \{ \langle e, \sigma, x, y, s \rangle : \Phi^\sigma_{e,s}(x) = y \}
\]

are computable and the set

\[
\{ \langle e, \sigma, x \rangle : \Phi^\sigma_e(x) \downarrow \}
\]

is computably enumerable.
1.9 \( \Delta_2 \)-sets

We conclude our review of basic concepts of computability theory by looking at the sets which are Turing reducible to the halting problem. As we have shown above, any c.e. set \( A \) is \( T \)-reducible to \( K \), but there are also non-c.e. sets with this property. Here we first review the characterization of the sets \( T \)-below \( K \) in terms of the arithmetical hierarchy. Then we look at the characterization in terms of computable approximations.

**Definition 1.36 (Arithmetical Hierarchy)** (a) A set \( A \) is in \( \Sigma_0 \) (\( \Pi_0 \)) if \( A \) is computable. For \( n \geq 1 \), \( A \) is in \( \Sigma_n \) if there is a computable relation \( B \subseteq \mathbb{N}^{n+1} \) such that, for any \( x \geq 0 \),

\[
x \in A \iff \exists y_1 \forall y_2 \exists y_3 \ldots Q y_n B(x, y_1, \ldots, y_n)
\]

and \( A \) is in \( \Pi_n \) if there is a computable relation \( B \subseteq \mathbb{N}^{n+1} \) such that, for any \( x \geq 0 \),

\[
x \in A \iff \forall y_1 \exists y_2 \forall y_3 \ldots \hat{Q} y_n B(x, y_1, \ldots, y_n)
\]

(where in either case the quantifiers on the right hand side are alternating).

(b) A set \( A \) is in \( \Delta_n \) if \( A \) is in \( \Sigma_n \) and \( \Pi_n \) (\( n \geq 0 \)).

If \( A \in \Sigma_n(\Pi_n, \Delta_n) \) then we also say that \( A \) is a \( \Sigma_n \) (\( \Pi_n \), \( \Delta_n \)) set. Note that by part (vi) of the Characterization Lemma (Lemma 1.2), a set \( A \) is c.e. iff \( A \in \Sigma_1 \). This observation is extended by Post’s Theorem which we state here without proof.

**Theorem 1.37 (Post’s Theorem)** (a) The following are equivalent (\( n \geq 0 \)).

(i) \( A \) is \( \Sigma_{n+1} \)

(ii) \( A \) is c.e. in a \( \Pi_n \)-set.

(iii) \( A \) is c.e. in a \( \Sigma_n \)-set.

(iv) \( A \) is c.e. in \( \emptyset^{(n)} \).

(b) \( A \) is \( \Delta_{n+1} \) if and only if \( A \leq_T \emptyset^{(n)} \).

So, in particular, a set \( A \) is \( T \)-reducible to the halting problem \( K (= \emptyset') \) if and only if \( A \) is a \( \Delta_2 \)-set. Alternatively, the \( \Delta_2 \)-sets can be described in terms of computable approximations.

A **computable approximation** of a set \( A \) is a strong array \( \{A_s\}_{s \geq 0} \) of finite sets such that \( A = \lim_s A_s \), i.e., such that, for any number \( x \), \( A(x) = \lim_s A_s(x) \) (which in turn means that there is a number \( s_x \) such that, for all \( s \geq s_x \), \( A(x) = A_s(x) \)).
Lemma 1.38 (Shoenfield’s Limit Lemma) The following are equivalent.

(i) $A$ is a $\Delta_2$-set.

(ii) There is a computable approximation of $A$.

(iii) There is a computable function $a : \mathbb{N} \times \mathbb{N} \to \{0, 1\}$ such that

$$\forall x \left[ \lim_{s} a(x, s) \text{ exists } & \lim_{s} a(x, s) \right]$$

(1.5)

Proof (Sketch). “(i) $\Rightarrow$ (ii)”: Assume that $A$ is a $\Delta_2$-set. Then, by Post’s Theorem, $A \leq_T K$. So we may fix an index $e$ such that $A = \Phi^K_e$ and we may let $\{K_s\}_{s \geq 0}$ be a computable enumeration of $K$. Then, for $A_s$ defined by

$$A_s = \{ x \leq s : \Phi^K_{es}(x) = 1 \},$$

$\{A_s\}_{s \geq 0}$ is a strong array of finite sets and, by the Use Lemma, $\{A_s\}_{s \geq 0}$ is a computable approximation of $A$.

“(ii) $\Rightarrow$ (iii)”: Given a computable approximation $\{A_s\}_{s \geq 0}$ of $A$, it suffices to let $a(x, s) = A_s(x)$.

“(iii) $\Rightarrow$ (i)”: Assume that $a$ is a computable function such that (1.5) holds. By Post’s Theorem, it suffices to show that $A \leq_T K$. Let

$$\hat{A} = \{ (x, s) : \exists t \geq s [a(x, t) \neq a(x, t + 1)] \}.$$

Note that $\hat{A}$ is a $\Sigma_1$-set, hence (by Post’s Theorem) $T$-reducible to $K$. So it suffices to show that $A \leq_T \hat{A}$. But this is immediate since, by (1.5), for any $x$ there is a least number $s_x$ such that $(x, s_x) \not\in \hat{A}$, and $A(x) = a(x, s_x)$. □

As an application of the Limit Lemma we give the following lowness criterion for c.e. sets.

Lemma 1.39 (Lowness Lemma) Let $A$ be a c.e. set and let $\{A_s\}_{s \geq 0}$ be a computable enumeration of $A$ such that, for all $e \geq 0$,

$$\exists^* s [\Phi^A_e(s) \downarrow] \Rightarrow \Phi^A_e(e) \downarrow$$

(1.6)

holds. Then $A$ is low.

Proof (Sketch). By Exercise 1.30 it suffices to show that $\hat{K}^A = \{ e : \Phi^A_e(e) \downarrow \}$ is Turing reducible to the halting set $K$. So, by Post’s Theorem and by the Limit
Lemma, it suffices to give a computable function $a : \mathbb{N} \times \mathbb{N} \to \{0, 1\}$ such that, for any number $e$, $\hat{K}^A(e) = \lim_s a(e, s)$. We claim that $a$ defined by

$$a(e, s) = \begin{cases} 1 & \text{if } \Phi^A_{e,s}(e) \downarrow \\ 0 & \text{otherwise} \end{cases}$$

has the required properties. Obviously $a$ is computable. It remains to show that $\hat{K}^A$ is the limit of $a$. In order to do so, it suffices to show that, for any given $e$,

$$\Phi^A_e(e) \downarrow \Rightarrow \exists s_e \forall s \geq s_e \Phi^A_{e,s}(e) \downarrow$$

and

$$\Phi^A_e(e) \uparrow \Rightarrow \exists s_e \forall s \geq s_e \Phi^A_{e,s}(e) \uparrow$$

hold. But the former is immediate by the Use Lemma (Lemma 1.34) while the latter is immediate by (1.6). □
C.e. Turing degrees and the priority method I

In this chapter we review some finite-injury priority arguments. The priority method was invented by Friedberg and, independently, Muchnik in 1956 in order to show that there are T-incomparable c.e. sets (i.e., incomparable c.e. T-degrees) thereby giving a positive solution to Post’s Problem (which asked for the existence of a c.e. Turing degree $a$ with $0 < a < 0'$). We start with the proof of this theorem. Next we give an alternative solution of Post’s Problem by showing that there is a noncomputable c.e. set which is low. Then we combine the proof of the Friedberg-Muchnik Theorem with the permitting method thereby showing that there is an incomparable pair of c.e. T-degrees below any given nonzero c.e. T-degree. Finally, we look at a further strengthening of this result, namely Sacks’s splitting theorem, which implies that any nonzero c.e. Turing degree is the join of two lesser c.e. T-degrees.

2.1 The Friedberg-Muchnik Theorem

**Theorem 2.1 (Friedberg [Fr56] and Muchnik [Mu56])** There are c.e. sets $A_0$ and $A_1$ such that $A_0 \upharpoonright TA_1$, i.e., $A_0 \not\leq_T A_1$ and $A_1 \not\leq_T A_0$.

Note that sets $A_0$ and $A_1$ as above can be neither computable nor complete.

**Corollary 2.2** There is a c.e. T-degree $a$ such that $0 < a < 0'$.

**Corollary 2.3** The partial ordering $(\mathcal{R}_T, \leq)$ of the c.e. Turing degrees is not a total ordering.

**Proof of Theorem 2.1.** It suffices to give computable enumerations $\{A_{0,s}\}_{s \geq 0}$ and $\{A_{1,s}\}_{s \geq 0}$ of c.e. sets $A_0$ and $A_1$ such that $A_0$ and $A_1$ are T-incomparable.

More intuitively, we effectively enumerate $A_0$ and $A_1$ in stages $s$ and let $A_{0,s}$ and $A_{1,s}$ be the finite parts of $A_0$ and $A_1$ enumerated by the end of stage $s$. In order
to make $A_0$ and $A_1$ T-incomparable it suffices to meet the requirements
\[ \mathcal{R}_{2e} : A_0 \neq \Phi^A_e \quad \text{and} \quad \mathcal{R}_{2e+1} : A_1 \neq \Phi^A_e \]
for $e \geq 0$.

Before we describe the actual construction of $A_0$ and $A_1$ we first discuss the strategy for meeting a requirement $\mathcal{R}_{2e}$. We want to meet $\mathcal{R}_{2e}$ by diagonalization, i.e., we want to pick some $x$ and ensure that $A_0(x) \neq \Phi^A_e(x)$. For this it is crucial that, by the Use Lemma (Lemma 1.34), if $\Phi^A_e(x)$ is defined (note that otherwise diagonalization is achieved automatically) then, for all sufficiently large $s$,
\[ \Phi^A_e(x) = \Phi^{A_{s,s}}_{e,s}(x) = \Phi^{A_{s,s,\prime}}(x). \]

So if (at some stage of the construction) we pick $x$ such that $x$ is not yet in $A_0$ and we commit ourselves to put $x$ into $A_0$ only for the sake of ensuring $A_0(x) \neq \Phi^A_e(x)$ then we can afford to wait for a stage $s$ such that $\Phi^{A_{s,s}}_{e,x}(x) = 0$. (Namely, if there is no such stage then, as observed above, $\Phi^A_e(x) \neq 0$. So, since we do not put $x$ into $A_0, A_0(x) = 0 \neq \Phi^A_e(x)$.) Now at a stage $s$ such that $\Phi^{A_{s,s}}_{e,x}(x) = 0$ we can diagonalize by first putting $x$ into $A_0$ and second restraining all numbers $y \leq \Phi^{A_{s,s}}_{e,s}(x)$ from $A_1$ at all stages $> s$. The latter will ensure that
\[ A_1 \upharpoonright \Phi^{A_{s,s}}_{e,x}(x) + 1 = A_{1,s} \upharpoonright \Phi^{A_{s,s,\prime}}_{e,x}(x) + 1 \]
whence, again by the Use Lemma, $\Phi^A_e(x) = \Phi^{A_{s,s,\prime}}_{e,s}(x) = 0$. So, by $x \in A_0, A_0(x) = 1 \neq \Phi^A_e(x)$.

The strategy for meeting requirement $\mathcal{R}_{2e+1}$ is symmetric where now $A_1$ plays the role of $A_0$ and $A_0$ plays the role of $A_1$. This role change, however, may lead to some conflict between the strategy for meeting an even index requirement, say $\mathcal{R}_{2e}$, and the strategy for meeting an odd index requirement, say $\mathcal{R}_{2e+1}$. Here is the scenario: Assume that we fix diagonalization candidates $x$ and $x'$ for $\mathcal{R}_{2e}$ and $\mathcal{R}_{2e+1}$, respectively. Then a computation $\Phi^{A_{s,s,\prime}}_{e,x}(x) = 0$ shows up and we put $x$ into $A_0$ and restrain all numbers $\leq \Phi^{A_{s,s}}_{e,s}(x)$ from $A_1$ in order to meet $\mathcal{R}_{2e}$. Now $\Phi^{A_{s,s,\prime}}_{e,s}(x)$ might be greater or equal to $x'$. So if, at a later stage $s', \Phi^{A_{s',s}}_{e,x}(x') = 0$ then in order to meet $\mathcal{R}_{2e+1}$ we want to put $x'$ into $A_1$ (and impose the appropriate restraint on $A_0$). But this will (or might) destroy our previously completed attack for meeting $\mathcal{R}_{2e}$. This conflict is resolved by giving the requirement with the lesser index higher priority. I.e., if $2e' + 1 < 2e$ then we put $x'$ into $A_1$ thereby destroying the previous attack on $\mathcal{R}_{2e}$ (i.e., $\mathcal{R}_{2e}$ is injured) which will force us to start another attack on $\mathcal{R}_{2e}$. On the other hand, if $2e < 2e' + 1$ then we destroy the attack on $\mathcal{R}_{2e+1}$ via $x'$ (i.e., $\mathcal{R}_{2e+1}$ becomes injured) which will force us to start another attack on $\mathcal{R}_{2e+1}$ where now the diagonalization witness is chosen big enough so that it will not be
restrained. (Note that by our convention on approximations of Turing functionals $\varphi_{e,s}(x) \leq s$ (if defined). So if the new diagonalization candidate will be bigger than $s$ then it will not be restrained.)

We now give the actual construction of $A_0$ and $A_1$. The construction proceeds in stages $s$. The parts enumerated into $A_i$ by the end of stage $s$ are denoted by $A_{i,s}$ ($i = 0, 1$). Simultaneously with $A_{0,s}$ and $A_{1,s}$, at stage $s$ we describe the status of any requirement $\mathcal{R}_{2e+i}$ ($i = 0, 1$) at stage $s$. $\mathcal{R}_{2e+i}$ may have a (unique) follower $x$ at stage $s$. Once a follower has been appointed it may exist forever (in which case we call it permanent) or it may be eventually cancelled. (In the latter case, a new follower will be appointed later.) The follower $x$ of requirement $\mathcal{R}_{2e+i}$ at stage $s$ (if any) is called realized if $\Phi^{A_{i,s}}_{e,s}(x) = 0$. Requirement $\mathcal{R}_{2e+i}$ is called satisfied at stage $s$ if it has a follower $x$ at the end of stage $s$ and $x$ is realized and $x \in A_{i,s}$. (An $\mathcal{R}_{2e+i}$-follower $x$ is just a diagonalization candidate for meeting requirement $\mathcal{R}_{2e+i}$ as described above. The fact that $x$ is realized expresses that the attack can be completed by putting $x$ into $A_i$, respectively. So the fact that $\mathcal{R}_{2e+i}$ is satisfied at stage $s$ expresses that the attack has been completed.)

THE CONSTRUCTION OF $A_0$ AND $A_1$

Stage $s = 0$. Let $A_{0,0} = A_{1,0} = \emptyset$. No requirement has a follower at stage 0.

Stage $s + 1 > 0$. Requirement $\mathcal{R}_{2e+i}$ ($i \leq 1$) requires attention at stage $s + 1$ if $2e+i \leq s$ and one of the following holds.

(i) $\mathcal{R}_{2e+i}$ does not have any follower at the end of stage $s$.

(ii) $\mathcal{R}_{2e+i}$ has the follower $x$ at the end of stage $s$, where $x$ is realized at stage $s$ and $x \notin A_{i,s}$.

If no requirement requires attention then do nothing. (I.e., since no number is put into $A_0$ or $A_1$, $A_{i,s+1} = A_{i,s}$ ($i \leq 1$). Moreover, since no follower is cancelled or appointed, any requirement $\mathcal{R}_n$ has the same follower at the end of stage $s + 1$ as at the end of stage $s$ (if any).)

Otherwise, fix $2e+i$ ($i \leq 1$) minimal such that $\mathcal{R}_{2e+i}$ requires attention. Say that $\mathcal{R}_{2e+i}$ receives attention or is active at stage $s + 1$, and distinguish the following cases. If (i) holds then appoint $s + 1$ as a follower of $\mathcal{R}_{2e+i}$. If (ii) holds then put the follower $x$ of $\mathcal{R}_{2e+i}$ into $A_i$ (i.e., $A_{i,s+1} = A_{i,s} \cup \{x\}$) and cancel all followers of requirements $\mathcal{R}_n$ with $2e+i < n$.

This completes the construction of the sets $A_0$ and $A_1$. In order to show that the constructed sets have the required properties, first observe that, by effectivity
of the construction, \( \{ A_{i,t} \}_{t \geq 0} \) is a computable enumeration of \( A_i \) whence \( A_i \) is c.e. \((i = 0, 1)\). So it only remains to show that all requirements \( \mathcal{R}_n, n \geq 0 \), are met. This is established by proving the following claim.

**Claim.** Every requirement \( \mathcal{R}_n \) requires attention at most finitely often and is met.

The proof of the claim is by induction on \( n \). Fix \( n \) and assume the claim to be correct for all \( n' < n \). By symmetry, w.l.o.g. \( n = 2e \) and, by inductive hypothesis, we may fix a stage \( s_0 > n \) such that no requirement \( \mathcal{R}_{n'} \), with \( n' < 2e \) requires attention after stage \( s_0 \). So requirement \( \mathcal{R}_{2e} \) receives attention whenever it requires attention after stage \( s_0 \) and if \( \mathcal{R}_{2e} \) has a follower at a stage \( s \geq s_0 \) this follower will be permanent. But this implies that \( \mathcal{R}_{2e} \) can require attention at most twice after stage \( s_0 \) (namely once via (i) and once via (ii)). So \( \mathcal{R}_{2e} \) requires attention at most finitely often.

It remains to show that \( \mathcal{R}_{2e} \) is met. Note that \( \mathcal{R}_{2e} \) has a follower \( x \) at the end of stage \( s_0 + 1 \) (since if there is no follower at the end of stage \( s_0 \), \( \mathcal{R}_{2e} \) requires and receives attention at stage \( s_0 + 1 \) and a follower \( x \) is appointed) and, as observed above, this follower is permanent. We claim that \( A_0(x) \neq \Phi^A_{1}(x) \). Namely, if \( x \) does not enter \( A_0 \) then \( x \) will not become realized at any stage \( \geq s_0 + 1 \). So \( \Phi^A_{e,s}(x) \neq 0 \) for all sufficiently large \( s \) whence (by the Use Lemma) \( \Phi^A_{e}(x) \neq 0 \) whereas \( A_0(x) = 0 \). On the other hand, if \( x \) enters \( A_0 \) - say at stage \( t + 1 \) - then \( x \) is realized at stage \( t \), i.e., \( \Phi^A_{e,t}(x) = 0 \). Moreover, at stage \( t + 1 \) all followers of requirements of lower priority are cancelled and no follower of a higher priority requirement can be later enumerated into \( A_1 \) since otherwise \( x \) would be cancelled. So the only numbers which may enter \( A_1 \) after stage \( t \) are followers appointed after stage \( t \) hence greater than \( t \). By our convention on the use this implies that \( A_{1,t} \upharpoonright \Phi^A_{e,t}(x) + 1 = A_1 \upharpoonright \Phi^A_{e,t}(x) + 1 \) whence, by the Use Lemma, \( \Phi^A_{e}(x) = 0 \).

This completes the proof of the claim and of the theorem.

**Remark 2.4** In the proof of the Friedberg-Muchnik Theorem, the strategy for meeting requirement \( \mathcal{R}_{2e+i} \) \((i = 0, 1)\) may want to put a number \( x \) into the set \( A_i \) and (at the same time) may want to restrain certain numbers from \( A_{1-i} \) (in order to preserve the computation \( \Phi^A_{e,s}(x) = 0 \), i.e., in order to ensure \( \Phi^A_{e,i}(x) = \Phi^A_{e,s}(x) = 0 \)). In such a situation we say that requirement \( \mathcal{R}_{2e+i} \) (or the strategy for meeting \( \mathcal{R}_{2e+i} \)) is \( A_i \)-**positive** and \( A_{1-i} \)-**negative**. Conflicts among requirements arise if, for some of the sets \( A \) under construction, some requirements are \( A \)-positive while other requirements are \( A \)-negative. In the proof of the Friedberg-Muchnik Theorem such conflicts arise among the even index requirements and the odd index requirements.
EXERCISES

Exercise 2.5 (Sacks) A finite sequence \(a_1, \ldots, a_n\) \((n \geq 1)\) of elements of an upper semi-lattice \((P, \leq)\) is independent if

\[
\forall i \in \{1, \ldots, n\} \ [a_i \not\leq \bigvee_{j \in \{1, \ldots, n\} \setminus \{i\}} a_j]
\]

Extend the Friedberg-Muchnik Theorem by showing that, for any \(n \geq 2\), there is an independent sequence \(a_1, \ldots, a_n\) of c.e. Turing degrees.

2.2 A low noncomputable c.e. set

In this section we give an alternative solution to Post’s Problem by constructing a noncomputable c.e. set \(A\) which is low.

Theorem 2.6 (Lowness Theorem; see e.g. Soare [So87], Theorem VII.1.1) There is a noncomputable c.e. set \(A\) which is low.

Note that Theorem 2.6 implies Corollary 2.2: Namely, for a c.e. set \(A\) as in Theorem 2.6 and for \(a = \deg_A(A)\), \(0 < a \) since \(A\) is noncomputable and \(a < 0'\) since \(A\) is low (see Exercise 1.29).

Proof of Theorem 2.6. By a finite-injury argument we effectively enumerate a c.e. set \(A\) with the required properties in stages. We let \(A_s\) be the finite part of \(A\) enumerated by the end of stage \(s\). (So, formally, \(\{A_s\}_{s \geq 0}\) will be a computable enumeration of \(A\), thereby guaranteeing that \(A\) is c.e.)

The required properties of the c.e. set \(A\), namely to be noncomputable and low, are split into infinitely many finitary requirements: In order to make \(A\) noncomputable it suffices to meet the requirements

\[ P_e : A \neq \phi_e \]

for \(e \geq 0\). In order to make \(A\) low, by the Lowness Lemma (Lemma 1.39), it suffices to meet the requirements

\[ Q_e : \exists^\infty s [\Phi_{e,s}^A(e) \downarrow] \Rightarrow \Phi_e^A(e) \downarrow \]

for \(e \geq 0\).
The basic strategy for meeting requirement $\mathcal{P}_e$ is a simplified version of the Friedberg-Muchnik strategy. We pick a previously unused number $x$ (called a follower of $\mathcal{P}_e$), wait for for a stage $s$ such that $\varphi_{e,s}(x) = 0$, and put $x$ into $A$ at stage $s + 1$ thereby guaranteeing that $A(x) = 1 \neq 0 = \varphi_{e,s}(x) = \varphi_e(x)$. (Here the computation $\varphi_{e,s}(x) = 0$ does not use an oracle. So $\varphi_{e,s}(x) = 0$ immediately implies $\varphi_e(x) = 0$. Hence here we do not have to impose any restraints on an oracle set in order to preserve the computation.) Note that if there is no stage $s$ with $\varphi_{e,s}(x) = 0$ then $\varphi_e(x) \neq 0$ and $x$ is never put into $A$. So $A(x) \neq \varphi_e(x)$ in this case too.

The basic strategy for meeting requirement $\mathcal{Q}_e$ is as follows. If there is a stage $s$ such that $\Phi_{e,s}^1(e) \downarrow$ then, by restraining all numbers $\leq \varphi_{e,s}^1(x)$ (i.e., all numbers used in the computation) from $A$ after stage $s$, we preserve this computation, i.e., guarantee (by the Use Lemma) that $\Phi_e^1(e) = \Phi_{e,s}^1(e) \downarrow$.

Note that the noncomputability requirements $\mathcal{P}_e$ are (A-)positive while the lowness requirements $\mathcal{Q}_e$ are (A-)negative. The conflicts which may arise between a noncomputability requirement and a lowness requirement are resolved by assigning priorities to the requirements, $\mathcal{R}_{2e} = \mathcal{P}_e$ and $\mathcal{R}_{2e+1} = \mathcal{Q}_e$, and by giving requirements with lesser index higher priority. So a nonlowness requirement $\mathcal{R}_{2e+1} = \mathcal{Q}_e$ which wants to preserve a computation by imposing a (finite) restraint, may cancel all followers of noncomputability requirements $\mathcal{P}_{e'}$ for $e' > e$ which may interfere with this restraint. Correspondingly, a noncomputability requirement $\mathcal{P}_{e'}$ with $e' < e$ may ignore the restraint imposed by $\mathcal{Q}_e$ and it may injure $\mathcal{Q}_e$ by enumerating a follower $x$ restrained by $\mathcal{Q}_e$ into $A$. This procedure will guarantee that any requirement will be injured only finitely often. So, if after an injury we start with a new instance of the basic strategy, eventually there will be an instance of the basic strategy which will never be injured and which will succeed in meeting the requirement.

We now give the actual construction of $A$ where $A_e$ denotes the finite part enumerated into $A$ by the end of stage $s$. Simultaneously with $A_e$, at stage $s$ we describe the status of any requirement $\mathcal{R}_{2e+1}$ ($i = 0, 1$) at stage $s$. Requirement $\mathcal{R}_{2e} = \mathcal{P}_e$ may have a (unique) follower $x$ at stage $s$ (which may be permanent - i.e., may exist for ever once it is appointed - or may eventually be cancelled). Follower $x$ is realized at stage $s$ if $\varphi_{e,s} = 0$, and requirement $\mathcal{R}_{2e+1}$ is satisfied at stage $s$ if it has a follower $x$ at the end of stage $s$, $x$ is realized, and $x \in A_e$. Requirement $\mathcal{R}_{2e+1} = \mathcal{Q}_e$ has a restraint $r(e,s)$ attached to it at the end of any stage $s$ where $r(e,s) > 0$ indicates that $\mathcal{R}_{2e+1}$ is attempting to preserve the converging computation $\Phi_{e,s}^1(e)$ by restraining numbers $< r(e,s)$ from $A$.
2.2. A low noncomputable c.e. set

Stage $s = 0$. Let $A_0 = \emptyset$. No requirement $P_e$ has a follower at stage 0 and $r(e, 0) = 0$ for all $e \geq 0$. Declare all requirements to be initialized at stage 0.

Stage $s + 1 > 0$. Requirement $R_{2e} = P_e$ requires attention at stage $s + 1$ if $2e \leq s$ and one of the following holds.

(i) $R_{2e}$ does not have any follower at the end of stage $s$.

(ii) $R_{2e}$ has the follower $x$ at the end of stage $s$, where $x$ is realized at stage $s$ and $x \not\in A_s$.

Requirement $R_{2e+1} = Q_e$ requires attention at stage $s + 1$ if $2e + 1 \leq s$, $\Phi_{A_s}^e(x) \downarrow$, and $r(e, s) = 0$.

If no requirement requires attention then do nothing.

Otherwise, fix $2e + i$ ($i \leq 1$) minimal such that $R_{2e+i}$ requires attention. Say that $R_{2e+i}$ receives attention or is active at stage $s + 1$, and distinguish the following cases.

Case 1: $i = 0$. If (i) holds then appoint $s + 1$ as a follower of $R_{2e+i}$. If (ii) holds then put the follower $x$ of $R_{2e+i}$ into $A$ (i.e., $A_{s+1} = A_s \cup \{x\}$). In either case set $r(e', s + 1) = 0$ for all $e' \geq e$ and declare $Q_e$ to be initialized (injured).

Case 2: $i = 1$. Then let $r(e, s + 1) = s + 1$. Moreover, cancel all followers of requirements $P_{e'}$ where $e' > e$ and declare $P_{e'}$ to be initialized (injured).

(Note that any parameters from stage $s$ persist at stage $s + 1$ unless explicitly stated otherwise. So $A_{s+1} = A_s$ unless a follower is enumerated into $A$ at stage $s + 1$; a noncomputability requirement $P_e$ which has follower $x$ at the end of stage $s$ will have follower $x$ at the end of stage $s + 1$ unless $x$ is cancelled at stage $s + 1$; a noncomputability requirement $P_e$ which has no follower at the end of stage $s$ will have no follower at the end of stage $s + 1$ unless follower $s + 1$ is appointed at stage $s + 1$; and $r(e, s + 1) = r(e, s)$ unless the lowness requirement $Q_e$ or a noncomputability requirement $P_{e'}$ with $e' \leq e$ becomes active at stage $s + 1$ in which case $r(e, s + 1)$ is set to $s + 1$ and 0, respectively.)

This completes the construction of the sets $A$. Note that, by effectivity of the construction, $\{A_s\}_{s \geq 0}$ is a computable enumeration of $A$ whence $A$ is c.e. So it only remains to show that all requirements $R_n$, $n \geq 0$, are met. This is established by proving the following claims.
Claim 1. Every requirement $\mathcal{R}_n$ requires attention at most finitely often.

Proof of Claim 1. The proof is by induction on $n$. Fix $n$ and assume the claim to be correct for all $n' < n$. Fix a stage $s_0 > n$ such that no requirement $\mathcal{R}_{n'}$, with $n' < n$ requires attention after stage $s_0$. Then requirement $\mathcal{R}_n$ receives attention whenever it requires attention after stage $s_0$ and $\mathcal{R}_n$ is not initialized (injured) after stage $s_0$. So, for $\mathcal{R}_n = \mathcal{P}_e$, any follower existing at a stage $s \geq s_0$ will be permanent. But this implies that $\mathcal{R}_n = \mathcal{P}_e$ can require attention at most twice after stage $s_0$ (namely once via (i) and once via (ii)). Similarly, for $\mathcal{R}_n = \mathcal{Q}_e$, $\mathcal{R}_n$ can require attention after stage $s_0$ at most once. Namely, if $\mathcal{Q}_e$ requires hence receives attention at stage $s+1 > s_0$, then $r(e, s+1) = s+1$. Since $\mathcal{Q}_e$ is not initialized after stage $s_0$ it follows that $r(e, s') = r(e, s+1) > 0$ for all $s' \geq s+1$. So $\mathcal{Q}_e$ will not require attention at any stage $s' + 1 > s+1$.

Claim 2. If requirement $\mathcal{P}_e$ is satisfied at some stage then $\mathcal{P}_e$ is met.

Proof of Claim 2. If requirement $\mathcal{P}_e$ is satisfied at stage $s$ then, by definition, there is a realized follower $x$ at the end of stage $s$ such that $x \in A_e$. So $A(x) = A_e(x) = 1 \neq 0 = \varphi_{e,s}(x) = \varphi_e(x)$. So $\mathcal{P}_e$ is met.

Claim 3. Every requirement $\mathcal{P}_e$ is met.

Proof of Claim 3. By Claim 1 fix a stage $s_0 > 2e$ such that neither $\mathcal{P}_e$ nor a higher priority requirement requires attention after stage $s_0$. Then $\mathcal{P}_e$ has a follower $x$ at the end of stage $s_0$ (since otherwise $\mathcal{P}_e$ would require attention via clause (i) at stage $s_0 + 1$ contrary to choice of $s_0$) and $x$ will be permanent (since, by choice of $s_0$, $\mathcal{P}_e$ is not initialized after stage $s_0$).

First assume that $x$ is in $A$. Since a number can become a follower of at most one requirement, it follows that there is a stage $s+1 \leq s_0$ at which $\mathcal{P}_e$ requires and receives attention via clause (ii). So $\mathcal{P}_e$ is satisfied at the end of stage $s+1$. It follows by Claim 2 that $\mathcal{P}_e$.

Finally assume that $x$ is not in $A$. Then $\varphi_{e,s}(x) \neq 0$ for all stages $s \geq s_0$ since otherwise $\mathcal{P}_e$ will require attention at stage $s+1$ contrary to choice of $s_0$. So $A(x) = 0 \neq \varphi_e(x)$. Hence $\mathcal{P}_e$ is met in this case too.

Claim 4. Every requirement $\mathcal{Q}_e$ is met.

Proof of Claim 4. By Claim 1 fix a stage $s_0 > 2e + 1$ such that neither $\mathcal{Q}_e$ nor a higher priority requirement requires attention after stage $s_0$. Then $r(e, s) = r(e, s_0)$ for all $s \geq s_0$.

First assume that $r(e, s_0) = 0$. Then $\Phi_{e,s_0}(e) \uparrow$ for all $s \geq s_0$ since otherwise $\mathcal{Q}_e$ will require attention at stage $s+1$ contrary to choice of $s_0$. So $\mathcal{Q}_e$ is met trivially.

Finally assume that $r(e, s_0) > 0$, say $r(e, s_0) = s+1$. Then, by construction, $s < s_0$, $\mathcal{Q}_e$ requires and receives attention at stage $s+1$, and, by assumption and
by choice of $s_0$, $r(e, s') = s + 1$ for all $s' \geq s$ (so $Q_e$ is not initialized (i.e., not injured) after stage $s$). It follows that $\Phi_{e, s}^A(s) \downarrow$ and, by our convention on the use, that $\Phi_{e, s}^A(x) < s$. Moreover, (by $Q_e$ receiving attention at stage $s + 1$) all followers of requirements $P_e'$ with $e < e'$ are cancelled at stage $s + 1$ and (by $Q_e$ not being initialized after stage $s + 1$) no follower of a requirement $P_e$ with $e' \leq e$ will enter $A$ after stage $s + 1$. Since no number is put into $A$ at stage $s + 1$ it follows that only followers $x'$ appointed after stage $s + 1$, hence followers $x' > s + 1$, can enter $A$ after stage $s$. So, in particular, $A_s \upharpoonright \Phi_{e, s}^A(x) + 1 = A \upharpoonright \Phi_{e, s}^A(x) + 1$. It follows with the Use Lemma that $\Phi_{e}^A(e) = \Phi_{e, s}^A(e) \downarrow$. So $Q_e$ is met in this case too.

Note that by Claims 3 and 4 all requirements are met. This completes the proof of the theorem.

2.3 Permitting

The proof of the Friedberg-Muchnik Theorem can be combined with the permitting technique in order to prove the following extension which in particular shows that there are no minimal nonzero T-degrees.

**Theorem 2.7 (Muchnik)** Let $C$ be a noncomputable c.e. set. There are T-incomparable c.e. sets $A_0$ and $A_1$ such that $A_0 \leq_T C$ and $A_1 \leq_T C$.

Let $C$ be a c.e. set and let $\{C_s\}_{s \geq 0}$ be a computable enumeration of $C$. We say that a c.e. set $A$ is T-reducible to $C$ (via $\{C_s\}_{s \geq 0}$) by permitting if there is a computable enumeration $\{A_s\}_{s \geq 0}$ of $A$ such that for any stage $s$ and any number $x$ entering $A$ at stage $s + 1$ (i.e., $x \in A_{s+1} \setminus A_s$) there is a number $y \leq x$ entering $C$ at stage $s + 1$ (i.e., $y \in C_{s+1} \setminus C_s$). (We say that $y$ permits $x$ to enter $A$ or that $C$ permits $x$ to enter $A$ via $y$.)

Note that, for $A$ and $C$ as above, $A$ is $C$-computable by the following procedure. In order to decide whether $x \in A$, using $C$ as an oracle find a stage $s$ such that $C_s \upharpoonright x + 1 = C \upharpoonright x + 1$. Then $x \in A$ iff $x \in A_s$.

**Proof of Theorem 2.7.** Fix a computable enumeration $\{C_s\}_{s \geq 0}$ of $C$. As in the proof of Theorem 2.1 we give computable enumerations $\{A_{i,s}\}_{s \geq 0}$ of c.e. sets $A_i$ ($i = 0, 1$) such that $A_0$ and $A_1$ meet the requirements

$$\mathcal{R}_{2e+1} : A_i \neq \Phi_{e}^{A_{i-1}}$$
for $e \geq 0$ and $i = 0, 1$. For meeting these requirements we use the strategy explained in the proof of Theorem 2.1, called the Friedberg-Muchnik strategy in the following. In addition to meeting the requirements $\mathcal{R}_{2e+i}$ we ensure that all numbers entering $A_0$ or $A_1$ are permitted by $C$ (w.r.t. the given enumeration $\{C_s\}_{s \geq 0}$ of $C$) thereby guaranteeing that $A_0, A_1 \leq_T C$.

The latter, however, may interfere with the Friedberg-Muchnik strategy as follows. If we want to put a follower $x$ of requirement $\mathcal{R}_{2e+i}$ into $A_i$ at some stage $s + 1$ (since $x$ is realized at stage $s$) then we are only permitted to do so if there is a number $y \leq x$ in $C_{s+1} \setminus C_s$. Since a follower may never be permitted, in such a situation we appoint another follower of $\mathcal{R}_{2e+i}$ (but we also cancel all followers of lower priority requirements thereby guaranteeing that the follower $x$ will be permanently realized from now on unless it becomes cancelled by some higher priority requirement). We then will argue that eventually some follower of $\mathcal{R}_{2e+i}$ will either never become realized or be permitted by $C$ since otherwise the set $C$ will be computable.

Using the terminology of the proof of Theorem 2.1, stage $s + 1$ of the construction is as follows.

Requirement $\mathcal{R}_{2e+i}$ ($i \leq 1$) requires attention at stage $s + 1$ if $2e + i \leq s$ and one of the following holds.

(i) There is no follower of $\mathcal{R}_{2e+i}$ or all followers $x$ of $\mathcal{R}_{2e+i}$ are realized and none of these followers is in $A_i$ and none of these followers is permitted by $C$.

(ii) $\mathcal{R}_{2e+i}$ has a follower $x$ at the end of stage $s$, where $x$ is realized at stage $s$, $x$ is permitted by $C$ (i.e., there is a number $y \leq x$ in $C_{s+1} \setminus C_s$), and $x \notin A_i$.

If no requirement requires attention then do nothing. Otherwise, fix $2e + i$ ($i \leq 1$) minimal such that $\mathcal{R}_{2e+i}$ requires attention. Say that $\mathcal{R}_{2e+i}$ receives attention or is active at stage $s + 1$, cancel all followers of requirements $\mathcal{R}_n$ with $2e + i < n$, and distinguish the following cases. If (i) holds then appoint $s + 1$ as a follower of $\mathcal{R}_{2e+i}$. If (ii) holds then let $x$ be the least follower as in (ii) and put $x$ into $A_i$.

To show that the construction is correct we first observe that the sets $A_0$ and $A_1$ are c.e. and T-reducible to $C$ by permitting. So it suffices to show that all requirements $\mathcal{R}_n$ require attention at most finitely often and are met. As in the proof of Theorem 2.1 this is done by induction on $n$.

Fix $n$ and assume the claim to be correct for all $n' < n$. By symmetry, w.l.o.g. $n = 2e$ and, by inductive hypothesis, we may fix a stage $s_0 > n$ such that no requirement $\mathcal{R}_{n'}$, with $n' < 2e$ requires attention after stage $s_0$. So requirement $\mathcal{R}_{2e}$
receives attention whenever it requires attention after stage $s_0$ and any follower $x$ of $\mathcal{R}_{2e}$ existing at a stage $s \geq s_0$ will be permanent.

