Undecidability and Definability for Parameterized Polynomial Time m-Reducibilities[1]

Peter Cholak[2]
Mathematics Department
Victoria University
P.O. Box 600, Wellington, New Zealand.
and
Mathematics Department
University of Michigan
Ann Arbor, MI , U.S.A.

Rod Downey[3]
Mathematics Department
Victoria University
P.O. Box 600, Wellington, New Zealand.
and
Mathematics Department
Cornell University
Ithaca, NY14853, U.S.A.

September 29, 2003

Abstract

In the setting of the parameterized reducibilities introduced by the second author and Mike fellows, we prove a number of decidability and definability results. In particular the undecidability of the relevant m-degree structures is proven. The relationship with classical notions is analysed, and this leads to a number of observations about classical constructions in the $PTIME$ degrees. Methods include $\mathbf{0}''$ and $\mathbf{0}'''$ priority arguments combined with speedup type arguments.

Introduction.

From among Anil Nerode's many superb contributions, one recurrent theme is the analysis of definability and decibility results in the structures of recursion theory. For instance the papers [MaN], [MN], [NSm], [NSh] and [NR1,2] are clearly of this ilk. In the present paper we wish to analyse the structures introduced by the first author and Mike Fellows[DF1-5], in the same spirit. This seems particularly apt in view of Anil's interest in polynomial time and polynomialy graded structures[NR3]. In these structures we plan to prove a number of decidability and definability results, as well as examining relationships with classical notions. Before we state specific results it is perhaps appropriate to include a brief recap of the Downey-Fellows setting, and its motivations as it is still rather novel.

While the NP completeness phenomenom is a good tool to explain the apparent intractability of many combinatorial problems, it is really a fairly coarse measure in the sense that from a practical viewpoint many NP complete problems can behave quite differently with to respect to the spectrum of solutions. Furthermore NP completeness does not say much about intractability in P. To be specific, many combinatorial problems have the prop-

---

[1]Dedicated to Anil Nerode on the occasion of his 60 th birthday

[3]partially supported by a Victoria University Postdoctoral Fellowship, and by an N.S.F. Postdoctoral Fellowship.

[3]Partially supported by the Victoria University IGC, the U.S.-New Zealand cooperative foundation grant INT 90-20558, Cornell University and the MSI at Cornell.

erty that their input consists of one or more parameters. Consider the following examples.

Example 1. The Vertex Cover problem takes as input a pair (G, k) consisting of a graph G and a positive integer k, and determines whether there is a set of k vertices in G having the property that every edge in G has at least one endpoint in this set.

Example 2. The Graph Genus problem takes as input a pair (G, k) as above, and determines whether the graph G embeds on the surface of genus k.

Example 3. The Planar Improvement problem takes as input a pair (G, k) as above, and determines whether G is a subgraph of a planar graph G' of diameter at most k.

Example 4. The Graph Linking Number problem takes as input a pair (G, k) as above, and determines whether G can be embedded in 3-space so that at most k disjoint cycles in G are topologically linked.

Example 5. The Dominating Set problem takes as input a pair (G, k) as above, and determines whether there is a set of k vertices in G having the property that every vertex of G either belongs to the set, or has a neighbor in the set.

Example 6. The Weighted CNF Satisfiability problem takes as input a pair (ϕ, k) where ϕ is a propositional (boolean) formula in conjunctive normal form, and k is a positive integer, and determines whether there is a weight k satisfying truth assignment to the variables of ϕ. (A truth assignment has *weight* k if it assigns exactly k variables the value *true* and all others the value *false*.)

With the exception of examples 3 and 4, the above problems are known to be NP-complete. Downey and Fellows[DF1-5] considered the question of what can be said about the complexity of these problems when the parameter k is held fixed. In many practical applications of computational problems having this form, efficient algorithms for a small range of parameter values may be quite useful, but NP completeness says *nothing* about the behavior of the fixed parameter version.

Note also that this question now is concerned with the structure of P. For k fixed there is the obvious algorithm running in time $O(n^k)$ which simply searches all possibilities. The question is whether it is possible to uniformly do better.

For each of examples 1–4 above, there is a constant α such that for every fixed parameter value k the problem can be solved in time $O(n^\alpha)$. For example 1, we may take $\alpha = 1$. This means that for each fixed k there is an algorithm A_k that determines whether there is a vertex cover of size k in an input graph G in time $C_k n$ [BG]. For examples 2–4 we may take $\alpha = 3$ by the deep results of Robertson and Seymour [RS1,RS2].

Examples 5 and 6 illustrate the contrasting situation where for fixed values of k we seem to be able to do no better than a brute force examination of all possible solutions. In both cases the best known algorithm is $O(n^k)$ for fixed k. (Actually, Fan Chung has shown that dominating set can be done in time $O(n^{.6k})$.)

The above observations lead Downey and Fellows to introduce reducions and notions of completeness and hardness to explain this fixed parameter (in)tractability.([DF1-5], also [ADF]). We remark that this framework seems to be widely applicable and provides a refined tool to measure the apparent differences in the fixed parameter behavior of many classically equivalent problems. To make the above precise we turn to the definitions of [DF1-5].

A *parameterized problem* is a set $L \subseteq \Sigma^* \times \Sigma^*$ where Σ is a fixed alphabet. In the interests of readability, and with no effect on our theory, we consider in this paper that a parameterized problem L is a subset $L \subseteq \Sigma^* \times N$. Furthermore, in this context we consider N as bring represented as tally sets, that is $N = \{1^n : n = 0, 1, 2...\}$. We simply write n for 1^n in these circumstances. We will tend to use k, i, j for members of N and x, y, z for strings. For $n \in N$ we write $L_k = \{y|(y, k) \in L\}$. We refer to L_x as the x^{th} *slice* of L.

Careful analysis of examples 1–4 in the introduction leads to three flavours of tractability.

