Fixed-Parameter Tractability and Completeness III: Some Structural Aspects of the W Hierarchy

Rod Downey*
Mathematics Department
Victoria University
Wellington, New Zealand

Michael Fellows†
Computer Science Department
University of Victoria
Victoria, B.C. Canada

July 7, 2010

Abstract

We analyse basic structural aspects of the reducibilities we use to describe fixed parameter tractability and intractability, in the model we introduced in earlier papers in this series. Results include separation and density, the latter for the strongest reducibility.

1. Introduction

*Downey's research partially supported by a grant from Victoria University IGC, by the United States / New Zealand Cooperative Science Foundation, and by the University of Victoria and Simon Fraser University during a visit to British Columbia in August, 1991.

†Fellows research supported by the United States Office of Naval Research, by the United States National Science Foundation, and by the National Science and Engineering Research Council of Canada.

A wide variety of natural computational problems have the property that their input consists of two 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. We consider 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.

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 $O(n^{k+1})$ for fixed k.

We are thus concerned with an issue in computational complexity that is very much akin to to the central issue in *P versus NP*. In the previous papers of this series [DF1,DF2,DF3] we have established the framework of a completeness theory with which to address the apparent fixed-parameter intractability of problems such as examples 5 and 6. In particular, we defined a hierarchy of classes of parameterized problems and showed that a variety of natural problems are complete for various levels of this hierarchy. Dominating Set, for example, is complete at the second level.

In this paper, we study structural aspects of the fixed-parameter complexity hierarchy. For the remainder of this section we briefly recap the main points of this theory.

Definition. 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 problem examples 1–4 above 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. Most reasonable variations of the above definitions can be seen to coincide with one of the three flavors offered. For example, if in (1) we require the sequence of algorithms Φ_x to be recursive, then equivalently we have (2). In Section 2 we will show that the three forms of fixed-parameter tractability defined above are distinct, even on the the recursive sets.

Problem 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 tecchniques. 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.

We define three flavors of problem reducibility corresponding to the three flavors of f.p. tractability.

Definition. Let A, B be parameterized problems. We say that A is *uni-*

formly 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

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

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

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

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

Note that the above are good definitions since whenever $A < B$ with $<$ any of the reducibilities, if B is f.p. tractable so too is A. Note also that the above definitions allow us to specify the notions of f.p. tractability we had before. Now nonuniformly f.p. tractabilility corresponds to being $\le_T^n \emptyset$. We will henceforth write $FPT(\le)$ as the f.p. tractable class corresponding to the reducibility $\le$. We next turn to the complexity classes of parameterized problems introduced in [DF1,DF2]. These classes correspond, in a finely resolved way, to the complexity of checking a solution, as measured by circuit depth.

Fix attention on any of the above reducibilities. We consider circuits in which some gates have bounded fan-in and some have unrestricted fan-in. It is assumed that fan-out is never restricted.

Definition. A Boolean circuit is of *mixed type* if it consists of circuits having gates of the following kinds.

(1) *Small gates*: *not* gates, *and* gates and *or* gates with bounded fan-in. We will usually assume that the bound on fan-in is 2 for *and* gates and *or* gates, and 1 for *not* gates.

(3) *Large gates*: *And* gates and *Or* gates with unrestricted fan-in.

We will use lower case to denote small gates (*or* gates and *and* gates), and upper case to denote large gates (*Or* gates and *And* gates).

Definition. The *depth* of a circuit C is defined to be the maximum number of gates (small or large), not counting *not* gates, on an input-output path in C. The *weft* of a circuit C is the maximum number of large gates on an input-output path in C.

Definition. We say that a family of circuits F has *bounded depth* if there is a constant h such that every circuit in the family F has depth at most h. We say that F has *bounded weft* if there is constant t such that every circuit in the family F has weft at most t. F is a *decision circuit family* if each circuit has a single output. A decision circuit C *accepts* an input vector x if the single output gate has value 1 on input x. The *weight* of a boolean vector x is the number of 1's in the vector.

Definition. Let F be a family of boolean circuits. We allow that F may have many different circuits with a given number of inputs. To F we associate the parameterized circuit problem $L_F = \{(C, k) : C$ accepts an input vector of weight $k\}$.

Definition. A parameterized problem L belongs to $W[t]$ if L reduces to the parameterized circuit problem $L_{F(t,h)}$ for the family $F(t, h)$ of mixed type decision circuits of weft at most t, and depth at most h, for some constant h.

Definition. A parameterized problem L belongs to $W[P]$ if L reduces to the circuit problem L_F where F is the set of all circuits (no restrictions).

Definition. We designate the class of fixed-parameter tractable problems FPT.

In the papers [DF1,DF2,DF3,ADF] we have identified many natural complete problems for these classes. We mention the following variant of Satisfiability.

Definition. A boolean expression X is termed *t-normalized* if:
(1) $t = 2$ and X is in product-of-sums (P-o-S) form,
(2) $t = 3$ and X is in product-of-sums-of-products (P-o-S-o-P) form,
(3) $t = 4$ and X is in P-o-S-o-P-o-S form,
... etc.

WEIGHTED t-NORMALIZED SATISFIABILITY
Input: A t-normalized boolean expression X and a positive integer k.
Question: Does X have a satisfying truth assignment of weight k?

Our analogue of Cook's Theorem is the following.

Theorem. *For any fixed reducibility,*
(1) (Downey-Fellows [DF1,2]) For $t \geq 2$*, Weighted t-Normalized Satisfiability is complete for* $W[t]$*.*
(2) (Downey-Fellows [DF3]) Weighted Satisfiability for 2CNF formulas is complete for $W[1]$*.*

The above leads to an interesting hierarchy

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

Note that if $P = NP$ then the hierarchy collapses. We conjecture that each of the containments is proper. Many natural problems are complete for various levels. For example, Independent Set is complete for $W[1]$ and Dominating Set is complete for $W[2]$.

In the present paper we explore some structural aspects of the W hierarchy. In Section 2 we present some basic preliminary results and review the needed background material in recursion theory and computational complexity. In Section 3 we turn to results concerned with embeddings into the relevant degree structures. In Section 4 we prove some density results akin to the well-known Ladner theorem for NP-completeness. In Section 5 we present some related results on relativizations.

2. The Basics

We shall need a little basic recursion theory (see Soare[So]). The reader should recall that $K_0 = \{\langle x, y\rangle : \phi_x(y)\downarrow\}$ encodes the halting problem. (Here ϕ_x denotes the x-th partial recursive function.) There is a natural notion of reducibility $\le_T$, called Turing reducibility, between languages and the equivalence classes are called *degrees (of unsolvability)*. The most complex (with respect Turing reducibility) recursively enumerable (r.e.) degree is the degree of K_0 above and this degree is denoted by $\mathbf{0}'$. We shall need the following result.

(2.1) Lemma*(Shoenfield limit lemma)* *$B \le_T A$ iff there is a recursive function $f(\ ,\ )$ such that, for all x,*
(i) $\lim_s f(x,s) = f(x)$ exists,(i.e. $f(x,s) \neq f(x,s+1)$ only finitely often) and
(ii) $f(x) = B(x)$.

Here we identify sets with their characteristic functions. Similarly it follows that a function g is recursive in $\mathbf{0}'$ (and we say g is Δ^0_2) iff $g(x) = lim_s G(x,s)$ for a recursive G with $G(x,s) \neq G(x,s+1)$ only finitely often. We shall additionally say that G is an r.e. function if $G(x,s+1) > G(x,s)$ whenever $G(x,s+1) \neq G(x,s)$. Such highly undecidable sets and functions are relevant to our studies as can be seen from the following theorem.

(2.2) Theorem (i) Suppose that $A \le^u_T B$(or $A \le^u_m B$) with A and B recursive. Then there exists an r.e. function f such that $A \le^u_T B$ (resp. $A \le^u_m B$) via f.
(ii) Suppose that $A \le^n_T B$ (or $A \le^n_m B$ with A and B recursive. Then there exists an r.e. function f such that $A \le^n_T B$ (resp. $A \le^n_m B$) via f.

Proof We do (i) for $\le^u_T$, the others being esentially similar. So suppose that A and B are recursive and $A \le^u_T B$. Then there is a procedure Φ, a constant α and a function g so that for all k
(2.3) $(\forall z)((\langle z,k\rangle \in A$ iff $\Phi(B^{(g(k))}; \langle z,k\rangle) = 1$ and runs in time $\le g(k)|z|^\alpha)$.