Now in order to show that $\mathcal{R}_{2e}$ requires attention only finitely often, for a contradiction assume that $\mathcal{R}_{2e}$ requires attention infinitely often. Then $\mathcal{R}_{2e}$ receives attention infinitely often and an infinite increasing sequence $x_1 < x_2 < x_3 \ldots$ of followers are appointed at stages $s_1 + 1 < s_2 + 1 < s_3 + 1 < \ldots$ after stage $s_0$, all of them permanent and none of them entering $A_0$. Moreover, when follower $x_{k+1}$ is appointed at stage $s_{k+1} + 1$ then follower $x_k$ is realized at stage $s_{k+1}$ and all followers of lower priority requirements are cancelled at stage $s_{k+1} + 1$ whence $x_k$ remains permanently realized from stage $s_{k+1} + 1$. So $x_k$ will not be permitted by $C$ after stage $s_{k+1} + 1$. It follows that

$$C_{s_{k+1} + 1} \upharpoonright x_k = C \upharpoonright x_k.$$ 

But since the sequences $\{x_k\}_{k \geq 0}$ and $\{s_k\}_{k \geq 0}$ are strictly increasing and computable, this implies that $C$ is computable contrary to choice of $C$.

It remains to show that $\mathcal{R}_{2e}$ is met. As in the proof of the Friedberg-Muchnik Theorem one can argue that if there is a permanently unrealized follower or a permanent follower in $A_0$ then $\mathcal{R}_{2e}$ is met. Then one argues that such a follower must exist. Namely, otherwise infinitely many followers will be appointed after stage $s_0$, all eventually permanently realized but none entering $A_0$. So, in particular $\mathcal{R}_{2e}$ receives (hence requires) attention infinitely often. But this contradicts the first part of the claim.

\[\square\]

**EXERCISES**

**Exercise 2.8** Show that, for any c.e. degree $b > 0$ and any $n \geq 2$, there is an independent sequence $a_1, \ldots, a_n$ of c.e. Turing degrees such that $a_1, \ldots, a_n < b$. (Compare with Exercise 2.5.)

### 2.4 Sacks’s splitting theorem

Theorem 2.7 shows that any nonzero c.e. T-degree bounds an incomparable pair of c.e. T-degrees. Sacks (1963) has strengthened this by showing that any nonzero c.e.
T-degree \( a \) splits, i.e., is the join of an incomparable pair of c.e. T-degrees \( a_0 \) and \( a_1 \). Moreover, the degrees \( a_0 \) and \( a_1 \) can be chosen to be low thereby subsuming the Lowness Theorem (Theorem 2.6) too.

Sacks proved this result by giving a similar result for set splittings and by observing that a splitting of a c.e. set into c.e. sets induces a degree splitting.

**Lemma 2.9 (T-Splitting Lemma)** Let \( A, A_0, A_1 \) be c.e. sets such that \( A = A_0 \cup A_1 \). Then

\[
\text{deg}_T(A) = \text{deg}_T(A_0) \lor \text{deg}_T(A_1).
\]

**PROOF.** We have to show that

\[
A_i \leq_T A \quad (i = 0, 1)
\]  

and

\[
\forall B \text{ c.e.} \ [A_0 \leq_T B \& A_1 \leq_T B \Rightarrow A \leq_T B]
\]  

For a proof of the former, w.l.o.g. assume \( i = 0 \). Then, given \( x, A_0(x) \) is computed from \( A \) as follows. First \( A \)-computably decide whether \( x \in A \). If not then, obviously, \( x \) is not in \( A_0 \). If \( x \in A \) then simultaneously enumerate \( A_0 \) and \( A_1 \) until \( x \) shows up in \( A_0 \) or \( A_1 \) (by \( x \in A = A_0 \cup A_1 \) exactly one of these must happen). Then \( x \in A_0 \) iff \( x \) shows up in \( A_0 \).

For a proof of (2.2) fix \( B \) and assume that \( A_0 \leq_T B \) and \( A_1 \leq_T B \). Then, for given \( x, A(x) \) can be computed from \( B \) as follows. By assumption, \( A_0(x) \) and \( A_1(x) \) can be computed from \( B \) and \( A(x) = \max\{A_0(x), A_1(x)\} \). \( \Box \)

**Theorem 2.10 (Sacks’s Splitting Theorem (for sets))** Let \( A \) and \( B \) be c.e. sets such that \( B \) is not computable. There is a splitting of \( A \) into disjoint c.e. sets \( A_0 \) and \( A_1 \), \( A = A_0 \cup A_1 \), such that

(i) \( A_0 \) and \( A_1 \) are low and

(ii) \( B \not\leq_T A_0 \) and \( B \not\leq_T A_1 \).

Before we prove Theorem 2.10 we give some corollaries.

**Corollary 2.11 (Sacks’s Splitting Theorem (for degrees))** For any c.e. T-degrees \( a \) and \( b \) such that \( b > 0 \) there are low c.e. degrees \( a_i \) such that \( a = a_0 \lor a_1 \) and \( b \not\leq a_i \) (\( i = 0, 1 \)).

**PROOF.** Immediate by Theorem 2.10 and Lemma 2.9. \( \Box \)

**Corollary 2.12** For any c.e. T-degree \( a > 0 \) there are low c.e. degrees \( a_i \) such that \( a_i < a \) and \( a = a_0 \lor a_1 \) (\( i = 0, 1 \)).
Note that, for \( a_0 \) and \( a_1 \) as in Corollary 2.12, \( a_0 \) and \( a_1 \) are incomparable. So Corollary 2.12 implies Theorem 2.7 and Theorem 2.6.

**Proof.** This follows from Corollary 2.11 by letting \( a = b \). \( \square \)

**Corollary 2.13** For any c.e. \( T \)-degree \( b \) such that \( 0 < b < 0' \) there is a low c.e. degree \( c \) such that \( b \mid c \).

**Proof.** Apply Corollary 2.11 to \( a = 0' \) and \( b \). We claim that \( a_0 \) or \( a_1 \) is incomparable with \( b \). For a contradiction, assume not. Then, by \( b \not\leq a_i, a_i \leq b \) \((i = 0, 1)\). It follows that \( a_0 \vee a_1 \leq b \). But, by \( 0' = a_0 \vee a_1 \), this implies \( 0' \leq b \) contrary to choice of \( b \). \( \square \)

**Proof of Theorem 2.10.** Note that, for computable \( A \), the claim is trivial. So w.l.o.g. we may assume that \( A \) is infinite and we may fix a computable one-to-one function \( a \) enumerating \( A \). Let \( \{A_i\}_{s \geq 0} \) be the computable enumeration of \( A \) induced by \( a \), i.e., \( A_i = \{a(0), a(1), \ldots, a(s-1)\} \), and let \( \{B_i\}_{s \geq 0} \) be any computable enumeration of \( B \).

The desired sets \( A_i \) are constructed in stages where we let \( A_i, s \) denote the finite part enumerated into \( A_i \) by the end of stage \( s \). By effectivity of the construction, \( \{A_i, s\}_{s \geq 0} \) will be a computable enumeration of \( A_i \). So the sets \( A_i \) will be c.e.

In order to guarantee that \( A_0 \) and \( A_1 \) split \( A \), at stage \( s + 1 \) of the construction we put \( a(s) \) either into \( A_0 \) or into \( A_1 \) (and no other number will enter \( A_0 \) or \( A_1 \) at stage \( s + 1 \)):

\[
\exists i \leq 1 \left[ A_{i,s+1} = A_{i,s} \cup \{a(s)\} \land A_{1-i,s+1} = A_{1-i,s} \right]
\]

So, by letting \( A_{i,0} = \emptyset \), \( A_s = A_{0,s} \cup A_{1,s} \) for all \( s \geq 0 \) and \( A = A_0 \cup A_1 \).

In order to make sure that the sets \( A_i \) are low and that \( B \) is not Turing reducible to \( A_i \) it suffices to meet the **lowness requirements**

\[
Q_{2e+i} : \exists s \left[ \Phi_{e,s}^{A_{e,i}}(e) \downarrow \right] \Rightarrow \Phi_{e}^{A_i}(e) \downarrow
\]

and the **diagonalization requirements**

\[
D_{2e+i} : B \neq \Phi_{e}^{A_i}
\]

for \( e \geq 0 \) and \( i \leq 0 \). We order these requirements by letting

\[
\mathcal{R}_{4e+i} = Q_{2e+i} \land \mathcal{R}_{4e+2+i} = D_{2e+i}.
\]
The strategy for meeting the lowness requirements is the strategy used in the proof of the Lowness Theorem (Theorem 2.6). Let

\[ r(4e + i, s) = r^Q(2e + i) = \begin{cases} 0 & \text{if } \Phi^{A_{e,s}}(e) \uparrow \\ \Phi^{A_{e,s}}(e) + 1 & \text{if } \Phi^{A_{e,s}}(e) \downarrow. \end{cases} \]

Then, in order to meet requirement \( R_{4e+i} = Q_{2e+i} \), it suffices to guarantee that for all sufficiently large \( s \), \( a(s) \) is not enumerated into \( A_i \) if \( a(s) < r^Q(2e + i, s) \) (thereby guaranteeing that \( A_{i,s+1} \upharpoonright r^Q(2e + i, s) = A_{i,s} \upharpoonright rQ(2e + i, s) \)):

**Claim 1. Assume that there is a stage \( s_0 \) such that \( r^Q(2e + i, s_0) > 0 \) and**

\[ \forall s \geq s_0 \ [A_{i,s+1} \upharpoonright r^Q(2e + i, s) = A_{i,s} \upharpoonright r^Q(2e + i, s)]. \]

**Then, for all \( s \geq s_0 \),**

\[ r^Q(2e + i, s) = \Phi^{A_{e,s}}(e) + 1 = \Phi^{A_{e,s_0}}(e) + 1 = r^Q(2e + i, s_0) \]

**and**

\[ \Phi^A(e) = \Phi^{A_{e,s}}(e) = \Phi^{A_{e,s_0}}(e) \downarrow. \]

Note that Claim 1 is proven as the corresponding observations in the proof of Theorem 2.6. Also note that the strategy for meeting requirement \( R_{4e+i} = Q_{2e+i} \) is \( A_i \)-negative since it wants to prevent \( A_i \) to change below \( r^Q(2e + i, s) \) after stage \( s \).

The diagonalization requirements \( D_{2e+i} \) look like the Friedberg-Muchnik requirements. There, however, both the set on the left hand side and the oracle set on the right hand side are under construction. Here, the set \( B \) on the left hand side is given and only the oracle set \( A_i \) is under construction. So here we cannot use a direct diagonalization argument as in case of the Friedberg-Muchnik requirements since we cannot enumerate numbers into \( B \).

Here the idea for meeting \( D_{2e+i} \) is as follows. If we see an agreement of \( B \) and \( \Phi^A_{e} \) on some initial segment then we try to preserve this agreement by holding the right hand side by imposing a restraint on oracle \( A_i \). Then we argue that the length of agreement has to be bounded (and so the restraint which has to be imposed). Namely if the length of agreement would go to infinity then the preservation of the computations on the right hand side would guarantee that \( B = \Phi^A_{e} \) and that the functional \( \Phi^A_{e} \) is computable. So \( B \) were computable contrary to choice of \( B \).

In order to explain this strategy, called the *Sacks’s Preservation Strategy*, in more detail, we will need some notation first.

The **length (of agreement) function** of \( D_{2e+i} \) is defined by

\[ l^Q(2e + i, s) = \mu y \ [B_i(y) \neq \Phi^{A_{e,s}}(y)] \]
A stage $s$ is called $2e + i$-expansionary if $s = 0$ or $s > 0$ and
\[ \text{ID}(2e + i, s) > \max_{t < s} \text{ID}(2e + i, t). \]

**Claim 2.** Assume that there is a stage $s_0$ such that for all $2e + i$-expansionary stages $s \geq s_0$,
\[ A_i \upharpoonright s + 1 = A_i \upharpoonright s + 1. \]
Then $B \neq \Phi^A_i$ and there are only finitely many $2e + i$-expansionary stages.

**Proof.** Note that $B = \Phi^A_i$ implies that there are infinitely many $2e + i$-expansionary stages. So it suffices to show that there are only finitely many $2e + i$-expansionary stages.

For a contradiction assume that there are infinitely many $2e + i$-expansionary stages, and let $s_1 < s_2 < s_3 < \ldots$ be the $2e + i$-expansionary stages $> s_0$. Then
\[ \lim_{n \to \infty} \text{ID}(2e + i, s_n) = \omega \] (2.3)
and, by choice of $s_0$,
\[ A_{i,s_n} \upharpoonright s_n = A_i \upharpoonright s_n. \] (2.4)
By definition of the length function and by the Use Lemma (together with our convention on uses) this implies
\[ \forall s \geq s_n \left[ B_{s_n} \upharpoonright i(2e + i, s_s) = \Phi^A_{s_n} \upharpoonright i(2e + i, s_n) \right. \]
\[ \left. = \Phi^A_{s'} \upharpoonright i(2e + i, s_n) \right. \]
\[ = \Phi^A_{e} \upharpoonright i(2e + i, s_n) \] (2.5)
It follows that
\[ B \upharpoonright i(2e + i, s_n) = B_{s_n} \upharpoonright i(2e + i, s_n). \] (2.6)
Namely, otherwise, there must be a number $x < i(2e + i, s_n)$ and a stage $s > s_n$ such that $B(x) = B_s(x) = 1 \neq 0 = B_{s_n}(x)$. But, by (2.5), this implies that $B_{s'}(x) \neq \Phi^A_{e'}(x)$ for all $s' > s$. It follows that $i(2e + i, s') < x \leq i(2e + i, s_n)$ whence no stage $s' > s$ is $2e + i$-expansionary contrary to assumption.

So (2.6) holds. But, by (2.3), this implies that $B$ is computable (since $B(x) = B_{s_n}(x)$ for the least $n$ such that $i(2e + i, s_n) > x$) contrary to choice of $B$.

This completes the proof of Claim 2.

By Claim 2, we can meet requirement $\mathcal{R}_{4e+2+i} = D_{2e+i}$ by letting
\[ r(4e + 2 + i, s) = r^D(2e + i) = \max \{t \leq s : t \text{ is } 2e + i \text{-expansionary} \} \]
and by ensuring that there is a stage $s_0$ such that, for all $s \geq s_0$, $a(s)$ is not put into $A_i$ at stage $s + 1$ if $a(s) < r(4e + 2 + i)$. Namely, this ensures that the hypothesis
of Claim 2 holds. So, by Claim 2, $\mathcal{R}_{4e+2+i} = D_{2e+i}$ is met and the limit of the restraints $r(4e + 2 + i, s)$, $s \geq 0$, exists and is finite (since there are only finitely many $2e + i$-expansionary stages). Note that this strategy is $A_i$-negative.

The above discussion of the strategies for meeting the individual requirements leads to the following construction of the sets $A_i$ where we use the above introduced notation. Note that, in general, the strategy for meeting requirement $\mathcal{R}_n$ described above is $A_0$-negative if $n$ is even and $A_1$-negative otherwise.

Stage $s+1$

Fix $n$ minimal such that $a(s) < r(n, s)$. If there is no such $n$ or if $n$ is even then put $a(s)$ into $A_1$. Otherwise put $a(s)$ into $A_0$.

To show that the thus defined sets $A_i$ have the required properties, we first observe that $A = A_0 \cup A_1$ and that, by effectivity of the construction, $A_j$ is c.e. So it suffices to show that all requirements are met. This is established by the following claim.

Claim 3. For any $n \geq 0$, requirement $\mathcal{R}_n$ is met and $r(n) := \lim_{s \to \omega} r(n, s) < \omega$ exists.

Proof. The proof is by induction on $n$. Fix $n$ where (by symmetry) w.l.o.g. we may assume that $n$ is even, and assume the claim correct for $n' < n$. Then there is a stage $s_0$ such that for all $s \geq s_0$ and all $n' < n$, $r(n', s) = r(n)$ and $a(s) \geq \max_{n' < n} r(n')$. So, by construction, for all $s \geq s_0$,

$$a(s) < r(n, s) \Rightarrow a(s) \notin A_0,$$

i.e.,

$$A_{0, s+1} \upharpoonright r(n, s) = A_{0, s} \upharpoonright r(n, s)$$

Now, if $n = 4e$, then, by Claim 1, $\mathcal{R}_n = \Omega_{2e}$ is met and $r(n) = r(n, s_0) = r(n, s_0)$ for all $s \geq s_0$. Finally, if $n = 4e + 2$, then, (as observed above) by choice of $s_0$ and by definition of $r(n, s)$, the hypothesis of Claim 2 holds. So $\mathcal{R}_n = D_{2e}$ is met and there are only finitely many $2e + i$-expansionary stages whence $r(n) < \omega$ exists.

This completes the proof of the Splitting Theorem.

\[\square\]

EXERCISES

Exercise 2.14 (see e.g. Soare [So87], Chapter VII) Show that the lowness requirements in the proof of the Splitting Theorem are not needed: The strategy for meeting the diagonalization requirements $D_n$ ensures that the sets $A_i$ are low.
(Hint: Consider the Turing functional

\[ \Psi^X(e, x) = \begin{cases} B(0) & \text{if } x = 0 \text{ and } \Phi^X_e(e) \downarrow \\ \uparrow & \text{otherwise.} \end{cases} \]

and by Corollary 1.20 (relativized) fix a computable functions g such that \( \Phi^X_{g(e)}(x) = \Psi^X(e, x) \). Then argue that

\[ \Phi^A_{g_i}(e) \downarrow \Leftrightarrow \lim_{s \to \infty} l(2g(e) + i, s) > 0 \]

and apply the Limit Lemma.)
Chapter 3

Strong reducibilities

In this chapter we introduce the strongly bounded Turing reducibilities which will be in the center of the course and we compare these reducibilities with some of the other common reducibilities strengthening Turing reducibility.

We start with some general remarks and notation on reducibilities. For any reducibility $r$ we write $A \leq_r B$ if $A$ is $r$-reducible to $B$. We say that a reducibility $r$ is stronger than a reducibility $r'$ if

$$\forall A, B [A \leq_r B \Rightarrow A \leq_{r'} B].$$

I.e., $r$-reducibility is stronger than $r'$-reducibility if any $r$-reduction is (or can be replaced by) an $r'$-reduction. All reducibilities considered here are computable reducibilities which are stronger than Turing reducibility.

In the following we apply the notation previously introduced for Turing reducibility to arbitrary reducibilities $r$. So, for instance, we say that sets $A$ and $B$ are $r$-equivalent ($A =_r B$) if $A$ is $r$-reducible to $B$ and $B$ is $r$-reducible to $A$. So, if $r$ is a preordering, i.e., reflexive and transitive, then $r$-equivalence is an equivalence relation. In this case we call these equivalence classes $r$-degrees and denote the partial ordering induced by $r$-reducibility on the class $D_r$ of the $r$-degrees by $\leq_r$. Again we call an $r$-degree a c.e. $r$-degree if it contains a c.e. set and let $(R_r, \leq)$ denote the partial ordering of the $r$-degrees.

Note that for any reducibilities $r$ and $r'$ such that $r$ and $r'$ are preorderings and $r$ is stronger than $r'$, the $r$-degree of any set $A$ is contained in the $r'$-degree of $A$ and any $r'$-degree is the union of $r$-degrees, namely, $\text{deg}_r(A) = \bigcup \{\text{deg}_r(B) : B =_r A\}$. Since we consider only reducibilities $r$ which are stronger than $T$, it follows that any $r$-degree is contained in the corresponding $T$-degree.

Similarly, if $r$ is stronger than $r'$ then any $r$-hard set (if there is any) is $r'$-hard and any $r$-complete set is $r'$-complete.
3.1 Many-one reducibility and truth-table-type reducibilities

A set $A$ is many-one reducible (m-reducible) to a set $B$ ($A \leq_m B$) if there is a computable function $f$ such that

$$\forall x [x \in A \iff f(x) \in B]$$

(i.e., $A(x) = B(f(x))$ for all $x$). In this case we also say that $A$ is m-reducible to $B$ via $f$.

As one can easily check, m-reducibility is a preordering (exercise!) and an m-reduction may be viewed as a T-reduction which on any input $x$ allows only one oracle query $f(x)$ which has to be evaluated positively. Moreover, in contrast to T-reducibility (and all the other reducibilities introduced in the following), the class of c.e. sets is downward closed under m-reducibility, i.e., for any c.e. set $B$ and any set $A$ such that $A \leq_m B$, $A$ is c.e. too (exercise!).

Many-one reducibility is a special case of truth-table reducibility (tt). In a truth-table reduction all oracle queries and an evaluation function of the oracle answers to the queries are specified in advance. This has been formalized in various (equivalent) ways. For instance we may say that a tt-reduction consists of a computable function $g : \mathbb{N} \to \mathbb{N}^*$ (the selector function) and a computable function $h : \mathbb{N} \times \{0, 1\}^* \to \{0, 1\}$ (the evaluation function), and that a set $A$ is tt-reducible to a set $B$ via the reduction $(f, g)$ if for all numbers $x$

$$A(x) = h(x, B(y_0), \ldots, B(y_n)) \quad \text{where} \quad g(x) = (y_0, \ldots, y_n).$$

Note that a tt reduction $(g, h)$ can be simulated by a total oracle machine $M_e$, i.e., an oracle machine $M_e$ which terminates for all inputs and all oracle sets (i.e., the Turing functional $\Phi^A_e(x)$ computed by $M_e$ converges for all oracle sets $A$ and all inputs $x$) and which is nonadaptive in the sense that the oracle queries do not depend on the oracle. (Namely, $M^B_e$ first computes $g(x) = (y_0, \ldots, y_n)$ (without using oracle $B$), then, by querying the oracle about $y_0, \ldots, y_n$ produces $B(y_0), \ldots, B(y_n)$, and finally computes $h(x, B(y_0), \ldots, B(y_n))$ (without using the oracle).)

So, in particular, if $A \leq_n B$ then $A \leq_T B$ via a total oracle machine, i.e., $A = \Phi^B$ for a total Turing functional $\Phi$. (In fact, it is not hard to show that the converse is true too, i.e. truth-table reductions can be alternatively described by total Turing functionals.)

A number of strengthenings of truth-table reducibility have been introduced. For instance, a truth-table reduction $(g, h)$ is a $k$-tt-reduction if the selection function is of type $g : \mathbb{N} \to \mathbb{N}^k$ and $(g, h)$ is a bounded truth-table (btt) reduction if $(g, h)$
3.2. Bounded Turing reducibilities

is a $k$-tt-reduction for some $k \geq 1$. (For $k \geq 2$, $k$-tt-reducibility is not transitive while the 1-tt-, btt- and tt-reducibilities are preorderings.) Note that a many-reduction $f$ is a 1-tt-reduction $(f, h)$ where the evaluation function $h$ does not depend on $x$ and is just the identity function $h(x, i) = h_x(i) = i$.

The relations among the above truth table type reducibilities are as follows.

$$A \leq_m B \Rightarrow A \leq_{1-\text{tt}} B \Rightarrow A \leq_{\text{btt}} B \Rightarrow A \leq_{\text{tt}} B$$

Moreover, for sets in general, all implications are strict. For instance,

- for any set $A$, $A \leq_{1-\text{tt}} A$ whereas there are sets $A$ such that $A \not\leq_m A$ (e.g. $A = K$)
- for any set $A$, $\hat{A} = \{x : \{x, x + 1\} \cap A \neq \emptyset\} \leq_{\text{btt}} A$ whereas one can easily construct some $A$ such that $\hat{A} \not\leq_{1-\text{tt}} A$
- for any set $A$, $\tilde{A} = \{x : \{x, x + 1, \ldots, 2x\} \cap A \neq \emptyset\} \leq_{\text{tt}} A$ whereas one can easily construct some $A$ such that $\tilde{A} \not\leq_{\text{btt}} A$

In fact, by some straightforward finite-injury arguments we get the above separations also for c.e. sets with one exception: on the c.e. sets many-one-reducibility and 1-tt-reducibility coincide.

3.2 Bounded Turing reducibilities

An alternative way to strengthen Turing reducibility is to impose some bounds on the use functions of the reductions, i.e., on the size of the oracle queries.

**Definition 3.1** 1. Let $f$ be any function. A set $A$ is $f$-bounded Turing reducible to a set $B$ if $A$ is Turing reducible to $B$ via a reduction where the use function is bounded by $f$, i.e., if there is a number $e$ such that $A = \Phi^B_e$ and $\phi^B_e(x) \leq f(x)$ for all $x \geq 0$.

2. A set $A$ is bounded-Turing (bT-) reducible or weak-truth-table (wtt-) reducible to a set $B$ ($A \leq_{\text{bT}} B$) if $A$ is $f$-bounded Turing reducible to $B$ for some computable function $f$.

3. A set $A$ is identity-bounded-Turing (ibT-) reducible to a set $B$ ($A \leq_{\text{ibT}} B$) if $A$ is $f$-bounded Turing reducible to $B$ for the identity function $f(x) = x$. 
4. A set $A$ is $(i + k)$-bT-reducible to a set $B$ ($A \leq_{(i+k)\text{bT}} B$) if $A$ is $f$-bounded Turing reducible to $B$ for $f(x) = x + k$.

5. A set $A$ is computable Lipschitz (cl-) reducible to a set $B$ ($A \leq_{\text{cl}} B$) if $A$ is $(i + k)$-bT-reducible to $B$ for some $k \geq 0$.

We call a Turing functional $\Phi$ a bounded Turing (bT) functional if there is a total computable function $f$ such that, for any set $X$ and any number $x$ such that $\Phi^X(x) \downarrow$, $\phi^X(x) \leq f(x)$. cl- and ibT-functionals are defined correspondingly.

Note that in the definition of $A \leq_{\text{bT}} B$ we only require that there is Turing functional $\Phi$ and a computable function $f$ such that $A = \Phi B$ and the use function $\phi^B$ with oracle $B$ (but not necessarily the use function $\phi^X$ for other oracles $X$) is bounded by $f$. Still we can convert the Turing functional $\Phi$ reducing $A$ to $B$ into a bT-functional $\hat{\Phi}$ reducing $A$ to $B$. Namely, $\hat{\Phi}^X(x)$ simulates $\Phi^X(x)$ step by step unless $\Phi^X(x)$ asks an oracle query $y > f(x)$. In this case the oracle query is suppressed and the computation is aborted. Obviously, $\hat{\Phi}$ is a bT-functional (via the bound $f$) and, since the use function of $\Phi^B$ is bounded by $f$, $\Phi^B$ and $\hat{\Phi}^B$ agree. So $A \leq_{\text{bT}} B$ via the bT-functional $\hat{\Phi}$.

Similarly, any cl-reduction and any ibT-reduction is witnessed by a cl-functional and ibT-functional, respectively.

In the following we will refer to identity-bounded Turing reducibility and computable Lipschitz reducibility as the strongly bounded Turing (sbT) reducibilities. We will systematically look at these reducibilities in the following chapters.

Obviously,

$$A \leq_{\text{ibT}} B \Rightarrow A \leq_{\text{cl}} B \Rightarrow A \leq_{\text{bT}} B$$

holds. In fact the implications are strict, even if we restrict ourselves to c.e. sets.

**Theorem 3.2 (Downey et al. [DHL01], Barmpalias and Lewis [BL06])** Let $A$ be a noncomputable c.e. set.

(i) For $A + 1 = \{x + 1 : x \in A\}$, $A + 1 \leq_{\text{ibT}} A$ and $A + 1 =_{\text{cl}} A$.

(ii) For $2A = \{2x : x \in A\}$, $2A \leq_{\text{cl}} A$ and $2A =_{\text{bT}} A$.

**Proof.** We give the proof of (i). The proof of (ii) is similar.

Obviously, $A + 1 =_{\text{cl}} A$ and $A + 1 \leq_{\text{ibT}} A$. So it suffices to show that $A \not\leq_{\text{ibT}} A + 1$. For a contradiction assume that $A \leq_{\text{ibT}} A + 1$ and fix an ibT-functional $\Phi$ such that $A = \Phi^{A+1}$.

We claim that $A$ is self-reducible, i.e., there is a reduction $A = \hat{\Phi}^A$ such that on input $x$ all queries of $\hat{\Phi}^A(x)$ (if any) are less than $x$. Since self-reducible sets are computable this will complete the proof.
The functional $\hat{\Phi}$ is obtained from $\Phi$ as follows. For $x > 0$, $\hat{\Phi}^{A}(x)$ simulates $\Phi^{A+1}(x)$ replacing any oracle query $y > 0$ by the query $y - 1$ (note that $y \leq x$, hence $y - 1 < x$) and by answering the query $y = 0$ negatively. □

### 3.3 Comparing the strong reducibilities

Above we already have compared the different types of truth-table reducibilities with each other and the different types of bounded Turing reducibilities. This leaves to compare the different types of reducibilities.

**Theorem 3.3** For c.e. sets $A$ and $B$ the following and in general only the following relations hold.

\[
A \leq_{\text{tt}} B
\]

\[
\uparrow \quad \searrow
\]

\[
A \leq_{u} B \quad A \leq_{\text{cl}} B
\]

\[
A \leq_{\text{btt}} B \quad A \leq_{\text{ibT}} B
\]

\[
A \leq_{m} B
\]

Diagram 1

The positive implications are immediate by definition. For the strictness of the straight upwards arrows ($\uparrow$) and for the strictness of the arrow $\searrow$ see the two preceding sections. So it suffices to show that $\leq_{\text{cl}}$ does not imply $\leq_{m}$ and $\leq_{u}$ does not imply $\leq_{\text{btt}}$. We do this by constructing a pair of c.e. sets for which the relation with respect to the truth table reducibilities is just opposite to the relation with respect to the strongly bounded Turing reducibilities.

**Theorem 3.4 (Ambos-Spies [Am10])** There are noncomputable c.e. sets $A$ and $B$ such that

\[
A <_{r} B \text{ for } r \in \{\text{m}, \text{btt}, \text{tt}\}
\]

(3.1)

whereas

\[
B <_{r'} A \text{ for } r' \in \{\text{ibT}, \text{cl}\}.
\]

(3.2)
Proof (Sketch). By a finite injury argument enumerate c.e. sets $A$ and $B$ such that

$$B \subseteq \{2x^2 : x \geq 0\} \cup \{2x^2 + 1 : x \geq 0\} \tag{3.3}$$

$$x \in A_{at} \iff 2x^2 \in B_{at} \tag{3.4}$$

$$2x^2 + 1 \in B_{at} \implies x \in A_{at} \tag{3.5}$$

hold and such that the requirements

$$\mathcal{R}_e : \Phi_e \text{ total and nonadaptive } \implies \exists x (B(2x^2 + 1) \neq \Phi_e^A(2x^2 + 1))$$

are met where $\{\Phi_e\}_{e \geq 0}$ is an enumeration of the Turing functionals.

To show that this will guarantee that the sets $A$ and $B$ have the required properties, first note that (3.4) ensures that $A \leq_1 B$ (via $f(x) = 2x^2$) while the requirements $\mathcal{R}_e$ ensure that $B \not\leq_1 A$. So (3.1) holds and, by (3.1), $B$ is noncomputable. On the other hand, by (3.3) - (3.5), $B \leq_{ibT} A$ by permitting, in fact, $B \leq_{ibT} 2A$. So, by noncomputability of $B$, $A$ is noncomputable too, and, by the second part of Theorem 3.2, $B <_{cl} A$. So (3.2) holds too.

In the remainder of the proof we will explain the basic strategy for meeting requirement $\mathcal{R}_e$. Reserve an infinite computable set $R_e$ of diagonalization candidates for requirement $\mathcal{R}_e$. Pick $x \in R_e$ minimal such that $x$ is not restrained by any higher priority requirement and such that $x$ has not been enumerated into $A$ by some previous attack on $\mathcal{R}_e$. Pick $i = \Phi_A^{X_{s+1}}(2x^2 + 1)$. (Note that if no such stage $s$ exists then $\Phi_e$ is partial or adaptive.) Now, at stage $s + 1$, put $x$ into $A$ and $2x^2$ into $B$ and, if $i = 0$, put $2x^2 + 1$ into $B$ too. Moreover, by imposing a restraint on $A$, preserve the computation $\Phi_A^{X_{s+1}}(2x^2 + 1) = \Phi_A^{X_{s+1}}(2x^2 + 1)$. Note that the above action is consistent with (3.3) - (3.5) and ensures that $B(2x^2 + 1) \neq \Phi^A(2x^2 + 1)$ unless the restraint imposed at stage $s + 1$ is violated by some higher priority requirement. □

### 3.4 Bounded Turing reducibility vs. Turing reducibility

We conclude this chapter by showing that bounded Turing reducibility is strictly stronger than Turing reducibility on the c.e. set. First we show that the deficiency
set of a c.e. set is Turing equivalent to the set, then, by a priority argument, we construct a c.e. set $A$ which is not bT-equivalent to its deficiency set.\footnote{There are priority free proofs of this fact too. E.g. by using creativeness of the halting set $K$ one can show that $K$ is not bT-reducible to any $h$-simple set. Since deficiency sets are $h$-simple this implies $K \not\leq_{bT} D_K$ (see e.g. Exercise V.2.16 in Soare [So87]).}

**Lemma 3.5 (Dekker 1954)** Let $A$ be a noncomputable c.e. set, let $a(s)$ be a computable 1-1 enumeration function of $A$, and let $A_s = \{a(0), \ldots, a(s)\}$, $s \geq 0$, be the computable enumeration of $A$ induced by $a(s)$. Then the deficiency set of $A$ with respect to $a(s)$,

$$D = \{s : \exists t > s \ (a(t) < a(s))\} = \{s : A_s \upharpoonright a(s) \neq A \upharpoonright a(s)\},$$

is c.e. and Turing equivalent to $A$.

**Proof.** The proof that $D$ is c.e. and that $D$ is Turing (in fact truth-table) reducible to $A$ is straightforward. For a proof of $A \leq_T D$ we first observe that $D$ is infinite. (Namely, for any $s$ there is a stage $s' > s$ which is not in $D$, namely the stage $s'$ where $a(s') = \min \{a(t) : t > s\}$ is not in $D$.) So, given $x$, we can compute $A(x)$ with oracle $D$ as follows. Using $D$ as an oracle find the least stage $s$ such that $x < a(s)$ and $s \notin D$. (By infinity of $D$ such an $s$ exists.) Then $A_s \upharpoonright a(s) = A \upharpoonright a(s)$ and hence, by $x < a(s), A(x) = A_s(x)$. □

**Theorem 3.6** There is a noncomputable c.e. set $A$ and a computable 1-1 enumeration function $a(s)$ of $A$ such that $A$ is not bounded Turing reducible to the deficiency set $D$ of $A$ w.r.t. $a(s)$.

**Proof (Sketch).** The desired c.e. set $A$ and computable 1-1 enumeration function $a(s)$ of $A$ are defined by a priority argument. We define $a(s)$ at stage $s$ of the construction and let $A_s = \{a(0), \ldots, a(s)\}$ be the finite part of $A$ enumerated by the end of stage $s$ (where we let $a(0) = 0$).

Then it suffices to ensure that

$$A \not\leq_{bT} D$$

where

$$D = \{s : \exists t > s \ (a(t) < a(s))\}. \quad (3.6)$$

Note that the sets

$$D_s = \{s' < s : \exists t \ (s' < t \leq s \ \&\ a(t) < a(s'))\}$$

give a computable enumeration of $D$ where $D_s$ is completely determined by $A_s$.

Condition (3.6) is split up in the following requirements $(e_0, e_1 \geq 0)$.

$$\mathcal{R}_{(e_0, e_1)} : \text{If } \Phi^D_{e_0} \text{ and } \varphi_{e_1} \text{ are total and } \varphi^D_{e_0}(x) \leq \varphi_{e_1}(x) \text{ for all } x \text{ then } A \neq \Phi^D_{e_0}.$$
(To show that meeting these requirements implies (3.6) for a contradiction assume that all requirements are met but (3.6) fails. Then, by the latter, Aleq_bT D. So there are a Turing functional $\Phi$ and a total computable function $f$ such that $A = \Phi^0$ and $\Phi^0 \leq f$. Let $e_0$ and $e_1$ be indices of $\Phi$ and $f$, respectively. i.e., $\Phi_{e_0} = \Phi$ and $\Phi_{e_1} = f$. Then $\Phi_{e_0}^0$ is total (by $A = \Phi^0 = \Phi_{e_0}^0$), $\Phi_{e_1}$ is total, $\Phi_{e_0}^0(x) \leq \Phi_{e_1}(x)$ for all $x$ and $A = \Phi_{e_0}^0$ (again by $A = \Phi^0 = \Phi_{e_0}^0$). So requirement $\mathcal{R}_{(e_0,e_1)}$ is not met which gives the desired contradiction.)

Now the basic strategy for meeting requirement $\mathcal{R}_{(e_0,e_1)}$ is as follows.

1. Pick numbers $x$ and $x + 1$ which are not yet in $A$, which are not restrained by any higher priority requirement, and which are not used by any other strategy.

2. Wait for a stage $s$ such that $\varphi_{e_1,s}(x + 1) \downarrow$ and $x, x + 1, \varphi_{e_1,s}(x + 1) < s$. (Note that there will be such a stage $s$ unless $\varphi_{e_1}$ is not total in which case the requirement is trivially met.)

   Put $x$ into $A_{x+1}$ (i.e., set $a(s + 1) = x$) and restrain all numbers $< x$ from $A$ thereby ensuring $A | x = A_1 | x$.

   Note that this ensures that
   \[
   D_{s+1} | \varphi_{e_1,s}(x + 1) + 1 = D | \varphi_{e_1,s}(x + 1) + 1. \tag{3.7}
   \]

   This is shown as follows: Given $s' \leq \varphi_{e_1,s}(x + 1)$ such that $s' \in D$ we have to show that $s' \in D_{s+1}$. By $s' \in D$ there is a stage $t > s'$ such that $a(t) < a(s')$. Fix the least such $t$. We will show that $t \leq s + 1$ and hence that $s' \in D_{s+1}$.

   Distinguish the following two cases.

   If $a(s') > x + 1$ then, by $s' \leq \varphi_{e_1,s}(x + 1) < s$ and by $a(s + 1) = x$, $s' < s + 1$ and $a(s + 1) < a(s')$ whence $t \leq s + 1$.

   Otherwise, by choice of $x$ and $x + 1$, $a(s') < x$. Since, by $A | x = A_s | x$, $a(t') \geq x$ for all $t' \geq s$ it follows by $a(t) < a(s') < x$ that $t < s$ which completes the proof of the claim.)

3. Wait for a stage $s' > s$ such that $\Phi_{e_0,s'}^0(x + 1) = 0$. (Note that there will be such a stage $s'$ unless $\Phi_{e_0}^0(x + 1) \neq 0$ in which case the requirement is met since $x + 1$ will not enter $A$.)

   Put $x + 1$ into $A_{s'+1}$ (i.e., set $a(s' + 1) = x + 1$).

   Note that, by (3.7), this ensures that $A(x) = 1 \neq 0 = \Phi_{e_0}^0(x + 1)$.

   The combination of the strategies is standard and omitted here. (Note that if at some stage $s > 0$ no requirement puts one of its followers into $A$ then a big unused number $y$ is put into $A$, by letting $a(s + 1) = y$.)
Corollary 3.7 There are c.e. sets $A$ and $B$ such that $A \leq_T B$ but $A \not\leq_{bT} B$.