Definition. We say that a parameterized problem L is
(1) *nonuniformly fixed-parameter tractable* if there is a constant α and a sequence of algorithms Φ_x such that, for each $x \in N$, Φ_x computes L_x in time $O(n^\alpha)$;
(2) *uniformly fixed-parameter tractable* if there is a constant α and an algorithm Φ such that Φ decides if $(x, k) \in L$ in time $f(k)|x|^\alpha$ where $f : N \to N$ is an arbitrary function;
(3) *strongly uniformly fixed-parameter tractable* if L is uniformly fixed-parameter tractable with the function f recursive.

The reader familiar with classical recursion theory will note that these notions might be considered as analogues of piecewise recursive recursively enumerable sets. The problem in example 1 is strongly uniformly f.p.tractable (as are most examples of fixed-parameter tractability obtained without essential use of the Graph Minor Theorem). Example 2 can be shown to be strongly uniformly f.p. tractable by the methods of [FL2]. The reader should note that the graph minor theorem would only give nonuniform tractability and to get uniformity needs additional algebraic techniques. Example 3 can be shown to be uniformly f.p. tractable by the method of [FL1] (since the technique of [FL2] is not presently known to apply, we do not know a strongly uniform algorithm). Example 4 is at present only known to be nonuniformly f.p. tractable.

If $P = NP$ then examples 5 and 6 are also f.p. tractable. Thus aside from proving $P \neq NP$, a completeness program would seem to be the best we can do with respect to explaining the apparent fixed-parameter intractability of these problems.

Corresponding to the three notions of tractability there are three flavors of problem reducibility.

Definition. Let A, B be parameterized problems. We say that A is *uniformly P-reducible* to B if there is an oracle algorithm Φ, a constant α, and an arbitrary function $f : N \to N$ such that
(a) the running time of $\Phi(B; \langle x, k \rangle)$ is at most $f(k)|x|^\alpha$,
(b) on input $\langle x, k \rangle$, Φ only asks oracle questions of $B^{(f(k))}$ where

$$B^{(f(k))} = \bigcup_{j \leq f(k)} B_j = \{\langle x, j \rangle : j \leq f(k) \& \langle x, j \rangle \in B\}$$

(c) $\Phi(B) = A$.

If A is uniformly P-reducible to B we write $A \leq_T^u B$. Where appropriate we may say that $A \leq_T^u B$ *via* f. If the reduction is many:1 (an *m-reduction*), we will write $A \leq_m^u B$.

Definition. Let A, B be parameterized problems. We say that A is *strongly uniformly P-reducible* to B if $A \leq_T^u B$ via f where f is recursive. We write $A \leq_T^m B$ in this case.

Definition. Let A, B be parameterized problems. We say that A is *nonuniformly P-reducible* to B there is a constant α, a function $f : N \to N$, and a collection of procedures $\{\Phi_k : k \in N\}$ such that $\Phi_k(B^{(f(k))}) = A_k$ for each $k \in N$, and the running time of Φ_k is $f(k)|x|^\alpha$. Here we write $A \leq_T^n B$.

Following [DF1-5], we will henceforth write $FPT(\leq)$ as the f.p. tractable class corresponding to the reducibility $\leq$. These notions provide a platform to prove completeness type results along the lines of NP completeness. For instance, in [DF1-4] a hierarchy is defined based on the complexity of circuits needed to model the combinatorial problem. This hierarchy, the *weft* hierarchy, is of the form

$$FPT \subseteq W[1] \subseteq W[2] \subseteq ... \subseteq W,$$

where W denotes the ω level of the hierarchy. Downey and Fellows show that, for instance weight k satisfiability of boolean formulae in 3CNF form is complete for $W[1]$, for those in general CNF form is complete for $W[2]$. Call CNF product of sum.(PoS) Downey and Fellows also show that for any k, PoSoPoS... with k alternations is complete for $W[k]$ for $k \geq 2$. These authors conjecture that the hierarchy is an infinite hierarchy and provides a good framework for anyaysing the relative fixed parameter tractability of combinatorial problems. For instance, Dominating Set is $W[2]$ complete and Independent set is $W[1]$ complete. Many other things are $W[1]$ hard. Note that as opposed to NP we apparently get a lot of different degrees. A catalogue of known classifications can be found in [DF2,3].

Here we will be more concerned with technical development of these notions of reducibility and tractability. Firstly the notions differ as follows.

Theorem.[DF5] *$FPT(\leq_T^u)$ contains infinitely many incomparible $\leq_T^s$ degrees. Similarly $FPT(\leq_T^n)$ contains infinitely many incomparible $\leq_T^u$ degrees.*

This is proven by a priority argument showing the complexity of the structures. A number of other basic structural aspects of the orderings are addressed in [DF5]. A lot of the normal global structure theory seesmto go through, but local structure seems much more difficult. For instance it is still open if density holds for the structures $(REC, \leq)$, the recursive sets under the reducibility $\leq$, where $\leq$ is not strongly uniform. (See [DF5] for partial results and the strongly uniform case).

In this paper we shall delve much more deeply into the structure of the degree uppersemilattices. We concentrate on the degree structures for the uniform reducibilities making comments where relevant concerning the nonuniform reducibilities. The primary goal is to prove the undecidabiltiy of the structures $(REC, \leq)$, for each of the reducibilities $\leq$. The approach we use is along the lines of that used by Ambos-Spies, Nies and Shore[ANS] and Ambos-Spies and Nies[AN], to prove the undecidabiltiy of, respectively, the recursively enumerable *wtt*-degrees and the P-time m-degrees of recursive sets. This approach is to use an infinite independent set to define with parameters the lattice of Σ_n sets for some $n \geq 1$ in the relevant uppersemilattice using the ideal structure and an exact pair theorem. The result then follows by Herrmann[He]. Loosely speaking we follow this plan but of efforts are hampered by the complexity of the definition of the relevant reducibilities and the lack of general exact pair theorems in these structures. These problems necessitate the use of global embeddings('doing everything at once') rather than the use of local theorems as we shall see. The general technique we use will be Π_2 and Π_3 priority arguments involving the speedup technique.