We claim that $\mathbf{0}'$ can compute a value that works in place of $g(k)$ in the above. That is for each k, $\mathbf{0}'$ can compute $m = m(k)$ satisfying (2.3) with m in place of g. Call this $(2.3)'$. The reason is that the expression in the scope of the universal quantifier is recursive and hence the whole expression is $\leq_T K_0$. (For the reader who has forgotten this sort of thing, we briefly remind them that for each pair $\langle n,k\rangle$ we can enumerate a partial recursive function $\psi_{\langle n,k\rangle} = \phi_{h(n,k)}$ whose index $h(n,k)$ is given by the s-m-n theorem with $dom\psi_{\langle n,k\rangle}$ equal to N if there is some z with $\langle z,k\rangle \notin A$ but $\Phi(B^{(n)};\langle z,k\rangle) = 1$, or $\langle z,k\rangle \in A$ and $\Phi(B^{(n)};\langle z,k\rangle) = 0$, or $\Phi(B^{(n)};\langle z,k\rangle)$ not running in time $n|z|^\alpha$; and we have $\psi_{\langle n,k\rangle}$ the empty function otherwise. Now K_0 can decide if $\langle h(n,k),h(n,k)\rangle \in K_0$ and hence can compute the least n such that $dom\psi_{\langle n,k\rangle} = \emptyset$. For such an n we have that

$A_k = \Phi(B^{(n)})$ in running time $n|z|^\alpha$.

Now it is clear that we can such an $n(k)$ via a function where values only increase and hence we can take an m to perform the role of g that is r.e.. $\Box$.

As we mentioned earlier, one possible variation for the definition would be to consider a recursive collection $\Phi_{g(k)}$ of reductions with $\Phi_{g(k)}(B^{g(k)}) = A_k$ in running time $O(|z|^\alpha)$. This gives nothing new.

(2.4) Remark (i) Suppose A and B are recursive sets with $A \leq_T^n B$ via a recursive collection $\{\Phi_{g(k)} : k \in N\}$ of reductions all running in time $O(|z|^\alpha)$. Then $A \leq_T^u B$.
(ii) Furthermore,if the running time is recursively bounded (and hence can be taken to be $g(k)|z|^\alpha$), then $A \leq_T^s B$.

Proof (i). We shall define a single reduction Δ that takes the role of each of the $\Phi_{g(k)}$. On input $\langle z,k\rangle$, Δ first computes a stage $s = s(k)$ where $g(k)\downarrow$ in $|s|$ steps. Thereafter Δ simulates $\Phi_{g(k)}$. For (ii), use this and $g(k)+s(k)$ in place of $g(k)$ for the running time. $\Box$

We will now construct examples to show that the basic classes are indeed different for the various reducibilities.

(2.5) Theorem (i) $FPT(\leq_T^u) \subset FPT(\leq_T^n)$ **even for recursive sets.**
(ii) $FPT(\leq_T^s) \subset FPT(\leq_T^u)$

Proof (ii) We prove this by a simple diagonalization argument.Let $\{\langle\Phi_e,\phi_e\rangle : e \in N\}$ denote an enumeration of all pairs consisting of a procedure and a partial recursive function.We shall satisfy the requirements for $e \in N$:

$$R_{\langle e,n\rangle}\text{: Either } \phi_e \text{ is not total, or}$$
$$\text{for some } k, x,\ \Phi_e(\emptyset; \langle x, k\rangle) \neq A(\langle x, k\rangle)\text{,or}$$
$$\Phi_e(\emptyset; \langle x, k\rangle) \text{ does not run in time } \phi_e(|k|)|x|^n$$

Aditionally, we must ensure that $A \in FPT(\leq_T^u)$. We devote $A_{\langle e,n\rangle}$ to meeting $R_{\langle e,n\rangle}$. We ensure that at most one element of row $\langle e, n\rangle$ enters A, and if z enters $A_{\langle e,n\rangle}$ then z is of the form $\langle 1^m, \langle e, n\rangle\rangle$ for some m.

We shall build A in stages. At stage s we decide the fate of $\langle 1^s, k\rangle$ for all $k \in N$. At stage s, the construction runs as follows:
For each $\langle e, n\rangle \leq s$, if $R_{\langle e,n\rangle}$ is not yet declared satisfied, compute s steps in the computation of $\phi_e(\langle e, n\rangle)$. (Call this $\phi_{e,s}(\langle e, n\rangle)$.) If $\phi_{e,s}(\langle e, n\rangle) \uparrow$ do nothing for $\langle e, n\rangle$ at this stage keeping $m(\langle e, n\rangle, s) = m(\langle e, n\rangle, s-1)$. If $\phi_{e,s}(\langle e, n\rangle) \downarrow$ declare $R_{\langle e,n\rangle}$ as satisfied and perform the following diagonalization for $\langle e, n\rangle$. Run $\Phi_e(\emptyset; \langle 1^s, \langle e, n\rangle\rangle)$ for $\phi_e(\langle e, n\rangle)s$ many steps. If this does not halt in this many steps we need do nothing since the running time is wrong. If $\Phi_e(\emptyset; \langle 1^s, \langle e, n\rangle\rangle) \downarrow$ in $\phi_e(\langle e, n\rangle)s$ many steps set

$$A((\langle 1^s, \langle e, n\rangle\rangle) = 1 - \Phi_e(\emptyset; \langle 1^s, \langle e, n\rangle\rangle).$$

In either case set $m(\langle e, n\rangle, 1^s) = 2\phi_e(\langle e, n\rangle)(s + 1)$.It is clear that the diagonalization succeeds ensuring that $A \neq \Phi_e(\emptyset)$. Note that $A \in FPT(\leq_T^u)$ since for any k , $\langle z, k\rangle \in A$ iff z is of the form 1^t and $\langle 1^t, k\rangle$ is put into A at stage t. This can be decided in time $m(k, t)t$ and since $m(k, t) \neq m(k, t + 1)$ at most once we see that $A \in FPT(\leq_T^u)$.

(i) Again we use a simple diagonalization argument. Now we need a family of reductions $\{\Delta_k : k \in N\}$ with $\Delta_k(\emptyset)$ computing A_k. By the limit lemma we need to meet the following requirements.

$$R_{\langle e,n\rangle}\text{: Either } lim_s \phi(\langle e, n\rangle, s) \text{ fails to exist,or}$$
$$\Phi_e(\emptyset) \neq A_{\langle e,n\rangle} \text{ ,or}$$

it does not run in time $\phi_e(\langle e,n\rangle)|z|^\alpha$.

Here we are working with pairs consisting of a procedure and a binary recursive function. We are denoting by $\phi(p)$ the value of $lim_s\phi(p,s)$ if it exists.We shall additionally,and without loss of generality assume that ϕ_e is nondecreasing in both variables where defined. In the construction to follow a value can be *used* for $\langle e,n\rangle$.

At stage s, if $R_{\langle e,n\rangle}$ is not as yet declared satisfied and $\langle e,n\rangle \leq s$ find the least unused $j \leq s$,if any, such that $j = \phi_{e,s}(\langle e,n\rangle, t)\downarrow$ for some $t \leq s$. If either $\phi_{e,s}(\langle e,n\rangle,t)\uparrow$ for all $t \leq s$ or there is no unused j do nothing. If j and hence t exists declare j as used. Now compute js steps in the computation of $\Phi_e(\emptyset;\langle 1^s,\langle e,n\rangle\rangle)$.

It is clear that A is recursive. Now Δ_k is one of the following two reductions:
Either $\Delta_k = \Psi$ which, on input $\langle x,k\rangle$ says that $\langle x,k\rangle \notin A$, or $\Delta_k = \Delta$ which, on input $\langle x,k\rangle$ computes s where R_k is satisfied, and then has $\langle y,k\rangle \in A$ iff y is of the form 1^s and $1 - \Phi_{e(k)}(\emptyset;\langle 1^s,k\rangle) = 1$ and $\langle y,k\rangle \notin A$ otherwise. Note that the algorithm runs in constant time, so that $A \in FPT(\leq^n_T)$. $\square$

3. Embedding Type Results

In this section we shall analyse the general degree structures associated with the various reducibilities. That is, we look at $(REC,\leq^q_T)$, the recursive sets under $\leq^q_T$ for $q \in \{u,n,s\}$. We will concentrate on embedding type results such as Ladner[Ld], Ambos-Spies[AS], Melhorn[Me] etc, which eventually give enough definability to calculate the degree of the theory of $(REC,\leq^p_T)$ in Shinoda-Slaman[SS], and to get the undecidability result of Ambos-Spies and Nies[AN1,2]. We shall describe a basic technique that allows us to prove analogues of all these results. Local definability however presents special problems in our setting and we treat this in the next section, where we look, for instance, at density.

We begin with the easiest illustration of our technique.