Proof. Fix a c.e. set $A$ and a computable 1-1 enumeration function $a(s)$ of $A$ as in Theorem 3.6 and let $B$ be the deficiency set of $A$ w.r.t. $a(s)$. Then, by Lemma 3.5, $B$ is c.e. and $A \leq_T B$ while by Theorem 3.6, $A \not\leq_{bT} B$. □
The strongly bounded Turing reducibilities: cl-reducibility and ibT-reducibility

In this chapter we start our investigation of the strongly bounded Turing reducibilities ibT and cl on the c.e. sets. We first summarize some basic facts on these reducibilities to be used later. Then we point out the strong relations between ibT-reducibility and permitting. Before we present this material, however, we would like to give some hints at the origin of these reducibilities.

Though strongly bounded Turing reductions have been used in computability theory since its beginnings and though these reductions are quite common in the study of the c.e. sets - for instance the permitting method or c.e. set splittings yield ibT-reductions – the corresponding reducibilities themselves have been only recently introduced. A reason for the delay might be found in the fact that - in contrast to the “classical” reducibilities (like m, btt, tt, bT and T) - the strongly bounded Turing reducibilities are not computably invariant (as we will show in the next chapter).

Computable Lipschitz reducibility (originally called sw-reducibility where sw stands for strong weak-truth-table) was introduced by Downey, Hirschfeldt and LaForte [DHL01, DHL04] in 2001 in the context of algorithmic randomness. Namely, cl is not only a measure of the complexity in the sense of computability but also a measure of the complexity in the information theoretic sense. To understand this, note that by the strict bound on the use function, for a set A which is cl-reducible to a set B, the finite initial segment $A \upharpoonright n$ of $A$ can be computed from the corresponding initial segment $B \upharpoonright n$ of $B$ with the help of a constant number of additional bits. So the Kolmogorov complexity of $A \upharpoonright n$ is bounded by the Kolmogorov complexity of $B \upharpoonright n$ up to an additive constant. Moreover, Downey, Hirschfeldt and LaForte have shown that, on the computably enumerable sets, cl-reducibility coincides with Solovay reducibility which may be viewed as a relative measure of the speed by which a real number can be effectively approximated by rational numbers, and that cl-reducibility is closely related to other information theoretic reducibilities defined in terms of Kolmogorov complexity.

For a more complete account of the role played by cl-reducibility in the theory
of algorithmic randomness we refer the reader to the forthcoming monograph of Downey and Hirschfeldt [DH10]. We only want to add here that ibT-reducibility, which of course may be viewed as a special case of cl-reducibility, has some independent origins too. This reducibility has been introduced by Soare [So04] in 2004 in the context of some applications of computability theory to some problems in differential geometry.

4.1 Some basic facts on the sbT-reducibilities

We now give some basic facts on the sbT-reducibilities and fix some notation we will use later.

As one can easily show, the sbT-reducibilities ibT and cl are a preordering, i.e., reflexive and transitive (Exercise!). So for, \( r = \text{ibT}, \text{cl} \), \( r \)-equivalence (=, r) is an equivalence relation. Hence we obtain a partial ordering \( \leq \) (i.e., \( \leq \) is reflexive, transitive and antisymmetric) on the \( r \)-degrees, where the \( r \)-degree of a set \( A \) is given by the \( r \)-equivalence class of \( A \),

\[
\text{deg}_r(A) = \{ B : A =_r B \}
\]

and the partial ordering on the \( r \)-degrees is induced by \( \leq_r \), i.e.,

\[
\text{deg}_r(A) \leq \text{deg}_r(B) \iff A \leq_r B.
\]

Recall that an \( r \)-degree is c.e. iff it contains a c.e. set. As in case of Turing reducibility, we will denote \( r \)-degrees by boldface lower case letters \( a, b, c, \ldots \) and we let \( (R_r, \leq) \) denote the partial ordering of the c.e. \( r \)-degrees. Note that just as in the case of Turing reducibility - a set \( A \) is computable iff it is \( r \)-reducible to all (c.e.) sets. So the computable sets constitute an \( r \)-degree \( 0 \) and this degree is the least element of the partial ordering \( (R_r, \leq) \).

Also note that, for \( r = \text{ibT}, \text{cl} \) and for \( r = (i + k) \text{bT} \) \((k \geq 0)\), \( r \)-reducibility is invariant under finite variations. I.e., if \( A \leq_r B \), \( A =^* \hat{A} \) and \( B =^* \hat{B} \) then \( \hat{A} \leq_r \hat{B} \). (We will tacitly use this fact in the following.)

Recall that a Turing functional \( \Phi \) is an ibT-functional ((\( i + k \))bT-functional, cl-functional) if for any set \( X \) and any number \( x \) such that \( \Phi^X(x) \downarrow \), \( \Phi^X(x) \leq x \) \((\Phi^X(x) \leq x + k, \Phi^X(x) \leq x + k \) for some \( k \geq 0)\). In Section 3.2 we have already remarked that if \( A \leq_{\text{ibT}} B \) then there is an ibT-functional \( \hat{\Phi} \) such that \( A = \hat{\Phi}^B \) (and,
4.2. The strongly bounded Turing reducibilities and the permitting method

obviously, vice versa). Namely, if \( A \leq_{ibT} B \) via the Turing functional \( \Phi \) then \( A = \Phi^B \) and \( \Phi^B(x) \leq x \). So, for \( \Phi \) defined by

\[
\hat{\Phi}_X(x) = \begin{cases} 
\Phi_X(x) & \text{if } \Phi_X(x) \downarrow \text{ and } \phi_X(x) \leq x \\
\uparrow & \text{otherwise,}
\end{cases}
\]

\( \Phi \) is an ibT-functional\(^1\) and \( A = \Phi^B = \hat{\Phi}^B \). (The argument for \( r = (i+k)bT, \text{cl} \) is similar.)

In a similar way we can argue that the computable enumeration \( \{ \Phi \}_{e \geq 0} \) of the Turing functionals can be converted into a computable enumeration \( \{ \hat{\Phi}_e \}_{e \geq 0} \) of the ibT-functionals by letting

\[
\hat{\Phi}_e^X(x) = \begin{cases} 
\Phi_e^X(x) & \text{if } \Phi_e^X(x) \downarrow \text{ and } \phi_e^X(x) \leq x \\
\uparrow & \text{otherwise.}
\end{cases}
\]

In order to obtain a computable enumeration \( \{ \tilde{\Phi}_e \}_{e \geq 0} \) of the cl-functionals it suffices to let

\[
\tilde{\Phi}_e^X(x) = \begin{cases} 
\Phi_{e_0}^X(x) & \text{if } \Phi_{e_0}^X(x) \downarrow \text{ and } \phi_{e_0}^X(x) \leq x + e_1 \\
\uparrow & \text{otherwise}
\end{cases}
\]

Note that by \( e_1 \leq \langle e_0, e_1 \rangle \), \( \hat{\Phi}_e \) is an \((i+e)bT\)-functional.

In the following we fix such a computable enumeration \( \{ \hat{\Phi}_e \}_{e \geq 0} \) of the ibT-functionals and a computable enumeration \( \{ \Phi_e \}_{e \geq 0} \) of the ibT-functionals such that \( \hat{\Phi}_e \) is an \((i+e)bT\)-functional.

Moreover, in the following \( \hat{Phi}, \hat{Psi} \) etc. will denote ibT-functionals and \( \tilde{Phi}, \tilde{Psi} \) etc. will denote cl-functionals.

### 4.2 The strongly bounded Turing reducibilities and the permitting method

For our further analysis of the strongly bounded Turing reducibilities on the c.e. sets we will need some observations relating the permitting technique and splittings of c.e. sets to these reducibilities.

\(^1\)We assume that it is clear from the above explicit definition of \( \hat{\Phi} \) that \( \Phi \) is a Turing functional. The intuitive, operational description of \( \hat{\Phi} \) behind the above definition is as follows: on input \( x \), \( \hat{\Phi}_X^X(x) \) simulates \( \Phi^X(x) \) step-by-step unless \( \Phi^X(x) \) asks an oracle query \( y \) which is greater than \( x \). In this case \( \Phi^X(x) \) stops the simulation and does not terminate (e.g., goes into an infinite loop).

The permitting method introduced in Section 2.3 is a fundamental quite common technique for constructing a c.e. set A below a given c.e. set B. It actually gives an ibT-reduction (but, in general, not a tt-reduction, hence not an m-reduction). By an obvious modification we obtain a basic technique for obtaining \((i + k)bT\)-reductions.

**Definition 4.1** Let A and B be c.e. sets and \(k \geq 0\). Then \(A \leq_T B\) by \(k\)-permitting if there are computable enumerations \(\{A_s\}_{s \geq 0}\) and \(\{B_s\}_{s \geq 0}\) of A and B, respectively, such that

\[
\forall x \forall s \ (x \in A_{as} \Rightarrow \exists y \leq x + k \ (y \in B_{as})))
\]

(4.1) holds. In particular, \(A \leq_T B\) by **permitting** if \(A \leq_T B\) by 0-permitting.

**Proposition 4.2** Let A and B be c.e. sets such that \(A \leq_T B\) by \(k\)-permitting. Then \(A \leq_{(i+k)bT} B\). In particular, if A and B are c.e. sets such that \(A \leq_T B\) by permitting then \(A \leq_{ibT} B\).

Conversely, any \((i + k)bT\)-reduction from a c.e. set A to a c.e. set B can be represented by a \(k\)-permitting if we replace A and B by some ibT-equivalent subsets. In the following we state this observation in a somewhat more general form.

**Lemma 4.3 (Representation Lemma)** Let A and \(B_0, \ldots, B_m\) \((m \geq 0)\) be noncomputable c.e. sets and let \(k \geq 0\) such that \(B_j \leq_{(i+k)bT} A\) for \(j \leq m\). There are c.e. subsets \(\hat{A}\) of A and \(\hat{B}_j\) of \(B_j\), and computable 1-1 enumerations \(\{a(n)\}_{n \geq 0}\) and \(\{b_j(n)\}_{n \geq 0}\) of \(\hat{A}\) and \(\hat{B}_j\), respectively, such that, for \(j \leq m\),

(i) \(\hat{A} =_{ibT} A\) and \(\hat{B}_j =_{ibT} B_j\) and

(ii) \(\forall n \ (a(n) \leq \min\{b_0(n), \ldots, b_m(n)\} + k)\)

Note that (in the context of the above lemma) \(\hat{B}_j \leq_T \hat{A}\) by \(k\)-permitting via the enumerations \(\{\hat{B}_{j,s}\}_{s \geq 0}\) and \(\{\hat{A}\}_{s \geq 0}\) given by \(\hat{B}_{j,s} = \{b_j(0), \ldots, b_j(s-1)\}\) and \(\hat{A}_s = \{a(0), \ldots, a(s-1)\}\).

**Proof of Lemma 4.3.** Fix computable enumerations \(\{A_s\}_{s \geq 0}\) and \(\{B_{j,s}\}_{s \geq 0}\) of A and \(B_j\), respectively, such that \(A_{s+1} \neq A_s\) and \(B_{j,s+1} \neq B_{j,s}\) \((s \geq 0)\), and let \(\Psi_j\) be an \((i + k)bT\)-functional such that \(B_j = \Psi_j^A\) \((j \leq m)\). Define the length (of agreement) function \(l\) by

\[l(s) = \max\{x \leq s : \forall y < x \forall j \leq m \ (B_{j,s}(y) = \Psi_j^A(y))\}.
\]

Then

\[
\lim_{s \to \infty} l(s) = \omega
\]
whence there are infinitely many expansionary stages s, i.e., stages s such that
\[ \forall t < s \ (l(t) < l(s)). \]

Call an expansionary stage s critical if, for some \( j \leq m \), there is a number \( x < l(s) \) such that \( x \in B_j \setminus B_{j,s} \), and say that criticalness of s is witnessed by t if t > s and \( B_{j,s} \upharpoonright l(s) \neq B_{j,t} \upharpoonright l(s) \). Note that if criticalness of s is witnessed by t then criticalness of s is witnessed by all \( t' \geq t \). Moreover, by noncomputability of the sets \( B_j \), there are infinitely many critical expansionary stages. Finally, the set of all pairs \((s,t)\) such that s is critical and criticalness of s is witnessed by t is computably enumerable. So we can define a computable ascending sequence of expansionary stages \( s_0 < s_1 < s_2 < \ldots \) such that \( s_n \) is critical and \( s_{n+1} \) witnesses criticalness of \( s_n \).

Now let
\[ a(n) = \mu x \ (x \in A_{s_{n+1}} \setminus A_{s_n}) \quad \text{&} \quad b_j(n) = \mu x \ (x \in B_{j,s_{n+1}} \setminus B_{j,s_n}) \]
and let \( \hat{A} = \{ a(n) : n \geq 0 \} \) and \( \hat{B}_j = \{ b_j(n) : n \geq 0 \} \). Then, obviously, \( a \) and \( b_j \) are computable one-to-one functions enumerating \( \hat{A} \) and \( \hat{B}_j \), respectively, and \( \hat{A} \subseteq A \) and \( \hat{B}_j \subseteq B_j \). So it only remains to show that \((i)\) and \((ii)\) hold.

For a proof of \((ii)\), given \( n \) and \( j \leq m \) such that \( b_j(n) = \min\{ b_0(n), \ldots, b_m(n) \} \), it suffices to show that
\[ a(n) \leq b_j(n) + k. \quad (4.2) \]

Now, by definition of \( b_j \) and by choice of \( j \),
\[ b_j(n) \in B_{j,s_{n+1}} \setminus B_{j,s_n} \quad \text{&} \quad b_j(n) < l(s_n). \]

So
\[ 0 = B_{j,s_n}(b_j(n)) = \Psi_{j,s_n}^{A_{s_n}}(b_j(n)) \]
and, since \( s_{n+1} \) is expansionary,
\[ 1 = B_{j,s_{n+1}}(b_j(n)) = \Psi_{j,s_{n+1}}^{A_{s_n+1}}(b_j(n)). \]
Since \( \Psi_j \) is an \((i + k)\)bT functional it follows that
\[ A_{s_n} \upharpoonright b_j(n) + k + 1 \neq A_{s_{n+1}} \upharpoonright b_j(n) + k + 1. \]
So, by definition of \( a \), \((4.2)\) holds.

Finally, for a proof of \((i)\), let
\[ \hat{A}_n = \{ a(0), \ldots, a(n - 1) \} \quad \text{&} \quad \hat{B}_{j,n} = \{ b_j(0), \ldots, b_j(n - 1) \}. \]
Then
\[ \mu x \left( x \in \hat{A}_{n+1} \setminus \hat{A}_n \right) = \mu x \left( x \in A_{s_{n+1}} \setminus A_{s_n} \right) \]
and
\[ \mu x \left( x \in \hat{B}_{j,n+1} \setminus \hat{B}_{j,n} \right) = \mu x \left( x \in B_{j,s_{n+1}} \setminus B_{j,s_n} \right). \]
So \( \hat{A} = \text{ibT} A \) and \( \hat{B}_j = \text{ibT} B_j \) by permitting. \( \square \)

If we split a c.e. set \( A \) into two disjoint c.e. sets \( A_0 \) and \( A_1 \) then the Turing degree of \( A \) is the least upper bound (join) of the Turing degrees of the parts \( A_0 \) and \( A_1 \) (see Lemma 2.9). Conversely, given any join \( \text{deg}_T(A) = \text{deg}_T(B) \lor \text{deg}_T(C) \), this join can be represented by the splitting of the disjoint union \( B \oplus C \) into the sets \( B \oplus \emptyset \) and \( \emptyset \oplus C \). While these observations easily carry over to the bT-degrees, for the strongly bounded Turing degrees we obtain only one direction.

**Lemma 4.4 (Splitting Lemma)** Let \( A_0, \ldots, A_m \) (\( m \geq 1 \)) be pairwise disjoint c.e. sets and let \( A = A_0 \cup \cdots \cup A_m \). Then, for \( r \in \{ \text{ibT}, \text{cl} \} \),
\[ \text{deg}_r(A) = \text{deg}_r(A_0) \lor \cdots \lor \text{deg}_r(A_m). \]

**Proof.** For \( j \leq m, A_j \leq_{\text{ibT}} A_0 \cup \cdots \cup A_m = A \) by permitting. So, given \( B \) such that \( A_j \leq_{(i+k_j)\text{bT}} B \), it suffices to show that \( A_0 \cup \cdots \cup A_m \leq_{(i+k)\text{bT}} B \) where \( k = \max k_j \). But this is obviously true. \( \square \)
Chapter 5

Shifts and the c.e. degrees under the sbT-reducibilities

In this chapter we give some first results on the partial orderings $R_r, \leq)$ of the c.e. degrees under the strongly bounded Turing reducibilities $r = \text{ibT}, \text{cl}$ which can be obtained by applying bounded and computable shifts, a notion which was introduced by Ambos-Spies, Ding, Fan and Merkle [ADFM10] in order to generalize the observations made in Theorem 3.2. So we use shifts in order to show that the partial orderings $R_r, \leq)$ do not only have no minimal nonzero elements but also do not have any maximal elements. So, in particular, for the sbT-reducibilities $r$, there is no greatest c.e. degree, i.e., there are no $r$-complete sets. (This distinguishes these reducibilities from all of the other transitive reducibilities which we have discussed here.) Along the same lines we will show that computable invariance fails for the sbT-reducibilities as badly as possible.

We then discuss automorphisms of the partial orderings $R_r, \leq)$. We show that the nontrivial bounded shifts introduce nontrivial automorphisms of the partial ordering $R_{\text{ibT}}, \leq)$ of the c.e. ibT-degrees thereby showing that this structure is not rigid. We also show, however, that no computable shift induces a nontrivial automorphism of the partial ordering $R_{\text{cl}}, \leq)$ of the c.e. cl-degrees and we leave the question of rigidity here open.

5.1 Bounded and computable shifts

The sets $A + 1$ and $2A$, used in Theorem 3.2 to separate the bounded Turing reducibilities, shift the elements of the set $A$ to the right where in case of $A + 1$ the shift is bounded (by 1) while in case of $2A$ the shift is computable but unbounded. Ambos-Spies, Ding, Fan and Merkle [ADFM10] have defined shifts of this type in a greater generality.

Definition 5.1 (Ambos-Spies, Ding, Fan and Merkle [ADFM10]) (a) $A (\text{computable})$
shift $f$ is a strictly increasing (computable) function $f : \omega \to \omega$. A shift $f$ is non-trivial if $f(x) > x$ for some (hence for almost all) $x$, and $f$ is unbounded if, for any number $k$, there is a number $x$ such that $f(x) - x > k$.

(b) For any set $A$ and any shift $f$, the $f$-shift of $A$ is defined by

$$A_f = f(A) = \{ f(x) : x \in A \}.$$ 

Note that, for any shift $f$, $x \leq f(x)$ and $f(x) - x$ is nondecreasing in $x$. So a shift $f$ is unbounded if and only

$$\lim_{n \to \infty} (f(n) - n) = \sup_{n \to \infty} (f(n) - n) = \infty.$$ 

Also note that, for any $k \geq 0$, $f(x) = x + k$ is a bounded shift and $A_f = A + k$.

In fact, for any bounded shift $f$, $A_f = A + k$ for some $k \geq 0$. So, in particular, any bounded shift $f$ is computable. Moreover, by invariance of the sbT-reducibilities under finite variations, in this context we may identify the bounded shifts with the shifts $f(x) = x + k$.

Also note that $A + (k + 1) = (A + k) + 1$ whence the observation in Theorem 3.2 that, for noncomputable $A$, $A + 1 \leq_{ibT} A$ extends to all nontrivial bounded shifts:

$$\cdots \leq_{ibT} A + 3 \leq_{ibT} A + 2 \leq_{ibT} A + 1 \leq_{ibT} A = A + 0 \quad (5.1)$$

Moreover, any bounded shift $f(x) = x + k$ can be inverted: If we let

$$A - k = \{ x - k : x \in A \& x \geq k \}$$

then

$$(A + k) - k = A = (A - k) + k \quad (hence \, A =_{ibT} (A + k) - k =_{ibT} (A - k) + k) \quad (5.2)$$

So we obtain the dual of (5.1) if we replace $A + k$ by $A - k$:

$$A = A - 0 \leq_{ibT} A - 1 \leq_{ibT} A - 2 \leq_{ibT} A - 3 \cdots \leq_{ibT} \quad (5.3)$$

(namely $A - 1 \leq_{ibT} A - 2$ since $A - 1 = B + 1$ for $B = A - 2$ etc.).

We now generalize part (i) of Theorem 3.2 as follows.

**Lemma 5.2 (Bounded-Shift Lemma [ADFM10, Am10])** Let $f$ be a nontrival bounded shift and let $A$ be a noncomputable c.e. set.

(i) $A_f =_{m} A$ and $A_f =_{cl} A$.

(ii) $A_f \leq_{ibT} A$.

(iii) For any c.e. set $B$ such that $A_f \cap B = \emptyset$ and such that $A \leq_{ibT} A_f \cup B$, $A \leq_{ibT} B$. 

5.1. Bounded and computable shifts

PROOF. Part (i) is straightforward and, obviously, $A_f \leq_{ibT} A$. By the latter, part (ii) follows from (iii) by letting $B$ be the empty set. This leaves (iii). Here the proof is quite similar to the proof of part (i) of Theorem 3.2. For the sake of completeness, we give the proof.

Fix $B$ such that $A_f \cap B = \emptyset$ and assume that $A = \Phi^{A_f \cup B}$ for the ibT functional $\Phi$. Moreover, by nontriviality of $f$, fix $y_0$ such that $y < f(y)$ for all $y \geq y_0$. In order to show $A \leq_{ibT} B$, we give an inductive procedure for computing $A(x)$ from $B \upharpoonright x + 1$. Given $x$ and the initial segment of $B \upharpoonright x + 1$ as an oracle, it suffices to compute $(A_f \cup B)(y)$ for the queries $y \geq y_0$ occurring in the computation of $\Phi^{A_f \cup B}(x)$ in order of appearance. Fix such a query $y$. Since $\Phi$ is an ibT-reduction, $y \leq x$. So, using $B \upharpoonright x + 1$ as an oracle, we can decide whether $y \in B$. If so, $(A_f \cup B)(y) = 1$. Otherwise, by disjointness of $A_f$ and $B$, $(A_f \cup B)(y) = A_f(y)$. In order to compute the latter, first compute the preimage $x'$ of $y$ under the shift $f$. Then $A(x') = A_f(y)$ and, by $y \geq y_0$, $x' < y \leq x$. So $A(x')$ can be computed from $B \upharpoonright x' + 1$ (hence from $B \upharpoonright x + 1$) by inductive hypothesis. $\square$

For unbounded computable shifts $f$, we obtain an analog of Lemma 5.2 with cl and bT in place of ibT and cl, respectively, which generalizes part (ii) of Theorem 3.2.

Lemma 5.3 (Computable-Shift Lemma [ADFM10, Am10]) Let $f$ be a computable shift and let $A$ be a noncomputable c.e. set.

(i) $A_f =_m A$ and $A_f =_{bT} A$.

(ii) $A_f \leq_{ibT} A$. Moreover, if $f$ is unbounded then $A \nleq_{cl} A_f$ (whence $A_f \nleq_{ibT} A$ and $A_f \nleq_{cl} A$).

(iii) If $f$ is unbounded then, for any c.e. set $B$ such that $A_f \cap B = \emptyset$ and such that $A \leq_{cl} A_f \cup B$, $A \leq_{cl} B$.

The proof of the Computable-Shift Lemma is similar to the proof of the Bounded-Shift Lemma and is left as an exercise.
5.2 Upwards and downwards density of the nonzero c.e. sbT-degrees

As Theorem 3.2 above shows, computable shifts witness the fact that the partial orderings of the c.e. ibT-degrees and the c.e. cl-degrees do not have minimal nonzero elements.

Lemma 5.4 (Downey et al. [DHL01], Birmalas and Lewis [BL06]) Let \( r = \text{ibT}, \text{cl} \).
For any nonzero c.e. r-degree a there is a c.e. r-degree b such that \( 0 < b < a \).

Proof. This is immediate by Theorem 3.2 since for any noncomputable c.e. set A the sets \( A + 1 \) and \( 2A \) are noncomputable and c.e. again. \( \square \)

Since as we have seen above, a bounded shift can be inverted, for the c.e. ibT degrees we also obtain the dual result.

Lemma 5.5 The partial ordering \( (\mathbb{R}_{\text{ibT}}, \leq) \) of the c.e. ibT-degrees does not have maximal elements. I.e., for any c.e. ibT-degree a there is a c.e. ibT-degree b such that \( a < b \).

Proof. Given a c.e. set A it suffices to show that there is a c.e set \( \hat{A} \) such that \( A <_{\text{ibT}} \hat{A} \). But his is immediate by (5.3) (just let \( \hat{A} = A - 1 \)). \( \square \)

Though, in contrast to the bounded case, where we can shift \( \omega \) to the left by giving up a finite amount of information (which will not matter), there will be no room to shift \( \omega \) unboundedly to the left. Still Ambos-Spies et al. [ADFM10] and, independently, Bélanger [Be09] have shown, that computable unbounded shifts can be also applied in order to prove the analog of Lemma 5.5 for the cl-degrees. For the cl-degrees the nonexistence of maximal c.e. degrees was first shown by Birmalas [Ba05] but Birmalas’s original proof was based on a quite sophisticated nonuniform argument. By using the fact, that any infinite c.e. set contains an infinite computable subset, however, there is a very simple proof of this fact using shifts.

Theorem 5.6 (Birmalas [Ba05]) The partial ordering \( (\mathbb{R}_{\text{cl}}, \leq) \) of the c.e. cl-degrees does not have maximal elements. I.e., for any c.e. set A there is a c.e. set \( \hat{A} \) such that \( A <_{\text{cl}} \hat{A} \).

Proof (Ambos-Spies et al. [ADFM10] and Bélanger [Be09]). Let \( \hat{A} \) be any c.e. set. If A is computable then, for any noncomputable c.e. set \( \hat{A} \), \( A <_{\text{ibT}} \hat{A} \).
5.3. The strongly bounded Turing reducibilities are not computably invariant

So w.l.o.g. we may assume that \( A \) is noncomputable, hence infinite, and we may fix an infinite computable subset \( C \) of \( A \). Then, obviously, \( A =_{ibT} A \setminus C = A \cap \overline{C} \) and, by noncomputability of \( A, \overline{C} \) is infinite.

Now, intuitively, we can effectively compress the set \( A \setminus C \) to a set \( \hat{A} \) by using the space in \( C \). Then \( A \setminus C \) will be a computable unbounded shift of \( \hat{A} \) whence, by the Computable-Shift Lemma, \( A =_{cl} A \setminus C <_{cl} \hat{A} \).

More formally, let \( f : \omega \to \overline{C} \) enumerate \( C \) in order, and let \( \hat{A} = f^{-1}(A \cap \overline{C}) \) be the preimage of \( A \cap \overline{C} \) under \( f \). Then \( f \) is a computable shift and \( \hat{A}_f = A \cap \overline{C} \). Moreover, by infinity of \( C \), \( f \) is unbounded. So \( A \cap \overline{C} <_{cl} \hat{A} \) by the Computable-Shift Lemma. By \( A =_{ibT} A \cap \overline{C} \) this implies the claim. \( \square \)

5.3 The strongly bounded Turing reducibilities are not computably invariant

The argument used in the above proof of Theorem 5.6 can be adapted to show that the strongly bounded Turing reducibilities are not computably invariant. Here a reducibility \( r \) is computably invariant if for any noncomputable set \( A \) and any computable permutation \( p \) (i.e., computable bijection \( p : \mathbb{N} \to \mathbb{N} \)), \( A \) and \( p(A) \) are \( r \)-equivalent. As the following Theorem show, computable invariance of the sbT-reducibilities fails on the c.e. sets (almost) as badly as possible.

**Theorem 5.7 (Ambos-Spies [Am10])** For any noncomputable c.e. set \( A \) there are c.e. sets \( A_\prec \) and \( A_\succ \) such that the following hold.

(i) \( A, A_\prec \) and \( A_\succ \) are computably isomorphic.

(ii) For \( r \in \{ibT, cl\} \), \( A_\prec <_r A <_r A_\succ \).

**Proof.** Given a noncomputable c.e. set \( A \) it suffices to define sets \( A_\prec \) and \( A_\succ \) such that \( A, A_\prec \), and \( A_\succ \) are computably isomorphic and, for \( r \in \{ibT, cl\} \), \( A_\prec <_r A <_r A_\succ \).

The set \( A_\prec \) is defined as follows. Fix an infinite computable subset \( C \) of \( A \). Then, by noncomputability of \( A, \overline{C} \) is infinite too. So the enumeration \( f \) of \( C \) in order is a computable unbounded shift, and \( C \) is the disjoint union of the infinite computable sets \( C_0 = f(C) \) and \( C_1 = f(\overline{C}) \). Now let

\[
A_\prec = (A \cap \overline{C})_f \cup \overline{C_1}.
\]
Note that $A_\leq =_{ibT} (A \cap \overline{C})_f$ since $(A \cap \overline{C})_f$ is contained in $C_1$ and $C_1$ is computable. Moreover, by the Computable-Shift Lemma, $(A \cap \overline{C})_f <_r A$ since $A =_{ibT} A \cap \overline{C}$. So $A_\leq <_r A$.

It remains to give a computable permutation $p : \omega \to \omega$ such that $p(A) = A_\leq$. Let

$$p(n) = \begin{cases} f(n) & \text{if } n \in \overline{C} \\ g(n) & \text{otherwise} \end{cases}$$

where $g$ maps the $n$th element of $C$ (in order of magnitude) to the $n$th element of $\overline{C}_1$. Since the restriction of $f$ to $\overline{C}$ is a computable bijection $\overline{C} \to C_1$ and since $g$ is a computable bijection $\overline{C} \to \overline{C}_1$, $p$ is a computable permutation. Moreover, for $n \in C$, $n \in A$ and $p(n) = g(n) \in A_\leq$ and, for $n \in \overline{C}$, $p(n) = f(n) \in C_1$ whence $n \in A$ iff $n \in A \cap \overline{C}$ iff $p(n) = f(n) \in (A \cap \overline{C})_f$ iff $p(n) \in f(n) \in A_\leq$.

For the definition of $A_\geq$, first fix two disjoint infinite computable subsets $C_0$ and $C_1$ of $A$, and let $C = C_0 \cup C_1$. Then $A =_{ibT} A \cap \overline{C}$. Moreover, the computable function $f : \omega \to \overline{C}_0$ enumerating the complement of $C_0$ in order is an unbounded computable shift. So, for $\hat{A} = f^{-1}(A \cap \overline{C})$, $A \cap \overline{C} = (\hat{A})_f <_r \hat{A}$ by the Computable-Shift Lemma. Now let

$$A_\geq = \hat{A} \cup f^{-1}(C_1).$$

Then $A_\geq =_{ibT} \hat{A}$ since $\hat{A}$ and $f^{-1}(C_1)$ are disjoint and the latter is computable. So, by the above, $A <_r A_\geq$.

It remains to give a computable permutation $p : \omega \to \omega$ such that $p(A) = A_\geq$. By choice of $f$, $f^{-1} \upharpoonright \overline{C}$ is a computable one-to-one mapping of $\overline{C}$ onto the computable set $\omega \setminus f^{-1}(C_1)$. Moreover, since $f^{-1}(C_1)$ and $C$ are computable and infinite, we may fix a computable bijection $g : C \to f^{-1}(C_1)$. Then we obtain the desired permutation $p$ by letting $p(x) = g(x)$ for $x \in C$ and $p(x) = f^{-1}(x)$ for $x \in \overline{C}$. Note that, for $x \in C$, $x \in A$ and $p(x) = g(x) \in f^{-1}(C_1) \subseteq A_\geq$. For $x \in \overline{C}$, $p(x) = f^{-1}(x)$ and $x \in A$ iff $f^{-1}(x) \in \hat{A}$ iff $f^{-1}(x) \in A_\geq$. \hfill \Box

OPEN PROBLEM. Is the following variant of Theorem 5.7 true? For any noncomputable c.e. set $A$ there is a c.e. set $A_1$ such that $A$ and $A_1$ are computably isomorphic and, for $r \in \{ibT, cl\}$, $A \mid_r A$. 
5.4 Shifts and Automorphisms

Here we give another interesting application of computable shifts due to Ambos-Spies [Am10]. The (nontrivial) bounded shifts induce (nontrivial) automorphisms of the partial ordering $\mathcal{R}_{ibT}, \leq$ of the c.e. ibT-degrees. So this degree structure is not rigid. As shown in [Am10] too, however, this approach for defining nontrivial automorphisms, however, cannot be extended to the structure of the c.e. cl-degrees. Namely, no unbounded computable shift $f$ induces an automorphism of $\mathcal{R}_{cl}, \leq$. So the question, whether the partial ordering of the c.e. cl-degrees is rigid too, is left open.

In order to show that the bounded shifts induce automorphisms of $\mathcal{R}_{ibT}, \leq$, we first observe that the bounded shifts are ibT-degree invariant and preserve the ordering $\leq_{ibT}$.

**Lemma 5.8 (Invariance Lemma for Bounded Shifts)** For any $k \geq 0$ and any c.e. sets $A$ and $B$,

$$A \leq_{ibT} B \iff A+k \leq_{ibT} B+k.$$  

We omit the straightforward proof.

**Lemma 5.9** For $k \geq 0$ let $f_k : \mathcal{R}_{ibT} \to \mathcal{R}_{ibT}$ be defined by $f_k(a) = a+k$ where $a+k = \deg_{ibT}(A+k)$ for any c.e. set $A \in a$. Then $f_k$ is well defined and $f_k$ is an automorphism of $\mathcal{R}_{ibT}, \leq$.

Recall that an automorphism $f$ of a partial ordering $(P, \leq)$ is a bijection $f : P \to P$ such that

$$\forall a,b \in P (a \leq b \iff f(a) \leq f(b)).$$

**Proof.** By Lemma 5.8, $a+k$ hence $f_k$ is well defined and

$$\forall a,b \in \mathcal{R}_{ibT} [a \leq b \iff f_k(a) \leq f_k(b)].$$

By the latter, $f_k$ preserves ordering and non-ordering, and $f_k$ is one-to-one. It remains to show that $f_k$ is onto, i.e., that for any given c.e. ibT-degree $a$ there is a c.e. ibT-degree $b$ such that $f_k(b) = a$. Since, for any c.e. set $A$, $(A-k)+k = A$ hence $(A-k)+k = \deg_{ibT}(A-k)$ will do.

**Theorem 5.10** The partial ordering $\mathcal{R}_{ibT}, \leq$ of the c.e. ibT-degrees is not rigid. In fact, the automorphism group of $\mathcal{R}_{ibT}, \leq$ is infinite.
PROOF. By Lemma 5.9, for any \( k \geq 0 \), \( f_k \) is an automorphism of \( (R_{ibT}, \leq) \). On the other hand, by definition of \( f_k \) and by (5.1),

\[
\cdots < f_3(a) < f_2(a) < f_1(a) < f_0(a) = a.
\]

So the automorphisms \( f_k, k \geq 0 \), are pairwise different. \( \square \)

Note that the nontrivial automorphisms \( f_k (k \geq 1) \) are push-down automorphisms, i.e., map any nonzero degree to a strictly lesser degree. Correspondingly, the inverse automorphisms \( f_k^{-1} \) (which, of course, are induced by the functions mapping a c.e. set \( A \) to \( A - k \)) are push-up automorphisms, i.e., map any nonzero degree to a strictly greater degree.

**OPEN PROBLEM.** It is an interesting open question whether there are also automorphisms of \( (R_{ibT}, \leq) \) which map some degree to an incomparable one (or all nonzero degrees to incomparable ones).

Ambos-Spies [Am10] has also shown that the above argument, showing that nontrivial bounded shifts yield nontrivial automorphisms of \( (R_{ibT}, \leq) \), cannot be adapted to unbounded computable shifts in order to get nontrivial automorphisms of \( (R_{cl}, \leq) \).

**Theorem 5.11 (Ambos-Spies [Am10])** Let \( f \) be an unbounded computable shift. There is a c.e. set \( A \) such that, for all c.e. sets \( B \), \( A \nleq_{cl} B_f \).

So if an unbounded computable shift \( f \) is cl-degree invariant (which, as shown in [Am10] too, is not always the case) then the induced function \( f \) on the c.e. cl-degrees is not onto, hence not an automorphism.

**OPEN PROBLEM.** Is the partial ordering \( (R_{cl}, \leq) \) rigid.
Joins and meets in the c.e. sbT-degrees

For the “classical” reducibilities \( r = m, btt, tt, bT, T \) we have previously discussed here, the partial ordering \((\mathbb{R}_r, \leq)\) of the c.e. \( r \)-degrees is an upper semi-lattice but not a lattice. So, any pair \((a, b)\) of c.e. \( r \)-degrees has a join (l.u.b.) \( a \lor b \) in the partial ordering \((\mathbb{R}_r, \leq)\) whereas there are pairs \((a, b)\) of c.e. \( r \)-degrees for which the meet \( a \land b \) does not exist, i.e., which do not have a greatest lower bound (g.l.b.) in \((\mathbb{R}_r, \leq)\).

Just as in case of Turing reducibility (see Lemma 1.12), for all of these reducibilities \( r \) the join of the \( r \)-degrees of c.e. sets \( A \) and \( B \) is represented by the effective disjoint union \( A \oplus B = 2A \cup 2B + 1 \) of \( A \) and \( B \). For the strongly bounded Turing reducibilities \( r' = ibT, cl \), however, this is not true. For instance, for any noncomputable c.e. set \( A \),

\[
A \oplus A = 2A \cup 2A + 1 \leq_{r'} 2A <_{r'} A
\]

(where the last inequality follows from the Computable-Shift Lemma). So here the questions of joins becomes nontrivial and, as we will show, joins will not always exist whence for the strongly bounded Turing reductions \( r' \), the partial ordering \((\mathbb{R}_{r'}, \leq)\) is not an upper semi-lattice.

In this chapter we have a closer look at joins and meets in the sbt-degrees. In particular, by a finite-injury priority argument we construct a pair of c.e. degrees without join. (In Chapter 8 we will give an alternative, priority-free proof by constructing a so-called maximal pair.) Then we look at the relations between joins and meets in the bounded Turing reducibilities. So we show that for any c.e. sets \( A \) and \( B \) the existence of the join of the degrees \( A \) and \( B \) in the ibT-degrees implies the existence in the cl-degrees which in turn implies the existence in the bT-degrees. Moreover, we obtain the corresponding results for meets. We will apply the latter to the fact that the partial ordering of the c.e. bT-degrees is not a lower semi-lattice in order to show that, for the strongly bounded Turing reducibilities \( r' \), \((\mathbb{R}_{r'}, \leq)\) is not a lower semi-lattice too. The transfer technique used in the proof of this result turns out to be quite powerful. So we will later show that noncategoricity and undecidability of the c.e. bT-degrees are inherited by the c.e. sbt-degrees by such transfer arguments.
6.1 Pairs without joins in the c.e. bt-degrees

Theorem 6.1 Let \( r = \text{ibT, cl} \). There are c.e. \( r \)-degrees \( a \) and \( b \) such that \( a \lor b \) does not exist.