In the appendix we make some remarks about the possibility of using transfer techniques from the P-time degrees. These results are of independent interest since they are very much concerned with the limits of the standard construction techniques of, for instance, minimal pairs for P.

Notation is standard and follows Soare[So], and Balcazaar et al[BDG].

2. The Weak Exact Pair Theorem.

In this section we shall prove a definability result that is crucial to our investigations. We do not prove a general exact pair theorem (indeed it may be possible to show that there is none) but will prove one for the special sets we work with. Before we state the theorem we need some preliminaries. We will work with the reducibility $\leq_m^u$. Now from [DF5], we know that if A and B are recursive sets with $A \leq_m^u B$ then there is an r.e. function f and a m-procedure Φ and an n such that $A \leq_m^u B$ via f in the following sense:

$$\Phi(B; \langle x, k\rangle) = A(\langle x, k\rangle) \text{ in running time } f(k)|x|^n, \text{ and with use } B^{(f(k))}.$$

Similar characterisations hold for the other reducibilities. With this we get the following result.

(2.1) Lemma. *Let A be a recursive set. Then for $q \in \{m, T\}$,*
(i) $\{W : W \leq_q^s A\}$ is a Σ_3^0 set.
(ii) $\{W : W \leq_q^u A\}$ is a Σ_4^0 set.
(iii)$\{W : W$ recursive and $W \leq_q^n A\}$ is a Π_5^0 set.

Proof. We prove (ii). By the definition, we see that

$$W \leq_q^u A \text{ iff } \exists e \exists f \exists n \forall k \exists s \exists u \forall z (\Phi_e(A; \langle z, k\rangle) = W(\langle z, k\rangle) \text{ in time } u|z|^n \text{ with use } A^{(u)}.$$

This is clearly Σ_4^0. □.

Our plan is to first deal with $\leq_m^u$ (and $\leq_m^s$), and only later deal with the more intricate nonuniform reducibility where we get the undecidability but with weaker definability results. Besides the elements of the proof are mainly present in the uniform case, where they are not obscured by technical considerations as they are in the nonuniform case.

By (2.1) to deal with finitely generated ideals in the structure $(REC, \leq_q^u)$ we need to deal with Σ_4^0 sets. What we do is deal with the following. We call a set B *row finite* if for all k, $|\{z : B(\langle z, k\rangle) \neq 0\}| < \infty$. We say that A is *row bounded* if there is a fixed bound on the number of members of a row. We call this number the norm of the bound. So a row bounded set of norm 1 has at most one member per row.

(2.2) Lemma. *Suppose that* **I** *is an ideal(i.e.* **I** *is closed under join and initial segments) in $(REC, \leq_q^u)$ and* **I** *is generated by $\{A_i : i \in Q\}$ with Q a Σ_4^0 set and $\{A_i : i \in \omega\}$ is a recursive collection of row finite recursive sets. Then there is a collection $C = \{B_i : i \in M\}$ of row finite recursive sets, with M recursive and such that* **I** *equal to the ideal generated by C.*

Proof. Assume that **I** is given as above. Then we know that there is a recursive relation R such that

$$j \in Q \text{ iff } \exists e \forall x \exists s \forall t R(e, j, x, s, t).$$

We suggest that the reader think of the set $\{j : \exists e(\Phi_e(K) = W_j\}$ the collection of complete r.e. sets as a canonical representation for Q. To build B_j we try to copy an A_k while it seems reasonable. If we fail then the set B_j we build will be FPT. (Actually weaker hypotheses suffice for the theorem. For instance if the collection has the property that for any k and A_i, for all j, we have $A_i^{(k)} \leq_q^u A_j$.) For definiteness we fix $q = m$. To do this we need the auxiliary function

$$l(e, j, s) = \max\{x : \forall y \leq x \exists u \leq s \forall t \leq s(R(e, j, y, u, t) \text{ holds })\}$$

Call the least u for y at stage s, $u(y, s)$, and let $U(y, s) = \Sigma_{y' \leq y} u(y' s)$. We shall say that a stage is (e, j)-expansionary if $l(e, j, s) > ml(e, j, s) =_{\text{def}} \max\{l(e, j, t) : t < s\}$. To build B_j

we proceed as follows. At stage s for all z with $|z| \leq s$, if $l(e,j,s) > k$, declare that $\langle z, (p_k)^r \rangle$ into $B_{e,j}$ for all $r \geq k + U(k,s)$ iff $\langle z, k \rangle \in A_j$, where p_k denotes the k-th prime. We argue that if $j \in Q$ then for some e, $B_{e,j} \equiv_m^u A_j$, and and for all e, $B_{e,j} \leq_m^u A_j$; and if $j \notin Q$ then for all e, $B_{e,j} \in FPT$. First suppose that j is in Q. Then there is an e such that for all x, $\exists s \forall t R(e,j,x,s,t)$. As above, this means that for all x, $\forall t R(e,j,x,u(x),t)$ holds where $u(x) = \lim_s u(x,s)$, and similarly $U(x) = \lim_s U(x,s)$. Then for each k we see that for all z,

$$\langle z, k \rangle \in A_j \text{ iff } \langle z, (p_k)^{k+U(k)} \rangle \in B_{e,j}.$$

Hence $A_j \leq_m^u B_{e,j}$. Evidently the construction directly ensures that $B_{e,j} \leq_m^u A_j$ always. Finally if $j \notin Q$, then for each e, if there are infinitely many (e,j)-expansionary stages it can only be that for some least x, $u(x,s)$ fails to have a limit. This means that for all $x' \geq x$, $U(x,s) \to \infty$. It follows that we can for each r compute a stage $s(r)$ such that for all $t > s(r)$, $U(x,t) > r$. Let also v be the stage where for all $y < x$, $u(y,v)$ has reached its limit. W.l.o.g. we may suppose that $s(r) > v$. Then after stage $s(r)$, if row i is not coding row y of A_j for some $y < x$ then this row will be henceforth empty. It therefore follows that $B_{e,j} \in FPT$, giving the result. □

We next turn to the main technical tool as follows.