(3.1) Theorem *If* $\mathbf{C}$ *is any complexity class generated by a superpolynomial function* f*, then there exist recursive sets* A *and* B *in* $\mathbf{C}$ *such that* $A \not\leq^n_T B$

and $B \not\leq_T^n A$. *(We write this as* $A|_T^n B$.*) Furthermore* A *and* B *can be chosen so that for all* k*,* A_k *and* B_k *are in* P*-time.*

Proof In **C** we build A and B to meet the requirements below.

$$R_{2\langle e,n\rangle}\text{:For } k = 2\langle e,n\rangle \text{ either}$$
$$\phi_e(k,s) \text{ has no limit, or}$$
$$\text{there is an } x \text{ such that } \Phi_{\phi_e(k)}(A;\langle x,k\rangle) \neq B(\langle x,k\rangle).$$

$$R_{2\langle e,n\rangle+1}\text{:Same as } R_{2\langle e,n\rangle} \text{ but with } A \text{ and } B \text{ reversed.}$$

Fix a recursive superpolynomial f for **C**. To meet the requirements we employ a priority argument. We first describe the basic module: that is the method whereby we meet a *single* requirement, $R_{2\langle e,n\rangle}$, say.

Again we shall have a notion of *used*. Again we employ row $2\langle e,n\rangle$ to meet $R_{2\langle e,n\rangle}$. Let $k = 2\langle e,n\rangle$. Initially we await a stage s_0 such that for some $t_0 \leq s_0$ and some corresponding least unused j_0 we have

$$j_0 = \phi_{e,s_0}(k,t_0)$$

We then declare j_0 as used. Note that j_0 is our current guess for the final value of $\phi_e(k)$. (Further note that this value may not exist.) Now await a stage $u_0 > s_0$ where
$$f(|\langle 1^{u_0},k\rangle|) > j_0|\langle 1^{u_0},k\rangle|^n.$$
As f is superpoly such a stage must u_0 must exist. Further note that we can wait and arrange matters so that we can see this in time $O(|x|^{n+1})$. See if $\Phi_{j_0}(A_{u_0-1};\langle 1^{u_0},k\rangle)$ halts in at most j_0u_0 many steps.(Here A_t denotes the portion of A that we have decided by stage t.) We say that $R_{2\langle e,n\rangle}$ *recieves attention via* $\langle j_0,u_0\rangle$. If the computation does indeed halt then set

$$B_{u_0}(\langle 1^{u_0},k) = 1 - \Phi_{j_0}(A_{u_0-1};\langle 1^{u_0},k\rangle)$$

and declare that $A[m_0] = A_{u_0-1}[m_0]$ where $m_0 = u(\Phi_{j_0}(A_{u_0-1};\langle 1^{u_0},k\rangle)$, the maximum length of an element queried in the computation.(This is called the *use* of the computation.We remind the reader that here we are using the notation that $Q[x] = \{z : z \in Q \text{ and } |z| \leq |x|\}$.) If the computation does not halt, do nothing.

By the above process, if $R_{2\langle e,n\rangle}$ recieves attention via $\langle j_0, u_0\rangle$ then j_0 is not a possible value of $lim_s \phi_e(k,s)$ if indeed $\Phi_{\phi_e(k)}(A) = B$. But it follows that either $\Phi_{\phi_e(k)}(A) \neq B$ or $R_{2\langle e,n\rangle}$ recieves attention infinitely often. The latter case means that it has no limit.

The reader should realize that there is the usual priority conflict in the above. A R_{2d} requirement usually requires us to change B and preserve A, to preserve any disagreement made. An R_{2d+1} type requirement asks us to change A and preserve B. The idea with such arguments is to define a priority ordering that allows all of the requirements to be met. In this particular construction, it is easiest to break the R_j into infinitely many subrequirements of the form:

$R_{2\langle e,n\rangle,m}$: Either $\phi_e(k,s)$ changes value at least k times, or
$\Phi_{\phi_e(k)}(A)$ does not run in time $\phi_e(k)|x|^n$,or
there is some x with $\Phi_{phi_e(k)}(A; \langle x,k\rangle) \neq B(\langle x,k\rangle)$.

We have similar requirements of type $R_{2\langle e,n\rangle+1,m}$. We then use the finite injury method to combine strategies. In the formal construction to follow we use the convention that all computations etc halting at stage s use elements of length below s.

(3.2)Definition We say that $R_{2\langle e,n\rangle,m}$ *requires attention* at stage $s+1$ if $\langle 2\langle e,n\rangle, m\rangle$ is least so that (i),(ii) and (iii) below all hold.
(i) No requirement is currently under attack.
(ii) $Count(2\langle e,n\rangle, s) = m-1$.
(iii) There is some unused $j \leq s$ such that for some $t \leq s$, $\phi_{e,s}(2\langle e,n\rangle, t) = j$. For the least such j declare that $R_{2\langle e,n\rangle,m}$ to require attention *via* j.

Construction
Stage 0. Do nothing
Stage $s+1$ If no requirement is currently under attack see if there is an R_q for $q \leq s$ which requires attention.If no such R_q exists do nothing.If R_q, for $q = \langle 2\langle e,n\rangle, m\rangle$, say, requires attention, declare j as used, and that R_q to be currently under attack with parameter j. Reset $Count(2\langle e,n\rangle, s) = m$.

If some $R_{2\langle e,n\rangle,m}$ is currently under attack and has parameter j (say), see if $f(s) > j|\langle 1^s, 2\langle e,n\rangle\rangle|^n$. If not do nothing. If so declare $R_{2\langle e,n\rangle,m}$ to be no longer under attack *at stage* $t = j|\langle 1^s, 2\langle e,n\rangle|^n + 1$. (This will protect any possible $\Phi_j(A_{s-1}; \langle 1^s, 2\langle e,n\rangle\rangle)$ computation that halts in at most $j|\langle 1^s 2\langle e,n\rangle\rangle|^n$ many steps.)

Now see if $\Phi_j(A_{s-1}; \langle 1^s, 2\langle e,n\rangle\rangle)$ halts in at most $j|\langle 1^s, 2\langle e,n\rangle\rangle|^n$ many steps. If not do nothing else. If so then set

$$B(\langle 1^s, 2\langle e,n\rangle\rangle) = 1 - \Phi_j(A_{s-1}; \langle 1^s, 2\langle e,n\rangle\rangle).$$

Proceed analogously for the $R_{2\langle e,n\rangle+1,m}$ type requirements.

End of Construction

To see that the construction succeeds, if ever we attack some R_k we eventually conclude this attack. We get to set $B(\langle 1^s, k'\rangle)$ (or $A(\langle 1^s, k'\rangle)$ as the case may be) and hence by the assumption that $|s| > u(\Phi_j(A_{s-1}; (\langle 1^s, k'\rangle)$ it must be that

$$\Phi_j(A_{s-1}; \langle 1^s, k'\rangle) = \Phi_j(A; \langle 1^s, k'\rangle),$$

and hence $\Phi_j(A) \neq B$. Note also that if we argue by priorities, we see that each R_k will be attacked if necessary and hence we either force $\phi_e(k', s)$ to change infinitely often, or the running time is wrong. □

Simple variations of the above technique can be used to improve (2.5).

(3.3) Theorem *(i)* $FPT(\leq_T^n)$ *contains infinitely many problems pairwise incomparible with respect to* $\leq_T^u$*).*
(ii) $FPT(\leq_T^u)$ *contains infinitely many problems pairwise incomparible with respect to* $\leq_T^s$*.*

Proof sketch of (e.g.) (ii) We build $\{A_i : i \in N\}$ in stages to meet the following requirements:

$R_{i,j,e,n}$:If $i \neq j$ then either ϕ_e is not total, or
the running time of $\Phi_e(A_i; \langle x,k\rangle)$ exceeds $\phi_e(k)|x|^n$, or
for some x, k ,$\Phi_e(A_i; \langle x,k\rangle) \neq A_j(\langle x,k\rangle)$.

Additionally we must ensure that for all i, $A_i \in FPT(\leq_T^u)$. Again we use row $k = \langle i, j, e, n\rangle$ to meet $R_{i,j,e,n}$. assume $i \neq j$. We will describe the basic module. We wait for a stage s where $\phi_{e,s}(k)\downarrow$. At stage s, if no other R_q is under attack, $R_{i,j,e,n}$ asserts control (as with (3.1), assuming that $\langle i, j, e, n\rangle$ is least), and sees if $\Phi_e(A_{i,s-1}; \langle 1^s, k\rangle)$ halts in fewer than $\phi_e(k)s^n$ many steps. If so we set

$$A_j(\langle 1^s, k\rangle) = 1 - \Phi_e(A_{i,s-1}; \langle 1^s, k\rangle).$$

In either case it declares R_e as satisfied and declares it to be under attack until stage $\phi_e(k)s^n+1$. (Again this is to protect the $\Phi_e(A_{i,s-1}; \langle 1^s, k\rangle)$ computation from A_i enumeration.)

It is clear that $A_i \in FPT(\leq_T^u)$ for all i giving (ii) ,and (i) is essentially similar. □

The idea of using the curent guess as to the values of the constants to meet the requirements for constructions analysing the structure of $(REC, \leq_T^q)$ for $q \in \{u, s, n\}$ is very flexible and allows us to show that a lot of classical $\leq_T^p$ have analogues in our setting.

