The Quasilinear Isomorphism Challenge

Kenneth W. Regan[*] Jie Wang[†]

Abstract

We show that all the basic facts about the Berman-Hartmanis Isomorphism Conjecture carry over from polynomial to quasilinear time bounds. However, we show that the connection between unique-accepting NTMs and one-way functions, which underlies much subsequent work on the B-H conjecture, breaks down for quasilinear time under relativizations. Hence we have a whole new ballgame, on a tighter playing field.

1 Why Quasilinear Time?

Polynomial-time algorithms have been widely accepted as efficient algorithms in computer science. In practice, many "polynomial-time" algorithms have small exponent: $O(n)$, $O(n^2)$, up to $O(n^3)$ or $O(n^4)$ at most, and usually the constant factors hidden in the O notation are reasonable.[1] The concept of polynomial time is basically independent of machine model (e.g. RAM or Turing machine), and of differences in encoding conventions for problems, such as adjacency matrix versus edge-list for graphs. Polynomial time bounds have nice closure properties: The sum or product or composition of two polynomials is a polynomial, so one can sequence or compose two polynomial-time routines, or insert one into the body of a "FOR $i := 1$ TO n" loop or a binary search, and obtain a new polynomial-time routine. A host of polynomial-based complexity theory papers have been built on these properties. However, free use of these constructions can quickly bump up the exponent past what a practitioner would call reasonable. An n^{100}-time algorithm is "polynomial" but not desirable. These caveats have motivated a handful of theoretical papers on time bounds less than "polynomial."

Of all such time bounds, the one with the best balance of importance, convenience, attractiveness, and attention in the literature appears to be *quasilinear* time, namely time $qlin = n(\log n)^{O(1)}$. Natural algorithmic primitives that run in qlin-time and are believed not to be in linear time include: sorting, integer multiplication and division, Fast Fourier Transforms, computing $ab + c$ in finite fields, and universal hashing. (For evidence of lower bounds see [FP74, MNT93].) The Hennie-Stearns reduction from k Turing machine tapes to two runs in time $O(n \log n)$, and this extends to show that qlin-time problems have qlin-sized circuits, as observed by Schnorr [Sch76, Sch78]. Quasilinear time bounds are closed under addition and composition, though not under product; divide-and-conquer and binary search over polynomial-sized domains are OK, but nesting inside "FOR $i := 1$ TO n" loops is verboten.

[*]Department of Computer Science, State Univ. of NY at Buffalo, 226 Bell Hall, Buffalo NY 14260-2000. Internet: regan@cs.buffalo.edu.

[†]Department of Mathematics, University of North Carolina at Greensboro, Greensboro, NC 27412. Internet: wangjie@hamlet.uncg.edu. Supported in part by NSF Grant CCR-9396331 and by UNC Greensboro under grant NFG-291362.

[1]For many problems the exponents have been improved over the years, though often at steep costs in constant factors. There is also the "Robertson-Seymour phenomenon" of problems whose low-exponent algorithms apparently *must* have stunningly high constants; for a survey, see Johnson [Joh87].

As with polynomial time, machines running in qlin-time can be efficiently enumerated. Some machine-independence may be lost because the class DQL of languages accepted in qlin-time by deterministic TMs appears to be smaller than the analogous class DNLT for log-cost RAMs. However, Gurevich and Shelah [GS89] showed that DNLT [2] is the same for many RAM-like models, and more strikingly, that the analogous nondeterministic classes NQL for TMs and NNLT for RAMs are *equal*. Schnorr [Sch78] showed that Cook's Theorem can be made to run with quasilinear time overhead, which improves the quadratic or cubic simulations of the proofs in [Coo71, GJ79, HU79]. So *SAT* and $3SAT$ are complete for NQL under DQL many-one reductions ($\leq_m^{ql}$). The standard reduction from $3SAT$ to *Independent Set* runs in qlin-time *if* one uses the edge-set encoding for graphs; with similar provisions, *Clique* and *Hamiltonian Circuit* and *Vertex Cover* and *Graph 3-Colorability* are likewise NQL-complete. Indeed, the catalogues of Dewdney [Dew81, Dew82, Dew89] show that "most" of the familiar NP-complete problems belong to NQL and are complete under $\leq_m^{ql}$, and related observations have been made by by Stearns and Hunt [SH90] and Grandjean [Gra93, Gra94].