(2.3) Theorem. *(Weak Exact Pair Theorem.) Let $\{A_i : i \in Q\}$ be a Σ_4^0 collection of row finite recursive sets. Then the ideal* $\mathbf{I}$ *of degrees in $(REC, \leq_q^u)$ generated by this collection has an exact pair. That is, there is a pair of recursive sets C, D such that* $\mathbf{a} \in \mathbf{I}$ iff $\mathbf{a} \leq_m^u \mathbf{c}, \mathbf{d}$, *where* $\mathbf{c}$ *denotes the uniform q-degree of C and* $\mathbf{D}$ *the uniform q-degree of D.*

Proof Before we turn to the proof of this result, we will take a little time out to discuss a central technique in the rest of the paper, the so-called *speedup* technique. This technique has been used by Ambos-Spies, Slaman and others and lies at the heart of many of the deepest embedding and decidability results in the area. It would seem appropriate for us to discuss this in the context of the classical exact pair theorem for the P-time m-degrees (Ambos-Spies[AS*]). Recall that there we had a recursive collection $\{B_i : i \in \omega\}$ of recursive sets and needed an exact pair for this collection. Thus we construct C and D so that $B_i \leq_m^P C, D$ for all i, and so that we meet the requirements below.

$$R_e: \text{ If } \Gamma_e(C) = \Delta(D) \text{ then for some } i,\ \Gamma_e(C) \leq_m^P \oplus_{j \leq i} B_j.$$

Here $\{\Gamma_e, \Delta_e : e \in \omega\}$ denotes the collection of all consisting of two P-time m-reducions with functions γ_e and δ_e respectively. Following an old idea of Clifford Spector, we shall build C and D so that they code B_i into row i. So we ensure that for almost all z, $z \in B_j$ iff $\langle j, z \rangle \in C($ and $)D$. Then clearly, for all i, $B_i \leq_m^P C, D$. By abuse of notation, we shall write this as $B_i =^* C_i$.

We can assume that the B_i (and the A_i) are given as the range of P-time functions with domain ω in unary notation. For $E = B_i$ (or A_i) we write $E_s = \{f(1^0), ..., f(1^s)\}$. We can also ask that if $|f(1^y)| > |f(1^x)|$, then for all $z > x$, $|f(1^z)| > |f(1^x)|$. We call this a *P-standard enumeration.* So we assume that we have such enumerations af all relevant sets. For a parameter p, we will have a stage $m(p)$ such that for all z if $|z| \leq p$ then for all $i \leq p$, $A_i(z) = A_{i,m(p)}(z)$ and $B_i(z) = B_{i,m(p)}(z)$. Moreover when $m(p) \downarrow$ it does so at a stage where it is P-time to compute it.

The problems all stem from meeting the R_e in this environment. For a single R_e alone we do this as follows. We give this requirement priority $2e+1$. We give the coding requirement asking that $B_i =^* C_i$ priority $2i$. The basic idea for a single R_e is the following. At stage s we will begin the following cycle. R_e will *assert control* of the construction till stage $p'(s)$ described below. We will be given a parameter $p(s)$ Let q_e be a polynomial bound on the time for the computations of Δ and Γ. We wait till stage $p'(s) = m(2^{m(q_e(m(p(s))))})$ and see if we can legally force a disagreement. That is we see if there are sets $C(2e+1, s)$, $D(2e+1, s)$ such that
(i) $\Gamma_e(C(2e+1, s); z) \neq \Delta_e(D(2e+1, s); z)$ with $|z| \leq p(s)$,
(ii) $C(2e+1, s)(\langle x, j\rangle) = C_s(\langle x, j\rangle)$ if $|x| \leq q_e(m(p(s))$, $C(2e+1, s)(\langle x, j\rangle) = B_j(x)$ if $j \leq e$, and
(iii) as for (ii) with D in place of C.

If such sets exist, then R_e will and ask that *with priority 2e+1*, $C_{s+1} = C(2e+1, s)$ and $D_{s+1} = D(2e+1, s)$ thus forcing a disagreement between $\Gamma_e(C)$ and $\Delta_e(D)$. This requirement will of course preserve the relevant use of the computation, with priority $2e+1$. It then releases control of C and D at stage $p'(s)$ and outputs this as the next parameter. In the actual construction, we consider the requirements in reverse order. So after Q_0 has been considered, we begin to consider the requirements $Q_s, ..., Q_0$, one at a time. Here Q_{2e} is a requirement trying to ensure that $A_e =^* C_e$, D_e and Q_{2e+1} is R_e. The Q_{2e} will simply ask that A_e will be copied into C_e and D_e. So it will assert control till from stage $p(s)$ till stage $p'(s) = m(p(s))$. The R_e as above will take the given parameter $p(s)$ and the stage number $p(s)$ and assert control until a stage $p'(s)$ as above. If it sees a way of forcing a disagreement, it will, then releasing control only of C and D up to the relevant uses. Note that in this case it can injure the higher coding action we amy have initiated but this is okay as we now think we are satisfied and will be unless a higher priority R_e action needs to be attended. In the other case, R_e simply releases all control at stage $p'(s)$ passing on to Q_j for $j < 2e+1$. So there will be a sequence of stages where, in turn various requirements have their say with C and D each in turn generating their incarnation of C and D. When we get to Q_0, we know the highest priority R_e that acts since the last time we were here, and its version will be the correct one. (If any.) In this case we declare that R_e is *satisfied*. Since it is clear that each Q_e can be injured at most finitely often, we see that they all are met. This is clear for the Q_{2e}, and to see it for the R_e, suppose once this requirement has priority (i.e. will not again be injured by any higher priority R_j), then any action we take will be permanent. Suppose this happens at stage s_0. Hence if we suppose that R_e never is declared satisfied, then for all choices of C and D provided they agree with A_j for $j \leq e$ and $C[s_0]$, and $D[s_0]$, (as they will by choice of s_0), they must give the same answer. This means that $\Gamma_e(C) \leq_m^u \oplus_{j \leq e} B_j$. The point here is that R_e asserts control at stage $p(s)$ and does so till stage $p'(s)$. Now as it has priority, the only R_j that can change things are of lower priority and these can only change C and D on things below $p(s)$ and R_e has control till these have become linear time. Thus in linear time with oracle $\oplus_{j \leq e} B_j$ we can figure out $\Gamma_e(C)$ simply by usine the empty extension on rows above e.