For instance using this idea it is quite straightforward to show that each complexity class properly containing P contains minimal pairs, that is $A, B \notin FPT$ such that if $C \leq A, B$ then $C \in FPT$. We can also construct recursive $D \notin FPT$ such that if $E \oplus G \equiv D$ then E and G do not form a minimal pair . (Analogue of Downey[Do]). This would seem to indicate that the central strategies of Shinoda-Slaman[SS] ought to extend to be able to verify the following conjecture:

(3.4) Conjecture: (i) The degree of the (first order) theory of $(P(N), \leq_T^q$) for $q \in \{s, u, n\}$ is that of second order arithmetic, and the degree of $(REC, \leq_T^q)$ is that of first order arithmetic. Thus they are as 'complicated as possible'.
(ii) All recursively presentable lattice can be embedded into $(REC, \leq_T^q)$.

Again similar comments apply for the structures $(REC, \leq_m^q)$. In this case we would be looking at the analogue of Ambos-Spies and Nies[AN]:

(3.5) Conjecture: The theory of $(REC, \leq_m^q)$ is undecidable for any $q \in \{s, u, n\}$.

In both cases we remark that there are some differences since the partial orderings generated by the reducibilities need one or two more quantifiers in the non-strongly uniform cases so an approach more like that uses for the r.e. wtt degrees ([ANS]) may be necessary. In the next section we examine local embeddings where we must work below, say, a given degree. Here analogues do not always work .

4. Density

From the last section,it would seem that most results from $(REC, \leq_T^p)$ ought to lift to our setting. However if we look at the structure $\{B : B \leq_T^u A\}$ for a *given* A, this is not in general true. First we shall examine tha analogue of Ladner's result that the polynomial degrees of recursive sets are dense, which in particular show that if $P \neq NP$ then there are NP languages that are neither polynomial time nor NP- complete. The analogous fact for the weft hierarchy would be that *if* $W[t] \neq W[t+1]$ *then there exist infinitely many intermediate problems between* $W[t]$ *and* $W[t+1]$. This is indeed true for strong uniform reducibility as we now see.

(4.1) Theorem. *If A and B are recursive with $A <_q^s B$, then there exists a set C with $A <_q^s A \oplus C <_q^s B$,where $q \in \{m, T\}$.*

Proof We begin by briefly recalling the construction of Ladner[Ld]. Recall that this worked as follows. There were given recursive sets $A \not\leq B$ (working with $\leq_m^p$, say).Let $\{z_n : n \in N\}$ be a standard P-time length/lexicographic P-time ordering of Σ^*. We can assume that A and B are given as the range of p-time functions with domain N in unary notation. We write $A_s = \{f(1^0), ..., f(1^s)\}$ if $f(N) = A$ in this sense. We can also ask that if $|f(1^y)| > |f(1^{y-1})|$ then for all $z > y$, $|f(z)| \geq |f(1^y)|$. We call this a *P-standard enumeration.* So we will assume that we have such enumerations of A and B. Recall also for a reduction Δ on a set E , $u(\Delta(E; x))$ denotes the length of the longest element used in the computation. Let $\{\Phi'_e : e \in N\}$ denote a standard enumeration of all P-time m-procedures.

We must build C to satisfy the requirements:

$$R'_{2e}: \ \Phi'_e(A \oplus C) \neq B$$

$$R'_{2e+1}: \ \Phi'_e(A) \neq C$$

additionally ensuring that $C \leq^p_m B$. For the sake of the R'_j we define a polynomial time relation $R(n)$ on $N = \{1\}^*$. Then we declare that $x \in C$ iff $R(|x|) = 0$ and $x \in B$. Clearly this makes $C \leq^p_m B$.

Now we meet the R'_j *in order* by 'delayed' diagonalization. So we begin with R'_0. We set at each stage s, $R(s) = 1$ until a stage t is found where (i) - (iv) below hold. (Here we consider s, t etc as being in N.)
(i) $\Phi'_{0,t}(A_t \oplus \emptyset; z_n) \downarrow$ in less than t steps.
(ii) $A_t[q] = A[q]$ if $|q| < u(\Phi'_0(A_t \oplus \emptyset; z_n)$.
(iii) $B_t[z_n] = B[z_n]$.
(iv) $\Phi'_{0,t}(A_t \oplus \emptyset; z_n)(= \Phi'_0(A \oplus \emptyset; z_n)) \neq B(z_n) = B_t(z_n)$.

At stage t we say that we have diagonalized R'_0 at z_n, this being found by *looking back for an A- and a B- certified disagreement.*

The idea is then to move to R'_1 and then to R'_2 etc. For R'_1 we set $R(t+1) = 0$, causing C to look like B locally. So we keep $R(u)$ for $u > t$ equal to zero until a stage v is found with some $m \leq v$ and

$$\Phi'_{0,v}(A_v; z_m) \neq C_v(z_m),$$

via $A-$ and $B-$ certified computations. We then move to R'_2 setting $R(v+1)$ to be 1 again. Thus the set C so constructed looks like B with 'holes' in it.

Keeping the above ideas in mind we turn to the result at hand. Now we are given $A < B$ with $\leq$ either $\leq^s_T$ or $\leq^s_m$. Again we must construct C, now to meet the following requirements

$R_{2\langle e,n\rangle}$: Either ϕ_e is not total,
or $(\exists k)(B_k \neq \Phi_e(A \oplus C^{(\phi_e(k))}))$
or $(\exists x, k)(\Phi_e(A \oplus C^{(\phi_e(k))}; \langle x, k\rangle)$ does not run in time $\phi_e(k)|x|^n$.

$R_{2\langle e,n+1\rangle}$: Either ϕ_e is not total,

$$\text{or } (\exists k)(C_k \neq \Phi_e(A^{(\phi_e(k))}))$$
$$\text{or } (\exists x, k)(\Phi_e(A^{(\phi_e(k))}; \langle x, k\rangle) \text{ does not run in time } \phi_e(k)|x|^n).$$

To aid the discussion we will use several conventions. First, if $\phi_{e,s}(k)\downarrow$, then the computation $\Phi_e(E^{(\phi_e(k))}; \langle x, k\rangle)$ cannot call any y of the form (k', z) for $k' > \phi_e(k)$. Also since we get a win for free if $\phi_{e,s}(k)\downarrow$ and the running time of $\Phi_e(E^{(\phi_e(k))}; \langle x, k\rangle)$ exceeds $\phi_e(k)|x|^n$, we shall assume that in the above the third option does not pertain to R_j and concentrate on the first two. This is because if the running time exceeds the bounds during the construction, we can *cancel* the relevant requirement. The argument to follow is a priority one with the Ladner strategy embedded.

Without loss of generality we can take ϕ_e to strictly increasing. Again there will be long intervals with $C(\langle x, k\rangle)$ equal to $\emptyset$ and long intervals where it looks like B, for 'many' k. We have problems, since, for instance, we cannot decide if ϕ_e is total. We first focus on the satisfaction of a single $R_0 = R_{\alpha\langle e,n\rangle}$. We then describe the basic module for an odd type requirement ,and finally describe the coherence mechanism whereby we combine strategies.

The Basic R_0-Module.

To meet R_0 above, we perform the following cycle. We have a parameter $k(0, s)$ that is nondecreasing in s and such that $\lim_s k(0, s) = k(0)$ exists. This is meant to be the number of "rows" devoted to R_0. It remains constant until we change it.

1. (Initialization.) Pick $k(0, 0) = 1$.
2. Wait until a stage s occurs with one of the following holding:

2(a). (Win.) "Looking back" we see a disagreement. That is, as with the Ladner argument, we see an $n < s$ with $z_n \in \{\langle x, j\rangle : j < k(0, s)\}$.

$$\Phi_{e,s}(A \oplus C^{(\phi_e(k(0,s)-1))}; z_n) \neq B(z_n)$$

via A- and B-certified computations, or

2(b). Not (2a) and $\phi_{e,s}(k(0, s))\downarrow$.

Comment If s does not occur then $\phi_e(k(0,s)) \uparrow$ and hence ϕ_e is not total. In this case we call $k(0,s)$ a *witness to the nontotality* of ϕ_e.

If 2(a) pertains, we declare R_0 to be *satisfied* (forever) and end its effect (forever). If 2(b) pertains, then we perform the following action.

3. R_0 *asserts control of* $C^{(\phi_e(k(0,s)))}$. That is, R_0 asks that for all $t \geq s$, until 2(b) pertains, we promise to set $C^{(\phi_e(k(0,s)))}(y) = 0$ for all y with $|y| = t$ and $y \in (\Sigma^*)^(\phi_e(k(0,s)))$. This can be achieved *via* a restraint $r(n,k)$.

4. Reset $k(0,s+1) = k(0,s) + 1$ and go to 2.

The Outcomes of the Basic R_0 Module.