A sketch of the proof - which is a finite injury argument - can be found on the slides of this lecture. We omit the proof here since below we will prove a stronger fact. Namely we will show that there are so-called maximal pairs in the c.e. strongly bounded Turing degrees, namely pairs of c.e. sbT-degrees which do not have any upper bounds in the c.e. sbT-degrees, hence in particular do not have a least upper bound (see Theorem 8.3).

6.2 Comparing joins and meets in the ibT- and cl-degrees

If a reducibility \( r \) is stronger than a reducibility \( r' \) on the c.e. sets, i.e., if for any c.e. sets \( A \) and \( B \), \( A \leq_r B \) implies \( A \leq_{r'} B \) then, obviously, \( \text{deg}_r(A) \leq \text{deg}_{r'}(B) \) implies \( \text{deg}_{r'}(A) \leq \text{deg}_{r'}(B) \). In general, however, this does not imply that any join \( \text{deg}_r(A) \lor \text{deg}_r(B) = \text{deg}_r(C) \) in the c.e. \( r \)-degrees yields the corresponding join \( \text{deg}_{r'}(A) \lor \text{deg}_{r'}(B) = \text{deg}_{r'}(C) \) in the c.e. \( r' \)-degrees. I.e., joins in the c.e. \( r \)-degrees may not be preserved in the c.e. \( r' \)-degrees, and similarly for meets.

If, for example, we let \( r = \text{bt} \) and \( r' = \text{T} \) then joins are preserved since for any sets \( A \) and \( B \), the effective disjoint union \( A \oplus B \) of \( A \) and \( B \) represents the join of the degrees of the degrees of \( A \) and \( B \) in both, in the bt-degrees and in the T-degrees. Meets in the c.e. bt-degrees, however, are not preserved in the c.e. T-degrees. For instance, as observed (in ??), there are noncomputable c.e. sets \( A \) and \( B \) such that \( \text{deg}_{\text{bt}}(A) \land \text{deg}_{\text{bt}}(B) = 0 \) but \( A =_T B \) (whence \( \text{deg}_T(A) \land \text{deg}_T(B) = \text{deg}_T(A) > 0 \)).

As we will show in the following, however, joins and meets in the c.e. ibT-degrees are preserved in the cl-degrees and joins and meets in the c.e. cl-degrees are preserved in the bt-degrees.

We will show the former in this section, the latter in the next section.

We start with some observations on the convertibility of cl-reductions into ibT-reductions.

Proposition 6.2 Let \( k \geq 0 \) and let \( A \) and \( B \) be c.e. sets such that \( A \leq_{(i+k)\text{ibT}} B \). Then, for any \( k', k'' \geq 0 \) such that \( k \leq k' + k'' \), \( A + k' \leq_{\text{ibT}} B - k'' \). So, in particular, for
6.2. Comparing joins and meets in the ibT- and cl-degrees

\[ k' \geq k, \quad A + k' \leq_{ibT} B \quad \text{and} \quad A \leq_{ibT} B - k'. \]

Lemma 6.3 (cl-ibT-Conversion Lemma) (a) Let \( A, B_0, \ldots, B_n \) be c.e. sets such that \( A \leq_{cl} B_0, \ldots, B_n \). There is a c.e. set \( \hat{A} =_{cl} A \) such that \( \hat{A} \leq_{ibT} B_0, \ldots, B_n \).

(b) Let \( A_0, \ldots, A_n, B \) be c.e. sets such that \( A_0, \ldots, A_n \leq_{cl} B \). There is a c.e. set \( \hat{B} =_{cl} B \) such that \( A_0, \ldots, A_n \leq_{ibT} \hat{B} \).

Proof. For a proof of (a), fix \( k \) minimal such that \( A \leq (i_k + k)_{ibT} B_0, \ldots, B_n \) and let \( \hat{A} = A + k \). Then, by the Bounded-Shift Lemma, \( \hat{A} =_{cl} A \), and by Proposition 6.2, \( \hat{A} \leq_{ibT} B_0, \ldots, B_n \).

The proof of (b) is similar: Given \( k \) such that \( A_0, \ldots, A_n \leq (i_k + k)_{ibT} B \), the set \( \hat{B} = B - k \) will have the required properties. \( \square \)

Lemma 6.4 (ibT-Meet Lemma) Let \( A, B_0, \ldots, B_n \) \((n \geq 0)\) be c.e. sets such that

\[
\deg_{ibT}(A) = \deg_{ibT}(B_0) \land \cdots \land \deg_{ibT}(B_n). \tag{6.1}
\]

Then

\[
\deg_{cl}(A) = \deg_{cl}(B_0) \land \cdots \land \deg_{cl}(B_n). \tag{6.2}
\]

Proof. Since \( A \) is a lower bound of \( B_0, \ldots, B_n \) with respect to ibT-reducibility and since ibT-reducibility is stronger than cl-reducibility, \( A \) is a lower bound of \( B_0, \ldots, B_n \) with respect to cl-reducibility too. So given a c.e. set \( C \) such that \( C \leq_{cl} B_0, \ldots, B_n \), it suffices to show that \( C \leq_{cl} A \). By the first part of the cl-ibT-Conversion Lemma, there is a c.e. set \( \hat{C} \) such that \( \hat{C} =_{cl} C \) and \( \hat{C} \leq_{ibT} B_0, \ldots, B_n \). It follows with (6.1) that \( \hat{C} \leq_{ibT} A \). So, by \( \hat{C} =_{cl} C \), \( C \leq_{cl} A \). \( \square \)

Lemma 6.5 (ibT-Join Lemma) Let \( A, B_0, \ldots, B_n \) \((n \geq 0)\) be c.e. sets such that

\[
\deg_{ibT}(A) = \deg_{ibT}(B_0) \lor \cdots \lor \deg_{ibT}(B_n).
\]

Then

\[
\deg_{cl}(A) = \deg_{cl}(B_0) \lor \cdots \lor \deg_{cl}(B_n).
\]

Proof. This easily follows from the second part of the cl-ibT-Conversion Lemma just as the ibT-Meet Lemma followed from the first part. \( \square \)
6.3 Comparing joins and meets in the cl- and bT-degrees

Having seen that joins and meets in the c.e. ibT-degrees are preserved in the c.e. cl-degrees, we now turn to the corresponding question for the cl-degrees and the bt-degrees. Again we start with some conversion lemma.

Lemma 6.6 (bt-ibT-Conversion Lemma)  
(a) Let \( A, B_0, \ldots, B_n \) be c.e. sets such that \( A \leq_{bt} B_0, \ldots, B_n \). There is a c.e. set \( \hat{A} =_{bt} A \) such that \( \hat{A} \leq_{ibT} B_0, \ldots, B_n \).

(b) Let \( A, B \) be c.e. sets such that \( A \leq_{bt} B \). There is a c.e. set \( \hat{B} =_{bt} B \) such that \( A \leq_{ibT} \hat{B} \).

Proof.  
(a) Fix bt-reductions \( A = \Gamma^j \), let \( f_j \) be computable bounds on the use functions of these reductions, let \( f \) be any strictly increasing computable function which dominates the functions \( f_j \) (\( j \leq n \)), and let \( \hat{A} = A_f \) be the \( f \)-shift of \( A \). Then, as one can easily check, \( \hat{A} \) has the required properties.

(b) Since the claim is trivial for computable \( A \), w.l.o.g. we may assume that \( A \) is not computable hence infinite. Fix an infinite computable subset \( C \) of \( A \) let \( f \) enumerate \( C \) in order (note that \( f \) is an unbounded computable shift), and let \( \hat{B} = (A \setminus C) \cup B_f \). Again, one can easily check, that \( \hat{B} \) has the required properties.

\[ \square \]

Remark. In the second part of the bt-ibT-Conversion Lemma we cannot replace the set \( A \) by a pair \( A_0, A_1 \) or a finite sequence sequence \( A(j) \ldots A_n \) (\( n \geq 1 \)) of c.e. sets (even if we replace ibT by cl). (This follows form the existence of maximal pairs (which we will show later) and the fact that the partial ordering of the c.e. bt-degrees is an upper semi-lattice.) So, here the conversion lemma only implies the following meet lemma but not the corresponding join lemma.

Lemma 6.7 (cl-Meet Lemma)  
Let \( A, B_0, \ldots, B_n \) (\( n \geq 0 \)) be c.e. sets such that

\[ \deg_{cl}(A) = \deg_{cl}(B_0) \land \cdots \land \deg_{cl}(B_n). \quad (6.3) \]

Then

\[ \deg_{bt}(A) = \deg_{bt}(B_0) \land \cdots \land \deg_{bt}(B_n) \quad (6.4) \]

Proof. This follows from the first part of the bt-ibT-Conversion Lemma just as the ibT-Meet Lemma followed from the first part of the cl-ibT-Conversion Lemma.

\[ \square \]

Though it cannot be deduced from the bt-ibT-Conversion Lemma we still obtain the dual of Lemma 6.7 for joins but the argument we have to use is a bit more sophisticated.
Lemma 6.8 (cl-Join Lemma; Ambos-Spies, Bodewig, Kräling and Yu) Let $A, B_0, \ldots, B_n$ ($n \geq 0$) be c.e. sets such that

$$\deg_{cl}(A) = \deg_{cl}(B_0) \lor \cdots \lor \deg_{cl}(B_n). \quad (6.5)$$

Then

$$\deg_{bt}(A) = \deg_{bt}(B_0) \lor \cdots \lor \deg_{bt}(B_n). \quad (6.6)$$

Proof. Since cl-reducibility is stronger than bt-reducibility (and since the partial ordering of the c.e. bt-degrees is an upper semi-lattice), (6.5) implies

$$\deg_{bt}(A) \geq \deg_{bt}(B_0) \lor \cdots \lor \deg_{bt}(B_n).$$

So, by

$$\deg_{bt}(B_0) \lor \cdots \lor \deg_{bt}(B_n) = \deg_{bt}(B_0 \oplus \cdots \oplus B_n), \quad (6.7)$$

it suffices to prove

$$A \leq_{bt} B_0 \oplus \cdots \oplus B_n. \quad (6.7)$$

W.l.o.g. we may assume that the sets $A, B_0, \ldots, B_n$ are not computable. By (6.5) fix $k \geq 0$ such that $B_i \leq_{(i+k)bt} A$ ($i \leq n$). Then, by the Representation Lemma, w.l.o.g. we may assume that there are one-to-one computable enumerations $a$ and $b_i$ of $A$ and $B_i$, respectively, such that

$$\forall s \geq 0 [a(s) \leq \min(b_0(s), \ldots, b_n(s)) + k]. \quad (6.8)$$

Fix an infinite computable subset $R$ of $A$ and let $r(n)$ be the $n$th element of $R$ in order. Note that, by noncomputability of $A$, $\overline{R}$ is infinite whence $r$ is an unbounded computable shift. Define the c.e. set $D \subseteq \overline{R}$ by

$$D = \{a(s) : a(s) \notin R \& \min(b_0(s), \ldots, b_n(s)) < r(a(s)).\}$$

We claim that

$$\forall i \leq n (B_i \leq_{cl} D \cup (A)_r). \quad (6.9)$$

Fix $i$ and $x$. It suffices to compute $B_i(x)$ from $(D \cup (A)_r) \upharpoonright (x+k+1)$. Note that, by (6.8), $x \in B_i$ if and only if there is a number $y \leq x+k$ such that $y \in A$ and, for the unique $s$ such that $y = a(s)$, $x = b_i(s)$. Now, by distinguishing three possible cases for $y = a(s) \in A$, namely (i) $y \in R$, (ii) $y \in A \setminus R$ and $\min(b_0(s), \ldots, b_n(s)) < r(a(s))$ and (iii) $y \in A \setminus R$ and $\min(b_0(s), \ldots, b_n(s)) \geq r(a(s))$, it follows from the definition of $D$, that $x \in B_i$ if and only if

(i) there is a number $y \leq x+k$ in $R \subseteq A$ such that, for the unique $s$ such that $y = a(s), x = b_i(s)$ or
(ii) there is a number \( y \leq x + k \) in \( D \subseteq (A \setminus R) \) such that, for the unique \( s \) such that \( y = a(s) \), \( x = b_i(s) \) or

(iii) There is a number \( y \) in \( A \setminus R \) such that \( r(y) \leq x \) and, for the unique \( s \) such that \( y = a(s) \), \( x = b_i(s) \).

Note that, by computability of \( R \), (i) is decidable (without oracle) while (ii) and (iii) can be decided with oracle \( D \upharpoonright (x + k + 1) \) and \( (A)_r \upharpoonright x \), respectively. By disjointness of \( D \) and \( (A)_r \), this implies the claim.

Now (6.7) is established as follows. By (6.5) and (6.9), \( A \leq_{cl} D \cup (A)_r \). So, by the Computable-Shift Lemma, \( A \leq_{cl} D \) hence \( A \leq_{bt} D \). On the other hand, \( D \leq_{bt} B_0 \oplus \cdots \oplus B_n \) since, by definition, in order to compute \( D(x) \) it suffices to know \( B_0 \upharpoonright r(x), \ldots, B_n \upharpoonright r(x) \) where \( r \) is computable.

\[ \square \]

6.4 Pairs without meets in the c.e. bt-degrees

As some first applications of the meet lemmas we show that the fact that the partial ordering of the c.e. bt-degrees is not a lower semi-lattice carries over to the partial orderings of the c.e. cl and ibT-degrees and that minimal pairs in the c.e. bt, cl and ibT-degrees coincide. Here a pair \( (a, b) \) is a minimal pair if \( a, b > 0 \) and \( a \land b = 0 \).

**Theorem 6.9** (Downey and Hirschfeldt [DH10]) For \( r \in \{ibT, cl\} \), there are c.e. \( r \)-degrees \( a \) and \( b \) such that \( a \land b \) does not exist.

**Proof.** This theorem was originally proven by Downey and Hirschfeldt by a direct construction. Since Jockusch [Jo81] has shown that there is a pair of c.e. sets \( A \) and \( B \) such that \( deg_{bt}(A) \land deg_{bt}(B) \) does not exist, we obtain the following simple alternative proof. Namely, from Jockusch’s result it follows by Lemmas 6.7 and 6.4 that \( deg_{cl}(A) \land deg_{cl}(B) \) and \( deg_{ibT}(A) \land deg_{ibT}(B) \) do not exist too. \[ \square \]
Chapter 7

The theories of the c.e. sbT-degrees: noncategoricity and undecidability

In this chapter we transfer some important results from the bounded Turing degrees to the strongly bounded Turing degrees. In particular we show that, for \( r = \text{ibT}, \text{cl} \), the first order theory \( \text{Th}(R_r, \leq) \) of the c.e. \( r \) degrees realizes infinitely many one-types, hence is not countably categorical, and that \( \text{Th}(R_r, \leq) \) is undecidable.\(^1\)

In the first part of this chapter, we start with the observation that minimal pairs in the c.e. ibT-, cl- and bT-degrees coincide (Minimal Pair Lemma). Then we observe that joins in the c.e. strongly bounded Turing degrees represented by disjoint sets (locally) have the distributivity property (Distributivity Lemma). This observation will be used in the Embedding Lemma where configurations are described which yield embeddings of the finite Boolean algebras into the partial orderings of the c.e. ibT- and cl-degrees. We then show that the above observations give us the embeddability of all finite distributive lattices into the partial orderings of the c.e. ibT- and cl-degrees by using some results on minimal pairs in the c.e. Turing degrees. We complete this first part with some observations on nonbounding and nontop degrees, i.e., degrees we do not bound a minimal pair and which are not the join of a minimal pair.

Then, by using the above results and the Transfer Lemmas, we prove the above mentioned noncategoricity and undecidability results, by transferring corresponding results for the c.e. bT-degrees from the literature to the c.e. strongly bounded Turing degrees.

\(^1\)The proof of the latter (as presented in the original paper Ambos-Spies [Am10]) is included in these lecture notes. In the lecture itself we will only present the results on 1-types in some detail.
7.1 Minimal pairs, embedding distributive lattices, and nonbounding degrees

We first observe that minimal pairs in the c.e. bT, cl and ibT-degrees coincide. Recall that \((a, b)\) is a minimal pair if \(a, b > 0\) and \(a \land b = 0\).

**Lemma 7.1 (Minimal-Pair Lemma [ADFM10])** For c.e. sets \(A\) and \(B\) the following are equivalent.

(i) The pair \((\text{deg}_{bT}(A), \text{deg}_{bT}(B))\) is a minimal pair of bT-degrees.

(ii) The pair \((\text{deg}_{cl}(A), \text{deg}_{cl}(B))\) is a minimal pair of cl-degrees.

(iii) The pair \((\text{deg}_{ibT}(A), \text{deg}_{ibT}(B))\) is a minimal pair of ibT-degrees.

**Proof.** The implications \((i) \Rightarrow (ii)\) and \((ii) \Rightarrow (iii)\) are immediate by the fact that \(\leq_{cl}\) is stronger than \(\leq_{bT}\) and \(\leq_{ibT}\) is stronger than \(\leq_{cl}\). The implication \((iii) \Rightarrow (ii)\) and \((ii) \Rightarrow (i)\) hold by the ibT-Meet Lemma and the cl-Meet Lemma, respectively. □

It might be of interest to note that we cannot expand Lemma 7.1 by adding Turing reducibility: As mentioned before, there is a bT-minimal pair \((A, B)\) such that \(A =_T B\) (whence the pair \((\text{deg}_{T}(A), \text{deg}_{T}(B))\) of Turing degrees is not minimal). For halves of minimal pairs, however, Ambos-Spies [Am85] has shown that a c.e. set \(A\) is half of a bT-minimal pair if and only if \(A\) is half of a T-minimal pair. So a c.e. set \(A\) is half of a T-minimal pair iff \(A\) is half of a bT-minimal pair iff \(A\) is half of an cl-minimal pair iff \(A\) is half of an ibT-minimal pair.

In order to extend some existence results for minimal pairs to embedding results for distributive lattice we first need some facts on distributivity in the c.e. degrees.

As Lachlan has shown (see Stob [St83]), the upper semi-lattice of the c.e. bT-degrees is distributive, i.e., satisfies the following distributivity law for upper semi-lattices:

\[
\forall a_0, a_1, b \ [b \leq a_0 \lor a_1 \implies \exists b_0, b_1 \leq a_0, a_1 \ (b = b_0 \lor b_1)]. \tag{7.1}
\]

Note that a lattice is distributive in the common sense if and only if it satisfies (7.1). Moreover, no nondistributive lattice can be embedded into any distributive upper semi-lattice. Recently, Ambos-Spies, Bodewig, Kr"aling and Yu [ABKY10] have shown that the nonmodular five-element lattice \(N_5\) can be embedded into the partial
orderings of the c.e. ibT- and cl-degrees. (We will present this result in Chapter 10.) So there are joins in these degree structures for which (7.1) fails. Any join represented by a disjoint union of c.e. sets, however, has the distributive splitting property.

**Lemma 7.2 (Distributivity Lemma)** Let $A_0, \ldots, A_m$ $(m \geq 1)$ be pairwise disjoint c.e. sets, let $A = A_0 \cup \cdots \cup A_m$, and let $B$ be a c.e. set such that $B \leq_{ibT} (i+k) A$ $(k \geq 0)$. There is a c.e. set $\hat{B} = ibT A$ and a splitting $\hat{B} = B_0 \cup \cdots \cup B_m$ of $\hat{B}$ into pairwise disjoint c.e. sets $B_j$ such that $B_j \leq_{ibT} A_j$ $(j \leq m)$.

**Proof.** Choose $\hat{A}, \hat{B}, \{a(n)\}_{n \geq 0}$ and $\{b(n)\}_{n \geq 0}$ as in the Representation Lemma and let $B_j = \{b(n) : a(n) \in A_j\}$. □

**Lemma 7.3 (Embedding Lemma)** Let $r \in \{ibT, cl\}$ and let $A_0, \ldots, A_{n-1}$ $(n > 0)$ be noncomputable pairwise disjoint c.e. sets such that

$$\forall i, j < n (i \neq j \Rightarrow \deg_r(A_i) \land \deg_r(A_j) = 0).$$

Then $f_r : POWER(\{0, \ldots, n-1\}) \rightarrow R_r$ defined by

$$f_r(\alpha) = deg_r(A_\alpha) \text{ where } A_\alpha = \bigcup_{i \in \alpha} A_i$$

defines a lattice embedding of the $n$-atom Boolean algebra

$$\mathcal{B}_n = POWER(\{0, \ldots, n-1\}), \subseteq$$

into the partial ordering $(R_r, \leq)$ of the c.e. $r$-degrees which preserves the least element.

**Proof.** Since $\leq_{ibT}$ is stronger than $\leq_{cl}$ and since, by the ibT-Meet and ibT-Join Lemmas, joins and meets in the ibT-degrees constitute joins and meets in the cl-degrees, given $\alpha, \beta \subseteq \{0, \ldots, n-1\}$ it suffices to show

$$\alpha \subseteq \beta \Rightarrow A_\alpha \leq_{ibT} A_\beta \text{ (ordering)}$$

$$\alpha \not\subseteq \beta \Rightarrow A_\alpha \not\leq_{cl} A_\beta \text{ (non-ordering)}$$

$$\deg_{ibT}(A_\alpha) \lor \deg_{ibT}(A_\beta) = \deg_{ibT}(A_{\alpha \cup \beta}) \text{ (joins)}$$

$$\deg_{ibT}(A_\alpha) \land \deg_{ibT}(A_\beta) = \deg_{ibT}(A_{\alpha \cap \beta}) \text{ (meets)}$$

Moreover, by the Minimal-Pair Lemma, we may assume that (7.2) holds for both, $r = ibT$ and $r = cl$.

Now, (7.3) and (7.5) are immediate by the Splitting Lemma.
Theories of the C.E. sbT-Degrees: Noncategoricity and Undecidability

For a proof of (7.4), given \( \alpha \) and \( \beta \) such that \( \alpha \not\subseteq \beta \), fix \( i \in \alpha \setminus \beta \). By (7.3), it suffices to show that \( A_i \not\leq \text{cl} A \beta \). For a contradiction assume that \( A_i \leq \text{cl} A \beta \). Then by the Distributivity Lemma, there is a splitting of \( A_i \) into pairwise disjoint c.e. sets \( A_{i,j} \), \( j \in \beta \), such that \( A_{i,j} \leq \text{cl} A_j \) and, by the Splitting Lemma,

\[
\text{deg}_{\text{cl}}(A_i) = \bigvee_{j \in \beta} \text{deg}_{\text{cl}}(A_{i,j}).
\]

(7.7)

Hence, for \( j \in \beta \), \( A_{i,j} \leq \text{cl} A_i, A_j \) and, by \( i \neq j \), \( (A_i, A_j) \) is an cl-minimal pair. So \( A_{i,j} \) is computable. It follows with (7.7) that \( A_i \) is computable too. But this contradicts the choice of the sets \( A_i \).

Finally, for a proof of (7.6), by (7.3) it suffices to show that, for any c.e. set \( B \),

\[
[B \leq \text{ibT} A \alpha \& B \leq \text{ibT} A \beta] \Rightarrow B \leq \text{ibT} A_{\alpha \cap \beta}
\]

holds. So fix \( B \) such that \( B \leq \text{ibT} A \alpha \) and \( B \leq \text{ibT} A \beta \). By the former and by the Distributivity Lemma there are pairwise disjoint c.e. sets \( B_i \leq \text{cl} A_i \), \( i \in \alpha \), such that \( B = \bigcup_{i \in \alpha} B_i \). It follows by the Splitting Lemma that \( B_i \leq \text{cl} B \) whence, by \( B \leq \text{ibT} A \beta \), \( B_i \leq \text{cl} A \beta \). So, again by the Distributivity Lemma, there are pairwise disjoint c.e. sets \( B_{i,j} \leq \text{cl} A_j \) (\( i \in \alpha \), \( j \in \beta \)) such that \( B_i = \bigcup_{j \in \beta} B_{i,j} \). Hence, by the Splitting Lemma, \( B_{i,j} \leq \text{cl} A_i, A_j \) and \( B \) is the disjoint union

\[
B = \bigcup_{(i,j) \in \alpha \times \beta} B_{i,j}.
\]

By the former, for \( i \neq j \), \( B_{i,j} \) is computable since \( (A_i, A_j) \) is an cl-minimal pair. So

\[
B = \text{cl} \bigcup_{i \in \alpha \cap \beta} B_{i,i}
\]

where \( B_{i,i} \leq \text{cl} A_i \). Hence, by the Splitting Lemma, \( B \leq \text{cl} A_{\alpha \cap \beta} \). \( \square \)

**Theorem 7.4** Let \( r \in \{ \text{ibT}, \text{cl} \} \). Any finite distributive lattice \( L \) is embeddable (as a lattice) into the partial ordering of the c.e. \( r \)-degrees by a map which preserves the least element.

**Proof.** For any finite distributive lattice \( L \) there is some \( n \geq 2 \) such that \( L \) can be embedded into the \( n \)-atom Boolean algebra \( \mathcal{B}_n \) by a map which preserves the least element. So, given \( n \geq 2 \) and \( r \in \{ \text{ibT}, \text{cl} \} \), it suffices to embed \( \mathcal{B}_n \) into the partial ordering \( (\mathcal{R}_r, \leq) \) by a map which preserves the least element. By the Embedding Lemma, the latter can be done by giving pairwise disjoint c.e. sets \( A_0, \ldots, A_{n-1} \) which are pairwise \( r \)-minimal pairs.

We obtain such sets by various results in the literature. For instance, Thomason [Th71] has shown that, for any \( n \geq 0 \), there are c.e. T-degrees \( a_i \) (\( i < n \)) which are...
pairwise T-minimal pairs. Since any T-minimal pair is a fortiori an r-minimal pair for \( r \in \{ \text{ibT}, \text{cl} \} \), it suffices to choose any pairwise disjoint c.e. sets \( A_i \in a_i (i < n) \).

We close this section with some results on bounds of minimal pairs. A c.e. r-degree \( a \) is called bounding if there is a minimal pair \((a_0, a_1)\) of c.e. r-degrees such that \( a_0, a_1 \leq a \), and \( a \) is called nonbounding otherwise. A c.e. r-degree \( a \) is called a top (of a minimal pair) if there is a minimal pair \((a_0, a_1)\) of c.e. r-degrees such that \( a_0 \lor a_1 = a \), and \( a \) is called a nontop otherwise.

Lemma 7.5 (Nonbounding Lemma) For any noncomputable c.e. set \( A \) the following are equivalent.

(i) \( \text{deg}_{\text{ibT}}(A) \) is bounding.

(ii) \( \text{deg}_{\text{cl}}(A) \) is bounding.

(iii) \( \text{deg}_{\text{ibT}}(A) \) is bounding.

Proof. For the proof of (i) \( \Rightarrow \) (ii) assume that \( \text{deg}_{\text{ibT}}(A) \) is bounding. Then there are noncomputable c.e. sets \( B, C \leq_{\text{ibT}} A \) such that \((\text{deg}_{\text{ibT}}(B), \text{deg}_{\text{ibT}}(C))\) is a minimal pair. By the bT-ibT-Conversion Lemma there are c.e. sets \( \hat{B}, \hat{C} \leq_{\text{cl}} A \) such that \( \hat{B} \equiv_{\text{bT}} B \) and \( \hat{C} \equiv_{\text{bT}} C \). By the latter, \((\text{deg}_{\text{ibT}}(\hat{B}), \text{deg}_{\text{ibT}}(\hat{C}))\) is a minimal pair too. It follows by (the trivial direction of) the Minimal-Pair Lemma that \( \text{deg}_{\text{cl}}(A) \) bounds the minimal pair \((\text{deg}_{\text{cl}}(\hat{B}), \text{deg}_{\text{cl}}(\hat{C}))\).

The proof of (ii) \( \Rightarrow \) (iii) is similar to the proof of (i) \( \Rightarrow \) (ii) (now applying the cl-ibT-Conversion Lemma).

Finally, for a proof of (iii) \( \Rightarrow \) (i), assume that \( \text{deg}_{\text{ibT}}(A) \) is bounding. Fix noncomputable c.e. sets \( B, C \leq_{\text{ibT}} A \) such that \((\text{deg}_{\text{ibT}}(B), \text{deg}_{\text{ibT}}(C))\) is a minimal pair. Then a fortiori \( B, C \leq_{\text{bT}} A \) and by (the nontrivial direction of) the Minimal-Pair Lemma, \((\text{deg}_{\text{bT}}(B), \text{deg}_{\text{bT}}(C))\) is a minimal pair too. So \( \text{deg}_{\text{bT}}(A) \) is bounding.

\[ \square \]

Lemma 7.6 (Nontop Lemma) For any noncomputable c.e. set \( A \) the following hold.

\( \text{deg}_{\text{bT}}(A) \) is a nontop

\[ \Downarrow \]

\( \text{deg}_{\text{cl}}(A) \) is a nontop

\[ \Downarrow \]

\( \text{deg}_{\text{ibT}}(A) \) is a nontop
7. The theories of the c.e. sbT-degrees: noncategoricity and undecidability

Proof. The proof is by contraposition. If \( \deg_{\text{ibT}}(A) \) is a top, say \( \deg_{\text{ibT}}(A) = \deg_{\text{ibT}}(A_0) \lor \deg_{\text{ibT}}(A_1) \) for the ibT-minimal pair \( (\deg_{\text{ibT}}(A_0), \deg_{\text{ibT}}(A_1)) \) then, by the Minimal Pair Lemma, \( (\deg_{\text{cl}}(A_0), \deg_{\text{cl}}(A_1)) \) is an cl-minimal pair and, by the ibT-Join Lemma, \( \deg_{\text{cl}}(A) = \deg_{\text{cl}}(A_0) \lor \deg_{\text{cl}}(A_1) \). So \( \deg_{\text{cl}}(A) \) is a top too.

The proof that, for any top \( \deg_{\text{cl}}(A), \deg_{\text{ibT}}(A) \) is a top is similar (using the cl-Join Lemma in place of the ibT-Join Lemma).

□

7.2 The theories of the partial orderings of the c.e. sbT-degrees are not \( \omega \)-categorical

Ambos-Spies and Soare [AS89] have shown that the theory of the c.e. bT-degrees realizes infinitely many 1-types. They showed that, for any number \( n \geq 2 \), there is an \( n \)-bounding c.e. bT-degree which is not \( (n + 1) \)-bounding, where a c.e. \( r \)-degree \( b \) is \( n \)-bounding if there are \( n \) c.e. \( r \)-degrees \( a_0, \ldots, a_{n-1} < b \) which are pairwise minimal pairs. Ambos-Spies and Soare deduce this result from the following technical lemma (which is proven by a quite sophisticated \( 0'' \)-priority argument) by exploiting distributivity of the upper semi-lattice of the c.e. bT-degrees.

Lemma 7.7 (Ambos-Spies and Soare [AS89]) For any \( n \geq 2 \) there are c.e. bT-degrees \( a_0, \ldots, a_{n-1} > 0 \) such that

\[
\forall i, j < n (i \neq j \Rightarrow a_i \land a_j = 0) \quad (7.8)
\]

\[
\forall i < n (a_i \text{ is nonbounding}) \quad (7.9)
\]

As we will show next, our preceding observations on the relations between the strongly bounded Turing reducibilities and bounded Turing reducibility together with the local distributivity phenomena in the c.e. ibT and cl-degrees established in the preceding section will allow us to deduce the existence of infinitely many 1-types in the strongly bounded Turing degrees of the c.e. sets from Lemma 7.7 too.

Theorem 7.8 (Ambos-Spies [Am10]) For \( r \in \{\text{ibT}, \text{cl}\} \), the elementary theory \( \text{Th}(R_r, \leq) \) of the partial ordering of c.e. \( r \)-degrees realizes infinitely many 1-types. So \( \text{Th}(R_r, \leq) \) is not \( \mathfrak{K}_0 \)-categorical.
7. The theories of the partial orderings of the c.e. sbT-degrees are undecidable

PROOF. Given $n \geq 2$ it suffices to show that there is a c.e. $r$-degree $b$ which is $n$-bounding but not $(n+1)$-bounding (hence not $m$-bounding for $m > n$). By Lemma 7.7 fix c.e. bT-degrees $a_0, \ldots, a_{n-1} > 0$ such that (7.8) and (7.9) hold, choose pairwise disjoint c.e. sets $A_i \in a_i$ ($i < n$), let $B = A_0 \cup \cdots \cup A_{n-1}$, and let $b$ be the $r$-degree of $B$.

Then $b$ is $n$-bounding since, by the Splitting Lemma, $b$ bounds the c.e. $r$-degrees $\hat{a}_i = \deg_r(A_i)$ ($i < n$), and, by Lemma 7.1 and by (7.8), the degrees $\hat{a}_i$ are pairwise minimal pairs.

It remains to show that $b$ is not $(n+1)$-bounding. For a contradiction assume that there are noncomputable c.e. sets $C_0, \ldots, C_n \leq_r B$ such that

$$\forall j, j' \leq n (j \neq j' \Rightarrow \deg_r(C_j) \land \deg_r(C_{j'}) = 0).$$

(7.10)

By the Distributivity Lemma, split $C_j$ into pairwise disjoint c.e. sets $C_{j,i}$ such that $C_{j,i} \leq_r A_i$ ($i \leq n, j < n$). Then, by the Splitting Lemma and by noncomputability of $C_j$, for all $i < n$ $C_{j,i} \leq_r C_j$ and there is some $i < n$ such that $C_{j,i}$ is noncomputable. Let $i_j$ be the least such $i$. By the pigeon hole principle, fix $j \neq j'$ such that $i_j = i_{j'}$. It follows with (7.10) that $(\deg_r(C_{j,i_j}), \deg_r(C_{j',i_j}))$ is a minimal pair bounded by the $r$-degree $\hat{a}_{i_j}$. So $\hat{a}_{i_j}$ is bounding. It follows with Lemma 7.5 that the bT-degree $a_{i_j}$ is bounding too. But this contradicts (7.9).

7.3 The theories of the partial orderings of the c.e. sbT-degrees are undecidable

Our final result is the undecidability of the theories of the partial orderings of the c.e. ibT and cl-degrees. Just as in case of the preceding theorem on 1-types, our proof will be based on a proof of the undecidability of the theory of the partial ordering of the c.e. bT-degrees, namely on the proof of the undecidability of the $\Pi_4$-theory of this structure given in Lempp and Nies [LN95]. We will use the main technical lemma on the c.e. bT-degrees of [LN95] together with a sufficient condition for a partial ordering to have an undecidable theory given there too. We first state these results from [LN95].

Lemma 7.9 (Main Lemma of Lempp and Nies [LN95]) Let $n \geq 1$. There are noncomputable c.e. set $A_i$, $B_j$ and $D_{i,j}$ ($i, j < n$) such that

$$D_{i,j} \leq_{bT} A_i, B_j,$$

(7.11)
deg_{ST}(A_i) and deg_{ST}(B_j) are nontops, \( (7.12) \)

\[ \forall i, i' < n \ (i \neq i' \Rightarrow (deg_{ST}(A_i), deg_{ST}(A_{i'})) \text{ is a bT-minimal pair}), \] \( (7.13) \)

and

\[ \forall j, j' < n \ (j \neq j' \Rightarrow (deg_{ST}(B_j), deg_{ST}(B_{j'})) \text{ is a bT-minimal pair}). \] \( (7.14) \)

**Lemma 7.10 (Undecidability Lemma; [LN95])** Let \( P = (P, \leq) \) be a partial ordering with least element 0. Suppose that there is a first order formula \( \varphi(x,y) \) in the language of partial orderings such that, for any \( n \geq 1 \), there are elements \( a_i, b_j, d_{i,j} \) of \( P \) \((i, j < n)\) such that

(i) \[ 0 < d_{i,j} \leq a_i, b_j, \]

(ii) for any \( I \subseteq \{0, \ldots, n-1\} \times \{0, \ldots, n-1\} \), the supremum of the elements \( d_{i,j} \) of \( P \) with \((i, j) \in I\), \( \lor_{(i,j) \in d_{i,j}} \) exists in \((P, \leq)\),

(iii) for \( \hat{d}_{i,j} = \lor_{(i', j') \neq (i,j)} d_{i',j'} \), the infimum \( a_i \land b_j \land \hat{d}_{i,j} \) of \( a_i, b_j, \hat{d}_{i,j} \) exists and \( a_i \land b_j \land \hat{d}_{i,j} = 0 \), and

(iv) there are elements \( \hat{a}, \hat{b} \in P \) such that, for \( A = \{a_0, \ldots, a_{n-1}\} \) and \( B = \{b_0, \ldots, b_{n-1}\} \),

\[ P \models \varphi(a, \hat{a}) \iff a \in A \text{ and } P \models \varphi(b, \hat{b}) \iff b \in B. \]

Then the elementary theory \( \text{Th}(P, \leq) \) is undecidable.

Actually, Lemma 7.10 is stated in [LN95] (see Theorem 2.1 there) only for upper semi-lattices (and there it is stated as a criterion for proving \( \Pi_1 - \text{Th}(P, \leq) \) to be undecidable by imposing some bound on the complexity of the formula \( \varphi \)). But the existence of the joins required in the proof of Theorem 2.1 of [LN95] is guaranteed by clause (ii) above (which is not present in [LN95]). The idea of the proof of Lemma 7.10 is as follows. Given a finite bipartite graph \( G \) with left side \( \{0, \ldots, n-1\} \), right side \( \{0', \ldots, (n-1)\} \), and edge relation \( E \subseteq \{0, \ldots, n-1\} \times \{0', \ldots, (n-1)\} \), \( G \) can be defined in \((P, \leq)\) with parameters \( \hat{a}, \hat{b}, \) and \( \hat{c} = \lor_{(i,j) \in E} d_{i,j} \) (which exists by (ii)). Namely, by representing the left and right parts of the vertex set of \( G \) by \( A = \{a_0, \ldots, a_{n-1}\} \) and \( B = \{b_0, \ldots, b_{n-1}\} \), respectively, these parts are definable from \( \hat{a} \) and \( \hat{b} \) by the formula \( \varphi \). Finally, the edge relation becomes definable from \( \hat{c} \) by

\[ (i, j') \in E \Leftrightarrow \exists u \leq \hat{c} \ (u \neq 0 \ & \ u \leq a_i, b_j). \]

Since the theory of finite bipartite graphs with left and right sides of the same size is hereditarily undecidable, the above interpretation implies undecidability of \( \text{Th}(P, \leq) \). For details see [LN95].
In order to derive from Lemma 7.9 the necessary facts on the c.e. ibT and cl-degrees which will allow us to argue that the partial orderings \((R_{ibT}, \leq)\) and \((R_{cl}, \leq)\) satisfy the hypotheses of Lemma 7.10, we have to provide sufficient distributivity in these structures. So we will show first that splittings of the ibT and cl-degrees of sufficiently scattered sets are distributive.