We contribute the observation that all these problems are *qlin-isomorphic*, meaning that they are images of each other under bijections from Σ^* to Σ^* that are computable and invertible in qlin-time. After showing in Section 2 that all the fundamental results of Berman and Hartmanis [BH77] carry over to quasilinear time, we formulate a "Quasilinear Time Isomorphism Conjecture." However, in Section 3 we show that most of the subsequent work on the original Berman-Hartmanis conjecture does *not* carry over, because the connection to one-way functions breaks down under relativizations. We construct an oracle C such that $\mathrm{NQL}^C = \mathrm{UQL}^C = \mathrm{DQL}^C$, and yet quasilinear one-way functions exist (even in a strong sense)! We believe that our notion of a *strong qlin-one-way function* is important in itself. Thus our answer to "why quasilinear time?" here is not merely more chart-making for the theoretician, but a meaningful new twist on a famous problem.

2 Quasilinear Time Isomorphisms

The original theory in Berman and Hartmanis' paper is based around the notion of a function f being *length-increasing*, namely for all x, $|f(x)| \geq |x|+1$. Instead we define: f is *length-doubling* if for all x, $|f(x)| \geq 2|x|$.

Theorem 2.1 *Let f and g be one-one, qlin-time computable and invertible reductions of A to B and B to A respectively such that $f \circ g$ and $g \circ f$ are length-doubling. Then A and B are qlin-isomorphic.*

Proof. Following [BH77], let

$$\begin{aligned} R_1 &= \{(g \circ f)^k(x) : k \geq 0, x \notin g(\Sigma^*)\}, \\ R_2 &= \{g \circ (f \circ g)^k(x) : k \geq 0, x \notin f(\Sigma^*)\}, \\ S_1 &= \{(f \circ g)^k(x) : k \geq 0, x \notin f(\Sigma^*)\}, \\ S_2 &= \{f \circ (g \circ f)^k(x) : k \geq 0, x \notin g(\Sigma^*)\}. \end{aligned}$$

We have $\Sigma^* = R_1 \cup R_2$ and also $\Sigma^* = S_1 \cup S_2$. It is easy to see that ϕ defined by $\phi(z) = f(z)$ if $z \in R_1, g^{-1}(z)$ if $z \in R_2$ is a qlin-time isomorphism between A and B, where $\phi^{-1}(z) = g(z)$ if $z \in S_1, f^{-1}(z)$ if $z \in S_2$. Here ϕ and ϕ^{-1} are qlin-time computable because $f \circ g$ and $g \circ f$ are length-doubling. ∎

[2] We have added a 'D' to their notation 'NLT' for deterministic "nearly-linear time."

Given a language A, define Z to be a *qlin-padding function* for A if Z is one-one and computable in qlin time, and for all x, $Z(x) \in A$ iff $x \in A$.

Lemma 2.2 *Let f be a one-one, qlin-time computable and invertible reduction from A to B. Assume also that either A or B has a qlin padding function Z such that for all x, $|Z(y)| > |y| \log |y| + 1$. Then there is a function $f' : A \to B$ such that f' is one-one, length-doubling, and both computable and invertible in qlin time.*

Proof. Following [BH77], first suppose Z is a qlin padding function for A. Given f as above, let $q(n)$ be a *qlin* time bound in which f and f^{-1} can be computed. Then there is a fixed $r \geq 1$ such that for all x, $|Z^r(x)| > q(2|x|)$, where Z^r means composition of Z for r times. It follows that $|f \circ Z^r(x)| > 2|x|$. The desired f' is $f \circ Z^r$.