We claim that 2(b) cannot occur infinitely often and hence $\lim_s k(0,s) = k(0)$ exists. Note that we have only reset $k(0,s)$ if 2(b) pertains in step 3. So suppose $k(0,s) \to \infty$ and hence $\phi_e(k(0,s)) \to \infty$. Then for each q and almost all y , we have $C(\langle q,y\rangle) = 0$.

We write $A =^* B$ to denote that the symmetric difference of A and B is finite. So $C_q =^* \emptyset$ for all q. Furthermore, for all q, we can compute a stage $h(q)$ where

$$[\forall t > h(q)](C_q(\langle y,q\rangle) = 0 \quad \text{for all } y \text{ with } |y| > h(q))$$

where $h(q)$ is the stage where R_0 asserts control of row q.

Finally, we know that for all k,

$$\Phi_e((A \oplus C)^{(\phi_e(k))}) = B_k$$

This allows us to get a reduction $\Delta(A) = B$. For each input $\langle y,k\rangle$, Δ simply computes $B(\langle y,k\rangle)$ for all y with $|y| \leq h(k)$, and $C(\langle z,k'\rangle)$ for all k', z with $k' \leq \phi_e(k)$ and $|z| \leq h(k)$. Then Δ simulates $\Phi_e(A^{(\phi_e(k))}; \langle y,k\rangle)$ if $|y| > h(k)$ with the exception that, if Φ_e calls some $\langle r,k'\rangle$ with $|r| \leq h(k)$ (and necessarily $k' \leq \phi_e(k)$), then Δ uses the table of values for C to provide the answer.

Note that the computations of $\Delta(A;\langle x,k\rangle)$ and $\Phi_e(C;\langle x,k\rangle)$ must agree and hence $\Delta(A) = B$, a contradiction. Thus 2(b) can pertain only finitely often. It follows that there are two outcomes.

Outcome $(0, f)$: 2(a) occurs for some t. Then we win R_0 with finite effect. (*Comment:* Once R_0 is met in this way, say at stage t, then we are completely free to do what we like with all y for which $|y| > t$ without injuring R_0.)

Outcome $(0, \infty)$: 2(a) does not occur. Then ϕ_e is not total. Note that the effect of R_0 is in this case infinite and for some $k = \lim_s k(0, s) - 1$, we will have

$$C^{(\phi_e(k))} =^* \emptyset$$

and furthermore, there is a reduction Δ_0 with time bound $\phi_e(k)|x|^n$ for which

$$\Delta_0(A^{(\phi_e(k))}) = B^{(k)}$$

Note that for the basic module, Δ_0 is simply Φ_e.

The Basic Module for R_1.

This is essentially the same as for R_0 except that for R_1 we wish to set $C(\langle x, k\rangle) = B(\langle x, k\rangle)$. Herein is the basic conflict: an even-indexed requirement R_j asks that lots of rows look like $\emptyset$ and an odd-indexed R_j asks for them to look like B.

Combining Strategies.

We cannot perform a delayed diagonalization as in the proof of Ladner's theorem, since we cannot know if $\phi_e(k)$ is defined. The combination of strategies needs the priority method. Let us consider a module for R_1 that works in the outcomes of R_0. We cannot know if this outcome is $(0, f)$ or $(0, \infty)$. Instead we have a strategy based on a guess as to R_0's behavior. Basically R_0 always believes that $k(0, s)$ is $k(0)$, that is, that the current value is the final one. Let $e = e(0)$, $n = n(0)$, $f = e(1)$ and $m = n(1)$.

Whilst R_1 believes that $\phi_e(k(0, 0)) \uparrow$, R_1 acts as if R_0 is not there. So if $k(0, 0) = k(0)$ and $\phi_e(k(0, 0)) \uparrow$ then we win R_1 for the same reasons as we did for R_0. On the other hand, if $\phi_e(0) \downarrow$ for some least stage s, then R_0 will assert control of $C^{(\phi_e(k(0,0)))}$. For the sake of R_1 we have probably been setting $C(0, x) = B(0, x)$ for all x with $|x| < s$. Since R_0 has higher priority than R_1, R_1 must release its control of C_0 (and indeed of C_j for $j \leq \phi_e(k(0))$) until a stage, if any, occurs where 2(a) pertains to R_0 so that R_0 is satisfied and releases control forever (or it becomes inactive because of a time bound being exceeded). Note that if 2(a) pertains at t, then R_1 is free to reassert control of C_0 for all y of the form $\langle y, 0\rangle$ with $|y| > t$. Also, in this case, as R_1 is the requirement of highest overall priority remaining, its control cannot be violated and hence it will be met.

On the other hand, while R_0 can hope that 2(a) will pertain to R_1, R_0 may have outcome $(0, \infty)$ and R_0 will never release control of C_0. The key idea at this point is that we begin anew with a version of R_1 believing that $k(0, s+1) = k(0)$. That is, R_0 will *never again act*.

This version of R_1 can only work with C_q for $q > \phi_e(k(0, s)) = \phi_e(k(0, 0))$. Some care is needed since potentially we need all of B to meet R_1.

An elegant solution to this difficulty is to *shift* B into C above $\phi_e(k(0, s))$. Thus R_1 will ask that

$$C(\langle x, q\rangle) = B(\langle x, q - \phi_e(k(0, s)) - 1\rangle)$$

for $q > \phi_e(k(0, s))$. It does so until either $k(0, t)$ is reset again, or 2(a) pertains, or the time bounds are exceeded. In the latter cases, it reverts to the $(0, f)$-strategy. In the first case it begins anew on $q > \phi_e(k(0, t))$. Since this restart process only occurs finitely often, it follows that we eventually get a *final* version of R_1 whose actions will not be disturbed.

Thus there is a final version of R_1 that is met as follows. As $\lim_s k(0, s) = k(0)$ exists, there is a value r and a stage s_0 so that for $q \geq r$ and $s > s_0$, R_1 is not initialized at stage s and can assert control on C_q if it so desires. If R_0 has outcome $(0, f)$, then $r = 0$, otherwise $r = \phi_e(k(0) - 1) + 1$. So we know that if R_1 fails then for all j there is a stage $h(j)$ (computable from the parameters r and s_0) where for y with $|y| > h(j)$

$$C(\langle y, r + j\rangle) = B(\langle y, j\rangle) \text{ and}$$

$$\Phi_f(A; \langle y, r + j\rangle) = C(\langle y, r + j\rangle).$$

Thus if R_1 fails again we can prove there is a reduction $\Delta(A) = B$ with running time $O(|z|^m)$ and computable constants.This is a contradiction.

The outcomes for R_1 are thus either $(1, \infty)$ and $(1, f)$. In the former case we know that for a finite number of rows j and for almost all y, $C(\langle y, j\rangle) = B(\langle y, j\rangle)$. But we also know that for such rows there is a reduction Δ_f such that

$$\Delta_f(A; \langle y, j\rangle) = C(\langle y, j\rangle) \text{ in time } O(|y|^m) \text{ and computable constants.}$$

We continue in the obvious way with the inductive strategies. Consider eg R_2. It is confronted with at worst a finite number of rows permanently controlled by R_0 and a finite number by R_1. However, in each case we know that there is a reduction from a computable number of rows af A to these rows, and hence a reduction

$$\Psi_2(A; \langle y, j\rangle) = C(\langle y, j\rangle)$$

for all j cofinally under the control of either R_0 or R_1. Therefore to argue that R_2 is met, we get to use Ψ_2 to help construct a reduction from A to B. That is, for R_i, let $e = e(i)$ and $n = n(i)$.Then inductively we have a reduction and constants $p(2), m(2)$ and $r(2)$ with

$$\Psi_2(A^{m(2)}; \langle x, j\rangle) = C(\langle x, j\rangle)$$

for all $j \leq p(q)$ running in time $m(2)|x|^{r(2)}$. Futhermore, we have a stage s_2 such that for all $k < 3$, R_k ceases further activity.

Thereafter R_2 is free to assert control over any row$q$ of C for $q > p(2)$. If we suppose that R_2 fails, then for each such q, R_2 will eventually assert control of C_q at some stage $h(q)$ to make $C(\langle x, q\rangle) = 0$ for all x with $|x| > h_2(q)$ and we have $\Phi_{e(2)}(C) = B$.

Now to get a reduction Δ from A to B we go as for R_0 except that now if $\Phi_{e(2)}$ makes an oracle question of $\langle y, j\rangle$ for $j \leq p(2)$,we use Ψ_2 to answer this question.Thus we get a reduction Δ_2 that runs in time $O(|x|^{r(2)+n(2)})$, with computable constants and correct use. Thus again $B \leq A$,a contradiction.