Turning now to the construction at hand, it turns out to be not too dificult to modify the ideas above to our setting. Now for the $\{A_i : i \in \omega\}$, which can be taken to be a recursive collection by the previous lemma, we now meet the requirements:

$R_{e,n}$: Either $\lim_s \phi_e(k,s) =_{\text{def}} \phi_e(k)$ fails to exist for some k
or $\exists x, k(\Gamma_e(C; \langle x, k\rangle)$ or $\Delta_e(D; \langle x, k\rangle)$ does not run in time $\phi_e(k)|x|^n)$,
or the use exceeds $C^{(\phi_e(k))}$ or $D^{(\phi_e(k)))}$.
or $\Gamma_e(C) \neq \Delta(D)$,
or for some i, $\Gamma_e(C) \leq_q^u \oplus_{j \leq i} A_j$.

This time we shall code A_i into $C_{i,j}$ and $D_{i,j}$ for $j \in \omega$. That is we ensure that we meet

$$P_i : A_{i,j} =^* C_{i,j} =^* D_{i,j} \text{ for all } j.$$

Then the construction runs as follows. We can split the P_i into $P_{i,j}$ trying to achieve $A_{i,j} =^* C_{i,j} =^* D_{i,j}$. This is achieved by direct coding, as above, subject to finite injury. We will have $R_{e,n}$ respect $P_{i,j}$ for e.g. $i, j < \langle e, n\rangle$. Now in the construction $R_{e,n}$ will have a parameter $p(s)$ as above together with a parameter $v(e,n,s)$ which will either be defined or currently undefined. It represents our current guess as to how many rows the reductions Γ_e and Δ_e seem to be working on. Also associated will be parameters $t(e,n,j,s)$ for $j \leq v(e,n,s)$. These represent the current guess as to the value of $\phi_e(j)$. When $R_{e,n}$ asserts control, it first calculates a new $v(e,n,s)$. to do this, it finds the largest v such that
(i) for all $g \leq v$, $\phi_{e,s}(g,t) \downarrow$ for some $t < s$ with $t'(e,n,g,s)$ the largest such t.
(ii) if $g \leq v$ and $g \leq v(e,n,s)$ then $\phi_{e,s}(g, t(e,n,g,s)) = \phi_{e,s}(g, t'(e,n,g,s))$.

We let $v(e,n,s+1) = v$. While $R_{e,n}$ is in control we will only allow this value to stay the same. We let $t(e,n,s+1) = t'(e,n,s)$. Now $R_{e,n}$ asserts control till stage $p'(s) = m(2^{m(\max_{g \leq v(e,n,s+1)} \phi_{e,s}(g,t(e,n,g,s+1))m(p(s)^n))})$. We again see if we can legally force a disagreement. Are there sets $C' = C(2\langle e,n\rangle + 1, s)$, $D' = D(2\langle e,n\rangle, s)$ such that
(i) $\Gamma_e(C'; \langle x, g\rangle) \neq (D'; \langle x, g\rangle)$ for some $x \leq p(s)$ and $g \leq v(e,n,s+1)$, the computations running in time $\phi_e(g, t(e,n,g,s+1))|x|^n$ and only using numbers from the first $\phi_e(g, t(e,n,g,s))$ rows of C and D.
(ii)$E'(\langle x,k,j\rangle) = E(\langle x,k,j\rangle)$ for $E = C$ or D for $|x| \leq p(s)$,
or if $|x| \leq \max_{g \leq v(e,n,s+1)} \phi_e(e,s)(g, t(e,n,g,s))m(p(s))^n$ and $k, j \leq \langle e, n\rangle$.

If at stage $p'(s)$ we see such configurations then as before $R_{e,n}$ will request them as the real versions of D and C. Otherwise it releases as before. Note that if we do not declare that $R_{e,n}$ is satisfied as above, then if $\liminf_s v(e,n,s) = \infty$, then all the $\phi_e(g,s)$ reach a limit. Now the argument goes through as before. $\square$

(2.4) Corollary. *Let* $\mathbf{I}$ *be an ideal of* $(REC, \leq_q^r)$ *with* $q \in \{m, T\}$ *and* $r \in \{s, u\}$*. Suppose that* $\mathbf{I}$ *is generated by as set of degrees of row finite sets. Then* $\mathbf{I}$ *has an exact pair iff* $\mathbf{I}$ *can be generated by a* Σ_n^0 *set of degrees of row finite sets where* $n = 3$ *if* $r = s$ *and* $n = 4$ *otherwise, iff* $\mathbf{I}$ *can be generated by a recursive collection of degrees of row finite sets.*

3. Undecidability.

In this section we use the definability results of the last section to get the undecidability if the degree structures. We will do the easier m-degree case first, indicating the modifications needed for the T-degree case later. So fix the underlying structure as $(REC, \leq_m^u)$. It is not difficult to see that as usual this is a distributive uppersemilattice(usl). Following [AN], we call $\mathbf{a}$ a *nontop* if it is not the top of a minimal pair. That is for all $\mathbf{c}$, $\mathbf{d}$, if $\mathbf{c} \vee \mathbf{d} = \mathbf{a}$ and $\mathbf{c} \wedge \mathbf{d} = \mathbf{0}$, then either $\mathbf{c} = \mathbf{a}$, or $\mathbf{d} = \mathbf{a}$. (Recall that $\mathbf{c}$ and $\mathbf{d}$ form a minimal pair if $\mathbf{c} \wedge \mathbf{d} = \mathbf{0}$.)