Alternatively suppose Z is a qlin padding function for B. Since f is qlin-time computable and invertible, f must be *qlin-honest*, meaning that there is a quasilinear function h such that for all x, $h(|f(x)|) \geq |x|$. Hence there is a quasilinear function q such that for all x, $q(|f(x)|) \geq 2|x|$. Then for some fixed r and all x, $|Z^r(x)| > q(|x|)$. The desired f' is $Z^r \circ f$. ∎

The following condition suffices for the existence of qlin-time invertible reductions.

Lemma 2.3 *Let A be a set for which two qlin-time computable functions $S_A(\cdot,\cdot)$ and $D_A(\cdot)$ exist with the following properties: (a) $(\forall x)(\forall y)[S_A(x,y) \in A \iff x \in A]$. (b) $(\forall x)(\forall y)[D_A(S_A(x,y)) = y]$. Then if f is any qlin-time reduction of some set C to A, then $f'(x) = S_A(f(x), x)$ is one-one and qlin-time computable and invertible and reduces C to A.*

Proof. The proof is exactly the same as that of Lemma 5 in [BH77], using the fact that quasilinear functions are closed under composition. ∎

Theorem 2.4 *Let $A \leq_m^{ql} B$ and $B \leq_m^{ql} A$. Let A have a qlin padding function Z_A satisfying Lemma 2.2 and functions S_A and D_A satisfying Lemma 2.3. Then B is qlin-isomorphic to A iff B has functions S_B and D_B satisfying Lemma 2.3.*

Proof. As in Theorem 7 of [BH77]. ∎

Lemma 2.5 *$3SAT$ has a padding function Z_{3SAT} satisfying Lemma 2.2 and functions S_{3SAT} and D_{3SAT} satisfying Lemma 2.3.*

Proof. We follow Theorem 8 of [BH77]. Let A denote $3SAT$. Given a string w, check if w is a $3SAT$ Boolean formula. If yes, let $x_1, ..., x_r$ be variables appearing in w. If not, let $r = 0$. Let y be a binary string and $y(j)$ be the jth digit of the string y. Let $S_A(w, y) = w \wedge (x_{r+1} \vee \bar{x}_{r+1}) \wedge z_1 \wedge z_2 \wedge \cdots \wedge z_{|y|}$, where z_j is a literal: $z_j = x_{r+1+j}$ if $y(j) = 1$, $z_j = \bar{x}_{r+1+j}$ otherwise. Thus, w is in $3SAT$ iff $S_A(w, y)$ is in $3SAT$. $S_A(\cdot,\cdot)$ is clearly qlin-time computable. Let D_A be the function that examines a string to determine whether it has a suffix of the proper form and if so translates it appropriately. Clearly, $D_A(\cdot)$ is qlin-time computable. Both $S_A(\cdot,\cdot)$ and $D_A(\cdot)$ satisfy Lemma 2.3. The desired Z_A is defined by $Z_A(w) = S_A(w, 0^{|w|\log|w|+1})$, and this satisfies Lemma 2.2. ∎

In consequence,

Theorem 2.6 *An* NQL*-complete set B is qlin-isomorphic to $3SAT$ iff it has two qlin-time com-*

putable functions S_B and D_B satisfying Lemma 2.3. ∎

We note that the alternative formulation of Mahaney and Young [MY85] also carries over to quasilinear time. Say a language B has *qlin-binary padding* if there is a one-one function $q_B(\cdot,\cdot)$, computable and invertible in qlin-time, such that for all $x, y \in \Sigma^*$, $x \in B \iff q_B(x,y) \in B$.

Theorem 2.7 *An* NQL*-complete set B is qlin-isomorphic to $3SAT$ iff B has qlin-binary padding.*

Proof. Let A be $3SAT$ with its qlin-binary padding function q_A, and take q_B as above for B. Let $A \leq_m^{ql} B$ via f, and let $B \leq_m^{ql} A$ via g. Then for all $y \in \Sigma^*$ define:

$$f_1(y) = \begin{cases} q_B(f(y), q_A(q_A(x,z), w0^{|w|})) & \text{if } y = q_A(x, q_A(z,w)) \\ q_B(f(y), q_A(y, y1^{|y|})) & \text{otherwise,} \end{cases}$$

$$g_1(y) = \begin{cases} q_A(g(y), q_A(q_A(x,z), w0^{|w|})) & \text{if } y = q_B(x, q_A(z,w)) \\ q_A(g(y), q_A(y, y1^{|y|})) & \text{otherwise.} \end{cases}$$