**Definition 7.11** A set \(A\) is *strongly scattered* if there is a computable set \(R\) and a nondecreasing and unbounded computable function \(l : \omega \to \omega\) such that \(A \subseteq R\) and
\[
\forall n \in R \left( [n - 2l(n), n + 2l(n)] \cap R = \{n\} \right) \tag{7.15}
\]
holds. A c.e. \(r\)-degree \(a\) is *strongly scattered* if there is a strongly scattered c.e. set \(A \in a\).

**Lemma 7.12** Let \(r \in \{ibT, cl\}\). Suppose that \(A, B_0, B_1\) are c.e. sets such that \(A\) is strongly scattered and \(\text{deg}_r(A) = \text{deg}_r(B_0) \lor \text{deg}_r(B_1)\). There are pairwise disjoint c.e. sets \(\hat{B}_i\) such that
\[
\hat{B}_i \leq_r B_i \ (i \leq 1) \tag{7.16}
\]
and
\[
A =_r \hat{B}_0 \cup \hat{B}_1. \tag{7.17}
\]

**Proof.** We give the proof for \(r = cl\). The proof for \(r = ibT\) is similar.

Fix a computable set \(R\) and a nondecreasing and unbounded computable function \(l : \omega \to \omega\) such that \(A \subseteq R\) and (7.15) holds, and fix \(k \geq 0\) minimal such that \(B_0, B_1 \leq_{(i+k)ibT} A\). W.l.o.g. we may assume that \(l(n) > k\) (by letting \(l(n) = k + 1\) for the finitely many numbers \(n\) such that \(l(n) \leq k\) and by omitting these numbers \(n\) from \(R\) and \(A\)). Moreover, since any c.e. subset of \(A\) will be strongly scattered via \(R\) and \(l\) too, by the Representation Lemma, we may assume that there are computable one-to-one functions \(a, b_0\) and \(b_1\) enumerating \(A, B_0\) and \(B_1\), respectively, such that, for \(b(n) = \min(b_0(n), b_1(n))\),
\[
\forall n (a(n) \leq b(n) + k). \tag{7.18}
\]
Note that by \(a\) being a one-to-one enumeration of \(A\) and by choice of \(R\) and \(l\), for any numbers \(n\) and \(n'\),
\[
n \neq n' \Rightarrow [a(n) - k, a(n) + l(a(n))] \cap [a(n') - k, a(n') + l(a(n'))] = \emptyset \tag{7.19}
\]
holds.

Now, let
\[
\hat{B}_0 = \{ b_0(n) : n \geq 0 & b_0(n) \leq b_1(n) & b_0(n) < a(n) + l(a(n)) \},
\]
\[ \hat{B}_1 = \{ b_1(n) : n \geq 0 \& b_1(n) < b_0(n) \& b_1(n) < a(n) + l(a(n)) \}, \]

and

\[ \hat{A} = \{ a(n) + l(a(n)) : n \geq 0 \}. \]

Note that, by definition of the sets \( \hat{B}_1 \) and by (7.18), for any \( n \) such that \( b_i(n) \in \hat{B}_i \), \( b_j(n) \in [a(n) - k, a(n) + l(a(n)) - 1] \) and \( b_{i-1}(n) \notin \hat{B}_{i-1} \). So, by (7.19), the sets \( \hat{B}_0, \hat{B}_1, \hat{A} \) are pairwise disjoint. Moreover, \( \hat{A} = A_f \) for the computable unbounded shift \( f \) defined by \( f(n) = n + l(n) \), and \( \hat{B}_1 \leq_{\text{ibT}} B_i \) by permitting whence (7.16) holds.

For a proof of (7.17), note that, by the Splitting Lemma and by (7.16),

\[ \text{deg}_{\text{cl}}(\hat{B}_0 \cup \hat{B}_1) = \text{deg}_{\text{cl}}(\hat{B}_0) \lor \text{deg}_{\text{cl}}(\hat{B}_1) \leq \text{deg}_{\text{cl}}(B_0) \lor \text{deg}_{\text{cl}}(B_1) = \text{deg}_{\text{cl}}(A). \]

It remains to show that \( A \leq_{\text{cl}} \hat{B}_0 \cup \hat{B}_1 \). Since \( \text{deg}_{\text{cl}}(A) = \text{deg}_{\text{cl}}(B_0) \lor \text{deg}_{\text{cl}}(B_1) \) it suffices to show that \( B_i \leq_{\text{cl}} \hat{B}_0 \cup \hat{B}_1 \) \( (i = 0, 1) \). In fact, since the sets \( \hat{B}_0, \hat{B}_1, \hat{A} \) are pairwise disjoint and \( \hat{A} \) is a computable unbounded shift of \( A \), by the Computable-Shift Lemma it suffices to show \( B_i \leq_{\text{cl}} \hat{B}_0 \cup \hat{B}_1 \cup \hat{A} \). But the latter follows from the fact that if \( x = b_i(n) \) enters \( B_i \) at stage \( n \) then \( b_0(n) \leq b_i(n) \) and \( b_0(n) \) enters \( \hat{B}_0 \) at stage \( n \) or \( b_1(n) \leq b_i(n) \) enters \( \hat{B}_1 \) at stage \( n \) or \( a(n) + l(a(n)) \leq b_i(n) \) and \( a(n) + l(a(n)) \) enters \( \hat{A} \) at stage \( n \). So \( B_i \leq_{\text{ibT}} \hat{B}_0 \cup \hat{B}_1 \cup \hat{A} \) by permitting. \( \square \)

**Theorem 7.13 (Undecidability Theorem; Ambos-Spies [Am10])** Let \( r \in \{ \text{ibT}, \text{cm} \} \). The elementary theory \( \text{Th}(R_r, \leq) \) of the partial ordering of the c.e. \( r \)-degrees is undecidable.

**Proof.** Let \( \phi(x, y) \) express that \( x \) is minimal with the property that \( 0 < x < y \) and there is a \( z \) such that \( 0 < z < y \) and \( 0 = x \land z \) and \( y = x \lor z \). (i.e., \( x \) is a minimal element of the set of elements of the open interval \((0, y)\) which possess a complement in the closed interval \([0, y]\)). Then, given \( n \geq 1 \), it suffices to give c.e. \( r \)-degrees \( a_i, b_j \) and \( d_{i, j} \) \((i, j < n)\) satisfying the conditions (i) – (iv) of Lemma 7.10 in the partial ordering \( (R_r, \leq) \).

By Lemma 7.9 fix c.e. sets \( A_i, B_j \) and \( D_{i, j} \) \((i, j < n)\) such that (7.11) to (7.14) hold. Since, for any c.e. set \( C \) and any infinite computable set \( R \) there is a c.e. set \( \hat{C} \subseteq R \) which is bT equivalent to \( C \), w.l.o.g. we may assume that there are pairwise disjoint, infinite computable sets \( R^A_i, R^B_j \) and \( R^D_{i, j} \) such that \( A_i \subseteq R^A_i, B_j \subseteq R^B_j \) and \( D_{i, j} \subseteq R^D_{i, j} \) \((i, j < n)\) and such that there is a nondecreasing and unbounded computable function \( l \) such that (7.15) holds for

\[ R = \bigcup_{i < n} R^A_i \cup \bigcup_{j < n} R^B_j \cup \bigcup_{i, j < n} R^D_{i, j}. \]
Moreover, since (by (7.11)) $D_{i,j} \leq_{bT} A_i, B_j$, as in the proof of the bT-ibT-Conversion Lemma, for any sufficiently fast growing computable shift $f$, $D_{i,j} =_{bT} (D_{i,j})_f \leq_{bT} A_i, B_j$. So, by choosing $f$ so that $f(R^D_{i,j}) \subseteq R^D_{1,j}$, w.l.o.g. we may assume that

$$D_{i,j} \leq_{bT} A_i, B_j. \quad (7.20)$$

Finally, let $a_i = \deg_r(A_i)$, $b_j = \deg_r(B_j)$, and $d_{i,j} = \deg_r(D_{i,j})$, and let $\hat{a} = \deg_r(A)$ and $\hat{b} = \deg_r(B)$ where $A = A_0 \cup \cdots \cup A_{n-1}$ and $B = B_0 \cup \cdots \cup B_{n-1}$, and call the given sets the canonical representatives of the thus defined $r$-degrees.

For verifying conditions (i) - (iv) of Lemma 7.10 we start with some observations.

Since the sets $A_i$, $B_j$ and $D_{i,j}$ are pairwise disjoint, it follows by the Splitting Lemma that, for any finite collection $S_1, \ldots, S_m (m \geq 1)$ of these sets,

$$\deg_r(S_1 \cup \cdots \cup S_m) = \deg_r(S_1) \lor \cdots \lor \deg_r(S_m).$$

Moreover, since $S_1 \cup \cdots \cup S_m$ is contained in $R$, it follows that $S_1 \cup \cdots \cup S_m$ (hence $\deg_r(S_1 \cup \cdots \cup S_m)$) is strongly scattered. So, for any nonempty, finite collection $S$ of the degrees $a_i, b_j, d_{i,j}$ the join of $S$ exists and the join is strongly scattered and represented by the union of the canonical representatives of the members of $S$. We will tacitly use these facts in the following.

Next we observe that, by Lemma 7.6 and Lemma 7.1, in (7.12) - (7.14) we may replace $bT$-reducibility by $r$-reducibility:

$$\deg_r(A_i) \text{ and } \deg_r(B_j) \text{ are nontops}, \quad (7.21)$$

$$\forall i, i' < n (i \neq i' \Rightarrow (\deg_r(A_i), \deg_r(A_{i'})) \text{ is an } r\text{-minimal pair}), \quad (7.22)$$

and

$$\forall j, j' < n (j \neq j' \Rightarrow (\deg_r(B_j), \deg_r(B_{j'})) \text{ is an } r\text{-minimal pair}). \quad (7.23)$$

Moreover, by (7.22) and (7.23) and by the Embedding Lemma, for any nonempty $\alpha$ which is strictly contained in $\{0, \ldots, n-1\}$ and for $\overline{\alpha} = \{0, \ldots, n-1\} \setminus \alpha$,

$$0 < \deg_r(A_{\alpha}), \deg_r(A_{\overline{\alpha}}) \& \hat{a} = \deg_r(A_{\alpha}) \lor \deg_r(A_{\overline{\alpha}}) \& \deg_r(A_{\alpha}) \land \deg_r(A_{\overline{\alpha}}) = 0 \quad (7.24)$$

and

$$0 < \deg_r(B_{\alpha}), \deg_r(B_{\overline{\alpha}}) \& \hat{a} = \deg_r(B_{\alpha}) \lor \deg_r(B_{\overline{\alpha}}) \& \deg_r(B_{\alpha}) \land \deg_r(B_{\overline{\alpha}}) = 0. \quad (7.25)$$
We are now ready to establish conditions (i) - (iv) of Lemma 7.10. Condition (i) is immediately by noncomputability of the sets \( D_{i,j} \) and by (7.20), and condition (ii) is immediate by the preceding observations on joins.

For a proof of (iii) fix \( i, j < n \) and let \( \hat{d}_{i,j} = \vee_{(i',j') \neq (i,j)} d_{i',j'} \). Then \( \hat{d}_{i,j} \) is represented by the set \( \bigcup_{(i',j') \neq (i,j)} D_{i',j'} \). So, in order to show that \( a_i \land b_j \land \hat{d}_{i,j} = 0 \), it suffices to show that any given c.e. set \( E \) with

\[
E \leq_r A_i, B_j, \bigcup_{(i',j') \neq (i,j)} D_{i',j'}
\]

is computable. Now since

\[
\bigcup_{(i',j') \neq (i,j)} D_{i',j'} = \bigcup_{i' \neq i, j' < n} D_{i',j'} \cup \bigcup_{j' \neq j} D_{i,j'}
\]

by the Splitting Lemma, \( E \) can be split into disjoint c.e. sets

\[
E_0 \leq_r E, \bigcup_{i' \neq i, j' < n} D_{i',j'} \tag{7.26}
\]

and

\[
E_1 \leq_r E, \bigcup_{j' \neq j} D_{i,j'} \tag{7.27}
\]

and it suffices to show that \( E_0 \) and \( E_1 \) are computable. This is done as follows. Note that, by (7.20), \( \bigcup_{i' \neq i, j' < n} D_{i',j'} \leq_r A_{\{0, \ldots, n-1\} \setminus \{i\}} \) and, by choice of \( E \), \( E \leq_r A_i \). So \( E_0 \) is computable by (7.26) and (7.24). Computability of \( E_1 \) follows in a similar way from (7.27) and (7.25) by observing that \( \bigcup_{j' \neq j} D_{i,j'} \leq_r B_{\{0, \ldots, n-1\} \setminus \{j\}} \) and \( E \leq_r B_j \).

Finally, for a proof of (iv) it suffices to show that

\[
(R_{i,\leq}) = \varphi(a, \hat{a}) \iff a \in \{a_0, \ldots, a_{n-1}\}.
\]

(The proof of the corresponding claim for \( b, \hat{b}, b_0, \ldots, b_{n-1} \) in place of \( a, \hat{a}, a_0, \ldots, a_{n-1} \) is symmetric.) Note that, by (7.24), \( 0 < a_i < \hat{a} \) and \( a_i \) has a complement in \( \{0, \hat{a}\} \). So, by definition of \( \varphi \), it suffices to show that, for any c.e. \( r \)-degrees \( e \) and \( f \) in \( \{0, \hat{a}\} \) such that \( e \lor f = \hat{a} \) and \( e \land f = 0 \), there is some \( i < n \) such that \( a_i \leq e \).

Since \( \hat{a} = e \lor f \) and \( \hat{a} \) is strongly scattered it follows from Lemma 7.12 and the Distributivity Lemma, that, for any c.e. \( r \)-degree \( x \leq \hat{a} \) there are c.e. \( r \)-degrees \( x_0 \leq e \) and \( x_1 \leq f \) such that \( x = x_0 \lor x_1 \). So, in particular, for \( i < n \), there are c.e. \( r \)-degrees \( a_{i,0} \leq e \) and \( a_{i,1} \leq f \) such that \( a_i = a_{i,0} \lor a_{i,1} \). Note that, by the former and by \( e \land f = 0 \), \( a_{i,0} \land a_{i,1} = 0 \). Since \( a_i \) is a nonstop it follows that \( a_{i,0} = a_i \) or \( a_{i,1} = a_i \). Now, if the former happens for some \( i < n \) then we are done since \( a_{i,0} \leq e \). Otherwise, however, \( a_i \leq f \) for all \( i < n \) whence \( \hat{a} = a_0 \lor \cdots \lor a_{n-1} \leq f \) contradicting the choice of \( f \). So this case cannot occur.
This completes the proof of the theorem. □

OPEN PROBLEM For $r = bT, T$, the theory of the partial ordering of the c.e. $r$-degrees is not only undecidable but the theory has the highest possible degree, namely the degree of first order arithmetic. The question, whether this is also true for the strongly bounded Turing reducibilities $r = ibT, cl$ is open.
Maximal pairs in the c.e. sbT-degrees

One of the most striking differences between the c.e. sbT-degrees and the c.e. degrees under the classical reducibilities - like m, tt, bT and T - is the existence of so-called maximal pairs, i.e., pairs of c.e. sbT-degrees \( a \) and \( b \) such that there is no c.e. sbT-degree which is above both \( a \) and \( b \). The existence of such maximal pairs was independently shown by Barmpalias [Ba05] and Fan and Lu [FL05]. (In fact, maximal pairs were first introduced for the class of the cl-degrees of the left-computable reals - a degree structure which properly contains the c.e. cl-degrees - and the existence of maximal pairs in this structure was shown by Yu and Ding [YD04].)

In this chapter we present some recent results on maximal pairs. Before we prove the existence of maximal pairs in the sbT-degrees we first observe that the maximal pairs in the c.e. cl-degrees and in the c.e. ibT-degrees coincide. Then, by some variations of the maximal pair construction, we show that there is a pair of m-complete sets \( A \) and \( B \) such that the pair \((\deg_{sbT}(A), \deg_{sbT}(B))\) is maximal and we show that above any c.e. sbT-degree there is a maximal pair of c.e. sbT-degrees. Finally, we summarize some characterizations of the Turing degrees containing (halves of) maximal pairs of c.e. sbT-degrees in terms of array noncomputability.

All of the above extensions of the basic construction of a maximal pair are due to Ambos-Spies, Ding, Fan and Merkel [ADFM10].

8.1 Existence of maximal pairs

**Definition 8.1** A pair \((a, b)\) of c.e. \( r \)-degrees is a maximal pair (of c.e. \( r \)-degrees) if there is no c.e. \( r \)-degree \( c \) such that \( a \leq c \) and \( b \leq c \).

A pair \((A, B)\) of c.e. sets is an \( r \)-maximal pair if \((\deg_r(A), \deg_r(B))\) is a maximal pair of c.e. \( r \)-degrees.
Note that a maximal pair is incomparable and does not possess any upper bound (in the c.e. degrees). So, for a reducibility \( r \) such that there is a maximal pair of c.e. \( r \)-degrees there are no \( r \)-complete c.e. sets and the partial ordering of the c.e. \( r \)-degrees is not an upper semi-lattice. Hence, for instance, there are no \( r \)-maximal pairs for \( r = m, bT, T \).

As we will show now, however, maximal pairs exist in the c.e. sbT-degrees. By the following lemma it suffices to show this for the ibT-degrees.

**Lemma 8.2** Let \( A \) and \( B \) be c.e. sets. Then \( (A, B) \) is a \( cl \)-maximal pair if and only if \( (A, B) \) is a \( ibT \)-maximal pair.

**Proof.** Since \( \leq_{ibT} \) is stronger than \( \leq_{cl} \), any \( cl \)-maximal pair is an \( ibT \)-maximal pair too. The proof of the nontrivial implication is by contraposition. Assume that the pair \( (A, B) \) is not \( cl \)-maximal. Then there is a c.e. set \( C \) such that \( A, B \leq_{cl} C \).

It follows by part (b) of the \( cl \)-ibT-Conversion Lemma (Lemma 6.3) that there is a c.e. set \( \hat{C} \) such that \( A, B \leq_{ibT} \hat{C} \). So the pair \( (A, B) \) is not \( ibT \)-maximal too. \( \square \)

**Theorem 8.3** (Barmpalias [Ba05], Fan and Lu [FL05]) For \( r = ibT, cl \), here is an \( r \)-maximal pair of c.e. sets.

**Proof.** By Lemma 8.2 it suffices to construct a pair of c.e. sets \( A \) and \( B \) which is \( ibT \)-maximal. So, given a computable enumeration \( \{ \hat{\Phi}_{e} \}_{e \geq 0} \) of the ibT-functionals, it suffices to meet the requirements

\[
\mathcal{R}_e : A \neq \hat{\Phi}_{e_1}^{W_0} \lor B \neq \hat{\Phi}_{e_2}^{W_0}
\]

for all \( e = \langle e_0, e_1, e_2 \rangle \).

We effectively split \( \mathbb{N} \) into infinitely many intervals \( I_e \) such that

\[
|I_e| > |\bigcup_{\epsilon < e} I_{\epsilon}|
\]

whence

\[
2 \cdot |I_e| > |\bigcup_{\epsilon \leq e} I_{\epsilon}|
\]

(for all \( e \geq 0 \)). Then \( A \) and \( B \) are defined on \( I_e \) in such a way that \( \mathcal{R}_e \) is met:

\[
\exists x \in I_e \ (A(x) \neq \hat{\Phi}_{e_1}^{W_0}(x) \lor B(x) \neq \hat{\Phi}_{e_2}^{W_0}(x)).
\]

The basic idea for achieving this goal is as follows. Whenever (for the current approximations) \( A = \hat{\Phi}_{e_1}^{W_0} \) and \( B = \hat{\Phi}_{e_2}^{W_0} \) on the interval \( I_e \) then we put a new number \( x \) from intervals \( I_e \) either into \( A \) or \( B \). This will guarantee that requirement \( \mathcal{R}_e \) is met unless a new number \( y \leq x \) (hence a number from the intervals \( \bigcup_{e \leq e} I_{\epsilon} \) is
8.1. Existence of maximal pairs

enumerated into $W_{e_0}$ later. Now the sizes of the interval $I_e$ are chosen so that eventually $W_{e_0}$ cannot respond to an attack and the requirement will be met. Namely since we can put numbers $x$ from $I_e$ into $A$ or $B$, we can make $2|I_e|$ many attacks whereas $W_{e_0}$ can use only $\big|\bigcup_{c < e} I_c\big|$ numbers for responding to these attacks. So, by (8.2), eventually $W_{e_0}$ runs out of numbers.

We now describe the construction more formally.

The intervals $I_e$ are defined as follows. Let $\{x_e\}_{e \geq 0}$ be a computable sequence such that $x_0 = 0$ and $x_{e+1} > 2x_e$ for $e \geq 0$ (e.g., $x_{e+1} = 2x_e + 1$), and let $I_e = [x_e, x_{e+1})$ ($e \geq 0$). Note that, for any $e > 0$, (8.1) and hence (8.2) hold.

Now, at stage $s + 1$ of the construction of $A$ and $B$, requirement $\mathcal{R}_e$ ($e = (e_0, e_1, e_2)$) requires attention if $s > e$,

$$\forall x \in I_e \ [A_s(x) = \Phi^{W_{e_0}}_{e_1, s}(x) \land B_s(x) = \Phi^{W_{e_0}}_{e_2, s}(x)],$$

(8.4)

and

$$A_s \cap I_e \neq I_e \lor B_s \cap I_e \neq I_e.$$  

(8.5)

If $\mathcal{R}_e$ requires attention then the greatest number $x \in I_e \setminus A_s$ is put into $A_{s+1}$ and if there is no such number then the greatest number $x \in I_e \setminus B_s$ is put into $B_{s+1}$. (We could also take the least or any other number $x \in I_e \setminus A_s$ or $x \in I_e \setminus B_s$ and put it into $A_{s+1}$ or $B_{s+1}$, respectively. In particular, we could enumerate numbers into $A$ and $B$ alternatingly, using up the numbers from $I_e$ in decreasing or increasing order.)

Note that the above action ensures that

$$A_{s+1}(x) = 1 \neq 0 = \Phi^{W_{e_0}}_{e_1, s}(x)$$

(or similarly for $B$ and $e_2$ in place of $A$ and $e_1$). Hence the attack is successful unless a number $y \leq x$ is enumerated into $W_{e_0}$ later which allows to adjust the value of $\Phi^{W_{e_0}}_{e_1, s}(x)$. So we can show that requirement $\mathcal{R}_e$ is met as follows.

For a contradiction assume that $\mathcal{R}_e$ is not met. Then, in particular,

$$\forall x \in I_e \ [A(x) = \Phi^{W_{e_0}}_{e_1, 0}(x) \land B(x) = \Phi^{W_{e_0}}_{e_2, 0}(x)]$$

whence (8.4) holds for all sufficiently large $s$. Since a number from the interval $I_e$ is put into $A$ or $B$ only for the sake of meeting $\mathcal{R}_e$, it follows that, for $p = 2|I_e|$, $\mathcal{R}_e$ requires attention $p$ times. Let $s_0 + 1 < s_1 + 1 < \cdots < s_{p-1} + 1$ be the stages at which $\mathcal{R}_e$ requires attention and let $s_p$ be the least stage $s > s_{p-1} + 1$ such that (8.4) holds. Then, for $m < p$, (8.4) holds for $s = s_m$ and for $s = s_{m+1}$ and, by $\mathcal{R}_e$ requiring attention at stage $s_m + 1$,

$$A_{s_m+1}(x) = 1 \neq 0 = \Phi^{W_{e_0, s_m}}_{e_1, s_m}(x) \lor B_{s_m+1}(x) = 1 \neq 0 = \Phi^{W_{e_0, s_m}}_{e_2, s_m}(x)$$
for some \( x \in I_e \). So \( \hat{\Phi}_{e, s}^{W_0, s+1}(x) \neq \Phi_{e, s}^{W_0, s}(x) \) or \( \hat{\Phi}_{e+2, s+1}^{W_0, s}(x) \neq \Phi_{e+2, s}^{W_0, s}(x) \) whence a new number \( y \leq x \) has to enter \( W_0 \) at a stage \( t \) with \( s_m < t \leq s_m+1 \). Note that, by \( y \leq x \) and \( x \in I_e \), \( y \) has to be in \( I_e' \) for some \( e' \leq e \). So at least \( p \) elements from the intervals \( I_{e'} \), \( e' \leq e \), have to enter \( W_0 \). But, by \( p = 2|I_e| \) and by (8.2), this is impossible.

Note that in the above proof the strategies for meeting the individual requirements act independently of each other and there are no injuries. Since Theorem 8.3 implies Theorem 6.1 this also gives an injury-free proof for the existence of c.e. \( r \)-degrees \( a \) and \( b \) such that \( a \lor b \) does not exist \((r = \text{iT}, \text{cl})\).

### 8.2 Variations of the maximal pair construction

The above construction of an sbT-maximal pair \((A, B)\) can be modified in various ways. For example, we can assign the interval \( I_{2e} \) (in place of interval \( I_e \)) to requirement \( R_e \) and reserve the odd intervals \( I_{2e+1} \) for some coding. In this way we can construct an sbT-maximal pair \((A, B)\) of m-complete sets. It suffice to mix the maximal pair construction with the following coding procedure. Given a computable 1-1 enumeration function \( k(s) \) of the m-complete halting set \( K \), put \( x_{2k(s)+1} \) (i.e., the least element of the interval \( I_{2k(s)+1} \)) into \( A \) and \( B \) at stage \( s \). This does not interfere with the strategies for meeting the maximal pair requirements and guarantees that \( K \leq_m A, B \) via \( f(n) = x_{2n+1} \).

**Theorem 8.4** For \( r = \text{iT}, \text{cl} \), there is an \( r \)-maximal pair of m-complete c.e. sets.

By another variation of the maximal-pair technique we can show that there is an sbT-maximal pair above any c.e. set.

**Theorem 8.5** For \( r = \text{iT}, \text{cl} \) and for any c.e. set \( C \) there are c.e. sets \( A \) and \( B \) such that \( C \leq_r A, B \) and the pair \((A, B)\) is \( r \)-maximal.

The proof is based on the observation that in any c.e. itT-degree there are c.e. sets of low density. We first establish this fact.

The upper density \( \alpha \) and the lower density \( \beta \) of a set \( A \) are defined by

\[
\alpha = \limsup_{n \to \infty} \frac{|A \upharpoonright n|}{n} \quad \text{and} \quad \beta = \liminf_{n \to \infty} \frac{|A \upharpoonright n|}{n}.
\]
Lemma 8.6 Let $\hat{A}$ be a noncomputable c.e. set and let $k \geq 1$. There is a c.e. set $A$ and a strictly increasing computable sequence of numbers, $\{x_n\}_{n \geq 0}$, such that

$$A = \text{ibT} \hat{A} \text{ and } A \subseteq \hat{A},$$

(8.6)

$$x_0 = 0 \text{ and } \forall n \ (x_{n+1} - x_n > k \cdot x_n),$$

(8.7)

and

$$\forall n \geq 1 \ (|A \upharpoonright x_n| < \frac{1}{k} x_n).$$

(8.8)

Proof (Sketch). We distinguish two cases depending on the upper density $\alpha$ of $\hat{A}$.

Case 1: $\alpha < \frac{1}{k}$ Then w.l.o.g. we may assume that $|\hat{A} \upharpoonright n| < \frac{1}{k} \cdot n$ for all $n \geq 1$. (Note that this can be achieved by omitting a finite part of $\hat{A}$.)

Let $A = \hat{A}$ and let $\{x_n\}_{n \geq 0}$ be any computable sequence satisfying (8.7). Then, obviously, $A$ and $\{x_n\}_{n \geq 0}$ have the required properties.

Case 2: $\alpha \geq \frac{1}{k}$ Fix rational numbers $q_-$ and $q_+$ such that $q_- < \alpha < q_+$ and $q_+ - q_- < \frac{1}{k}$. W.l.o.g. we may assume that $|\hat{A} \upharpoonright n| < q_+ \cdot n$ for all $n \geq 1$. Moreover, by $q_- < \alpha$, for any number $m$ there are infinitely many numbers $m' > m$ such that

$$|\hat{A} \cap [m, m')| > q_- \cdot m'.

So, for any computable enumeration $\{\hat{A}_s\}_{s \geq 0}$ of $\hat{A}$, there are computable functions $s(m)$ and $f(m) > k \cdot m$ such that

$$|\hat{A}_{s(m)} \cap [m, f(m)]| > q_- \cdot f(m).$$

Now let $x_0 = 0$ and $x_{n+1} = f(x_n)$ and let $A = \hat{A} \setminus R$ for the computable subset

$$R = \bigcup_{n \geq 0} \hat{A}_{s(x)} \cap [x_n, f(x_n)]$$

of $\hat{A}$.

Then, as one can easily check, the set $A$ and the sequence $\{x_n\}_{n \geq 0}$ have the required properties. □

Proof of Theorem 8.5. By Lemma 8.2, it suffices to consider $r = \text{ibT}$. Fix a c.e. set $C$ and a computable enumeration $\{C_s\}_{s \geq 0}$ of $C$. By Lemma 8.6 w.l.o.g. we may assume that there is a strictly increasing computable sequence $\{x_n\}_{n \geq 0}$ such that $x_0 = 0$,

$$\forall n \geq 0 \ (x_{n+1} - x_n > 4 \cdot x_n),$$

(8.9)

and

$$\forall n \geq 1 \ (|C \upharpoonright x_n| < \frac{1}{4} \cdot x_n).$$

(8.10)
Now the construction of $A$ and $B$ is as follows. We combine the basic construction of a $\text{ibT}$-maximal pair with some direct coding. The maximal-pair part of the construction is just as in the proof Theorem 8.3 using the intervals $I_e$ defined by $I_e = [x_e, x_e+1)$ for $x_e$ as above. The only difference is that now, when attacking requirement $\mathcal{R}_e$, we put numbers into $A$ and $B$ alternatingly, always using the greatest number from $I_e$ not yet put into $A$ and $B$, respectively. Coding is as follows. In the second step of stage $s+1$, having completed the maximal-pair part of stage $s+1$, fix the least number $y \in C_{s+1} \setminus C_s$ (if any) and fix $e$ such that $y \in I_e$. Let $y_A$ be the greatest number $\leq y$ in $I_e$ which has not entered $A$ by the end of step 1 of stage $s+1$ of the construction, let $y_B$ be the greatest number $\leq y$ in $I_e$ which has not entered $B$ by the end of step 1 of stage $s+1$ of the construction, and put $y_A$ into $A$ and $y_B$ into $B$. (So $C \leq \text{cl} A, B$ by permitting.)

In order to show that the construction is correct, it suffices to argue that there are sufficiently many numbers in $I_e$ which can be put into $A$ and $B$ for the sake of meeting $\mathcal{R}_e$ and for the sake of coding $C$ into $A$ and $B$, respectively. (Since for meeting requirement $\mathcal{R}_e$ and for coding $C$ we always use the greatest possible numbers, it will follow that sufficiently small numbers for coding $C$ will be available.) Since the argument is symmetric, we only consider the case of $A$.

First note that we have to make at most $x_{e+1} + 1$ attacks on $\mathcal{R}_e$ since a new attack has to be started only if the previous attack is invalidated by a new number $< x_{e+1}$ entering $W_{e_1}$. So, by using numbers from $A$ and $B$ alternatingly in the attacks, at most $\frac{1}{2} x_{e+1} + 1$ numbers from $I_e$ are put into $A$ for the sake of meeting $\mathcal{R}_e$. On the other hand, for coding $C \cap I_e$ into $A$, by (8.10), at most $\frac{1}{4} x_{e+1}$ numbers from $I_e$ have to enter $A$. So it suffices to show that $\frac{3}{4} x_{e+1} + 1 \leq |I_e|$. But this is immediate by $|I_e| = x_{e+1} - x_e$ and by (8.9). □

OPEN PROBLEM Is every c.e. $\text{ibT}$-degree $c$ the meet of a maximal pair $(a, b)$ of c.e. $\text{ibT}$-degrees.

8.3 Maximal pairs and array noncomputability

Ambos-Spies, Ding, Fan and Merkle [ADFM10] have characterized maximal pairs in terms of array noncomputability.

Theorem 8.7 (Ambos-Spies, Ding, Fan and Merkle [ADFM10]) Let $r = \text{ibT}, \text{cl}$. For any c.e. set $C$ the following are equivalent.
(i) $\text{deg}_T(C)$ is array noncomputable.

(ii) There is a $r$-maximal pair $(A, B)$ such that $A =_T B =_T C$.

(iii) There is a $r$-maximal pair $(A, B)$ such that $A =_T C$.

Here array noncomputability is defined as follows.

**Definition 8.8 (Downey, Jockusch and Stob [DJS90])**

(i) A sequence of finite sets $\{F_n\}_{n \geq 0}$ is a very strong array (v.s.a.) if the following hold.

(a) There is a computable function $f$ such that $f(n)$ is the canonical index of $F_n$,

(b) $\bigcup_{n \geq 0} F_n = \omega$,

(c) $F_n \cap F_m = \emptyset$ if $m \neq n$, and

(d) $0 < |F_n| < |F_{n+1}|$ for all $n \geq 0$.

(ii) A c.e. set $A$ is array noncomputable (a.n.c.) with respect to the v.s.a. $\{F_n\}_{n \geq 0}$ (or $\{F_n\}_{n \geq 0}$-a.n.c. for short) if

$$\forall e \exists^* n \ (A \cap F_n = W_e \cap F_n).$$  

(8.11)

(iii) A c.e. set $A$ is array noncomputable (a.n.c.) if there is a v.s.a. $\{F_n\}_{n \geq 0}$ such that $A$ is $\{F_n\}_{n \geq 0}$-a.n.c. Otherwise, $A$ is array computable (a.c.).

(iv) A c.e. $r$-degree $a$ is array noncomputable (a.n.c.) if there is an a.n.c. set $A$ in $a$. Otherwise, $a$ is array computable (a.c.).

Downey, Jockusch and Stob [DJS90] have shown that the class of a.n.c. c.e. Turing degrees is closed upwards and contains the class $\mathbf{L}_2$ of the c.e. nonlow $\mathbf{2}$ Turing degrees. Moreover, there are low a.n.c. c.e. degrees, and the notion is non-trivial, i.e., there are c.e. array computable degrees $> 0$.

So, by Theorem 8.7, every non-low $\mathbf{2}$ c.e. set is Turing equivalent to c.e. sets $A$ and $B$ forming an $r$-maximal pair and there is an $r$-maximal pair of low c.e. sets $A$ and $B$ ($r = \text{ibT, cl}$). On the other hand, there is a noncomputable c.e. set $C$ such that there is no $r$-maximal pair $(A, B)$ such that $A$ or $B$ is Turing equivalent to $C$.

Ambos-Spies, Ding, Fan and Merkle [ADFM10] have also shown that any bT-complete set $A$ is half of an $r$-maximal pair $(A, B)$ whereas there is Turing complete set $A$ which does not have this property. In fact, any high c.e. Turing degree contains a set $A$ which is not half of any $r$-maximal pair of c.e. sets.

**OPEN PROBLEM.** Is there a c.e. set $C$ such that, for all c.e. sets $A =_T C$, $A$ is half of a ibT-maximal pair of c.e. sets?
C.e. Turing degrees and the priority method II

In this lecture we look at the infinite injury method, an extension of the finite injury method in which also infinitary requirements may occur. Just as there are numerous types of finite injury argument, there is a large variety of infinite injury arguments. In some cases only the negative requirements are infinitary, i.e., impose unbounded restraint; in other cases the positive requirements are infinitary, i.e., meeting such a requirement requires the enumeration of infinitely many numbers in one of the sets under construction; in some cases both happens, i.e., positive and negative requirements are infinitary. Here we only consider the case of infinitary negative requirements while the positive requirements are finitary. Though such arguments are considerably simpler than the general case, much of the new machinery introduced will also be the basic tools for handling more complex constructions. In particular the tree method, which allows us to model guesses about the possible outcomes of infinitary requirements is a standard tool for doing infinite injury arguments.

Here we will develop the basic format of a tree argument by constructing a minimal pair in the c.e. Turing degrees. In the following lectures we will use this minimal-pair technique for our further investigation of the c.e. degrees under the strongly bounded Turing reducibilities.

9.1 A minimal pair in the c.e. Turing degrees

It suffices to construct noncomputable c.e. sets \( A_0 \) and \( A_1 \) such that

\[
\forall C \left[ C \leq_T A_0, A_1 \Rightarrow C \text{ computable} \right]
\]

(9.1)

holds. The global goals of making the sets \( A_0 \) and \( A_1 \) noncomputable and satisfying (9.1) are split up into the noncomputability requirements

\[
\forall e+i : A_i \neq \varphi_e \quad (e \geq 0, i \leq 1)
\]
and minimal pair requirements

$$N_e : \Phi^{A_0}_{e_0} = \Phi^{A_1}_{e_1} \text{ total } \Rightarrow \Phi^{A_0}_{e_0} \text{ computable } \quad (e = \langle e_0, e_1 \rangle \geq 0)$$

where \( \{ \Phi^X_e \}_{e \geq x} \) and \( \{ \varphi_e \}_{e \geq x} \) are computable enumerations of the 0-1-valued Turing functionals and of the 0-1-valued computable partial functions. As usual we let \( \varphi^X_e \) denote the use function of \( \Phi^X_e \), and let \( \Phi^X_{e,s}, \varphi^X_e \) and \( \varphi_e \) at stage \( s \) denote the approximation of \( \Phi^X_e, \varphi^X_e \) and \( \varphi_e \) at stage \( s \). Moreover we adopt our convention that

$$\Phi^X_{e,s}(x) \downarrow \Rightarrow e, x, \varphi^X_{e,s}(x) < s.$$ 

Note that the noncomputability requirements guarantee that the sets \( A_0 \) and \( A_1 \) are noncomputable while the minimal pair requirements ensure (9.1). In order to make \( A_i \) computably enumerable we define a computable enumeration \( \{ A_i, s \}_{s \geq 0} \) of \( A_i \), i.e., effectively enumerate \( A_i \) in stages and let \( A_i, s \) be the finite part of \( A_i \) enumerated by the end of stage \( s \) (\( i = 0, 1 \)).

The priority ordering of the requirements is defined by

$$\mathcal{P}_0 < \mathcal{N}_0 < \mathcal{P}_1 < \mathcal{N}_1 < \ldots$$

where \( \mathcal{R} < \mathcal{R}' \) denotes that \( \mathcal{R} \) has higher priority than \( \mathcal{R}' \).

The basic strategy for meeting requirement \( \mathcal{P}_{2r+i} \) is essentially the Friedberg-Muchnik strategy. Since now on the right hand side there is a partial function, not a functional, however, we do not have to worry about restraining the oracle set here. An attack on \( \mathcal{P}_{2r+i} \) consists of (up to) two steps as follows.