We need the following machinery from [AN], [ANS]. Let $\mathbf{P}$ be an infinite usl with $\mathbf{0}$. A subset A of $\mathbf{P}$ is called *independent* if for all $a \in A$ for all $a_1, ..., a_n \in A - \{a\}$, we have that $a \not\leq a_1 \vee ... \vee a_n$. If $A \subset \mathbf{P}$ let $\langle A \rangle$ denote the ideal generated by A. Let $I(A) = \{\langle B \rangle : B \subseteq A\}$. The following is a slight generalization of [AN, lemma(2.2)].

(3.1) Lemma. *Let A be an infinite subset of P, such that for all $a,\ b \in A$, if $a \neq b$ then a and b form a minimal pair. Suppose further that $\langle A \rangle$ is locally distributive. That is, for all $b, a_1, ...a_n$, if $b \leq A_1 \vee ... \vee a_n$, then there exist $b_1, ..., b_n$ with $b_i \leq a_i$ for $i = 1, ..., n$ and $b = b_1 \vee ... \vee b_n$. Then for all distinct elements $a, a_1, ...a_n$ of A we have that $a \wedge (a_1 \vee ... \vee a_n) = \mathbf{0}$. Furthermore*

$$I(A) = \{I \in I(\mathbf{P}) : I \subseteq \langle A \rangle \wedge \forall x \in I \forall a \in A(x \neq \mathbf{0} \wedge x \leq a \rightarrow a \in I\}$$

The proof is the same. This gives the following slight restatement of [AN, Theorem (2.4)].

(3.2) Theorem. *Let* $\mathbf{P}$ *be an infinite usl and* $A \subseteq \mathbf{P}$*. Suppose that*
(i) $\langle A \rangle$ *is locally distributive,*
(ii) A is infinite,
(ii) $\forall a, a_1, ..., a_n \in A(a \wedge (a_1 \vee ... \vee a_n) = \mathbf{0})$*,*
(iii) $\forall a \in A$*(a is a nontop), and*
(iv) $\langle A \rangle$ *possesses an exact pair.*
Then

$$DI(A) =_{\text{def}} \{I \in I(A) : I \text{ has an exact pair } \}$$

is elementarily definable with parameters(e.d.p) in $(\mathbf{P}, \leq)$.

Note that the theorem applies to $(REC, \leq_m^r)$ as this is a distributive usl and hence certainly locally distributive. Since the lattice of Σ_4^0 sets under inclusion is heriditarily undecidable, by corollary (2.4) to establish the undecidability of $(REC, \leq_m^u)$ it suffices to construct a recursive sequence of recursive row finite sets $\{A_i : i \in \omega\}$ with degrees $\mathbf{A} = \{\mathbf{a_i} : i \in \omega\}$ with the properties below:

(3.3) $\forall \mathbf{a}, \mathbf{a_1}, ..., \mathbf{a_n} \in \mathbf{A}(\mathbf{a} \wedge (\mathbf{a_1} \vee ... \vee \mathbf{a_n} = \mathbf{0})$.

(3.4) $\forall \mathbf{a} \in \mathbf{A}$($\mathbf{a}$ is a nontop).

To achieve (3.3) and (3.4) we build the recursive sequence to meet the requirements below:

$P_{e,i,j,m}$: $A_i \neq \Psi_e(A_j)$, with time bound m,
that is, either $\lim_s \psi_e(k, s) =_{\text{def}} \psi_e(k)$ fails to exist for some k
or $\exists x, k(\Psi_e(A_j; \langle x, k\rangle)$ does not run in time $\psi_e(k)|x|^m)$,
or the use exceeds $A_j^{(\psi_e(k))}$
or $\exists x, k(\Psi_e(A_j; \langle x, k\rangle) \neq A_i(\langle x, k\rangle)$,

$R_{e,i,n}$: Either $\lim_s \phi_e(k, s) =_{\text{def}} \phi_e(k)$ fails to exist for some k
or $\exists x, k(\Phi_e(A_i; \langle x, k\rangle)$ or $\Delta_e(A_i; \langle x, k\rangle)$
or $\Gamma_e(\Phi_e(A_i) \oplus \Delta_e(A_i))(\langle x, k\rangle)$ does not run in time $\phi_e(k)|x|^n)$,
or the use exceeds $A_i^{(\phi_e(k))}$ (or $(\Phi_e(A_i) \oplus \Delta_e(A_i))^{(\phi_e(k)))}$,
or $\exists x, k(\Gamma_e(\Phi_e(A_i) \oplus \Delta(A_i); \langle x, k\rangle) \neq A_i(\langle x, k\rangle)$,
or $\Phi_e(A_i) \in FPT$ or $\Delta_e(A_i) \in FPT$,
or $Q_{e,i} \leq_m^u \Phi_e(A_i), \Delta_e(A_i)$ and $\forall j, m(R'_{e,i,n,j,m})$.

where

$R'_{e,i,n,j,m}$: $Q_{e,i} \neq \Psi_e(\emptyset)$ in time bound m.

$N_{e,i,j,n}$: Either $\lim_s \phi_e(k, s) =_{\text{def}} \phi_e(k)$ fails to exist for some k
or $\exists x, k(\Lambda_e(A_i; \langle x, k\rangle)$ or $\Theta_e(\oplus_{v \in D_j} A_v; \langle x, k\rangle)$ does not run in time $\phi_e(k)|x|^n)$,
or the use exceeds $A_i^{(\phi_e(k))}$ or $\oplus_{v \in D_j} A_v^{(\phi_e(k))}$,
or $\exists x, k(\Lambda_e(A_i; \langle x, k\rangle) \neq \Theta_e(\oplus_{v \in D_j} A_v; \langle x, k\rangle)$,
or $\Lambda_e(A_i) \in FPT$.

Here D_j denotes the j-th canonical finite set, $\{\Psi_e : e \in \omega\}$ is a listing of all machines, $\{\Gamma_e, \Delta_e, \Gamma_e : e \in \omega\}$ is a listing of all triples, and $\{\Lambda_e, \Theta_e : e \in \omega\}$ is a listing of all pairs.