The remaining details give no further insight and we leave them to the reader. □

What happens to $\leq_T^q$, $\leq_m^q$ for $q \in \{u, n\}$? Answer: The above proof will go through if we only consider reductions $E \leq_T^q F$ (or m reductions) *via a* *function $\phi(x)$ with an approximation $\phi(x, s)$ for which there is a recursive function g such that*

$$\{s : \phi(x, s) \neq \phi(x, s + 1)\} \leq g(x).$$

That is although we do not know the constants we do know in advance the maximum number of times that our approximation can change. In the context of the Robertson-Seymour application, we would not necessarily know the obstruction set but would know in advance a bound on the possible size of the set. Of course this won't in general happen, leaving the following question apparently open.

(4.3) Question Do any of the other reductions generate a dense structure on the recursive sets? Indeed, for any of the other reductions if, for some k, $W[k] \neq W[k + 1]$, then is there an infinite collection of classes between $W[k]$ and $W[k + 1]$?

We will briefly observe that we can easily get a density result for certain classes of sets with additional hypotheses. We also look at these questions for the degrees *at large* where we find substantial differences from the classical case. Before we do so we remark that (4.1) has many obvious variations : for instance one can put an infinite antichain between A and B, or recursively presentable lattices,etc. Again these results don't really give new insights and we omit them.

We begin with a weak density result. Some parameterized problems exhibit *concentrated* nondeterminism. A well known example of this is the following.

PLANAR k-COLOURABILITY
Input A planar graph G and an interger k.
Question Is G k-colourable?

Now for any $k \neq 3$ we know that this is linear time and hence certainly in FPT. Yet we also know that it is NP complete. Although it is not in the W- hierarchy unless $P = NP$, it shows that many natural problems are *concentrated* in the sense that for some k, B_k itself is not in P-time. With this motivating example in mind we have:

(4.4) Theorem *For $\leq$ any of the reductions, if A and B are recursive sets, with $A < B$ and such that $B_m \not\leq A$ for some m, then there is a C such that $A < C < B$.*

Proof We give the proof for $\leq_T^u$, the others being similar. Suppose that $B_m \not\leq_T^u A$. We define C_k to be empty for all $k \neq m$. Then we meet

$R_{2\langle e,n\rangle}$: Either $\lim_s \phi_e(m,s)$ does not exist,or
$\Phi_e(A^{(\phi_e(m))}) \neq C_m$, or
there exists x such that the running time
of $\Phi_e(A^{(\phi_e(m))}; \langle x, m\rangle)$ exceeds $\phi_e(m)|x|^n$.

$R_{2\langle e,n\rangle+1}$: Either $\phi_e(m,s)$ has no limit, or
$\Phi_e((A \oplus C)^{(\phi_e(m))}) \neq B_m$, or
the running time is wrong.

The proof is fairly similar to some of our earlier ones and we only sketch it. We first break the requirements down into

$R'_{2\langle e,n\rangle,q}$: Either $\phi_e(m,s)$ changes at least q times, or as before.

and similarly with the $R_{2\langle e,n\rangle+1,q}$. So for simplicity asume that the time bound for $R'_j = R'_{2\langle e,n\rangle,q}$ is not exceeded, nor is the use bound. We give R'_j priority j. We allow it to assert controll of C_m when it has the priority and it has seen $\phi_e(m,s)$ change exactly q times. At such a stage it will set $C(\langle x, m\rangle) = 0$ until we get a disagreement or we see $\phi_e(m,t) \neq \phi_e(m,s)$ for some $t > s$. Such a stage must exist by hypothesis. It is clear that either we win via some R'_j, q with finite effect, or for all q we see that $\phi_(m,s)$ changes q times and hence $\phi_e(m,s)$ has no limit. □

We remark also that density *fails* for the nonrecursive sets. This stands in contrast to the situation for $\leq_T^p$, as was realized by Shinoda. Shinoda[unpubl.] observed that the Ladner density argument can be modified to work for *any* sets A and B with $A <_T^p B$. Recall, for R_{2e} that we wish to keep $C(x)$ equal to 0 until a stage and a z_n are found where we *know* that $\Phi_e(A \oplus C; z_n) \neq B(z_n)$. Now if Φ_e has use bounded by $|x|^{n(e)} + e$, then by stage, say, $2^{2^{|x|^{n(e)}+e}}$, B can figure out, in P-time relative to B, if there is some z_n with $|z_n| = s$ and $\Phi_e(A \oplus C; z_n) \neq B(z_n)$. Thus we inductively promise that we won't switch until we see a stage t which occurs for the first z_n such that $\Phi_e(A \oplus C; z_n) \neq B(z_n)$, where $t = 2^{2^{|z_n|^{n(e)+e}}}$, and similarly for the requirements with odd indices.

This means that the whole construction can be made P-time in B giving the following result.

(4.5) Theorem *(Shinoda[unpubl.]) For any sets A and B, if $A <_T^p B$ then there exists C with $A <_T^p C <_T^p B$.*

Now we show that (4.5) fails for $\leq_T^p$.

(4.6) Theorem *There exists $A \notin FPT(\leq_T^s)$such that for all $B <_T^s A$, $B \in FPT(\leq_T^s)$.*

Proof We build an r.e. set $A = \bigcup_s A_s$ in stages to satisfy the following requirements.

$$N_{e,n}\text{: Either } \phi_e \text{ is not total, or}$$
$$(\exists x, k)(\Phi_e(A^{(\phi_e(k))}; \langle x, k\rangle) \text{ has running time exceeding } \phi_e(k)|x|^n, \text{ or}$$
$$\Phi_e(A) \in FPT(\leq_T^s), \text{ or}$$
$$A \leq_T^s \Phi(A).$$

$$P_e\text{: } \bar{A} \not\models W_e \text{ where } W_e \text{ denotes the } e\text{-th r.e. set.}$$

We will carefully describe the ideas for the modules before we describe the formal construction. As usual a $P_{e,n}$ or an $N_{e,n}$ ceases activity if the running time ever proves wrong. So we can assume only to be considering such good (Φ_e, ϕ_e) pairs.

We meet the $P_{e,n}$ as follows.We *pick* a fixed *follower* $z = z(e, n) = \langle 0, \langle e, n\rangle\rangle$ targetted for A. We wait until we see a stage s with $\langle 0, \langle e, n\rangle\rangle \in W_{e,s}$ and then put $\langle 0, \langle e, n\rangle\rangle$ into A. This is the basic module but needs modification to live with the $N_{e,n}$.

Now $N_{e,n}$ will have two outcomes. These are labelled $(\langle e, n\rangle, f\rangle$ and $(\langle e, n\rangle, \infty)$. It is the former if ϕ_e is not total, and the latter if ϕ_e is total.

The idea is the following. Associated with almost all elements targetted for A (i.e. of the form $\langle 0,j\rangle$) we have a current (e,n)-*state*. This will initially be f. We *raise* the state to ∞ if we see a stage s and a number q such that $\phi_{e,s}\downarrow$ and

$$\Phi_e(A_s^{j+1}) \neq \Phi(A_s^j)$$

where A_s^k is the result of putting $\langle 0,k\rangle, ..., \langle 0,s\rangle$ into A_s and changing nothing else. The idea is that we cancel, by enumeration into A all followers of the form $\langle 0,j'\rangle$ for $j' \leq s$ and $j' > j$. we then reassign $\langle 0,j\rangle$ the $P_{g,m}$ of highest priority not yet satisfied and having no follower of state ∞. We also promise that if $\langle 0,j\rangle$ enters at stage t then we also cancel $\langle 0,j'\rangle$ by enumeration for all $j < j' \leq t$. We call this the *dump*. For a single $N_{e,n}$ the idea succeeds. For consider the outcomes. Either for almost all $\langle 0,j\rangle$, $\langle 0,j\rangle$ is cancelled or has state ∞; or for almost all j, $\langle 0,j\rangle$ is cancelled or has state f. If the first option pertains we define a reduction Δ taking $\Phi_e(A)$ to A as follows. Suppose we have defined it for all $j \leq k$. Find the first r with $r > k$ and the first stage S where $\langle 0,r\rangle$ has state ∞, and is given it at stage s. Then for all r' with $k < r' < r$, $\langle 0,r'\rangle \in A$ if $\langle 0,r'\rangle \in A_s$. Also, there is some $q = q(r)$ such that

$$\Phi_e(A_s;q) \neq \Phi_e(A_s^r;q).$$

Now we define Δ. Inductively,we know that for each $\langle 0,k\rangle < \langle 0,r\rangle$ with state ∞ not yet in A will be asociated with a number $q(k)$ such that the value of $\Phi_e(A;q(k))$ determines if $q(k)$ enters A. Then Δ says that for $k < m < r$, $\langle 0,m\rangle \in A$ iff $\Phi_e(A;q(d)) \neq \Phi_e(A_s;q(d))$ for the least d with $q(d) \notin A_s$. Finally Δ says that $\langle 0,r\rangle \in A$ iff $\Phi_e(A;q(r)) \neq \Phi_e(A_s;q(r))$. This generates a very large constant but note that we get,for each row, $\Phi_e(A) \leq_T^s A$ via a constant time reduction.