1. Start the attack, say at stage \( s_1 + 1 \), by picking \( x = s_1 + 1 \) as follower.

2. Wait for a stage \( s_2 > s_1 \) such that \( \varphi_{x,s_2}(x) = 0 \). At stage \( s_2 + 1 \) put \( x \) into \( A_i \) and say that the attack is completed.

When a follower \( x \) of \( \mathcal{P}_{2r+i} \) is picked at stage \( s_1 + 1 \) then \( x \) has not been enumerated into \( A_i \) before, and \( x \) will be eventually enumerated into \( A_i \) if only if the attack is completed. So, no matter whether or not the attack will be completed, \( A_i(x) \neq \varphi_e(x) \) whence requirement \( \mathcal{P}_{2r+i} \) will be met. Moreover, performance of step 2 may be arbitrarily delayed since once \( \varphi_{x,s_2}(x) = 0 \) this value will be preserved at all stages \( s \geq s_2 \). So action for meeting requirement \( \mathcal{P}_{2r+i} \) can be limited to any infinite computable set \( S \) of stages.

The basic strategy for meeting requirement \( \mathcal{N}_e \ (e = \langle e_0, e_1, e_2 \rangle ) \) is as follows. Let

$$l(e,s) = \max \{ x : \forall y < x (\Phi^A_{e_0,s}(y) = \Phi^A_{e_1,s}(y) \downarrow) \}$$
9.1. A minimal pair in the c.e. Turing degrees

be the length (of agreement) function of \( N_e \) and call a stage \( s \) e-expansionary if \( s = 0 \) or if \( s > 0 \) and
\[
l(e, s) > \max\{l(e, t) : t < s\}.
\]

Note that if the hypothesis
\[
(*) \quad \Phi_{e_0}^{A_0} = \Phi_{e_1}^{A_1} \text{ total}
\]
of the minimal pair requirement \( N_e \) is true then
\[
\lim_{s \to \infty} l(e, s) = \infty \quad (9.2)
\]
and hence there are infinitely many e-expansionary stages.

Now requirement \( N_e \) (which is purely negative) imposes the following restraint on the enumeration of numbers into the sets \( A_0 \) and \( A_1 \). A number \( x \) can be enumerated into \( A_0 \) or \( A_1 \) at stage \( s + 1 \) only if either \( x \) is greater than the greatest e-expansionary stage \( \leq s \) or stage \( s \) is e-expansionary. Moreover, in the latter case, numbers less than or equal to the last e-expansionary stage may be enumerated only into one of the sets \( A_0 \) or \( A_1 \).

Note that, assuming \((*)\), this allows us to compute \( \Phi_{e_0}^{A_0}(x) \) for a given \( x \) as follows. Compute the least stage \( s \) such that \( s \) is e-expansionary and \( l(e, s) > x \) and let \( j = \Phi_{e_0}^{A_0}(x) = \Phi_{e_1}^{A_1} \). We claim that \( j = \Phi_{e_0}^{A_0}(x) \). Namely, if at stage \( s + 1 \) a number \( y \leq s \) goes into one of the sets \( A_i \) then no number \( \leq s \) is allowed to enter \( A_1\neg\neg \) at stage \( s \) or at any of the following stages until the next expansionary stage \( s' \) is reached. Since, by our convention, \( \Phi_{e_1}^{A_1}(x) < s \) it follows with the use principle that \( j = \Phi_{e_1}^{A_1}(x) = \Phi_{e_1}^{A_1}(x) \). Moreover, since \( s' \) is e-expansionary, \( j = \Phi_{e_0}^{A_0}(x) = \Phi_{e_1}^{A_1}(x) \). So if now a small number goes into \( A_1\neg\neg \) then \( A_i \) will not change below \( s' \) up to the next expansionary \( s'' \) and hence hold the computation up to this stage (and so on).

Note that the restraint imposed by this strategy goes to infinity on the non-e-expansionary stages while it is lifted (i.e. drops to 0) at the e-expansionary stages, though the restraint is lifted only for one of the sets \( A_0 \) or \( A_1 \) (which can be freely chosen). So action of a positive requirement \( \mathcal{P}_{e'} \) will be only delayed by \( N_e \) and not permanently blocked.

There is a synchronisation problem, however, if we consider more than one \( N \)-requirement, say \( N_e \) and \( N_{e'} \) where \( e < e' \). If we define e-expansionary and \( e' \)-expansionary stages as above then it might happen that no stage is simultaneously e-expansionary and \( e' \)-expansionary. So the combined restraint imposed by these two requirements may never drop back and some lower priority \( \mathcal{P} \)-requirements may never be met.
This problem can be overcome as follows. The $N$-strategy will remain correct if we limited the definition of $e$-expansionary stages to any computable set $S$, i.e., if we call a stage $s > 0$ an $S$-expansionary if $s \in S$ and

$$l(e,s) > \max\{l(e,t) : t < s \& t \in S \cup \{0\}\}.$$  

So if we assume that (*) holds (or, more generally, if there are infinitely many $e$-expansionary - this may happen also when (*) fails) then we can let $N_e$ work only within the set $S$ of the $e$-expansionary stages so that the set of the $(S)$-expansionary stages will be contained in the set of the $S$-expansionary stages and the restraints will drop back simultaneously.

Since we cannot decide, however, whether there are infinitely many $e$-expansionary stages, we have to guess whether this is true or not. We do this as follows: if $s$ is $e$-expansionary then the guess $g(s)$ is that there are infinitely many $e$-expansionary stages while otherwise the guess $g(s)$ is that there are only finitely many $e$-expansionary stages. If we denote the infinitary outcome with 0 and the finitary outcome with 1, then the above guessing procedure ensures that, for the true outcome 1, $g(s) = 1$ for all sufficiently large $s$ and hence $\lim_{s \to \infty} g(s) = 1$. On the other hand, for the true outcome 0, there will be infinitely many stages $s$ such that the guess $g(s)$ is correct but since, in general, there will also be infinitely many nonexpansionary stages there will also be infinitely many incorrect guesses. Hence in this case we only get that $\liminf_{s \to \infty} g(s) = 0$. So in general we can only argue that the correct outcome is $\liminf_{s \to \infty} g(s)$ (but $\lim_{s \to \infty} g(s)$ might not exist).  

we guess at stage $s + 1$ that there are infinitely many stages and we guess that there are only finitely many stages otherwise. So if we let 0 denote the infinitary guess and 1 the finitary guess then for the true outcome 1 the guesses converge to 1 (i.e., our guess will be 1 at all sufficiently large stages) whereas, for the true outcome 0, 0 is guessed infinitely often but there might be also infinitely many 1 guesses whence here we can only argue that the true outcome 0 is the liminf of the guesses. (This guessing procedure cannot be improved since the question whether a minimal pair requirement is infinitary cannot be solved relative to $K = \emptyset'$ whence there is no computable approximation of the fact.)

In order to model the above guesses at the nature of the higher priority $N$-requirements, we use the full infinite binary tree $T = \{0,1\}^\omega$ (called the priority tree). So a node $\alpha$ of $T$ (i.e., $\alpha \in T$) of length $|\alpha| = e + 1$, say $\alpha = \alpha(0) \ldots \alpha(e)$.
codes a guess at the outcomes of the first $e + 1$ negative requirements $N_0, \ldots, N_e$, where $\alpha(n) = 0$ codes the guess that $N_n$ is infinitary while $\alpha(n) = 1$ codes the guess that $N_n$ is finitary.

Some notation related to the priority tree will be useful for the following: For nodes $\alpha, \beta \in T$ we write $\alpha \sqsubseteq \beta$ if $\alpha$ is an initial segment of $\beta$ and $\alpha \sqsubset \beta$ if $\alpha$ is a proper initial segment of $\beta$. If $\alpha \sqsubseteq \beta$ then we call $\alpha$ a predecessor of $\beta$ and $\beta$ a successor of $\alpha$; and we say that $\alpha$ is below $\beta$ if $\alpha \sqsubset \beta$ and $\alpha$ is above $\beta$ (i.e., our trees grow upwards). We say that $\alpha$ is to the left of $\beta$ ($\alpha <_L \beta$) if there is a node $\gamma$ such that $\gamma \sqsubseteq \alpha$ and $\gamma \sqsubset \beta$. Finally we let $\alpha \leq \beta$ if $\alpha <_L \beta$ or $\alpha \sqsubseteq \beta$; and we let $\alpha < \beta$ or $\alpha \sqsubset \beta$.

Now, if the guess $\alpha$ seems to be correct at stage $s$ then $s$ is called an $\alpha$-stage and $s$ is called $\alpha$-expansionary if $s$ is $|\alpha|$-expansionary relative to the set of $\alpha$-stages. Formally, $\alpha$-stages and $\alpha$-expansionary stages are inductively defined as follows.

- Any stage $s$ is a $\lambda$-stage (where $\lambda$ is the empty string).
- An $\alpha$-stage $s$ is $\alpha$-expansionary if
  \[ l(|\alpha|, s) > \max \{ l(|\alpha|, t) : t < s \land t \alpha\text{-stage} \} \]
  (where $\max \emptyset = 0$).
- A stage $s$ is an $\alpha 0$ stage if $s$ is $\alpha$-expansionary and $s$ is an $\alpha 1$ stage if $s$ is an $\alpha$-stage and not $\alpha$-expansionary.

(Note that the question whether $s$ is an $\alpha$-stage or $\alpha$-expansionary can be decided at the end of stage $s$ of the construction.) The unique string $\alpha$ of length $s$ such that $s$ is an $\alpha$-stage is denote by $\delta_s$. So $\delta_s$ codes the guesses at the outcomes of the first $s$ minimal pair requirements at stage $s$. We call the nodes below $\delta_s$ accessible at stage $s + 1$.

Our observation that the current guesses, i.e., the strings $\delta_s$, approximate the true outcomes of the minimal pair requirements requirements (in the lim inf sense) can now be made more precise as follows by defining the true path $f \in \{0, 1\}^\omega$ of the construction giving the true outcomes of the $N$-requirements. The true path $f$ is inductively defined by

\[ f(n) = \begin{cases} 0 & \text{if there are infinitely many } (f \upharpoonright n)\text{-expansionary stages} \\ 1 & \text{otherwise.} \end{cases} \]

(Intuitively, the strategy $N_{f \upharpoonright e}$ for $N_e$ which works inside the $(f \upharpoonright e)$-stages will be infinitary (i.e., there will be infinitely many $S_{f \upharpoonright e}$-e-expansionary stages for the set $S_{f \upharpoonright e}$ of the $(f \upharpoonright e)$-stages) if and only if $f(e) = 0$, i.e., iff $((f \upharpoonright e) \sqsubset f)$.)
TRUE PATH LEMMA. For all numbers \( n \), \( \liminf_{s \to \infty} \delta_s \upharpoonright n = f \upharpoonright n \) (i.e., for any number \( n \) there are only finitely many stages \( s \) such that \( \delta_s \upharpoonright n <_L f \upharpoonright n \) and infinitely many stages \( s \) such that \( \delta_s \upharpoonright n = f \upharpoonright n \). (In other words: \( f \) is the leftmost path \( g \) through \( T \) such that, for any \( n \), \( \delta_s \upharpoonright n \sqsubseteq g \) for infinitely many \( s \).)

The proof of the True Path Lemma is by a straightforward induction on \( n \).

Finally, to each node \( \alpha \) of length \( e \) a strategy \( P_\alpha \) for meeting \( P_e \) is attached. This strategy respects the restraint imposed by the strategies for the higher priority requirements based on corresponding guesses. \( P_\alpha \) is only allowed to act if its guess \( \alpha \) seems to be correct, i.e., if \( \alpha \sqsubseteq \delta_s \), and \( P_\alpha \) is initialized (i.e., its current follower (if any) is cancelled) if \( \alpha \) is to the right of \( \delta_s \). This ensures that the strategy for meeting a minimal pair requirement \( N_{e'} \) based on the correct guess \( f \upharpoonright e' \) will succeed: The positive strategies to the left will be accessible only finitely often and there are only finitely many strategies below \( f \upharpoonright e' \) and each of this strategies will be initialized only finitely often and hence will act only finitely often. So the number of injuries from below and from the right is bounded. Nodes to the right cannot injure the strategy by initialization, and, finally, the strategies above will cause no injuries since they work with the correct guess.

After having explained the basic ideas we now give the formal construction.

At stage \( s = 0 \) we let \( A_{0,0} = A_{1,0} = \emptyset \) and initialize all \( P \)-strategies.

Stage \( s + 1 \) is as follows. A strategy \( P_\alpha, |\alpha| = 2e + i \), requires attention at stage \( s + 1 \) if \( \alpha \sqsubseteq \delta_s \) and

(i) \( P_\alpha \) has no follower or

(ii) \( P_\alpha \) has follower \( x \), \( x \not\in A_{i,s} \), and \( \varphi_{e,s}(x) = 0 \).

Fix \( \alpha \sqsubseteq \delta_s \) minimal such that \( P_\alpha \) requires attention. (Say that \( P_\alpha \) receives attention or becomes active.) If \( P_\alpha \) requires attention via \((i)\) then appoint \( s + 1 \) as follower. If \((ii)\) holds then put the follower \( x \) of \( P_\alpha \) into \( A_i \) and declare \( P_\alpha \) to be satisfied. In any case (also if no strategy requires attention), initialize all \( P_\beta \) such that \( \delta_s <_L \beta \).

This completes the construction. The correctness of the construction follows from the following claims.

Claim 1. For any \( e \geq 0 \), \( P_{f \upharpoonright e} \) requires attention at most finitely often and \( P_e \) is met.

Proof. The proof is by induction on \( e \). So fix \( e \) and assume the claim to be correct for \( e' < e \). Then we may fix a stage \( s_0 \) such that no strategy \( P_{f \upharpoonright e'} \) with \( e' < e \) requires attention after stage \( s_0 \). Moreover, by the True Path Lemma, we may fix \( s_1 \geq s_0 \) such that no \( \alpha \) with \( \alpha <_L f \upharpoonright e \) will be accessible after stage \( s_1 \). So \( P_{f \upharpoonright e'} \)
will not become initialized after stage $s_1$ and $\mathcal{P}_{f|e}$ will receive attention at any $f \upharpoonright e$ stage $s > s_1$ at which it requires attention. Moreover, by the True Path Lemma, $f \upharpoonright e$ will be accessible infinitely often. So $\mathcal{P}_{f|e}$ will eventually have a permanent follower $x$ and this follower will witness that $\mathcal{P}_e$ is met.

Claim 2. For any $e = \langle e_0, e_1 \rangle \geq 0$, $N_e$ is met.

Proof. W.l.o.g. we may assume that the hypothesis (*) of $N_e$ holds and hence that (9.2) holds. Let $\alpha = f \upharpoonright e$. Then, by the True Path Lemma, there are infinitely many $\alpha$-stages. So, by (9.2), there are infinitely many $\alpha$-expansionary stages whence $\alpha 0 \sqsubseteq f$. So, by the True Path Lemma and by Claim 1, we may fix $s_0$ such that no $\gamma <_L \alpha 0$ is accessible after stage $s_0$ and no strategy $\mathcal{P}_\beta$ with $\beta \sqsubseteq \alpha 0$ requires attention after stage $s_0$.

Now in order to compute $\Phi_{\infty}^{A_0}(x)$ for a given $x$ proceed as follows. Fix an $\alpha$-expansionary stage $s$ such that $s > s_0$ and $l(e, s) > x$. We claim that $\Phi_{\infty}^{A_0}(x) = i$ where

$$i = \Phi_{\infty}^{A_0}(x) = \Phi_{\infty}^{A_1}(x)$$ (9.3)

In order to show this it suffices to show that

$$\forall n \geq 1 \ [i = \Phi_{\infty}^{A_{0,n}}(x) = \Phi_{\infty}^{A_{1,n}}(x)]$$ (9.4)

where $s_1 < s_2 < s_3 < s_4 < \ldots$ are the $\alpha$-expansionary stages $\geq s$.

The proof is by induction $n$. For $n = 1$ the claim holds by (9.3) since $s_1 = s$. For the inductive step fix $n$ and assume the claim for $n$. In order to show that

$$i = \Phi_{\infty}^{A_{0,n+1}}(x) = \Phi_{\infty}^{A_{1,n+1}}(x)$$

it suffices to show that

$$A_{0,n+1} \upharpoonright s_n = A_{0,n} \upharpoonright s_n \text{ or } A_{1,n+1} \upharpoonright s_n = A_{1,n} \upharpoonright s_n$$

But this is shown as follows. Since at any stage at most one strategy acts, at stage $s_n + 1$ at most one of the two equalities can become destroyed. So it suffices to show that at a stage $t$ with $s_n + 1 < t \leq s_{n+1}$ no number $< s_n$ is enumerated into $A_0$ or $A_1$. Recall that no strategy $\mathcal{P}_\beta$ where $\beta$ is below or to the left of $\alpha 0$ will act after stage $s_0$ (and $s_0 < t$). Moreover no $\beta$ with $\alpha 0 \sqsubseteq \beta$ is accessible at such a stage $t$. So the only strategies $\mathcal{P}_\beta$ which may enumerate some number into $A_0$ or $A_1$ are those where $\alpha 0 <_L \beta$. But these strategies are initialized at stage $s_n + 1$. So any follower $y$ of such a strategy at stage $t - 1$ has to been appointed after stage $s_n$ whence $y > s_n$.

This completes the proof.
HISTORICAL REMARKS. The existence of minimal pairs was shown by Lachlan and, independently, by Yates in 1966. In their original proofs, however, Lachlan and Yates did not use a tree construction. A proof of the minimal pair theorem along the lines of Lachlan’s original proof can be found in Soare [So87]. Tree argument were introduced later by Lachlan in the context of $\Theta''$-arguments. For a more detailed discussion of this technique see Soare [So87] too. There also a proof of the minimal pair theorem using trees is given but there the tree is not a binary tree but an infinitely branching tree which does not only code the information whether a minimal pair requirement will be infinitary or not but it also monitors the outcomes of the finitary positive requirements and the size of the imposed restraints. So while in our proof there are still finite injuries along the true path (i.e., a $\mathcal{P}_\alpha$ strategy can be injured by a higher priority strategy $\mathcal{P}_\beta$ below it), the more sophisticated tree there allows to argue that there are no injuries along the true path.

9.2 Some applications of the minimal pair technique

Variants of the minimal pair technique play a central role in embedding results for lattices in the c.e. Turing degrees. Thomason (1971) and Lerman (unpublished) showed that every finite distributive lattice is embeddable in $R_T$. Lachlan (1972) and Lerman (unpublished) extended this to the countable case. While these results only required a straightforward extension of the minimal pair technique, the embeddability problem for nondistributive lattices turned out to be much more difficult and has not been completely resolved yet. Lachlan (1972) embeds the two five element nondistributive lattices, the nonmodular lattice $N_5$ and the modular lattice $M_3$ (see Figure 1 below). (These are of particular interest since every nondistributive lattice contains a copy of one of these lattices.) The embeddings required reductions where the use is not bounded computably. Systems of traces were introduced to describe these reductions. While for the embedding of the lattice $N_5$, a very simple trace model suffices, the trace system required for the embedding of $M_3$ is much more elaborate. A first nonembeddable lattice is given by Lachlan and Soare (1980), namely the $M_3$-lattice with a diamond on top (called $S_8$; see Figure 1 below). Later more nonembeddable lattices were found by Ambos-Spies and Lerman (1986) and by Lempp and Lerman (1997). Yet the embedding problem is still open. For a recent survey see Lempp, Lerman and Solomon [LLS06]. Moreover, all lattices which are known to be embeddable can also be embedded by a map
which preserve the least element. The question, however, whether embeddability and embeddability preserving $\mathbf{0}$ are equivalent in this setting is open.

The corresponding questions for the c.e. bT-degrees turned out to be quite simple. The embeddings of the distributive lattices in the c.e. T-degrees simultaneously are embeddings in the c.e. bT-degrees. Since, as mentioned before, the upper semi-lattice of the c.e. bT-degrees is distributive it follows that a lattice $\mathcal{L}$ can be embedded into $(\text{R}_{\text{bT}}, \leq)$ if and only if $\mathcal{L}$ is countable and distributive. Moreover all of these lattices can be embedded by a map preserving the least element.

Lachlan (1966) proved more results in his minimal pair paper, showing that, in the c.e. T-degrees, the situation with meets is very complex compared to the situation with joins. Lachlan showed that in contrast to Sacks’ Splitting Theorem, which in algebraic terms says that every nonzero c.e. degree is join-reducible, the dual fails: by the minimal pair theorem, $\mathbf{0}$ is meet-reducible (branching), there are non-zero meet-reducible degrees, but there are also incomplete non-meet-reducible (nonbranching) degrees.

In contrast, for the c.e. bounded Turing degrees the minimal pair construction can be easily modified in order to show that all incomplete c.e. bT-degrees are branching.

In the next chapter we look at some of the corresponding questions for the c.e. sbT-degrees.
Nondistributivity of the c.e. sbT-degrees

In this chapter we address the Embedding Problem for the c.e. strongly bounded Turing degrees, i.e., the question which finite lattices can be embedded into the partial orderings of the c.e. sbT-degrees. Since we have shown that all finite distributive lattices can be embedded in the c.e. sbT-degrees, a major step to the solution of the Embedding Problem is the answer to the question whether this degree structure is distributive. As we have pointed out in the previous section, the Embedding Problem for the distributive structure of the c.e. bT is solved (since all finite distributive lattice are embeddable and, by distributivity of the structure, no nondistributive lattices are embeddable) whereas the Embedding Problem for the nondistributive c.e. Turing degrees is still open (though it is known that all finite distributive lattices can be embedded, and a large variety of embeddable and nonembeddable finite nondistributive lattices has been found).

Our first main result is that, for \( r = \text{ibT, cl} \), the nomodular 5-element lattice \( N_5 \) can be embedded into the partial ordering \( (R_r, \leq) \) of the c.e. \( r \)-degrees (as a lattice and, in addition, by a map which preserves the least element) whence the c.e. degrees under the strongly bounded Turing reducibilities are nondistributive structures.

The question, whether the modular nondistributive 5-element lattice \( M_3 \) can be embedded in the c.e. sbT-degrees too, however, remains open. We only show that if there is such an embedding then it cannot preserve the least element. This might be considered as some evidence that the lattice \( M_3 \) cannot be embedded since in case of the c.e. Turing degrees all lattices which can be embedded there can be embedded by a map which preserves the least element. Ambos-Spies and Wang [AW10], however, have given an example of a (nondistributive) finite lattice which can be embedded into the c.e. sbT-degrees but which cannot be embedded by a map which preserves the least element.

Finally, we show that the finite lattices which can be embedded into the c.e. sbT-degrees and the finite lattices which can be embedded into the c.e. T-degrees are not the same. The proof is a bit curious since it exhibits two candidates of lattices of which we can show that one will do the separation but we do not know
which one. This result is based on some other new result which is of independent interest: all c.e. sbT-degrees are branching.

If not stated otherwise, all results in this chapter are taken from Ambos-Spies, Bodewig, Kräling and Yu [ABKY10].

10.1 The nonmodular 5-element lattice $N_5$ is embeddable into the c.e. sbT-degrees.

**Theorem 10.1** Let $r = \text{ibT}, \text{cl}$. There is an embedding of the 5-element nonmodular lattice $N_5$ into the partial ordering $(R, \leq)$ of the c.e. $r$-degrees (as a lattice) which preserves joins, meets, and the least element.

![Diagram](image)

**Figure 2**

**Proof (sketch).** We consider only the case of $r = \text{ibT}$. (The proof for $r = \text{cl}$ is similar, though at some point slightly more technical.) It suffices to construct c.e. sets $A$, $B$, $C$ and $D$ such that,

$$A \leq_{\text{ibT}} B \& B \leq_{\text{ibT}} D \& C \leq_{\text{ibT}} D$$  \hspace{1cm} (10.1)

$$B \not\leq_{\text{cl}} A$$  \hspace{1cm} (10.2)

$$\deg_{\text{ibT}}(B) \wedge \deg_{\text{ibT}}(C) = \deg_{\text{ibT}}(\emptyset)$$  \hspace{1cm} (10.3)

and

$$\deg_{\text{ibT}}(A) \lor \deg_{\text{ibT}}(C) = \deg_{\text{ibT}}(D)$$  \hspace{1cm} (10.4)
for all\( e \) and Friedberg-Muchnik strategy and will be described in more detail below.

The minimal pair requirements introduced in the preceding chapter. In particular, we define

\[
\begin{align*}
x \in A_{s+1} \setminus A_s &\Rightarrow \exists y \leq x (y \in B_{s+1} \setminus B_s) \\
x \in B_{s+1} \setminus B_s &\Rightarrow \exists y \leq x (y \in D_{s+1} \setminus D_s) \\
x \in C_{s+1} \setminus C_s &\Rightarrow \exists y \leq x (y \in D_{s+1} \setminus D_s)
\end{align*}
\]

(10.5)

The other conditions are split up into the requirements

\[
P_e : B \neq \hat{\Phi}_e^A
\]

\[
N_e : \hat{\Phi}_e^B = \hat{\Phi}_e^C \text{ total } \Rightarrow \hat{\Phi}_e^B \text{ computable}
\]

and

\[
Q_e : A = \hat{\Phi}_{e_1}^{W_{e_0}} \& C = \hat{\Phi}_{e_2}^{W_{e_0}} \Rightarrow D \leq_{ibT} W_{e_0}
\]

for all \( e = (e_0, e_1, e_2) > 0 \).

The strategy for meeting the nonordering requirement \( P_e \) is a variant of the Friedberg-Muchnik strategy and will be described in more detail below.

The minimal pair requirements \( N_e \) are met by the - purely negative - standard minimal pair strategy introduced in the preceding chapter. In particular, we define the length function

\[
l_N(e, s) = \max \{ x : \forall y < x (\hat{\Phi}_{e_0, s}^B(y) = \hat{\Phi}_{e_1, s}^C(y) \downarrow) \}.
\]

and call a stage \( s \) \( N_e \)-expansionary if \( l_N(e, s) \) is greater than \( l_N(e, t) \) for all \( t < s \). Since we deal with ibT-functionals here,

\[
\hat{\Phi}_{e_0}^B = \hat{\Phi}_{e_1}^C \text{ total } \Leftrightarrow \lim_{s \to \infty} l_N(e, s) = \infty \Leftrightarrow \limsup_{s \to \infty} l_N(e, s) = \infty.
\]

(10.6)

(Namely \( \hat{\Phi}_{e_0}^B(y) \) is defined if (and only if) there are infinitely many stages \( s \) such that \( \hat{\Phi}_{e_0, s}^B(y) \) is defined, and - if so - \( \hat{\Phi}_{e_0, s}^B(y) \) converges to \( \hat{\Phi}_{e_0}^B(y) \) (and similar for \( \hat{\Phi}_{e_1}^C(y) \)).) So here there are infinitely many \( N_e \)-expansionary stages \( s \) if and only if the hypothesis of \( N_e \) is true. (For Turing functionals only the implication \( \Leftarrow \) is true in general.)

The effect of the strategy for meeting \( N_e \) will be that “small” numbers are allowed to enter \( B \) or \( C \) only at expansionary stages and that at any stage only \( B \) or only \( C \) is allowed to change but not both.

The strategy for meeting a join requirement \( Q_e \) monitors the apparent length of agreement between \( A \) and \( \hat{\Phi}_{e_1}^{W_{e_0}} \) and between \( C \) and \( \hat{\Phi}_{e_2}^{W_{e_0}} \) at stage \( s \),

\[
l_Q(e, s) = \max \{ x : \forall y < x (A_s(y) = \hat{\Phi}_{e_1, s}^{W_{e_0}}(y) \& C_s(y) = \hat{\Phi}_{e_2, s}^{W_{e_0}}(y)) \}.
\]
Note that (as in case of the minimal pair requirements)
\[ A = \hat{\Phi}_{e_1}^{W_0} \land C = \hat{\Phi}_{e_2}^{W_0} \iff \lim_{s \to \omega} l_{Q_s}(e, s) = \infty \iff \limsup_{s \to \omega} l_{Q_s}(e, s) = \infty \quad (10.7) \]

Now a small number \( x \), i.e., a number \(< \max \{ l_{Q_s}(e, t) : t \leq s \} \), is allowed to enter \( D \) at stage \( s + 1 \) only if \( x < l_{Q_s}(e, s) \) and simultaneously a sufficiently small number \( y \) is simultaneously put into \( B \) or \( C \) so that restoration of the hypothesis of \( Q_e \) at \( y \) will force \( W_{e_0} \) to permit \( x \). (It suffices that \( y < l_{Q_s}(e, s) \) and \( x \geq z \) for the greatest number \( z \leq y \) such that \( z \) is not yet in \( W_{e_0} \) at stage \( s \).)

We now come back to the strategy for meeting a nonordering requirement \( \mathcal{P}_e \).

Following the Friedberg-Muchnik strategy we pick some unused \( x \) and wait for a stage \( s \) such that \( \hat{\Phi}_{e_2}^{A}(x) = 0 \). Then we want to put \( x \) into \( B \) and preserve the computation \( \hat{\Phi}_{e_2}^{A}(x) = 0 \) by restraining \( A_s \lceil x + 1 \). By the permitting requirement (10.5), putting \( x \) into \( B \) will require to put \( x \) (or a smaller number) into \( D \). The latter, however, is critical for the join requirements. In order to describe the problem, assume that requirement \( Q_e \) has higher priority and is infinitary.

HERE IS A PART MISSING!!!!

For synchronizing the strategies for meeting the individual minimal pair and join requirements we use the infinite binary tree \( T = \{0, 1\}^* \) as priority tree. We assign requirement \( N_e \) to the nodes \( \alpha \) of \( T \) of length \( 2e \) and requirement \( Q_e \) to the nodes \( \alpha \) of \( T \) of length \( 2e + 1 \). The outcome of an \( N_e \) or \( Q_e \)-requirement is 0 if its hypothesis is correct and 1 otherwise. These outcomes define the true path \( f \in 2^\omega \) through \( T \) where
\[ f(2e) = 0 \iff \hat{\Phi}_{e_0}^B = \hat{\Phi}_{e_1}^C \text{ total} \]
and
\[ f(2e + 1) = 0 \iff A = \hat{\Phi}_{e_1}^{W_0} \land C = \hat{\Phi}_{e_2}^{W_0}. \]

The guess \( \delta_s \) at the initial segment of \( f \lceil s \) at stage \( s \) is inductively defined by \( \delta_s(2e + i) = 0 \) if and only if either
\[ i = 0 \text{ and } l_N(e, s) > \max \{ l_N(e, t) : t < s \land \delta_t \lceil 2e + i = \delta_s \lceil 2e + i \}\]
or
\[ i = 1 \text{ and } l_Q(e, s) > \max \{ l_Q(e, t) : t < s \land \delta_t \lceil 2e + i = \delta_s \lceil 2e + i \}\]

\(^1\)Since we are dealing here with ibT-functional and not with general Turing functional, lim and lim sup of the length functions coincide and lim does not depend on the given enumeration. So here, for the true path \( f, f(e) \) only depends on the question whether the hypothesis of the requirement corresponding to \( e \) is true or not. For Turing functional this is not always the case. This is the reason why there we had to define the true path inductively based on the actual enumeration of the constructed sets.
where $e \geq 0$, $i \leq 1$, and $\max \emptyset = 0$.

10.2 Embeddings vs. 0-preserving embeddings

We now survey some results related to the question of embeddings which preserve the least element. We first give a criterion which implies that a lattice cannot be embedded into the c.e. sbT-degrees.

**Definition 10.2** Let $(P, \leq)$ be a partial ordering. For $b_0, b_1, b_2 \in P$, $(b_0, b_1, b_2)$ is a bad triple in $(P, \leq)$ if $b_0, b_1, b_2$ are pairwise incomparable and the following holds.

$$b_0 \wedge b_1 \text{ and } b_0 \wedge b_2 \text{ exist and } b_0 \wedge b_1 = b_0 \wedge b_2 \quad (10.8)$$

$$b_1 \lor b_2 \text{ exists and } b_0 \leq b_1 \lor b_2 \quad (10.9)$$

**Lemma 10.3 (Ambos-Spies and Wang [AW10])** Let $r = \text{iT}, \text{cl}$ and let $b_0, b_1, b_2$ be a bad triple in the partial ordering of the c.e. $r$-degrees. Then $b_0 \wedge b_1 \neq 0$. 

![Figure 3. A bad triple](image-url)
Proof. For a contradiction assume that $b_0 \wedge b_1 = 0$. Fix c.e. set $B_i \in b_i$. Then, by the Meet Lemma and the Join Lemmas, the bT-degrees of $B_0, B_1, B_2$ are a critical triple in the c.e. bT-degrees. But this, as one can easily check, contradicts distributivity of $(R_{bT}, \leq)$. □

**Corollary 10.4** There is no embedding of the lattice $M_3$ into the c.e. sbT-degrees which preserves the least element.

**Open Problem.** Is $M_3$ embeddable into the c.e. $r$-degrees for $r = ibT, cl$?

**Theorem 10.5 (Ambos-Spies and Wang [AW10])** Let $r = ibT, cl$. There is a finite lattice which can be embedded into the c.e. $r$-degrees but which cannot be embedded by a map which preserves the least element.

**Proof (Idea).** The lattice $\mathcal{L}$ shown in the figure below contains a critical triple $B_0, B_1, B_2$ such that $B_0 \wedge B_1$ is the least element of the lattice. So, by Lemma 10.3, $\mathcal{L}$ cannot be embedded into the c.e. $r$-degrees by a map which preserves the least element.

![Diagram](image)

On the other hand, by a variation of the proof on the embeddability of $N_5$ in the c.e. $r$-degrees, one can give an embedding of $\mathcal{L}$ in the c.e. $r$-degrees. □
10.3 Branching degrees

**Definition 10.6** Let \( r \) be any reducibility. A c.e. \( r \)-degree \( a \) is branching if there are c.e. \( r \)-degrees \( b_0 \) and \( b_1 \) such that \( a < b_0, b_1 \) and \( a = b_0 \land b_1 \).

**Theorem 10.7 (Ambos-Spies, Bodewig, Kräling, Yu)** For \( r = \text{ibT}, \text{cl} \), every c.e. \( r \)-degree \( a \) is branching. I.e., given a c.e. \( r \)-degree \( a \) there are c.e. \( r \)-degrees \( b_0 \) and \( b_1 \) such that \( a < b_0, b_1 \) and \( a = b_0 \land b_1 \). Moreover, the degrees \( b_0 \) and \( b_1 \) can be chosen so that \( b_0 \lor b_1 \) exists.

**Proof (Sketch).** We give the proof only for \( r = \text{ibT} \) which is considerably simpler. For both, \( r = \text{ibT}, \text{cl} \), the proof is nonuniform. But in case of \( r = \text{cl} \) the nonuniformity is even stronger. While for \( \text{ibT} \) it suffices to distinguish two cases, there we have to distinguish three cases. (Also note that -despite the fact that, by the \( \text{ibT-Meet Lemma} \), meets in the \( \text{ibT-degrees} \) also give meets in the \( \text{cl-degrees} \) - the proof for \( \text{ibT} \) does not give a proof for \( \text{cl} \) since we do not guarantee that the branches of \( a \) are not contained in the same \( \text{cl-degree} \) as \( a \).

Given a c.e. \( \text{ibT-degree} \ a \), it suffices to give c.e. sets \( A, B_0 \) and \( B_1 \) such that \( A \in a \),

\[
A \leq_{\text{ibT}} B_0, B_1 \tag{10.10}
\]

\[
B_0, B_1 \not\leq_{\text{ibT}} A \tag{10.11}
\]

and

\[
\forall C (C \leq_{\text{ibT}} B_0, B_1 \Rightarrow C \leq_{\text{ibT}} A) \tag{10.12}
\]

hold.

Choose \( A \in a \) such that there is a computable ascending sequence \( (x_n)_{n \geq 0} \) such that \( x_0 = 0 \), and, for all \( n \geq 0 \), \( x_n \notin A \) and \( x_{n+1} - x_n > x_n \). (Note that we obtain such a set together with such a sequence by taking any infinite c.e. set \( \hat{A} \in a \) where w.l.o.g. \( 0 \notin \hat{A} \), taking some infinite computable subset \( R \) of \( \hat{A} \), letting \( A = \hat{A} \setminus R \), and by defining a sequence \( (x_n)_{n \geq 0} \) on \( R \cup \{0\} \) with the required properties.) Then the intervals

\[
I_n = [x_n, x_{n+1}) \ (n \geq 0)
\]

give a computable partition of \( \omega \) where the least element of each interval is not an element of \( A \) and

\[
|I_n| > | \bigcup_{n' < n} I_{n'} |. \tag{10.13}
\]

Let

\[
I_{\text{even}} = \bigcup_{n \geq 0} I_{2n} \text{ and } I_{\text{odd}} = \bigcup_{n \geq 0} I_{2n+1}
\]
and, for the definition of $B_0$ and $B_1$, distinguish the following two cases.

Case 1. $A \cap I_{even} \upharpoonright_{ibT} A \cap I_{odd}$.

Then let

$$B_0 = (A \cap I_{even}) \cup (A \cap I_{odd}) - 1 \text{ and } B_1 = (A \cap I_{even}) - 1 \cup (A \cap I_{odd})$$

Since the least element of any interval is not an element of $A$,

$$(A \cap I_{odd}) - 1 \subseteq I_{odd} \text{ and } (A \cap I_{even}) - 1 \subseteq I_{even}$$

So the two parts of $B_0$ are disjoint and so are the two parts of $B_1$. By case assumption, by the Splitting Lemma, and by the Bounded-Shift Lemma, this easily implies (10.10) and (10.11). Moreover,

$$deg_{ibT}((A \cap I_{odd}) - 1 \cup (A \cap I_{even}) - 1) = deg_{ibT}(B_0) \lor deg_{ibT}(B_1).$$

So $b_0 \lor b_1$ exists.

For a proof of (10.12), fix a c.e. set $C$ such that $C \leq_{ibT} B_0, B_1$. Then, in a given $ibT$-reduction $C = \Phi^{B_0}$, for $x \in I_{even}$ we can answer all oracle queries in $\Phi^{B_0}(x)$ by using $A \upharpoonright x + 1$ in place of the oracle $B$. Namely, for $y \leq x$ such that $y \in I_{even}$, $B_0(y) = A(y)$ while for $y \leq x$ such that $y \in I_{odd}$ - say $y \in I_{2e+1}$ - $B_0(y) = 0$ if $y$ is the greatest element of $I_{2e+1}$ and $B_0(y) = A(y + 1)$ otherwise where in the latter case $y + 1 < x$ (since $y \leq x$ and $y, y + 1 \in I_{2e+1}$ whereas $x \in I_{even}$). So $C \cap I_{even} \leq_{ibT} A$. Since, by a similar argument using the fact that $C \leq_{ibT} B_1$, $C \cap I_{odd} \leq_{ibT} A$ too, (10.12) follows by the Splitting Lemma.