In the above we will adopt the convention that for any use/constant ϕ_e (or ψ_e), if $\phi_{e,s}(k,t)\downarrow$ (i.e. in $\leq s$ steps) and $\phi_{e,u}(k,v)\downarrow$ for $s\leq u$ and $t\leq v$, but $\phi_e(k,t)\neq\phi_e(k,v)$ then $\phi_e(k,t)<\phi_e(k,v)$. Uses are similarly monotone in the first variable. Furthermore if $\phi_{e,s}(k,t)\downarrow$ then $\phi_{e,s}(k',t)\downarrow$ for all $k'\leq k$.

The easiest requirements are the ones of the Friedberg type, namely the $P_q=P_{e,i,j,m}$. In the construction we will perform certain *cycles*. At the beginning of a cycle we will be aware of a highest priority *pending* action. The purpose of a cycle will be to determine if we need to replace this by another action, and finally in the last of the subcycles of the main cycle to determine what to do. (This is similar to (2.3) but more involved.) At the top of a (sub)cycle we first consider a unique P_q. This will have a unique row g set aside for its satisfaction. At stage s when P_q asserts control, it first finds the largest $t=t(s)$ if any such that $\psi_{e,s}(g,t)\downarrow$.

If no such t exists then P_q releases control immediately. If t exists then P_q controls until stage $s_1=\psi_e(k,t)s^m$. While P_q asserts control it asks that $A_{j,s}$ be extended by the empty extension. Now at stage s_1, P_q sees if $\Psi_{e,s_1}(A_{j,s_1};\langle 1^s,g\rangle)\downarrow$ and uses only $\psi_e(k,t)$ rows in the use. If not then P_e releases control of both A_j and A_i and the pending requirement remains the same. If the answer is yes then P_q becomes the pending requirement and requests us to set

$$A_i(\langle 1^s,g\rangle)=1-\Psi_e(A_{j,s_1};\langle 1^s,g\rangle),$$

and

$$A_j(z)=A_{j,s_1}(z) \text{ for all } z \text{ with } |z|\leq s_1$$

The next most difficult requirements are the minimal pair ones, but to understand their full action we really would really need to discuss the R type requirements first. Instead we will first discuss the basic module and then only later discuss the modifications needed in the full construction. So for the time being suppose that $N_{e,i,j,n}=N_d$ needed only to live with the P_q. Again we use a speedup argument. The construction is similar to that used in (2.3). This time, we need parameters as follows: we need a stage $p(s)$ where N_d first asserts control, a 'reference point' $r(s)$ indicating the portion of the sets A_i that have been finalized, (so that $A_i(z)$ for $|z|>r(s)$ is still to be decided), and a parameter $v(e,i,j,n,s+1)=v$ indicating the number of rows the pair Θ and Λ jointly compute. When N_d asserts control, it first computes a new value of v. To do this it finds the largest v such that

(i) for all $g\leq v$, $\phi_{e,s}(g,t)\downarrow$ for some $t<s$, with $t=t(e,i,j,n,s+1)$ the largest such t,
(ii)if $g\leq v$ and $g\leq v(e,i,j,n,s)$ then $\phi_e(g,t)=\phi_e(g,t(e,i,j,n,s)$.

Now as with (2.4) we let N_d asserts control until an appropriately huge stage where all the computations below $\langle p(s),p(s)\rangle$ have had a chance. So it asserts control until stage $p'(s)=2^{(\Sigma_{g\leq v}(\phi_e(g,t)m(e,s)^n))}$, where $m(e,s)$ denotes the maximum of the lengths of the uses $\delta_e(A_i;\langle y,k'\rangle),\theta_e(\oplus_{c\in D_j}A_c;\langle 1^y,k'\rangle$ for $y\leq s$ and $k'\leq v$. At this stage N_d will relinquish control, but will first see if there are possible configurations of the A_u that can be used to get a disagreement. Thus we see if there are incarnations of $A_{u,s+1}$, denoted by $A(u, s+1)$ with the following properties.

(i) For all z with $|z| \leq r(s)$ we have that $A(u, s+1) = A_u(z)(= A_{u,s}(z)$. that is this has been settled by now and future configurations must agree.
(ii) For all x and k with $k \leq d$, if $|x| \geq r(s)$, then $A(u, s)(x) = 0$.
(iii) A_u is *legal*. That is, if $A(u, s)(z) \neq 0$ then z is a follower. The point is that only on followers can A_u not be $\emptyset$.
(iii) For some x with $|x| \leq p(s)$ and $g \leq v(e, i, j, n, s)$ we have $\Lambda_e(A(i, s); \langle x, g\rangle) \neq \Theta_e(\oplus_{z \in D_j} A(z, s); \langle x, g\rangle)$, the computations running in time $\phi_e(g, t(e, i, j, n, s))|x|^n$ and only using numbers from the first $\phi_e(g, t(e, i, j, n, s)$ rows of the relevant $A(z, s)$ for $z \in D_j$.

(The reader should compare with (2.3)(i) and (ii) where C' and D' are defined.) If at stage $p'(s)$ we see such incarnations of the A_z then N_d will request that these are the real versions of the A_z. It will ask that it becomes the pending requirement of highest priority. Otherwise it releases as before. Note that only $p(s+1) = p'(s)$ is changed. The $r(s)$ remains the same.It will do so until we get to the end of all the subcycles.

Note that the above succeeds. For once the N_d has the priority, it will be the case that nothing will later be added to the first d rows. This means that the conditions (i) and (ii) will automatically be satisfied for all future configurations of the A_z. If a disagreement is spotted we will take it. This implies that if the action of this requirement is infinite then all future configurations must give the same answer. Then to compute the value of $\Lambda(A_i; \langle x, k\rangle)$ for $|x|$ sufficientlty large, we simply use the empty extension. This will work once the value of $phi_e(k, t)$ reaches its limit.

Finally we turn to the $R_w = R_{e,i,n}$. These are met using ideas along the lines of [Do], the simpler methods of Ambos-Spies et al[AHS] apparently not sufficing for our setting. This is where we use that we are dealing with m-reductions. Let f be the function witnessing Φ. That is Φ means that $x \in \Phi(A_i)$ iff $f(x) \in A_i$. Similarly let g represent Δ.