(This is well-defined since w, x, and z are implicit and q_A and q_B are one-one. The inner occurrences of q_A can be all changed to q_B without affecting the result.) The only difference from the formulation in [MY85] is our having $w0^{|w|}$ and $y1^{|y|}$ in place of $w0$ and $y1$ (or as written in [MY85], $2w$ and $2y{+}1$). This makes the recursion in [MY85] bottom out in $\log n$ rather than n steps. The remainder of the proof is as in [MY85] and is very similar to the above. ∎

Berman and Hartmanis [BH77] give the impression in their paper of trying to obtain linear-time versions of their results, but there seems to be no way to make the recursion bottom out in linear time, so it is really a quasilinear-time phenomenon.

We observe, with reference also to [Sch78, Dew82, Dew89, SH90], that all of the NP-complete languages used as evidence in [BH77] belong to NQL and are complete under qlin-time many-one reductions, and that the "polynomial time" padding functions in [BH77] all run in qlin-time. The same goes for the great many NP-complete problems in [GJ79] that belong to NQL. Hence they are all qlin-isomorphic. This emboldens us to make the

Conjecture *All* NQL*-complete languages are quasilinear time isomorphic.*

The truth of this conjecture implies NQL $\neq$ DQL, which seems to be just as hard as showing NP $\neq$ P. However, we actually suspect that this should be an Un-Conjecture, since the next section gives hope for refuting it absolutely, even under all relativizations.

3 Quasilinear One-Way Functions and Oracles

We note the following provision about oracle Turing machines M adopted in the standard references [WW86] and [BDG88] (see also [LL76, Wra78]): Whenever M enters its query state with the query string z on its query tape, z is *erased* when the oracle gives its answer. As shown in [NRS94], this makes quasilinear-time Turing reducibility transitive, and yields the theorem that the definition of the *quasilinear time hierarchy* DQL, NQL, $\sum_2^{ql} := \mathrm{NQL}^{\mathrm{NQL}}$, ..., by oracles coincides with the definition via quasilinear length-bounded quantifiers. The hierarchy QLH has the downward separation property (i.e., $\sum_k^{ql} = \prod_k^{ql} \implies \mathrm{QLH} = \sum_k^{ql}$), and the standard SAT-like $\sum_k^p$-complete languages are also complete for $\sum_k^{ql}$ under $\leq_m^{ql}$ reductions. The class UQL may be defined analogous to UP as the class of languages accepted by quasilinear-time NTMs N such that on all inputs x, $N(x)$ has at most one accepting computation. The same oracles $A = QBF$ and B constructed in [BGS75, HU79] to give $\mathrm{NP}^A = \mathrm{P}^A$ and $\mathrm{NP}^B \neq P^B$ also give $\mathrm{NQL}^A = \mathrm{DQL}^A$ and $\mathrm{UQL}^B \neq \mathrm{DQL}^B$ (hence $\mathrm{NQL}^B \neq \mathrm{DQL}^B$). Thus most of the familiar, cozy, well-developed world of polynomial-based complexity carries over to quasilinear time, even under relativizations.

There is, however, a very meaningful exception regarding one-way functions. The "weak" notion of polynomial one-way functions f used in structural complexity is that f is one-one, polynomially honest (meaning that f for some polynomial p and all x, $|f(x)| \geq p^{-1}(|x|)$), and polynomial-time computable, but f^{-1} is not polynomial-time computable. The basic fact is that such functions exist iff $\mathrm{P} \neq \mathrm{UP}$. If $\mathrm{DQL} \neq \mathrm{UQL}$, then taking N to accept a language L in $\mathrm{UQL} \setminus \mathrm{DQL}$, the function $f(x, y) := x0$ if y is a string of nondeterministic moves that makes N accept x, and $f(x, y) := (x, y)1$ otherwise, is one-one, quasilinearly honest, qlin-computable, but not qlin-invertible. That is to say, f is "qlin-one-way" in the above weak sense. However, the converse fails under relativizations—it is possible to have UQL = DQL and yet there exist functions that are "qlin-one-way" in a strong sense closer to how cryptographers use the term:

Theorem 3.1 *There exists an oracle C such that $\mathrm{NQL}^C = \mathrm{UQL}^C = \mathrm{DQL}^C$ (and also $\mathrm{NP}^C = \mathrm{P}^C$), and a linearly-honest one-one function f such that f is computable in linear time with oracle C, but every oracle machine M^C computing f^{-1} requires at least quadratic time on "many" inputs.*

The proof actually shows that the total number of bits in queries to the oracle must be quadratic, and the "query-erase" proviso makes this a lower bound on running time. Our point here is not to justify the query-erase proviso further, but to emphasize that the oracle gives a good explanation of why the *unrelativized* implication "UQL = DQL $\Longrightarrow$ all one-one honest qlin-computable functions are qlin-invertible" may not hold.

Proof. We begin by constructing a length-preserving function g such that no oracle allows g to be computed in less than quadratic time. This is done in [NRS94] by taking, for each n, a Kolmogorov-random string G_n of length $n2^n$. Then the first n bits of G_n define $g(0^n)$, the next n bits define $g(0^{n-1}1)$, and so on, up to the last n bits that define $g(1^n)$. Let D be any oracle set and suppose M is an OTM such that M^D computes g. Then the following determines the string G_n:

1. A string of length $2^n - 1$ that defines D on all strings of length up to n–1.
2. A string of some constant size k (independent of n) that specifies the code of M and "this discussion" (in the style of Li and Vitanyi [LV93]).
3. For each $x \in \{0, 1\}^n$, the string d_x giving the answers to all oracle queries of length $\geq n$ made by $M^D(x)$, *or* the string $g(x)$, whichever is shorter.

By the choice of G_n to be K-random (technically, so that $K(G_n|n) \geq n2^n$), this description must have length at least $n2^n$. Hence nearly all of the strings d_x representing answers to long queries made by M must have length at least n–1. (There is some slack because of the need to insert delimiters between successive strings d_x or $g(x)$ in item 3, costing roughly $2 \log n$ bits per delimiter.) Since each such query takes at least n steps to write and submit, M^D must take at least $n^2 - n$ steps on all these inputs.

Now define A to be the graph of g, namely $A := \{ (x, y) : g(x) = y \}$. And define the function f for all strings $z \in \Sigma^*$ by: if $z = xy$ such that $|x| = |y|$ and $(x, y) \in A$ then $f(z) = x0$, else $f(z) = z1$. Then f is one-one and linearly honest, and f is computable in linear time with oracle A. However, inverting f on an input of the form $x0$ requires computing $g(x)$.

Last, let $C := \{ \langle M, x, 0^n \rangle :$ the Turing machine M accepts x in space n with oracle set $A \}$, where queries count against the space bound. To show $\mathrm{NQL}^C = \mathrm{DQL}^C$, let the oracle NTM N^C run in quasilinear time $q(n)$ and accept some language L. Define a deterministic OTM M_L as follows: M_L uses one tape to cycle through all strings in $\{0,1\}^{q(n)}$ representing nondeterministic moves made by N. M_L has all the tapes of N, including its query tape, and some tapes on which it uses its oracle A to answer queries of the form $\langle M, z, 0^r \rangle$ made by N, by directly simulating of $M^A(z)$. The simulation of M onto, say, two tapes carries a linear space overhead, so each query by N can be answered in quasilinear space. Hence M^A runs in some quasilinear space bound q', and the mapping $x \mapsto \langle M_L, x, 0^{q'(|x|)} \rangle$ many-one reduces L to C in quasilinear time. Hence $L \in \mathrm{DQL}^C$ (with one query, in fact).