Case 2. Otherwise.

Then, by symmetry, we may assume that $A \cap I_{even} \leq_{ibT} A \cap I_{odd}$. Since the c.e. sets $A \cap I_{even}$ and $A \cap I_{odd}$ split $A$, it follows by the Splitting Lemma that

$$A =_{ibT} A \cap I_{odd}.$$

So, in order to guarantee (10.10), it suffices to let

$$B_0 \cap I_{odd} = B_1 \cap I_{odd} = A \cap I_{odd}. \quad (10.14)$$

In order to satisfy the other conditions we use a straightforward variant of the minimal pair technique, where the required diagonalization witnesses for satisfying (10.11) will be chosen from $I_{even}$.

Conditions (10.11) and (10.12) are split up into the requirements

$$\mathcal{P}_{2e+i} : B_i \neq \hat{\Phi}_e^A$$
(e \geq 0, i \leq 1) and
\[ N_e : \hat{\Phi}^{B_0}_{e_0} = \hat{\Phi}^{B_1}_{e_1} \text{ total } \Rightarrow \hat{\Phi}^{B_0}_{e_0} \leq_{ibT} A \]
(where \( e \geq 0 \) and \( e = (e_0, e_1) \)).

The strategy for meeting the requirements \( N_e \) is directly taken from the minimal pair construction. The only differences are that (1) the restraints do not apply to the odd intervals on which we have chosen the sets \( B_0 \) and \( B_1 \) to agree with \( A \) and that (2) here we may use \( A \) as an oracle.

Moreover, for satisfying the requirements \( P_{2e+i} \) we use the variant of the Friedberg-Muchnik strategy we have already used in the maximal pair construction and which does not require any restraints. To be more precise, if a strategy \( P_\alpha \) of \( P_{2e+i} \) becomes accessible, it requires to get an even interval \( I_{2n} \) assigned to it which has not yet been assigned to any other strategy and which contains only numbers which are bigger than the current stage. Once such an interval is assigned, \( P_\alpha \) will require attention at any stage \( s \) at which it is accessible, \( B_i \) and \( \Phi^A_{e} \) agree on \( I_{2n} \) at stage \( s \) and there is a number \( x \in I_{2n} \) left which has not yet put into \( B_i \). (As in the maximal pair construction we may argue that eventually a disagreement is achieved. Note that numbers from \( I_{2n} \) will not enter \( A \) and the length of \( I_{2n} \) is greater than the sum of the lengths of the previous intervals.)

The argument that \( N_e \) is met is almost the same as that in the construction of a minimal pair. In the proof of Claim 2 in the minimal pair construction only the following has to be changed:

Now, in order to compute \( \hat{\Phi}^{B_0}_{e_0}(x) \) for a given \( x \) from \( A \upharpoonright x + 1 \) we proceed as follows. We fix an \( \alpha \)-expansionary stage \( s \) such that \( s > s_0 \) and \( I(e, s) > x \) and such that \( A_s \upharpoonright x + 1 = A \upharpoonright x + 1 \). (Note that, by the latter, the unrestrained part of \( B_0 \) on the odd intervals cannot change after stage \( s \) on the use of the involved computations.) Then by an inductive argument as before we can show that \( \hat{\Phi}^{B_0}_{e_0}(x) = \hat{\Phi}^{B_0}_{e_0'}(x) \).

Finally, note that in this case of the construction the sets \( B_i \) are of the form \( B_i = (A \cap I_{odd}) \cup \hat{B}_i \) where \( \hat{B}_i \subseteq I_{even} \) and \( \hat{B}_0 \cap \hat{B}_1 = \emptyset \). So the join of the ibT-degrees of \( B_0 \) and \( B_1 \) exists and is represented by
\[ B = B_0 \cup B_1 = (A \cap I_{odd}) \cup \hat{B}_0 \cup \hat{B}_1. \]
10.4 Embeddability in the c.e. T-degrees vs. embeddability in the c.e. sbT-degrees

We conclude our discussion of embeddability questions for the c.e. strongly bounded Turing degrees by showing that, for \( r = \text{ibT, cl} \), the class of the finite lattices which can be embedded into the partial ordering \((R_r, \leq)\) (by a map which preserves the least element) and the class of the finite lattices which can be embedded into the partial ordering \((R_T, \leq)\) of the c.e. Turing degrees (by a map which preserves the least element) do not coincide. (Note that the class of the finite lattices which can be embedded into the partial ordering \((R_r, \leq)\) (by a map which preserves the least element) and the class of the finite lattices which can be embedded into the partial ordering \((R_T, \leq)\) (by a map which preserves the least element) differ from the class of the finite lattices which can be embedded into the partial ordering \((R_{bT}, \leq)\) of the c.e. bounded Turing degrees (by a map which preserves the least element) since the latter class consist of distributive lattices only while the two former classes have the nonmodular lattice \(N_5\) among there members.)

**Theorem 10.8** There is a lattice which can be embedded into \((R_T, \leq)\) by a map which preserves the least element which cannot be embedded into \((R_r, \leq)\) by a map which preserves the least element \((r = \text{ibT, cl})\).

**Proof.** Lachlan has shown that the modular nondistributive 5-element lattice \(M_3\) can be embedded into the c.e. Turing degrees by a map which preserves the least element, but by Corollary 10.4 above there is no such embedding into the c.e. sbT-degrees. \(\square\)

**Theorem 10.9 (Ambos-Spies, Bodewig, Kräling, Yu [ABKY10])** The class of the lattices which can be embedded into the partial ordering \((R_{bT}, \leq)\) of the Turing degrees does not coincide with the class of the lattices which can be embedded into the partial ordering \((R_r, \leq)\) \((r = \text{ibT, cl})\).

**Proof.** As pointed out above Lachlan has shown that the modular nondistributive 5-element lattice \(M_3\) can be embedded into the c.e. Turing degrees whereas Lachlan and Soare have shown that the lattice \(S_8\) (consisting of \(M_3\) with a diamond on top; see Figure 1 above) is not embeddable. On the other hand, for \( r = \text{ibT, cl} \) either \(M_3\) and \(S_8\) are embeddable or \(M_3\) and \(S_8\) are nonembeddable. Namely if \(M_3\) is embeddable then -since any c.e. \(r\)-degree is branching and in fact is the base of a diamond lattice - \(S_8\) is embeddable too. On the other hand if \(M_3\) is not embeddable then of course \(S_8\) is not embeddable too since \(M_3\) is a sublattice of \(S_8\). \(\square\)
Corollary 10.10 (Ambos-Spies, Bodewig, Kräling, Yu [ABKY10]) Let \( L = L(\leq, \vee, \wedge) \) and let \( r \in \{\text{ibT, cl}\} \). The existential theories \( \exists - \text{Th}(R_r), \exists - \text{Th}(R_{bT}) \) and \( \exists - \text{Th}(R_T) \) over the language \( L \) are pairwise different.
Comparing the theories of the c.e. ibT-degrees and the c.e. cl-degrees

In the past chapters we have seen many differences between the c.e. strongly bounded Turing degrees ($r = \text{ibT}, \text{cl}$) on the one side and c.e. Turing or bounded Turing degrees on the other side. In most cases, however, a property we could establish for one of the strongly bounded reducibility we could also establish for the other reducibility - mostly by quite similar proofs though in some cases the proof for the cl-degrees had been some more involved. The only example of a result we have obtained for only one of the reducibilities is the existence of nontrivial automorphisms for the c.e. ibT-degrees where we left the corresponding question for the c.e. cl-degrees open.

This leads to the question whether the partial orderings $(\mathcal{R}_{\text{ibT}}, \leq)$ and $(\mathcal{R}_{\text{cl}}, \leq)$ are elementarily equivalent (or even isomorphic). In this chapter we will show that this is not the case. Namely Ambos-Spies, Bodewig, Fan and Kräling [ABFK10] have obtained an elementary difference between the two degree structures by looking at cupping properties. Their result is based in part on some previous work by Ambos-Spies [Am10] on cupping properties of the c.e. ibT-degrees which also gives infinitely many 2-types in this structure, hence gives an alternative proof of the fact that the theory $\text{Th}(\mathcal{R}_{\text{ibT}}, \leq)$ of the partial ordering of the c.e. ibT-degrees is not $\aleph_0$-categorical (see Theorem 7.8). In contrast to the proof of Theorem 7.8 which was based on a quite sophisticated result on the bT-degrees, the alternative proof is selfcontained and requires only some fairly simple observations on bounded shifts. (In contrast to Theorem 7.8 which covered the ibT- and cl-degrees, however, the alternative proof applies only to the ibT-degrees.)
11.1 Cupping and noncupping in the c.e. ibT-degrees

**Definition 11.1** Let \((P, \leq)\) be a partial ordering and let \(a \in P\). An element \(b\) of \(P\) is \(a\)-cuppable (or cups to \(a\)) if there is some \(c \in P\) such that \(c < a\) and \(a = b \lor c\); and \(b\) is \(a\)-noncuppable if \(b \leq a\) and \(b\) is not \(a\)-cuppable.

Note that any \(a \in P\) is \(a\)-cuppable and if \((P, \leq)\) has a least element 0 then 0 is \(a\)-noncuppable for all \(a > 0\). Moreover, the class of the \(a\)-cuppable and \(a\)-noncuppable elements of \(P\) define a partition of the principal ideal \(P(A) = \{b \in P : b \leq a\}\) of \(P\).

For the partial ordering \((R_r, \leq)\) of the c.e. \(r\)-degrees and for any c.e. \(r\)-degree \(a\) we let \(\text{NCu}_r(a)\) denote the class of the \(a\)-noncuppable c.e. \(r\)-degrees \((r = \text{ibT, cl})\).

We first observe that in the c.e. ibT-degrees, for any degree \(a > 0\), the class of the \(a\)-noncuppable degrees is bounded by the +1-shift of \(a\).

**Theorem 11.2 (Ambos-Spies [Am10])** For any c.e. ibT-degree \(a > 0\),

\[
\text{NCu}_{\text{ibT}}(a) \subseteq \{b \in R_{\text{ibT}} : b \leq a + 1\}.
\]

For the proof we will need the following observation.

**Lemma 11.3 (Disjoint Sets Lemma)** (a) Let \(D\) and \(E\) be disjoint noncomputable c.e. sets such that \(D \leq_{\text{ibT}} E\). Then \(D \leq_{\text{ibT}} E + 1\).

(b) Let \(D\) and \(E\) be noncomputable c.e. sets such that \(D =_{\text{ibT}} E\). Then \(D \cap E \neq \emptyset\).

**Proof.** (a) By the Representation Lemma (and the Invariance Lemma) w.l.o.g. we may assume that there are one-to-one computable functions \(d\) and \(e\) which enumerate \(D\) and \(E\), respectively, and such that \(d(n) \geq e(n)\) for all \(n\). In fact, since \(D\) and \(E\) are disjoint, the latter implies \(d(n) > e(n)\), i.e., \(d(n) \geq e(n) + 1\). Since \(e(n) + 1\) is a computable one-to-one enumeration of \(E + 1\), it follows that \(D \leq_{\text{ibT}} E + 1\) by permitting. (b) This is immediate by part (a) and the Bounded-Shift Lemma. \(\square\)

**Proof of Theorem 11.2.** Given a noncomputable c.e. set \(A\) and a c.e. set \(B \leq_{\text{ibT}} A\) such that \(B \not\leq_{\text{ibT}} A + 1\), it suffices to give a c.e. set \(C <_{\text{ibT}} A\) such that

\[
\deg_{\text{ibT}}(A) = \deg_{\text{ibT}}(B) \lor \deg_{\text{ibT}}(C). \tag{11.1}
\]

By the Representation Lemma, w.l.o.g. we may assume that there are computable one-to-one enumerations \(\{a(n)\}_{n \geq 0}\) and \(\{b(n)\}_{n \geq 0}\) of \(A\) and \(B\), respectively, such that \(a(n) \leq b(n)\) for all \(n \geq 0\). Split \(A\) and \(B\) into c.e. sets

\[
A_0 = \{a(n) : a(n) = b(n)\} \quad \text{and} \quad A_1 = \{a(n) : a(n) < b(n)\}
\]
and
\[ B_0 = \{ b(n) : a(n) = b(n) \} \quad \text{and} \quad B_1 = \{ b(n) : a(n) < b(n) \}, \]
respectively. Note that \( A_0 = B_0 \). Hence, by the Splitting Lemma,
\[ \deg_{ibT}(A) = \deg_{ibT}(A_0) \lor \deg_{ibT}(A_1) = \deg_{ibT}(B_0) \lor \deg_{ibT}(A_1). \]

So, since (again by the Splitting Lemma) \( B_0 \leq_{ibT} B \), it suffices to show that \( A_1 <_{ibT} A \). (Then (11.1) will hold for \( C = A_1 \).)

For a contradiction assume that \( A_1 =_{ibT} A \). Then \( A_0 \leq_{ibT} A_1 \). Hence, by the Disjoint Sets Lemma, \( A_0 \leq_{ibT} (A_1) + 1 \). Since, by definition of \( A_1 \) and \( B_1 \), \( B_1 \leq_{ibT} (A_1) + 1 \) by permitting, it follows, by \( B_0 = A_0 \) and by the Splitting Lemma, that \( B \leq_{ibT} (A_1) + 1 \). Since \( (A_1) + 1 \leq_{ibT} A + 1 \), this contradicts the assumption that \( B \not\leq_{ibT} A + 1 \). \(\square\)

Ambos-Spies [Am10] has also shown that, for the degrees of sufficiently scattered sets, the converse of Theorem 11.2 is also true.

**Lemma 11.4** Let \( A \) be a noncomputable c.e. set such that \( A \subseteq 2\mathbb{N} \) or \( A \subseteq 2\mathbb{N} + 1 \). Then \( \deg_{ibT}(A + 1) \) does not cup to \( \deg_{ibT}(A) \).

**Proof.** Given a c.e. set \( C \leq_{ibT} A \) such that \( \deg_{ibT}(A) = \deg_{ibT}(A + 1) \lor \deg_{ibT}(C) \), it suffices to show that \( A \leq_{ibT} C \).

By the Representation Lemma, w.l.o.g. we may assume that there are computable one-to-one enumerations \( a(n) \) and \( c(n) \) of \( A \) and \( C \), respectively, such that \( a(n) \leq c(n) \). Split \( C \) into the disjoint c.e. sets
\[ C_0 = \{ c(n) : a(n) = c(n) \} \quad \text{and} \quad C_1 = \{ c(n) : a(n) < c(n) \}. \]

Then, by the Splitting Lemma, \( \deg_{ibT}(C) = \deg_{ibT}(C_0) \lor \deg_{ibT}(C_1) \). Since \( C_1 \leq_{ibT} A + 1 \) by permitting, it follows with (??) that
\[ \deg_{ibT}(A) = \deg_{ibT}(A + 1) \lor \deg_{ibT}(C_0). \]

Moreover, since \( C_0 \) is contained in \( A \) and since \( A \) is scattered, it follows that \( C_0 \) and \( A + 1 \) are disjoint. So, by the Splitting Lemma,
\[ A =_{ibT} A + 1 \cup C_0 \]
whence \( A \leq_{ibT} C_0 \) by the Bounded-Shift Lemma. Since, by the Splitting Lemma, \( C_0 \leq_{ibT} C \) it follows that \( A \leq_{ibT} C \). \(\square\)

Now by Theorem 11.2, by Lemma 11.4, and by downward closure of the non-cappable degrees we obtain:
**Theorem 11.5** Call a c.e. degree \( a \) scattered if it contains a c.e. set \( A \) such that \( A \subseteq 2\mathbb{N} \) or \( A \subseteq 2\mathbb{N} + 1 \), and let \( a \) and \( b \) be c.e. ibT-degrees such that \( a \) is scattered and \( b \leq a \). The following are equivalent.

(i) \( b \) cups to \( a \).

(ii) \( b \not\leq a + 1 \).

Now we can apply this observation in order to get a very elementary proof of the following special case of Theorem 7.8.

**Theorem 11.6** The elementary theory \( \text{Th}(R_{ibT}, \leq) \) of the partial ordering of c.e. ibT-degrees realizes infinitely many 2-types. So \( \text{Th}(R_{ibT}, \leq) \) is not \( \aleph_0 \)-categorical.

**Proof.** It suffices to give first order formulas \( \varphi_k(x, y) \) with two free variables \( x, y \) in the language of partial orderings \( (k \geq 1) \) such that, for the sets

\[
D_k = \{(a, b) \in R_{ibT}^2 : (R_{ibT}, \leq) \models \varphi_k(a, b)\},
\]

\( D_k \neq D_{k'} \) for any \( k, k' \geq 0 \) such that \( k \neq k' \).

Let the formula \( \varphi_1(x, y) \) express that \( y \) is the greatest element \( \leq x \) such that \( y \) does not cup to \( x \), and, for \( k \geq 2 \) define \( \varphi_k \) by

\[
\varphi_k(x, y) \equiv \exists y_1, \ldots, y_{k-1} \left( \varphi_1(x, y_1) \& \varphi_1(y_1, y_2) \& \ldots \& \varphi_1(y_{k-1}, y) \right).
\]

Then, by Lemma ??, for any scattered c.e. ibT-degree \( a \),

\[(R_{ibT}, \leq) \models \varphi_1(a, b) \iff b = a + 1.\]

Since, for any scattered degree \( a \), the degree \( a + 1 \) is scattered too, it follows by a straightforward induction on \( k \geq 1 \) that

\[(R_{ibT}, \leq) \models \varphi_k(a, b) \iff b = a + k.\]

Since, by the Bounded-Shift Lemma, for \( a > 0 \), the degrees \( a + k \) are pairwise different, this implies the claim. \( \square \)

### 11.2 Cupping and noncupping in the c.e. cl-degrees

By Theorem 11.2 and by the Bounded-Shift Lemma, for any c.e. ibT-degree \( a > 0 \), the class of the \( a \)-noncuppable degrees is bounded by a degree \( b < a \), namely, by
11.2. Cupping and noncupping in the c.e. cl-degrees

the \((+1)\)-shift \(a + 1\). Since this fact can be expressed by a first-order formula in the language \(\mathcal{L}(\leq)\) of partial orderings, in order to get an elementary difference between the partial orderings of the c.e. ibT- and the c.e. cl-degrees it suffices to construct a c.e. cl-degree \(a > 0\) for which the class of a-noncuppable degrees is not bounded by any degree strictly below \(a\). This has been recently done by Ambos-Spies, Bodewig, Fan and Kräling [ABFK10] by showing that the cl-degrees \(a\) of sufficiently scattered c.e. sets have this property.

**Definition 11.7** Let \(R_2 = \{2^m : m \geq 0\}\), call a set \(A\) **tally** if \(A \subseteq R_2\), and call a c.e. degree **tally** if it contains a tally c.e. set.

Note that, for any c.e. set \(A, \hat{A} = \{2^n : n \in A\}\) is tally, c.e. and wtt-equivalent to \(A\). So any c.e. wtt-degree contains a c.e. tally set.

**Theorem 11.8 (Ambos-Spies, Bodewig, Fan and Kräling [ABFK10])** Let \(a\) and \(b\) be c.e. cl-degrees such that \(a\) is tally and \(b < a\). There is an a-noncuppable c.e. cl-degree \(c\) such that \(c \leq a\) and \(c \not< b\).

For the proof of Theorem 11.8 we will need the following observation.

**Lemma 11.9** Let \(A\) and \(D\) be c.e. sets such that \(A\) is tally and \(D <_{cl} A\). There is a c.e. set \(\hat{D}\) such that \(\hat{D} \subseteq 2\mathbb{N} + 1\) and \(D \leq_{cl} \hat{D} <_{cl} A\).

**Proof.** W.l.o.g. we may assume that \(0, 1, 2, 3 \not\in A\) and (by replacing \(D\) by a bounded shift \(D + k\)) that \(D \leq_{ibT} A\). So, by the Representation Lemma, we may further assume that there are computable one-to-one enumerations \(\{a(n)\}_{n \geq 0}\) and \(\{d(n)\}_{n \geq 0}\) of \(A\) and \(D\), respectively, such that \(a(n) \leq d(n)\) \((n \geq 0)\). Fix a computable function \(\text{odd} : \mathbb{N} \to \mathbb{N}\) such that, for \(m \geq 2\) and \(y \in [2^m, 2^{m+1})\), \(\text{odd}(y)\) is an odd number in \([2^m, 2^{m+1} - 1)\), and \(|\text{odd}(y) - y| \leq 2\) (e.g., \(\text{odd}(y) = y + 1\) for even \(y \in [2^m, 2^{m+1} - 2)\), \(\text{odd}(y) = y\) for odd \(y \in [2^m, 2^{m+1} - 1)\), and \(\text{odd}(2^{m+1} - 1) = odd(2^{m+1} - 2) = 2^{m+1} - 3)\).

Then let \(\hat{D} = \{\hat{d}(n) : n \geq 0\}\) for the computable function \(\hat{d}\) defined by

\[
\hat{d}(n) = \begin{cases} 
\text{odd}(d(n)) & \text{if } a(n) \leq d(n) < 2a(n) \\
2a(n) - 1 & \text{otherwise (i.e., if } 2a(n) \leq d(n)).
\end{cases}
\]

Obviously, \(\hat{D}\) is c.e. and \(\hat{D} \subseteq 2\mathbb{N} + 1\). Moreover, the function \(\hat{d}\) is one-to-one and \(a(n) \leq \hat{d}(n) \leq d(n) + 2\). So \(D \leq_{cl} \hat{D} \leq_{cl} A\).

It remains to show that \(A \not\leq_{cl} \hat{D}\). For a contradiction assume \(A \leq_{cl} \hat{D}\). Split \(A\), \(D\), and \(\hat{D}\) into the c.e. sets

\[
A_0 = \{a(n) : a(n) \leq d(n) < 2a(n)\} \text{ and } A_1 = \{a(n) : 2a(n) \leq d(n)\},
\]
\[ D_0 = \{ d(n) : a(n) \leq d(n) < 2a(n) \} \text{ and } D_1 = \{ d(n) : 2a(n) \leq d(n) \}, \]
and
\[ \hat{D}_0 = \{ \hat{d}(n) : a(n) \leq \hat{d}(n) < 2a(n) \} \text{ and } \hat{D}_1 = \{ \hat{d}(n) : 2a(n) \leq \hat{d}(n) \}, \]
respectively. Note that, by the fact that any interval \([2^m, 2^{m+1} - 1]\) contains at most one element of \(D_0\) and by definition of \(\text{odd}(y)\), \(\hat{D}_0 =^c D_0\). Moreover, \(\hat{D}_1 = (A_1)_f\) for the computable unbounded shift \(f\) defined by \(f(0) = 0\) and
\[ f(2^m + k) = 2^{m+1} + k - 1 \ (m \geq 0, 0 \leq k < 2^m). \]

By the latter and by assumption,
\[ A \leq_{cl} \hat{D} = \hat{D}_0 \cup \hat{D}_1 = \hat{D}_0 \cup (A_1)_f \]
whence \(A \leq_{cl} \hat{D}_0\) by the Computable-Shift Lemma. By \(\hat{D}_0 =^c D_0\) and \(D_0 \leq_{cl} D\) this implies \(A \leq_{cl} D\) contrary to choice of \(A\) and \(D\).

**PROOF.** of Theorem 11.8 Fix c.e. sets \(A\) and \(B\) in \(a\) and \(b\), respectively, such that \(A\) is tally. By replacing \(B\) by a bounded shift \(B + k\), we may assume that \(B \leq_{\text{ibT}} A\). So, by the Representation Lemma, we may further assume that there are computable one-to-one enumerations \(\{a(n)\}_{n \geq 0}\) and \(\{b(n)\}_{n \geq 0}\) of \(A\) and \(B\), respectively, such that \(a(n) \leq b(n)\) \((n \geq 0)\).

It suffices to define a c.e. set \(C \leq_{cl} A\) such that \(C \nleq_{cl} B\) and \(\text{deg}_{cl}(C)\) does not cup to \(\text{deg}_{cl}(A)\). In the following we inductively define a computable one-to-one function \(c(n)\) enumerating such a c.e. set \(C\). We ensure that the function \(c(n)\) has the following properties.

\[ \forall n \ [a(n) \leq c(n) < 2a(n)] \quad (11.2) \]
\[ \forall n \ [c(n) \text{ even}] \quad (11.3) \]
\[ \forall e \exists n_e \forall n \geq n_e \ [c(n) > a(n) + e] \quad (11.4) \]

In addition we guarantee that the set \(C = \{c(n) : n \geq 0\}\) meets the requirements
\[ R_c : C \neq \Phi_e^B \]
\((e \geq 0)\) where \(\{\Phi_e\}_{e \geq 0}\) is a computable enumeration of the cl-functionals where w.l.o.g. the use function \(\hat{\Phi}_e(x)\) of \(\Phi_e\) is bounded by \(x + e\).

To show that this guarantees that \(C\) has the required properties, note that (11.2) implies that \(C \leq_{cl} A\) while the requirements \(R_c\) ensure that \(C \nleq_{cl} B\). It remains to
show that \( \text{deg}_{cl}(C) \) does not cup to \( \text{deg}_{cl}(A) \). For a contradiction assume that there is a c.e. set \( D <_{cl} A \) such that

\[
\text{deg}_{cl}(A) = \text{deg}_{cl}(C) \lor \text{deg}_{cl}(D).
\]

By Lemma 11.9, w.l.o.g. we may assume that \( D \subseteq 2\mathbb{N} + 1 \). Since, by (11.3), \( C \subseteq 2\mathbb{N} \), it follows that \( A <_{cl} C \cup D \) by the Splitting Lemma. So we may fix \( e \) such that \( A = \Phi^{C \cup D}_{e} \). Now in order to get the desired contradiction we show that this reduction can be converted into a cl-self-reduction of \( A \) relative to \( D \) whence \( A <_{cl} D \) contrary to choice of \( D \). This self-reduction is as follows.

Since \( A \) is tally it suffices to compute \( A(x) \) for \( x = 2^{m} \ (m \geq 0) \). In fact, by (11.4), we may fix a number \( m_{e} \) such that for any \( n \) such that \( a(n) = 2^{m} \) for some \( m \geq m_{e} \), \( c(n) > a(n) + e = 2^{m}\) and w.l.o.g. we may assume that \( m \geq m_{e} \). So in the computation \( \Phi^{C \cup D}_{e}(x) \) any even query \( y \) with \( y \geq x \) will be answered negatively since \( C \cap [2^{m}, 2^{m} + e] = \emptyset \) and \( y \leq \Phi_{e}(x) = 2^{m} + e. \) For an even query \( y < x \), compute \( m' < m \) such that \( y \in [2^{m'}, 2^{m'+1}] \). Then, by using \( A \upharpoonright 2^{m'} + 1 \) as an oracle, check whether \( 2^{m'} \in A \). If \( 2^{m'} \notin A \) then \( y \notin C \). Otherwise, \( y \in C \) if and only if \( c(n) = y \) for the unique \( n \) such that \( a(n) = 2^{m'} \).

Now the enumeration function \( c \) of \( C \) is inductively defined as follows. Given \( s \geq 0 \) and \( c(0), \ldots, c(s-1) \), let \( A_{s-1} = \{a(0), \ldots, a(s-1)\} \), \( B_{s-1} = \{b(0), \ldots, b(s-1)\} \) and \( C_{s-1} = \{c(0), \ldots, c(s-1)\} \). Say that requirement \( \mathcal{R}_{e} \) requires attention at stage \( s \) if \( e \leq s, a(s) + 3e + 1 \leq 2a(s) \),

\[
C_{s-1} \upharpoonright a(s) + 3e + 1 = \Phi^{B_{s-1}}_{e, s-1} \upharpoonright a(s) + 3e + 1,
\]

and \( b(s) \geq a(s) + 3e + 1 \). If no requirement requires attention then let \( c(s) = 2a(s) - 2 \). Otherwise, for the least \( e \) such that \( \mathcal{R}_{e} \) requires attention, let \( c(s) = a(s) + 2e \) and say that requirement \( \mathcal{R}_{e} \) is active at stage \( s \).

Obviously, the enumeration function \( c(n) \) is computable and one-to-one. So it suffices to show that \( c(n) \) satisfies (11.2) to (11.4) and that, for \( C = \{c(n) : n \geq 0\} \), the requirements \( \mathcal{R}_{e} \) are met. Now (11.2) and (11.3) are obvious. Moreover, for a proof of (11.4) note that, for sufficiently large \( s \), \( c(s) \leq a(s) + e \) only if a requirement \( \mathcal{R}_{e'} \) where \( e' \leq e \) is active at stage \( s \) hence requires attention at stage \( s \). So it suffices to prove the following claim.

Claim. Every requirement \( \mathcal{R}_{e} \) requires attention at most finitely often and is met.

The proof of the claim is by induction on \( e \). Fix \( e \) and, by inductive hypothesis, choose a stage \( s_{-1} > e \) such that no requirement \( \mathcal{R}_{e'} \) with \( e' < e \) requires attention after stage \( s_{-1} \).
Next observe that if $\mathcal{R}_e$ would require attention infinitely often then there were infinitely many stages $s$ such that (11.5) holds. Since $\lim_{s \to \infty} a(s) = \infty$ and since $\tilde{\Phi}_e$ is a $cl$-functional this would imply that $C = \tilde{\Phi}_e^B$, i.e., that $\mathcal{R}_e$ is not met.

So it suffices to show that $\mathcal{R}_e$ is met. For a contradiction assume that

$$C = \tilde{\Phi}_e^B.$$  \hfill (11.6)

Then, by using $B \upharpoonright 2^m + 3e + 1$ as an oracle, compute a stage $s_m > s_{m-1} > \cdots > s_{-1} > e$ such that

$$B_{s_{m-1}} \upharpoonright 2^m + 3e + 1 = B \upharpoonright 2^m + 3e + 1 \quad (11.7)$$

and

$$C_{s_{m-1}} \upharpoonright 2^m + 3e + 1 = \tilde{\Phi}_{e,s_{m-1}}^B \upharpoonright 2^m + 3e + 1. \quad (11.8)$$

We claim that $A(2^m) = A_{s_{m-1}}(2^m)$ whence $A(2^m)$ can be computed from $B \upharpoonright 2^m + 3e + 1$, i.e., $A \leq_{cl} B$ contrary to choice of $A$ and $B$.

For a contradiction assume that there is a stage $s^* \geq s_m$ such that $a(s^*) = 2^m$.

Note that, by (11.7) and by $\tilde{\Phi}_e(x) \leq x + e$, the $\tilde{\Phi}_e$ computations in (11.8) are $B$-correct whence, by (11.6),

$$\forall s \geq s_m \left[ C_{s-1} \upharpoonright 2^m + 3e + 1 = \tilde{\Phi}_{e,s-1}^B \upharpoonright 2^m + 3e + 1 = \tilde{\Phi}_e^B \upharpoonright 2^m + 3e + 1 \right]. \quad (11.9)$$

Now, since $a(s^*) = 2^m$ and $s^* \geq s_m$ it follows that (by (11.9)) (11.5) holds for $s^* = s$ and (by (11.7)) $b(s^*) \geq a(s^*) + 3e + 1$. Hence $\mathcal{R}_e$ requires attention and becomes active at stage $s^*$. So $c(s^*) = 2^m + 2e = a(s^*) + 2e$ is enumerated into $C$ at stage $s^*$ whence, by (11.9),

$$C(2^m + 2e) = C_{s^*}(2^m + 2e) \neq C_{s^*-1}(2^m + 2e) = \tilde{\Phi}_{e,s^*-1}^B(2^m + 2e) = \tilde{\Phi}_e^B(2^m + 2e).$$

But this contradicts (11.6).

This completes the proof of the claim and the proof of the theorem. \[\square\]

**Corollary 11.10** *The first order theory* $\text{Th}(R_{ibT}, \leq)$ *of the partial ordering of the c.e.* ibT*-degrees and the first order theory* $\text{Th}(R_{cl}, \leq)$ *of the partial ordering of the c.e. cl*-degrees are different.*

**Proof.** This is immediate by Theorems 11.2 and 11.8. \[\square\]
Nondensity of the strongly bounded c.e. degrees

This chapter has to be added!
Joins and meets: local vs. global structure

For any of the reducibilities \( r = \text{ibT}, \text{clT}, bT, T \) we are considering, the partial ordering \((R_r, \leq)\) of the c.e. \( r \)-degrees is a subordering of the partial ordering \((D_r, \leq)\) of all \( r \)-degrees.\(^1\) This leads to the question whether the greatest lower bound (meet) of two c.e. \( r \)-degrees \( a \) and \( b \) in the partial ordering \((R_r, \leq)\) (if any) coincides with the greatest lower bound of these degrees in the partial ordering \((D_r, \leq)\) (and similarly for least upper bounds, i.e., for joins).

We first discuss this question for meets and then have a quick look at joins.

### 13.1 Meets of c.e. degrees in the c.e. degrees and in the degrees in general

Note that if two c.e. \( r \)-degrees \( a \) and \( b \) have a meet \( c \) in the global degrees \((D_r, \leq)\) and a meet \( d \) in the c.e. degrees \((R_r, \leq)\) then \( d \leq c \). Moreover, \( c = d \) iff \( c \) is c.e.

For the Turing degrees Lachlan (196?) has shown that the meet of a pair of c.e. \( r \)-degrees \( a \) and \( b \) exists in \((D_r, \leq)\) if and only if it exists in \((D_r, \leq)\), and if the meet exists the it agrees in both structures. By the preceding remarks this is immediate by the following lemma.

**Lemma 13.1 (Lachlan’s Lemma)** Let \( A, B, C \) be sets such that \( A \) and \( B \) are c.e. and \( C \leq_T A, B \). There is a c.e. set \( \hat{C} \) such that \( C \leq_T \hat{C} \leq_T A, B \).

**Proof.** Fix T-functionals \( \Phi \) and \( \Psi \) such that \( C = \Phi^A = \Psi^B \), fix computable enumerations \( \{A_s\}_{s \geq 0}, \{B_s\}_{s \geq 0}, \{\Phi_s\}_{s \geq 0} \) and \( \{\Psi_s\}_{s \geq 0} \) of \( A, B, \Phi \) and \( \Psi \), respectively,

---

\(^1\)(\(R_r, \leq\)) is not an ideal of \((D_r, \leq)\), however. For Turing reducibility this has been shown by Shoenfield (195?). His proof can be combined with the permitting technique to show that, for any noncomputable c.e. set \( A \) there is a set \( B \) such that \( B \leq_{\text{ibT}} A \) (hence \( B \leq_T A \)) and \( \text{deg}^T(B) \) (hence \( \text{deg}^T(B) \)) is not c.e.
and let $\hat{C}$ be the deficiency set of the simultaneous reductions $\Phi^A = \Psi^B$ defined by

$$\hat{C} = \{ \langle x, s \rangle : \Phi^A_s(x) = \Psi^B_s(x) \downarrow \land \exists s' > s [\Phi^A_{s'}(x) = \Psi^B_{s'}(x) \downarrow \land \Phi^A_s(x) \neq \Phi^A_{s'}(x)] \}$$

$$= \{ \langle x, s \rangle : \Phi^A_s(x) = \Psi^B_s(x) \downarrow \land \exists s' > s [\Phi^A_{s'}(x) = \Psi^B_{s'}(x) \downarrow \land \Psi^B_s(x) \neq \Psi^B_{s'}(x)] \}$$

Obviously $\hat{C}$ is c.e. So it suffices to show that $C \leq_T \hat{C}$ and $\hat{C} \leq_T A, B$.

$C \leq_T \hat{C}$. Given $x, C(x)$ can be computed from $\hat{C}$ as follows. Using $\hat{C}$ as an oracle find $s$ minimal such that $\Phi^A_s(x) = \Psi^B_s(x) \downarrow$ and $\langle x, s \rangle \notin \hat{C}$. Then $C(x) = C_s(x)$.

$\hat{C} \leq_T A, B$. By symmetry, it suffices to show $\hat{C} \leq_T A$. Given a coded pair $\langle x, s \rangle$, $\hat{C}(\langle x, s \rangle)$ can be computed from $A$ as follows. Using $A$ as an oracle find the least stage $t$ such that $\Phi^A_{s'}(x) = \Psi^B_{t'}(x) \downarrow$ and $A \upharpoonright q^A_{s'}(x) + 1 = A_t \upharpoonright q^A_{t'}(x) + 1$. Then $\langle x, s \rangle \in C$ if and only if $s < t$, $\Phi^A_{s'}(x) = \Psi^B_{t'}(x) \downarrow$, and there is a stage $s'$ such that $s < s' \leq t$, $\Phi^A_{s'}(x) = \Psi^B_{t'}(x) \downarrow$ and $\Phi^A_{s'}(x) \neq \Phi^A_{s''}(x)$. \hfill $\square$

The proof of Lemma 13.1 fails to establish the corresponding fact for the reducibilities $r = bT, cl, ibT$. Namely, if in the above proof the functionals $\Phi$ and $\Psi$ are $r$-functionals (for $r = bT, cl, ibT$) then the reductions $\hat{C} \leq_T A, B$ will be $r$-reductions too since for computing $\hat{C}(\langle x, s \rangle)$ from $A$ only oracle $A \upharpoonright q^A_{s'}(x) + 1$ is needed (and similarly for $B$). The reduction $C \leq_T \hat{C}$, however, in general will only be a Turing reduction.

We obtain Lachlan’s Lemma for $bT$-reducibility (and, simultaneously for $T$-reducibility), however, if we replace the set $\hat{C}$, i.e., the deficiency set of the simultaneous reductions of $C$ to $A$ and $B$, by the set $\hat{C}$ which counts the number of occuring mind changes at the stages where the reductions agree and converge:

$$\hat{C} = \{ \langle x, n \rangle : \exists s_0 < s_1 < \ldots < s_n (\forall m \leq n [\Phi^A_{s_m}(x) = \Psi^B_{s_m}(x) \downarrow \land \\
\forall m < n [\Phi^A_{s_m}(x) \neq \Phi^A_{s_{m+1}}(x)]) \}$$

$$= \{ \langle x, n \rangle : \exists s_0 < s_1 < \ldots < s_n (\forall m \leq n [\Phi^A_{s_m}(x) = \Psi^B_{s_m}(x) \downarrow \land \\
\forall m < n [\Psi^B_{s_m}(x) \neq \Psi^B_{s_{m+1}}(x)]) \}$$

We leave the proof that for $bT$-functionals $\Phi$ and $\Psi$, $C \leq_{bT} \hat{C} \leq_{bT} A, B$ holds as an exercise. (Hint: use that the number of mind changes is bounded by the use function).