If the final state is f then we claim that $\Phi_e(A) \in FPT(\leq_T^s)$. To see this, as the final state is f for almost all j, there is a stage t and a k such that for all $j > k$, $\langle 0,j\rangle$ has state f and if $m < k$ then

$$\langle 0,m\rangle \in A \text{ iff } \langle 0,m\rangle \in A_s.$$

Now we know that for all $s > t$ and all y,j if $|y| \leq s$ and $\phi_{e,s}(j)\downarrow$, then

$$\Phi_e(A_s^j;y) = \Phi_e(A_s;y).$$

It follows that no matter whether $\langle 0, j\rangle$ enters or not, we always get the same answer. Thus to compute $\Phi_e(A; \langle 0, j\rangle)$ find the least stage $s > t, |y|$ where $\phi_e(j) \downarrow$ and compute $\Phi_e(A_s; y)$. This is the correct answer.

For more than one strategy we need to nest the states. To do this requires the so-called tree of strategies $\mathbf{0}''$ priority method. For more on this method we refer the reader to Soare[So].

Let $T = \{\infty, f\}^*$, with $\infty <_L f$ inducing the lexicographic ordering on T.We refer to members of T as *guesses.* We shall use the phrase 'initialize'. We take this to mean that all followers, etc currently associated with a requirement are no longer associated. We remind the reader that all computations are bounded by s at atage s.

(4.5) Definition We say that P_e requires attention at stage s if P_e is least such that one of the following holds:

(i) for some follower x of P_e, we have $x \in W_{e,s}$.
(ii) P_e has no follower at stage s.

Construction
Stage s find the least number of the form $\langle 0, j\rangle$ not yet in A such that for some k,

(i) $\langle 0, j\rangle$ has j-state $\sigma * (k, f)$
(ii) There is at least k elements of the form $\langle 0, q\rangle$ not yet in A with $q < j$ having state σ and (potentially followers) or followers.
(iii) $\phi_{e(k),s}(j) \downarrow$.
(iv)$\Phi_{e(k)}(A^{j+1}_{s-1}; y) \neq \Phi_e(k)(A^j_{s-1}; y)$ for some y with $|y| < s$.

If such $\langle 0, j\rangle$ exists, declare $\langle 0, j\rangle$ to have state $\sigma * (k, \infty)$ and put $\langle 0, j + 1\rangle, ..., \langle 0, s\rangle$ into A_s. Declare $\langle 0, j\rangle$ as unassigned and initialize any P_q not having a follower with state $\leq_L \sigma * (k, \infty)$. Now find the least $\langle 0, j'\rangle$ with $j' < j$ such that $\langle 0, j'\rangle$ has state $\leq_L \sigma * (k, \infty)$. For any $\langle 0, r\rangle$ with $j' < r < j$, declare $\langle 0, r\rangle$ as *no longer a potential follower EVER AGAIN* (and having no state henceforth.)

Now see if P_e requires attention. If (i) holds via $x = \langle 0, q\rangle$, put $\langle 0, q'\rangle$ into A for $q < q', s$. Initialize all P_k for $k > e$. If (ii) holds find the least $\langle 0, j\rangle$ not yet asigned to a requirement and still a potential follower, and assign $\langle 0, j\rangle$ to P_e.

End of Construction

Verification (sketch) Clearly A is r.e.. Let TP denote the true path of the construction,that is TP is in $[T]$ the collection of all paths through T and is the leftmost one visited infinitely often. Specifically $\lambda \subseteq TP$, and whenever $\sigma \subseteq TP$, then $\sigma{*}(0,\infty) \subseteq TP$ if $(\exists^\infty s)(s$ is a $\sigma{*}(0,\infty)-$stage) and otherwise $\sigma * (0, f) \subseteq TP$. We claim that TP exists and each P_e requires attention at most finitely often. This is easily proven by induction on $|\sigma|$. Thus if $\sigma \subseteq TP$ we can go to a stage t where, for all e with $e < |\sigma|$, P_e does not require attention after stage t and for no stage $s > t$ is s a τ-stage with $\tau \leq_L \sigma$ and $\tau \not\subseteq \sigma$. Now at the least σ-stage after t, P_e will be assigned a follower x with state σ if it does not already have one. This follower cannot be cancelled or initialized by choice of t and hence will succeed in meeting P_e.

Finally to see that all the $N_{e,n}$ are met, we can argue almost precisely as we did in the basic module. By the fact that there are only 2^{f+1} many f states, we can see that almost all j eventually get in the same f state. If the relevant state is of the form $\tau * (e, \infty)$ then $A \leq \Phi(A)$ and if this is of the form $\sigma = \tau * (e, 0)$ it is the case that $\Phi_e(A)$ is in $FPT(\leq_T^s)$. The only difference is that, in the latter case for no $(0, j)$ of state σ and no stage $s > t$ for some parameter t is it the case that $\phi_e(j)\downarrow$ and for some y with $|y| \leq s$,

$$\Phi_e(A_s^{j+1}) \neq (A_s^j).$$

Since the only things that survive have state σ it follows that $\Phi_e(A) \in FPT(\leq_T^s)$. $\square$

5. Oracle Results

In this section we shall explore oracle results. These results provide some evidence that methods that relativise are not sufficient to resolve questions such as $FPT(\leq_T^u) =?W[P]$. Of course the exact meaning of this imprecise statement is not quite clear in view of such results as Shamir's $IP = PSPACE$ result (Shamir[Sh]) which is known to fail relative to a random oracle. None the less we feel that the results of this section at least indicate that the relevant separation or collapse results will be hard. This is in the same spirit as Baker, Gill and Solovay[BGS]. We also believe that these oracle separation results support our thesis that the weft hierarchy is infinite.

We begin with an oracle result that supports the thesis that $FPT = ?W[P]$ is independent of $P = ?NP$. In view of this result we believe that it is unlikely that there is a proof that $P \neq NP$ implies $FPT \neq W[P]$ unless the hierarchies collapse.

(5.1) Theorem *There is a recursive oracle A relative to which $W[P] = FPT$ yet $P \neq NP$.*

Proof We do this for $\leq_T^s$ and observe that the obvious modifications work for the other reducibilities. Let Q_e denote the e-th P-time relation. Define K^B via

$\langle\langle x, e, 0^n\rangle, k\rangle \in K^B$ for some y with $|y| = |x|$, y has weight k and $Q_e^B(y)$ holds in n steps.

In view of the direct relationship between circuits and relations, it is clear that K^B is $W[P]$-complete.

Now let f be any recursive function from N to N. suppose we build a recursive set A such that for each k and all x with $|x| \leq f(k)$ we have $A(\langle x, k\rangle)$ can be computed in $g(k)$ many steps, and for all y with $|y| > f(k)$ we have $A(\langle y, k\rangle) = A(\langle y, k\rangle)$. We claim that $A \equiv_m^s B$, so that $W[P]^B = W[P]^A$. To see this for the k-th row for the reduction from A to B, say, we first compute $g(k)$ and $f(k)$. As A and B are recursive we can write the corresponding initial segments in a table. Otherwise $\langle x, k\rangle \in A$ iff $\langle x, k\rangle \in B$, and hence $A \equiv_m^s B$.

Now take any B with $W[P]^B = B$. That is, define B via

$\langle\langle x, e, 0^n\rangle, k\rangle \in B$ iff for some y with $|y| = |x|$, y has weight k and $Q_e(y)$ holds in n steps.

Now it will suffice to define f, g and A as above and ensure that $P^A \neq NP^A$.

We do this as follows. We must meet the requriements

$$R_k \ : \ \Gamma_k(A) \neq C,$$

where $C \in NP^A$ and Γ_k denotes the k-th P-time procedure with use q_k, say. We meet R_k via row $k+1$. We define C so that

$$\langle x, k\rangle \in C \text{ iff } (\exists y)[|y| = |x| \text{ and } \langle y, k\rangle \in C], \text{ and hence } C \in NP^A.$$

At stage k, we will have defined $f(i)$, $g(i)$ for $i \leq k$ and a restraint $r(k)$. Choose x so that $2^{|x|}$ exceeds $q_k(\langle x, k\rangle)$, and $|x| > r(k)$. Now compute $\Gamma_k(A_k^*; \langle x, k+1\rangle)$ where A_k^* is the result of setting A equal to B on all $\langle x, j\rangle$ for $j \leq k$ and $|x| > f(j)$, and setting $A(\langle x, k\rangle)$ equal to what we have decided at stages $\leq k$ for y with $|y| \leq f(j)$. (So, basically we've *decided* at stage k the precise contents of A on $\langle x, j\rangle$ for $j < k$.) Set $A_k^*(\langle z, r\rangle) = 0$ for $r \geq k+1$.