The underlying idea from [SS] and [Do], to recall a phrase of Ted Slaman, we call *bootstrapping*, and seems to be fundamental in all Δ_n proiority arguments for $n > 1$. We first drop all the parameterization approximation machinery, and consider only the general shape of the construction. The reader should be warned that the approximation machinery is what makes this a Π_4 aggument. We focus on the satisfaction of a single $R'_{d,m}$ type requirement. At any stage below the use corresponding to the triple $\langle \Gamma, \Phi, \Delta\rangle$ there will be at least two active 'followers' (actually configurations as we see) x and y which can potentially enter $A = A_i$. As with all such speedup type arguments the length of y will be very large and will in particulat exceed $2^\gamma(\rho(|x|))$ where $\rho(x)$ is the maximum of the uses $\phi(x)$ and $\delta(x)$. Note here we use lower case letters to denote the use of the corresponding functional. For simplicity suppose we are only considering the attack on $R'_{d,m}$ at row g, say. The attack may occur at cofinitely many rows but more on that later. For this fixed row g, we are trying to achieve hopefully $Q = Q_{e,i} \leq_m^u \Phi_e(A), \Delta_e(A)$ and for some p, $Q(p) \neq \Psi(\emptyset)(p)$.

References

[ADF] K. Abrahamson, R. Downey, and M. Fellows, "Fixed Parameter Tractability and Completeness IV: $W[P]$ and $PSPACE$," to appear.

[AS1] K. Ambos-Spies, "On the structure of polynomial time degrees" in STACS 84, Lecture Notes in Computer Science, vol 166, (1984) 198-208, Springer-Verlag.

[AS2] K. Ambos-Spies, "On the structure of the polynomial time degrees of recursive sets," Technical report 206 (1885), Abteilung Informatik, Universität Dortmund.

[AS3] K. Ambos-Spies, "Minimal pairs for polynomial time reducibilities," in *Computation and Proof Theory*, Lecture Notes in Computer Science, vol 270, (1987) 1-13

[AN] K. Ambos-Spies and A. Nies, "The Theory of The Polynomial Time Many-One Degrees is Undecidable,"to appear.

[ANS] K. Ambos-Spies, A. Nies, and R.A. Shore, "The Theory of the Recursively Enumerable Weak Truth Table Degrees is Undecidable,"to appear.

[AST] K. Aoki, J. Shinoda, and T. Tsuda, "On Π_2 Theories of hp-T Degrees of Low Sets," to appear.

[BDG] J. Balcazaaar, J. Diaz, and J. Gabarro," Structural Complexity" Volumes 1 and 2 Springer Verlag (1987,1989).

[BG] J. F. Buss and J. Goldsmith, "Nondeterminism Within P," to appear, SIAM J of Comput.

[Bo] H. L. Bodlaender, "On Disjoint Cycles," Technical Report RUU-CS-90-29, Dept. of Computer Science, Utrecht University, Utrecht, The Netherlands, August 1990.

[Do] R.Downey, "Nondiamond Theorems for Polymomial Time Reducibility," to appear, J.C.S.S.

[DF1] R. Downey and M.Fellows, "Fixed Parameter Tractability and Completeness, "to appear, Congressus Numerantium.

[DF2] R. Downey and M. Fellows, "Fixed Parameter Tractability and Completeness I: Basic Results," to appear.

[DF3] R. Downey and M. Fellows, "Fixed Parameter Tractability and Completeness II: On Completeness for $W[1]$," to appear.

[DF4] R.Downey and M.Fellows, "Fixed Parameter Intractability,"to appear Structures 1992.

[DF5] R. Downey and M. Fellows, "Fixed Parameter intractability II, " to appear.

[FL1] M. R. Fellows and M. A. Langston, "On Search, Decision and the Efficiency of Polynomial-Time Algorithms." In *Proc. Symp. on Theory of Computing (STOC)* (1989), 501-512.

[FL2] M. R. Fellows and M. A. Langston, "An Analogue of the Myhill-Nerode Theorem and Its Use in Computing Finite Basis Characterizations." In *Proc. Symp. Foundations of Comp. Sci. (FOCS)* (1989), 520-525.

[GJ] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (Freeman, San Francisco, 1979).

[He] E. Herrmann,"The undecidability of the elementary theory of the lattice of recursively enumerable sets (abstract),"in *Proceedings of the Second Frege Conference, Schwerin, DDR, 1984* (G. Wechsung, ed) Akademie-Verlag, Berlin, 1984, 66-72.

[La] R. Ladner, "On the Structure of Polynomial Time Reducibility," J.A.C.M. **22**(1975) 155-171.

[Me] K. Melhorn, "Polynomial and Abstract Subrecursive Classes," J.C.S.S. **12**(1976) 147-178.

[RS1] N. Robertson and P. D. Seymour, "Graph Minors XIII. The Disjoint Paths Problem," to appear.

[RS2] N. Robertson and P. D. Seymour, "Graph Minors XV. Wagner's Conjecture," to appear.

[Sh] A. Shamir, "$IP = PSPACE$,"FOCS **31**(1991) 145-152.

[Sho] J. Shinoda, Personal Communication, 1991.

[SS] J. Shinoda and T. Slaman, "On the Theory of $PTIME$ Degrees of Recursive Sets, J.C.S.S. **41** (1990) 321-366.

[ShS] R. Shore and T. Slaman, "The P-T-Degrees of Recursive Sets; Lattice Embeddings, Extensions of Embeddings, and the Two Quantifier Theory," in *Proc. Structures in Complexity* 1989.

[St] L. Stockmeyer, "Planar 3-Colourability is NP-Complete," SIGACT News,**5**(1973) 19-25.

[VV] L.G. Valiant and V.V. Vazirani, "NP is as easy as detecting unique solutions," *Theoretical*

Computer Science 47 (1986), 85-93.

Appendix