Just as the original proof, this variation of the proof of Lemma 13.1, however, does not work for the strongly bounded Turing reducibilities. Namely, for $sbT$-functionals $\Phi$ and $\Psi$, the reduction from $C$ to $\hat{C}$ in general will not be an sbT-reduction. In fact, as we will show below, Lachlan’s Lemma fails for the strongly bounded Turing reducibility $\Phi^A = \Psi^B$.
13.1. Meets of c.e. degrees in the c.e. degrees and in the degrees in general

bounded Turing reducibilities. Still we get the following variation of Lachlan’s Lemma which uniformly works for \( r = \text{ibT}, \text{cl}, bT, T \) but can be applied to disjoint sets \( A \) and \( B \) only. (Note that for \( r = \text{bT}, T \) this limitation is not relevant, since, for any c.e. sets \( A \) and \( B \) there are \( r \)-equivalent c.e. sets \( \hat{A} \) and \( \hat{B} \) such that \( \hat{A} \cap \hat{B} = \emptyset \). For \( r = \text{ibT}, \text{cl} \), however, this is not always true.)

**Lemma 13.2 (Generalized Lachlan’s Lemma)** Let \( r = \text{ibT}, \text{cl}, bT, T \) and let \( A, B, C \) be sets such that \( A \) and \( B \) are c.e., \( A \cap B = \emptyset \) and \( C \leq_r A, B \). There is a c.e. set \( \check{C} \) such that \( C \leq_r \check{C} \leq_r A, B \).

**Proof.** The proof resembles the proof of the Representation Lemma (Lemma 4.3). Fix \( r \)-functionals \( \Phi \) and \( \Psi \) such that \( C = \Phi^A = \Psi^B \), and fix computable enumerations \( \{A_s\}_{s \geq 0}, \{B_s\}_{s \geq 0}, \{\Phi_s\}_{s \geq 0} \) and \( \{\Psi_s\}_{s \geq 0} \) of \( A, B, \Phi \) and \( \Psi \), respectively. W.l.o.g. we may assume that \( A \) and \( B \) are infinite and that \( A_s \neq A_{s+1} \) and \( B_s \neq B_{s+1} \) for \( s \geq 0 \).

Define the length function \( l \) by

\[
l(s) = \max \{ x \leq s : \forall y < x (\Phi_s^A(y) = \Psi_s^B(y) \downarrow) \} \]

and call a stage \( s \) expansionary if \( l(t) < l(s) \) for all \( t < s \). Note that, by \( C = \Phi^A = \Psi^B \), \( \lim_{s \to \omega} l(s) = \omega \). So there are infinitely many expansionary stages, say

\[
0 = s_0 < s_1 < s_2 < s_3 < \ldots
\]

Let \( a(n) \) be the least element of \( A_{s_{n+1}} \setminus A_{s_n} \) and \( b(n) \) be the least element of \( B_{s_{n+1}} \setminus B_{s_n} \). Then \( \check{C} \) is defined by

\[
\check{C} = \{ \max \{a(n), b(n)\} : n \geq 0 \}.
\]

Obviously, \( \check{C} \) is c.e. So it suffices to show that

\[
\check{C} \leq \text{ibT} A, B \tag{13.1}
\]

and

\[
C \leq_r \check{C} \tag{13.2}
\]

Proof of (13.1). By symmetry, it suffices to prove \( \check{C} \leq \text{ibT} A \). Given \( x \), \( \check{C}(x) \) is computed from \( A \upharpoonright x + 1 \) as follows. For any \( y \in A \upharpoonright x + 1 \) compute the unique \( n_y \) such that \( y = a(n_y) \). (Note that \( n_y \) is unique since the function \( a(n) \) is 1-1.) Then let \( x \in \check{C} \) if and only if \( x \in \{ \max \{ a(n_y), b(n_y) \} : y \leq x \text{ and } y \in A \} \).

Proof of (13.2). We give the proof for \( r = \text{ibT} \). The proof for the other reducibilities \( r \) is similar. Given \( x \), it suffices to compute \( C(x) \) from oracle \( \check{C} \upharpoonright x + 1 \). This is done as follows. For any \( y \in \check{C} \upharpoonright x + 1 \) compute the unique \( n'_y \) such that
y = \max\{a(n'_x), b(n'_y)\} \quad \text{(note that uniqueness of } n'_y \text{ follows from } A \cap B = \emptyset \text{ and the fact that the functions } a(n) \text{ and } b(n) \text{ are 1-1), and let } n^* \text{ be the least number } n \text{ such that } n \text{ is greater than } x \text{ and greater than the maximum of these numbers } n'_x. \text{ We claim that}

\[ C(x) = \Phi_{s_n}^{A_{n}}(x) = \Psi_{s_n}^{B_{n}}(x) \downarrow. \]

(Note that \( \Phi_{s_n}^{A_{n}}(x) = \Psi_{s_n}^{B_{n}}(x) \downarrow \) by \( n^* > x \) and by choice of the stages \( s_n \).) It suffices to show that, for all \( n \geq n^* \), \( \Phi_{s_n}^{A_{n}}(x) = \Phi_{s_{n+1}}^{A_{n+1}}(x) \). For a contradiction assume that there is a number \( n \geq n^* \) such that \( \Phi_{s_n}^{A_{n}}(x) \neq \Phi_{s_{n+1}}^{A_{n+1}}(x) \). Since the stages \( s_n \) and \( s_{n+1} \) are expansionary it follows that \( \Psi_{s_n}^{B_{n}}(x) \neq \Psi_{s_{n+1}}^{B_{n+1}}(x) \) too. Since \( \Phi \) and \( \Psi \) are ibT-functionals it follows that numbers \( \leq x \) must enter \( A \) and \( B \) between stage \( s_n \) and \( s_{n+1} \) and hence that \( a(n), b(n) \leq x \). So, for \( y = \max\{a(n), b(n)\}, y \in \tilde{C} \uparrow x + 1 \) and, for the corresponding unique \( n'_x, n'_y = n \geq n^* \). But this contradicts choice of \( n^* \).

This completes the proof. \( \square \)

**Theorem 13.3** Let \( r \in \{\text{ibT, cl, bT, T}\} \). For any pair of disjoint c.e. sets \( A \) and \( B \), the meet of \( \deg_r(A) \) and \( \deg_r(B) \) exists is \( (R_r, \leq) \) if and only if the meet of \( \deg_r(A) \) and \( \deg_r(B) \) exists is \( (D_r, \leq) \). Moreover, if the meets exist then they agree.

As pointed out above, for \( r \in \{\text{bT, cl}\} \) this implies that a pair of c.e. \( r \) degrees has a meet in \( (R_r, \leq) \) if and only if it has a meet in \( (D_r, \leq) \) and if so the meet in both structures is the same. For \( r \in \{\text{ibT, cl}\} \), however, this not true in general.

Here we will only show that, for \( r \in \{\text{ibT, cl}\} \), Lachlan’s Lemma fails.

**Lemma 13.4** Let \( r \in \{\text{ibT, cl}\} \). There are sets \( A, B \) and \( C \) such that \( A \) and \( B \) are c.e., \( C \leq_r A, B \), and there is no c.e. set \( D \) such that \( C \leq_r D \leq_r A, B \).

**Proof (sketch).** Let \( r = \text{ibT} \). (The case of \( r = \text{cl} \) is similar.) We construct sets \( A, B, C \) with the desired properties by a finite injury argument. As usual we give computable enumerations \( \{A_i\}_{i \geq 0} \) and \( \{B_i\}_{i \geq 0} \) of the c.e. sets \( A \) and \( B \) where \( A_i \) and \( B_i \) are the finite parts of these sets enumerated by the end of stage \( s \) of the construction. Simultaneously we give a computable (nonmonotone) approximation \( \{C_i\}_{i \geq 0} \) of \( C \) (i.e., a computable strong array of finite sets \( C_i \)) such that, for any \( x \geq 0 \),

\[
\lim_{s \to \infty} C_s(x) = C(x),
\]

i.e., \( C_s(x) = C(x) \) for all sufficiently large \( s \).

Then in order to ensure that \( C \leq_{\text{ibT}} A, B \) it suffices to ensure that, for any \( x, s \geq 0 \),

\[
C_s(x) \neq C_{s+1}(x) \Rightarrow \exists y \leq x (y \in A_{s+1} \setminus A_s) \text{ and } \exists y' \leq x (y' \in B_{s+1} \setminus B_s)
\]
hence. Note that this will also ensure that \( \lim_{n \to \infty} C_s(x) \) exists.

In order to ensure that there is no c.e. set \( D \) such that \( C \leq_{ibT} D \leq_{ibT} A, B \) it suffices to meet the requirements

\[
\mathcal{R}_e : C = \Phi^{W_{e_0}}_{\langle e_1, e_2, e_3 \rangle} \Rightarrow W_{e_0} \neq \Phi^A_{e_2} \text{ or } W_{e_0} \neq \Phi^B_{e_3}
\]

for \( e = \langle e_0, e_1, e_2, e_3 \rangle \geq 0 \).

The strategy for meeting requirement \( \mathcal{R}_e \) (\( e = \langle e_0, e_1, e_2, e_3 \rangle \)) is as follows.

1. Pick unused numbers \( x - 1 \) and \( x \).
2. Wait for a stage \( s \) (if any) such that

\[
C_s(x) = \Phi^{W_{e_0}^{s'}}_{\langle e_1, s' \rangle} (x) \land W_{e_0} \upharpoonright x + 1 = \Phi^{s}_{e_2, s} \upharpoonright x + 1 = \Phi^{s}_{e_1, s} \upharpoonright x + 1
\]

Then put \( x \) into \( C \) and \( A \) and \( x - 1 \) into \( B \) at stage \( s + 1 \). Moreover, restrain \( A \upharpoonright x \).

3. Wait for a stage \( s' > s + 1 \) (if any) such that the computations have been recovered, i.e.,

\[
C_{s'}(x) = \Phi^{W_{e_0}^{s'}}_{\langle e_1, s' \rangle} (x) \land W_{e_0} \upharpoonright x + 1 = \Phi^{s'}_{e_2, s'} \upharpoonright x + 1 = \Phi^{s'}_{e_1, s'} \upharpoonright x + 1
\]

Then extract \( x \) from \( C \), put \( x - 1 \) into \( A \) and \( x \) into \( B \). Moreover, restrain \( B \upharpoonright x \).

Note that this strategy is compatible with our strategy for making \( C \) ibT-reducible to \( A \) and \( B \). To show that this strategy is successful, for a contradiction, assume that \( C = \Phi^{W_{e_0}}_{\langle e_1, e_2, e_3 \rangle} \) and \( W_{e_0} = \Phi^A_{e_2} = \Phi^B_{e_3} \). Then there must be stages \( s < s' \) as in steps 2 and 3 of the attack. By the described action of step 2, \( x \) must enter \( W_{e_0} \) by stage \( s' \) (note that by the restraint imposed on \( A \) it follows from \( W_{e_0} = \Phi^A_{e_2} \) that no number \( < x \) can enter \( W_{e_0} \) in order to restore \( 1 = C_s(x) = \Phi^{W_{e_0}^{s'}}_{\langle e_1, s' \rangle} (x) \). Now at stage \( s' + 1, C(x) \) is reset to 0, namely \( C_{s'+1}(x) = 0 \). So in order to restore the computation \( C(x) = \Phi^{W_{e_0}}_{\langle e_1, e_2, e_3 \rangle} \) the right hand side must change which requires \( W_{e_0} \upharpoonright x + 1 \) to change after stage \( s' \). By the restraint imposed on \( B \) at stage \( s' \), however, it follows from \( W_{e_0} \upharpoonright x + 1 = \Phi^{s'}_{e_2} \upharpoonright x + 1 \) and our assumption that \( W_{e_0} \upharpoonright x \) cannot change after stage \( s' \). In fact, since \( x \) entered \( W_{e_0} \) by stage \( s' \), this is true for \( W_{e_0} \upharpoonright x + 1 \). So the change of \( C(x) \) cannot be recorded by \( \Phi^{W_{e_0}}_{\langle e_1, e_2, e_3 \rangle} \) whence \( C(x) \neq \Phi^{W_{e_0}}_{\langle e_1, e_2, e_3 \rangle}(x) \) contrary to assumption.

Since the strategies for meeting the requirements \( \mathcal{R}_e \) are finitary, the combination of the requirements can be done in the standard finite-injury style. We leave this an exercise.\( \square \)
13.2 Joins of c.e. degrees in the c.e. degrees and in the degrees in general

For \( r = b'T, T \), the join of the \( r \)-degrees of sets \( A \) and \( B \) is represented by the disjoint union \( A \oplus B = 2A + 2B + 1 \) of these sets. Moreover, for c.e. sets \( A \) and \( B \), \( A \oplus B \) is c.e. again. This easily shows that the join of any two c.e. \( r \)-degrees exists in both, in the partial ordering \((R_r, \leq)\) of the c.e. \( r \)-degrees and in the partial ordering \((D_r, \leq)\) of all degrees, and agrees. So, in particular, the dual of Lachlan’s Lemma is true for these reducibilities.

**Lemma 13.5** For \( r \in \{b'T, T\} \) the following holds. Let \( A, B, C \) be sets such that \( A \) and \( B \) are c.e. and \( A, B \leq_r C \). There is a c.e. set \( \hat{C} \) such that \( A, B \leq_r \hat{C} \leq_r C \).

**Proof.** Let \( \hat{C} = A \oplus B \). \( \square \)

For \( r \in \{ib'T, cl\} \), however, the dual of Lachlan’s Lemma fails. By the existence of \( r \)-minimal pairs, this follows from the observation that, for any left-computable Martin-Löf random real \( C \), \( A \leq_r \) for all c.e. sets \( A \) (see Downey and Hirschfeldt [DH10]).

**Lemma 13.6** For \( r \in \{ib'T, cl\} \) the following holds. There are sets \( A, B, C \) such that \( A \) and \( B \) are c.e., \( A, B \leq_r C \) and there is no c.e. set \( \hat{C} \) such that \( A, B \leq_r \hat{C} \leq_r C \).
Further results and open problems

In our final lecture we summarize some of the results presented, mention some further results and give some directions for further research.

For details, see the slides of this lecture.
Partial orderings and lattices

In this appendix we shortly summarize the basic definitions and facts on partial orderings and lattices used in this course.

A.1 Partial orderings

Definition A.1 Let $P$ be any set and let $\leq \subseteq P \times P$ be a binary relation on $P$.

(i) $\leq$ is reflexive if, for any $a \in P$, $a \leq a$.

(ii) $\leq$ is transitive if, for any $a, b, c \in P$, $a \leq b$ and $b \leq c$ implies $a \leq c$.

(iii) $\leq$ is antisymmetric if, for any $a, b \in P$, $a \leq b$ and $b \leq a$ implies that $a = b$.

(iv) $\leq$ is total if, for any $a, b \in P$, $a \leq b$ or $b \leq a$.

Definition A.2 (Preorderings, partial orderings and linear orderings) (a) A preordering $(P, \leq)$ is a set $P$ together with a binary relation $\leq \subseteq P \times P$ such that $\leq$ is reflexive and transitive.

(b) A partial ordering (p.o.) $(P, \leq)$ is a set $P$ together with a binary relation $\leq \subseteq P \times P$ such that $\leq$ is reflexive, transitive and antisymmetric (i.e., a p.o. is an antisymmetric preordering).

(c) A total or linear ordering (l.o.) $(P, \leq)$ is a set $P$ together with a binary relation $\leq \subseteq P \times P$ such that $\leq$ is reflexive, transitive, antisymmetric and total (i.e., an l.o. is a total partial ordering).

Lemma A.3 Let $(P, \leq)$ be a preordering. Then

$$p \equiv q \iff p \leq q \& q \leq p$$
is an equivalence relation, i.e., $\equiv$ is reflexive, transitive and symmetric (i.e., $p \equiv q$ implies $q \equiv p$). Moreover, the set $([P], \leq')$ of the $\equiv$-classes $[p]$ of the elements of $P$ together with the relation

$$[p] \leq' [q] \iff p \leq q$$

is a partial ordering.\(^1\)

PROOF. Exercise! □

For a partial ordering $\leq$ we use the following abbreviations:

$$a < b \iff a \leq b & a \neq b$$

$$a \geq b \iff b \leq a$$

$$a \mid b \iff a \nleq b & b \nleq a$$

If $a \mid b$ then we say that $a$ and $b$ are incomparable; otherwise $a$ and $b$ are comparable.

Note that a p.o. $(P, \leq)$ is total if all $a, b \in P$ are comparable. Moreover, a partial ordering $(P, \leq)$ is called finite (infinite, countable, etc.) if $P$ is finite (countable, infinite, etc.).

Example A.4 (a) Examples of (infinite) linear orderings are the integers, the negative integers, the natural numbers (= nonnegative integers), the rational numbers, and the real numbers together with the usual orderings: $(\mathbb{Z}, \leq)$, $(\mathbb{Z}^-, \leq)$, $(\mathbb{N}, \leq)$, $(\mathbb{Q}, \leq)$, and $(\mathbb{R}, \leq)$. Moreover, any proper initial segment $(\mathbb{N} \upharpoonright n, \leq)$ of $\mathbb{N}$ together with the natural ordering is a finite l.o. ($n \geq 0$).

(b) The set $\{0, 1\}^*$ of all finite binary strings together with the relation

$$w \subseteq w' \iff w \text{ is an initial segment of } w'$$

is an infinite partial ordering but not an l.o.

$$(\{0, 1\}^*, \leq_L)$$

is also a p.o. (but not an l.o.) where

$$w \leq_L w' \iff w = w' \text{ or there are strings } u, v, v' \text{ s.t. } w = u0v \text{ und } w' = u1v'$$

(i.e., if we look at binary words as the nodes of the full infinite binary tree then $w \leq_L w'$ if $w$ is to the left of $w'$).

Finally, $\{0, 1\}^*$ is a linear ordering where

$$w \leq w' \iff |w| < |w'| \text{ oder } (|w| = |w'| \text{ und } w \leq_L w')$$

(where $|w|$ is the length of $w$).

---

\(^1\)The common computable reducibilities $\leq_r$ considered in computability theory are preorders where the corresponding equivalence classes are called $r$-degrees. So these reducibilities induce partial orderings on the corresponding degree classes.
(c) For any set $S$, the power set $\mathcal{P}(S)$ together with the inclusion relation $\subseteq$ is a partial ordering. The partial ordering $(\mathcal{P}(S), \subseteq)$ is finite if and only if $S$ is finite, and $(\mathcal{P}, \subseteq)$ is linear if and only if $S$ has at most one element.

Similarly, for any set $S$, $(\mathcal{P}^*, \subseteq^*)$ is a preordering, where

$$A \subseteq^* B \Leftrightarrow A \setminus B \text{ finite},$$

and $(\mathcal{P}^*(S), \leq^*)$ is a partial ordering where the elements of $\mathcal{P}^*(S)$ are the equivalence classes $[A]^* = \{ B \subseteq S : B =^* A \}$ of the subsets $A$ of $S$ under almost equality and

$$[A]^* \leq^* [B]^* \Leftrightarrow A \subseteq^* B.$$

**Definition A.5** Let $(P, \leq)$ be a partial ordering.

(a) An element $a$ of $P$ is the least (greatest) element of $(P, \leq)$ if $a \leq b$ ($a \geq b$) for all $b \in P$, and $a \in P$ is a minimal (maximal) element of $(P, \leq)$ if there is no $b \in P$ such that $b < a$ ($a < b$).

(b) $(P, \leq)$ is dense if, for any $a, b \in P$ such that $a < b$ there is a $c \in P$ such that $a < c < b$.

Note that the least and greatest elements of a partial ordering (if any) are unique (whereas there might be more than one maximal element or minimal element). For the least element of a partial ordering $(P, \leq)$ we write $0_{(P, \leq)}$ (or simply $0_P$ or 0 if $\leq$ or $(P, \leq)$ are known from the context). Similarly, $1_{(P, \leq)}$ (or simply $1_P$ or 1) denotes the greatest element of $(P, \leq)$. Note that if there is a least element (greatest element) then this is the unique minimal element (maximal element). Also note that in a linear ordering the only minimal element (if any) is the least element and the only maximal element (if any) is the greatest element. Finally, note that a dense p.o. is infinite.

**Example A.6** The linear ordering $(\mathbb{N}, \leq)$ possesses a least element (0) but no maximal elements (hence no greatest element). Similarly, the linear ordering $(\mathbb{Z}^-, \leq)$ of the negative integers possesses a greatest element (−1) but no minimal elements. The l.o.s $(\mathbb{Z}, \leq)$, $(\mathbb{Q}, \leq)$, and $(\mathbb{R}, \leq)$ possess neither minimal nor maximal elements. The l.o.s $(\mathbb{Q}, \leq)$ and $(\mathbb{R}, \leq)$ are dense whereas $(\mathbb{N}, \leq)$ and $(\mathbb{Z}, \leq)$ are not dense.

The p.o.s $(\{0, 1\}^*, \subseteq)$, $(\{0, 1\}^*, \leq_L)$, and $(\{0, 1\}^*, \leq)$ are not dense. $(\{0, 1\}^*, \subseteq)$ and $(\{0, 1\}^*, \leq)$ have a least element, namely the empty string $\lambda$ but no maximal elements. $(\{0, 1\}^*, \leq_L)$ possesses infinitely many minimal elements (namely the words $0^n$ for $n \geq 0$) and infinitely many maximal element (namely the words $1^n$ for $n \geq 0$) but neither a least nor a greatest element.
Exercise A.7 Analyse the p.o.s \((\mathcal{P}, \subseteq), (\mathcal{P}^- (S), \subseteq), \) and \((\mathcal{P}^* (S), \leq^*)\) w.r.t. to density, minimal and maximal element, and least and greatest elements where \(\mathcal{P}^- (S) = \mathcal{P}(S) \setminus \{\emptyset, S\}\). Consider the cases where \(S\) is finite and the case \(S = \mathbb{N}\).

A.2 Suborderings, embeddings, isomorphisms

Definition A.8 (Suborderings) A partial ordering \((Q, \leq_Q)\) is a subordering of a partial ordering \((P, \leq_P)\) if \(Q \subseteq P\) and \(\leq_Q\) is the restriction of \(\leq_P\) to \(Q\) (i.e., for \(a, b \in Q\), \(a \leq_Q b\) if and only if \(a \leq_P b\)).

Note that the suborder relation is reflexive and transitive. Also note that \((\mathbb{N}, \leq)\) and \((\mathbb{Z}^-, \leq)\) are suborders of \((\mathbb{Z}, \leq)\), \((\mathbb{Z}, \leq)\) is a suborder of \((Q, \leq)\), and \((Q, \leq)\) is a suborder of \((\mathbb{R}, \leq)\). Moreover, \((\mathcal{P}^- (S), \subseteq)\) (see Exercise A.7) is a subordering of \((\mathcal{P}(S), \subseteq)\). The p.o. \((\{0, 1\}^*, \leq_L)\) is not a suborder of \((\{0, 1\}^*, \leq)\), however, since there are words \(w\) and \(w'\) with \(w \leq w'\) but \(w \not\leq_L w'\).

Definition A.9 (Embeddings and isomorphisms) Let \((P, \leq_P)\) and \((Q, \leq_Q)\) be partial orderings.

(a) A map \(f : Q \to P\) is order preserving if
\[
\forall a, b \in Q \ [a \leq_Q b \Rightarrow f(a) \leq_P f(b)].
\]

(b) An (order) embedding of \((Q, \leq_Q)\) into \((P, \leq_P)\) is a one-to-one map \(f : Q \to P\) such that
\[
\forall a, b \in Q \ [a \leq_Q b \iff f(a) \leq_P f(b)].
\]

(c) An (order) isomorphism \(f\) from \((Q, \leq_Q)\) to \((P, \leq_P)\) is an embedding \(f\) of \((Q, \leq_Q)\) into \((P, \leq_P)\) such that \(f\) is onto (hence a bijection).

\((Q, \leq_Q)\) and \((P, \leq_P)\) are isomorphic \(((Q, \leq_Q) \cong (P, \leq_P))\) if there is an isomorphism from \((Q, \leq_Q)\) to \((P, \leq_P)\).

Note that any embedding is order preserving. But there are one-to-one (even bijective) order preserving functions which are not embeddings. (For instance, let \(Q = P = \{a, b\}\) and define \(\leq_Q\) and \(\leq_P\) by letting \(a \mid_Q b\) and \(a \not\leq_P b\). Then the identity function is order preserving but not an embedding (hence not an isomorphism).) Roughly speaking, embeddings do not only preserve ordering but also non-ordering.
As one can easily check, the isomorphism relation $\cong$ is an equivalence relation. The equivalence classes of linear (or partial) orders under $\cong$ are also called order types. The order types of $(\mathbb{N}, \leq)$, $(\mathbb{Z}^-, \leq)$ and $(\mathbb{Q}, \leq)$ are often denoted by $\omega$, $\omega^*$ and $\delta$, respectively. Orders and order types are concatenated in the obvious way. So, for example, $\omega^*\omega$ is the order type of $(\mathbb{Z}, \leq)$.

Note that if $f$ is an embedding of $(Q, \leq_Q)$ into $(P, \leq_P)$ then $(f(Q), \leq_P)^2$ is a subordering of $(P, \leq_P)$ which is isomorphic to $(Q, \leq_Q)$. Conversely, any subordering of $(P, \leq_P)$ is embeddable into $(P, \leq_P)$ via the identity function. Since sometimes we do not distinguish between isomorphic structures we sometimes call $(Q, \leq_Q)$ a substructure of $(P, \leq_P)$ if $(Q, \leq_Q)$ can be embedded into $(P, \leq_P)$.

**Definition A.10 (Chains and anti-chains)** Let $(P, \leq)$ be a partial ordering.

(a) Let $(I, \leq_I)$ be an (infinite) linear ordering. Then $(a_i : i \in I)$ is an (infinite) chain (of order type $(I, \leq_I)$) in $(P, \leq)$ if, for $i, j \in I$,

$$i <_I j \Rightarrow a_i < a_j$$

holds. A chain $(a_i : i \in I)$ is maximal if there is no $b \in P$ such that $b < a_i$ for all $i \in I$ and no $c \in P$ such that $a_i < c$ for all $i \in I$. A chain $(a_i : i \in I)$ is a skeleton of $(P, \leq)$ if, for any $p \in P$ there are $i, j \in I$ such that $a_i \leq p \leq a_j$.

If $(I, \leq_I) = (\mathbb{N}, \leq)$ then we call a chain $(a_i : i \in I)$ also an infinite ascending chain and call $(a_i : i \in I)$ a maximal ascending chain if there is no $c \in P$ such that $a_i < c$ for all $i \in I$. Similarly, if $(I, \leq_I) = (\mathbb{Z}^-, \leq)$ then we call a chain $(a_i : i \in I)$ also an infinite descending chain and call $(a_i : i \in I)$ a maximal descending chain if there is no $b \in P$ such that $b < a_i$ for all $i \in I$.

(b) An (infinite) subset $A$ of $P$ is an (infinite) anti-chain in $(P, \leq)$ if, for any $a, a' \in A$,

$$a \neq a' \Rightarrow a \nmid a'$$

holds. An anti-chain $A$ is maximal if there is no $p \in P \setminus A$ such that $p \mid a$ for all $a \in A$.

**Example A.11** Let $(P, \leq)$ be a linear ordering. Then, for any $p \in P$, the singleton $\{p\}$ is a maximal anti-chain. Moreover, $(P, \leq)$ is a skeleton (hence a maximal chain). In the partial ordering $(\mathbb{N}, \leq)$ any subset $A \subset \mathbb{N}$ is a chain and this chain is maximal if and only if $0 \in A$ and $A$ is unbounded. Moreover, $A \subset \mathbb{N}$ is a maximal ascending chain if $A$ is unbounded and $A \subset \mathbb{N}$ is a maximal descending chain if $0 \in A$. Finally, any maximal chain of $(\mathbb{N}, \leq)$ is a skeleton.

---

2 Strictly speaking we do not consider the order relation $\leq_P$ here but the restriction of $\leq_P$ to $f(Q) \times f(Q)$. But for ease of notation we denote this restriction of $\leq_P$ by $\leq_P$ again.
In the partial ordering \( (\{0,1\}^*, \leq_L) \), any infinite path (like \( \{0^n : n \geq 0\} \)) is a maximal anti-chain while, for instance, \( \{0^n1 : n \geq 0\} \) is a maximal descending chain.

For a finite set \( S = \{a_0, \ldots, a_n\} \) with at least two elements (\( n \geq 1 \)), the set of the singletons of \( S \), \( \{\{a_0\}, \ldots, \{a_n\}\} \) is a maximal anti-chain of \( (\mathcal{P}(S), \subseteq) \). Another example of a maximal anti-chain is \( \{\{a_i, a_j\} : 0 \leq i < j \leq n\} \). Examples of skeletons of \( (\mathcal{P}(S), \subseteq) \) are \( \{\emptyset, \{a_0\}, \{a_0, a_1\}, \ldots, \{a_0, \ldots, a_n\}\} \) and \( \{\emptyset, \{a_0, \ldots, a_n\}\} \).

(Note that in a linear ordering \( (P, \leq) \) with least and greatest elements 0 and 1, respectively, a set \( A \subseteq P \) is a skeleton of \( (P, \leq) \) if and only if \( \{0, 1\} \subseteq A \).)

### A.3 Automorphisms

**Definition A.12 (Automorphisms)** Let \( (P, \leq) \) be a partial ordering. An automorphism of \( (P, \leq) \) is an isomorphism from \( (P, \leq) \) to \( (P, \leq) \).

Note that the identity function \( id \) on \( P \) is an automorphism of \( (P, \leq) \), that the composition \( g \circ f \) of two automorphisms \( f \) and \( g \) of \( (P, \leq) \) is an automorphism of \( (P, \leq) \), and that the inverse \( f^{-1} \) of an automorphism \( f \) of \( (P, \leq) \) is an automorphism of \( (P, \leq) \). So the automorphisms of \( (P, \leq) \) constitute a group, the automorphism group of \( (P, \leq) \).

**Definition A.13 (Rigidity)** Let \( (P, \leq) \) be a partial ordering. An automorphism \( f \) of \( (P, \leq) \) is nontrivial if \( f \neq id \). \( (P, \leq) \) is rigid if there are no nontrivial automorphisms (i.e., if the automorphism group of \( (P, \leq) \) consists only of the identity function).

**Example A.14** \( (\mathbb{N}, \leq) \) rigid, whereas \( (\mathbb{Z}, \leq) \), \( (\mathbb{Q}, \leq) \) and \( (\mathbb{R}, \leq) \) are not rigid. For \( (\mathbb{Z}, \leq) \) the automorphisms are just the functions \( f_z(x) = x + z \) (for any given \( z \in \mathbb{Z} \)). For \( (\mathbb{Q}, \leq) \) one can construct nontrivial automorphisms by some back-and-forth arguments exploiting density of \( (\mathbb{Q}, \leq) \): For instance, given any maximal chains \( \{p_z : z \in \mathbb{Z}\} \) and \( \{q_z : z \in \mathbb{Z}\} \) of order type \( \omega \cdot \omega \) in \( (\mathbb{Q}, \leq) \), there is an automorphism \( f \) of \( (\mathbb{Q}, \leq) \) satisfying \( f(p_z) = q_z \) for all integers \( z \) (Exercise!).
A.4 Joins and meets

Definition A.15 (Joins and meets) Let \((P, \leq)\) be a partial ordering.

(a) An element \(b\) of \(P\) is called the join or least upper bound (l.u.b.) of elements \(a_0, \ldots, a_n\) of \(P\) \((n \geq 0)\) - and we write \(b = a_0 \lor \cdots \lor a_n\) - if, for all \(m \leq n, a_m \leq b\) (i.e., if \(b\) is an upper bound of \(a_0, \ldots, a_n\)) and if, for any \(c \in P\),
\[
a_0 \leq c \land \cdots \land a_n \leq c \Rightarrow b \leq c
\]
holds.

(b) An element \(b\) of \(P\) is called the meet or greatest lower bound (g.l.b.) of elements \(a_0, \ldots, a_n\) of \(P\) \((n \geq 0)\) - and we write \(b = a_0 \land \cdots \land a_n\) if, for all \(m \leq n, b \leq a_m\) (i.e., if \(b\) is a lower bound of \(a_0, \ldots, a_n\)) and if, for any \(c \in P\),
\[
c \leq a_0 \land \cdots \land c \leq a_n \Rightarrow c \leq b
\]
holds.

Definition A.16 (Ideals and filters) Let \((P, \leq)\) be a partial ordering.

(a) A subset \(I\) of \(P\) is an ideal in \((P, \leq)\) if \(I\) is closed under join and closed downward under \(\leq\), i.e., if
\[
\forall a, b \in I \ [a \lor b \text{ exists } \Rightarrow a \lor b \in I]\]  \hspace{1cm} (A.1)

and
\[
\forall a, b \in P \ [(b \in I \land a \leq b) \Rightarrow a \in I]\]  \hspace{1cm} (A.2)

hold; and \(I\) is a strong ideal if
\[
\forall a, b \in I \exists c \in I \ [a \leq c \land b \leq c]\]  \hspace{1cm} (A.3)

and (A.2) hold.

(b) A subset \(F\) of \(P\) is a filter in \((P, \leq)\) if \(F\) is closed under meet and closed upward under \(\leq\), i.e., if
\[
\forall a, b \in F \ [a \land b \text{ exists } \Rightarrow a \land b \in F]\]  \hspace{1cm} (A.4)

and
\[
\forall a, b \in P \ [(a \in F \land a \leq b) \Rightarrow b \in F]\]  \hspace{1cm} (A.5)

hold; and \(F\) is a strong filter if
\[
\forall a, b \in F \exists c \in F \ [c \leq a \land c \leq b]\]  \hspace{1cm} (A.6)

and (A.5) hold.

Note that for any \(a \in P\), \(P(\leq a) = \{b \in P : b \leq a\}\) is an ideal in \((P, \leq)\), called principal ideal (with top \(a\)). Principal filters \(P(\geq a)\) are defined dually.
A.5 Semi-lattices and lattices

**Definition A.17 (Semi-lattices and lattices)** A partial ordering \((P, \leq)\) is an upper semi-lattice (u.s.l.) if, for any \(a, b \in P\), the join \(a \vee b\) of \(a\) and \(b\) exists, and \((P, \leq)\) is a lower semi-lattice (l.s.l.) if, for any \(a, b \in P\), the meet \(a \wedge b\) of \(a\) and \(b\) exists. \((P, \leq)\) is a lattice if \((P, \leq)\) is an u.s.l. and a l.s.l.

Note that, for \(a \leq b\), \(a = a \wedge b\) and \(b = a \vee b\). So any total ordering is a lattice. The power set \((\mathcal{P}(S), \subseteq)\) of any set ordered by inclusion is a lattice too where joins (meets) are given by unions (intersections).

In a lattice \((P, \leq)\) the order relation is completely determined by the join and meet operations:

\[
a \leq b \iff a = a \land b \iff b = a \lor b.
\]

So lattices are sometime specified by giving the binary join and meet relations: \((P, \lor, \land)\). So, for instance the lattice \((\mathcal{P}(S), \subseteq)\) can be alternatively be given by \((\mathcal{P}(S), \cup, \cap)\).

**Definition A.18 (Distributive and modular lattices)** A lattice \((L, \lor, \land)\) is distributive if the distributivity law

\[
\forall a, b, c \in L \left[ a \lor (b \land c) = (a \lor b) \land (a \lor c) \right] \quad (A.7)
\]

holds.

A lattice \((L, \lor, \land)\) is modular if the modularity law

\[
\forall a, b, c \in L \left[ a \leq c \Rightarrow a \lor (b \land c) = (a \lor b) \land (a \lor c) \right] \quad (A.8)
\]

holds.

Obviously, \((A.7)\) implies \((A.8)\). So any distributive lattice is modular (but the converse is not true; see below). Moreover, the distributivity law \((A.7)\) is equivalent to the dual distributivity law

\[
\forall a, b, c \in L \left[ a \land (b \lor c) = (a \land b) \lor (a \land c) \right] \quad (A.9)
\]

TO BE ADDED: examples \(B_n, M_5, N_5\); boolean algebras; atoms; countable atomless BA.

Distributivity and modularity can be alternatively expressed in terms of the ordering relation which allows us to define distributivity also for partial orderings and semi-lattices and partial orderings.
Definition A.19 (Distributive partial orderings) Let $(P, \leq)$ be a partial ordering.

(a) An element $a$ of $P$ is distributive (in $(P, \leq)$) if, for any $b, c, d \in P$,

$$[a = b \lor c \land d \leq a] \Rightarrow \exists d_b, d_c \in P \ [d_b \leq b \land d_c \leq c \land d = d_b \lor d_c] \quad \text{(A.10)}$$

holds, and $a \in P$ is modular (in $(P, \leq)$) if, for any $b, c, d \in P$,

$$[a = b \lor c \land b \leq d \leq a] \Rightarrow \exists d_b, d_c \in P \ [d_b \leq b \land d_c \leq c \land d = d_b \lor d_c] \quad \text{(A.11)}$$

holds.

(b) $(P, \leq)$ is a distributive (modular) partial ordering if every element of $P$ is distributive (modular) in $(P, \leq)$.

Definition A.20 (Distributive upper semi-lattices) TO BE ADDED

TO BE ADDED: Compatibility of both definitions.

A.6 Lattice embeddings and representations of distributive lattices

TO BE ADDED: (1) definition (semi-)lattice embedding; order embedding not necessarily a lattice embedding (but order isos (and autos) are lattice isos (and autos)) (2) sublattices (3) lattice embeddings $f$ (alternative characterization): order and nonorder + join and meet requirements (4) Representations of finite (countable) distributive lattices (5) characterizations of the nondistributive/nonmodular lattices using the lattices $M_5$ and $N_5$.

A.7 First order logic and partial orderings

Languages $\mathcal{L}(\leq), \ldots, \mathcal{L}(\leq, \lor, \land, 0, 1)$, join / meet / least element / greatest element f.o. definable from ordering, theory of a structure, $\exists - Th$ etc, elementary equivalence vs. isomorphisms, types, categoricity, decidability, definability vs. invariance.
Bibliography


[ABFK10] Ambos-Spies, Klaus; Bodewig, Philipp; Fan, Yun; Kräling, Thorsten. The partial orderings of the computably enumerable ibT-degrees and cl-degrees are not elementarily equivalent (to appear).


[ADFM10] Ambos-Spies, Klaus; Ding, Decheng; Fan, Yun; Merkle, Wolfgang. Maximal Pairs of Computably Enumerable Sets in the Computable-Lipschitz Degrees (to appear).


[AW10] Ambos-Spies, Klaus; Wang, Wei. A lattice which can be embedded into the c.e. ibT-degrees but which cannot be embedded preserving 0. Manuscript.


[FL05] Fan, Yun; Lu, Hong. Some properties of sw-reducibility. Journal of Nanjing University (Mathematical Biquarterly) 2 (2005), 244 – 252.


[LN95] Lempp, Steffen; Nies, André. The undecidability of the $\Pi^0_4$-theory for the r.e. wtt and Turing degrees. J. Symbolic Logic 60 (1995), no. 4, 1118–1136.


[YD04] Yu, Liang; Ding, Decheng. There is no $SW$-complete c.e. real. J. Symbolic Logic 69 (2004), no. 4, 1163–1170.