Now if $\Gamma_k(A_k^*; \langle x, k+1\rangle) = 1$ define $\langle x, k+1\rangle = 0$ for all z with $|z| \leq r(k+1)$ with $r(k+1) \geq r(k)$ and also exceeding all uses seen so far. (This means that future actions will not affect these protected computations). If $\Gamma_k(A_k^*; \langle x, k+1\rangle) = 0$ then for some y with $|y| = |x|$ we have that $\langle y, k+1\rangle$ is not queried during the computation of $\Gamma_k(A_k^*; \langle x, k+1\rangle)$ computation because $2^{|x|}$ exceeds $q_k(|x|)$. Now put $\langle y, k+1\rangle$ into A and otherwise set $A(\langle z, k+1\rangle) = 0$ for all z with $|z| \leq r(k+1)$. Set $f(k+1) = r(k+1)$ and define $g(k+1)$ appropriately.

This ensures that $\Gamma_k(A) \neq C$ via the witness $\langle x, k+1\rangle$ since inductively all previous restraints are maintained, and hence all previous disagreements are also preserved. Thus $NP^A \neq P^A$) and yet $FPT^A(\leq_m^s) = W[P]^A$. □

(5.2) Corollary There exist recursive oracles A and B with $W[P]^A = W[P]^B = FPT^A = FPT^B$, $A \equiv_m^s B$, $NP^B = P^B$, yet $P^A \neq NP^A$.

Proof Take A and B as in the proof above. We claim that $NP^A = P^A$. This will then give the desired result. Define D via:

$$\langle x, e, 0^{p_e(|x|)}\rangle \in D \text{ iff some computation of } \Phi_e(B; x) \text{ accepts in } n \text{ steps.}$$

Then as usual D is NP^B complete. We claim that $D \in P^B$. To see this simply note that

$$\langle x, e, 0^{p_e(|x|)}\rangle \in D \text{ iff } (\exists j)[j \leq |x| \text{ and } \langle\langle x, e, 0^{p_{h(e)}(|x|)}\rangle j \in B],$$

where $Q_{h(e)}$ is the relation representing Φ_e. Hence $P^D = NP^D$. □

Now we turn to results separating the W hierarchy. Ideally we would like an oracle that separates the whole W hierarchy, showing it infinite. Unfortunately at this stage we don't know how to do this. Using a Baker,Gill and Solovay construction it is not difficult to show that $W[P]$ can be different from FPT in relativised worlds (indeed in random worlds). We do a little better. We construct an A such that $W[1]^A \neq FPT^A$. Before we do this we briefly describe how to interpret oracle results in circuits. This is a matter where there is no universal agreement. One natural idea that we use in our construction is to view an assignment of values to the inputs into oracle and gates as determining a word, and considering the gate as outputing one if the word is in A. For our purposes this seems a reasonable model for an oracle circuit.

To separate $W[P]$ from FPT we use the set C defined as follows.

$$\langle z, k\rangle \in C \text{ iff } (\exists y(|y| = |x| \text{ and } y \in A \text{ and } y \text{ has weight } k).$$

Now C is in $W[1]^A$ via the circuit such that, on input $\langle z, k\rangle$ the circuit to accept $\langle z, k\rangle$ consists of a single oracle gate with inputs $z_1, ..., z_{|z|}$ ordered left to right. Then $\langle z, k\rangle \in C$ iff there is a weight k work accepted by the gate (i.e. in A). It is routine then to build A to meet

$R_{e,n}$: Either ϕ_e is not total ,or
there is an $\langle x, k\rangle$ such that $\Phi_A; \langle x, k\rangle) \neq C(\langle x, k\rangle)$,or
the running time is incorrect.

To do this we assign $R_{e,n}$ some row $k = k(n)$ with $m^k > O(m^n)$. Then we meet $R_{e,n}$ as follows. Wait till $\phi_e(k)\downarrow$. Then find an x of sufficient length as not to injure other requirements, and so that $|x|^k$ exceeds $\phi_e(k)|x|^n$. We can then diagonalize via $\langle x, k\rangle$ in the standard way, using the string not addressed in the A computation of length $|x|$ (if it is the case that Φ_e is outputting 0), or doing nothing as the case may be but then restraining the result so as not to be disturbed by future actions. The strategies combine sequentially as usual and the result follows. Thus we have for each of the reducibilities (noting that the above can be easily modified for the others):

(5.3) Theorem *There is a recursive oracle A such that* $W[1]^A \neq FPT^A$ *(for any of the reducibilities).*

Now we address some remarks to the above. On the positive side it shows that in relativised worlds (with the model used) that $W[1]$ can differ from $mon - W[1]$ although they are the same in the real world. (Here $mon - W[t]$ denotes the version of $W[t]$ obtained by only considering circuits with no inverters.) Thus the give an example of the failure of relativization, or perhaps suggests that other relativization models might be more appropriate (more on this later.) Furthermore standard techniques would seem to show that this oracle failure will hold relative to a random oracle. This gives an amusing example of the failure of the battered random oracle hypothesis.

On the negative side, the above is rather unsatisfying. The separation is achieved not by analysing the structure of the circuits involved but rather by the manner by which a procedure can address information from an oracle. In a way it clearly demonstrates the shortcomings of the Baker-Gill-Solovay results too. We feel that it would be infinitely more satisfying to get a separation achieved more by the combinatorics and less on the oracle. This might well lead to real insight into the issues involved.

Open Questions, Generalizations, etc

There are several obvious generalizations one could pursue. One could look at parameterizations of the hierarchy, $PSPACE$, $\#P$ etc. This seems a very interesting exercise and we have some results with Karl Abrahamson [ADF] on games and the number of moves to win and $PSPACE$. Here one asks that the parameterization works in space $O(|x|^\alpha)$ for a fixed alpha. We also have some partial results on analogues of D_P. Here Valiant-Vazirani[VV] works for the analogue D_P for $W[P]$ but not at the $W[1]$ level and weight is lost in the procedure they use to take the hashed formula and convert to CNF.

Quite aside from the above we feel that central questions in the area include whether collapse propogates upward in the hierarchy (i.e. $W[t] = W[t+1]$ implies $W[t] = W[u]$ for all $u > t$.) nother question is to understand the exact relationship of the concepts here with classical notions such as

NP. We have some contributions in this area also to be found in [ADF].Also things such as f.p. crypto would seem very worthwhile. After all one really needs is feasibly one way functions.

Finally the degree structure of, in particular $W(\leq_T^u)$ and $W(\leq_T^n)$ remains to be explored.

References

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

[ADF2] K. Abrahamson, R. Downey, and M. Fellows,"Fixed Parameter Intractability II," in STACS'93, (1993) 374-385.

[AS] K. Ambos-Spies, "On the Structure of the Polynomial Degrees of Recursive Sets," Habilitationschrift, Universitat Dortmund, 1984.

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

[AN2] K. Ambos-Spies, A. Nies, and R.A. Shore, "The Theory of the Recursively Enumerable Weak Truth Table Degrees is Undecidable,"to appear.

[BGS] T. Baker, J. Gill, and R. Solovay, "Relativizations of the $P =?NP$ Question," Siam J. Comput. **4**(1975), 431-442.

[BDG] J. Balcazaar,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. 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.

[CD] P. Cholak and R. Downey,"Undecidability and Definability for Parameterized Polynomial Time Reducibilities,"(to appear) *Logical Methods* (ed.

Crossley, Remmel, Shore and Sweedler) Birkhauser, Boston.

[Do] R.Downey, "Nondiamond Theorems for Polymomial Time Reducibility," J.C.S.S. **45** (1992) 385-395.

[DF1] R. Downey and M.Fellows, "Fixed Parameter Tractability and Completeness, " Congressus Numerantium **87** (1992) 161-187.

[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 Tractability and Completeness," Monograph in Preparation.

[DF5] R. Downey and M. Fellows,"Feasible Parameterized Tractability," to appear, *Feasible Mathematics II* (ed. Clote and Remmel) Birkhauser, Boston.

[FL1] M. Fellows and M. Langston,"On Search, Decision, and the Efficiency of Polynomial Time Algorithms," in STOC'89, (1989) 501-512.

[FL2] M. Fellows and M. Langston,"An Analogue of the Myhill-Nerode Theorem and It's Use in Computing Finite Basis Characterizations,"in FOCS'89 (1989) 520-525.

[GJ] M. Garey and D. Johnson, *Computers and Intractability*, Freeman, San Francisco, 1979.

[La] R. Ladner,"On the Structure of Polynomial Tine Reducibility," JACM, **22** (1975) 155-171.

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

[RS1] N. Robertson and P. Seymour,"Graph Minors XIII: The Disjoint Paths Problem," to appear J. Comb. Th. B.

[RS2] N. Robertson and P. Seymour,"Graph Minors XV: Wagner's Conjecture," to appear, J. Comb. Th. B.

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

[Sho] J. Shinoda, Personal Communication.

[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," Theoretical Computer Science **97** (1992) 263-284.

[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.