Then f is still computable in linear time relative to C. However, because the computation of g requires quadratic time with *any* oracle set, f requires quadratic time to invert, on nearly all inputs of the form $x0$. A last note is that the oracle C can be made recursive via judicious use of *time-bounded* Kolmogorov complexity. ∎

Now we observe how this throws a spanner into the works of papers on relativizations of the original B-H conjecture. The result of Kurtz-Mahaney-Royer [KMR89] that the conjecture fails for random oracle sets R turns upon an analysis of the "Bennett and Gill function"

$$\xi^R(x) = R(x1)R(x10)\cdots R(x10^{3|x|}),$$

where $R(\cdot)$ stands for the characteristic function of R. However, for random R, $\xi^R(x)$ is not computable in quasilinear (or even subquadratic) time relative to R, owing to the query-erase proviso. The attempt to mimic the above proof by defining $f(x,y) := x0$ if $\xi^R(x) = y$ (etc.) has the same problem.

The theorem of Fenner-Fortnow-Kurtz [FFK92] that the conjecture holds relative to every "sp-generic" oracle set A uses the fact that $\mathrm{P}^A = \mathrm{UP}^A \implies$ every one-one length-increasing P-computable function is P-invertible. But the quasilinear analogue of this is falsified by (a length-increasing patch to) the above construction of f. Analogous parts of the paper by Homer and Selman [HS92] on P-inseparable sets also fail to carry over. So we may pose a

Challenge: Can these oracle results be shown for our qlin-isomorphism conjecture?

It is also interesting to consider other quasilinear m-degrees besides the complete one for NQL, and to explore analogues of the results of Ko-Long-Du [KLD86], Kurtz-Mahaney-Royer [KMR88], Fenner-Kurtz-Royer [FKR89], Ganesan [Gan92], or Wang [Wan90, Wan93, Wan94].

4 Conclusions

Isomorphism problems for other reducibilities stronger than polynomial-time have attracted much recent attention, and the results have tended to be *positive*. To wit, all languages that are NP-complete under first-order reductions are first-order isomorphic [ABI93], all complete for NP under one-way log-space reductions are polynomial-time isomorphic [AB93], and related results for PSPACE and NL-complete sets are given by Agrawal [Agr94]. We have stated the problem for a stronger reducibility that preserves most of the original landscape of [BH77], where it may be possible to give an absolute *negative* answer.[3]

[3] A negative answer of a different kind is shown in [WB94]: there are NP-complete problems that are not p-isomorphic with respect to their natural distributions on instances.

In approaching this, we still subscribe to the original idea of Joseph and Young [JY85] that *one-way* functions should be involved in a negative answer. One form of this idea is that if f is polynomial [quasilinear] one-way then $f(SAT)$ is still NP-complete [NQL-complete], but to compute an isomorphism between $f(SAT)$ and SAT may require the power to invert f. The oracle A of [HH91] for which $\mathrm{UP}^A = \mathrm{P}^A$ and the conjecture fails for NP^A is also moot for time *qlin*. Besides, what we're after is not a "weak" one-way function, but something that meets our second

Challenge: Prove that there exist linearly-honest one-one functions f that are computable in qlin-time, but require quadratic time to invert, on "many" inputs.

On the meaning of "many," the function f^{-1} in Theorem 3.1 has a large "hard set," namely strings ending in 0, but it also has a large "easy set" of strings ending in 1. We want the easy set to be "small" in some pertinent sense, such as for average-case complexity or cryptographic hardness. Even with strong notions of "many" and the stricter requirement that f be a length-preserving permutation, we see no way to construct an oracle relative to which such f do *not* exist. So this is an eminently fair challenge. Homer and Wang [HW89] studied sub-polynomial time one-way functions, and gave a combinatorial construction of some promise, but this does not work in qlin-time.

There are indeed candidates for one-way functions that take quasilinear time to compute and are believed to require *exponential* time to invert on "many" inputs. Integer multiplication vs. factoring provides one of them. We draw attention, however, to the lesser requirement that inversion take quadratic time, because on large inputs this might be just as good a notion of intractability. This notion of a one-way function seems natural and theoretically important in its own right. We look forward to further research on consequences of the existence (or nonexistence) of such functions. A more general point, supported by results in [FHOS93, Sel94, NRS94, Pap94], is that functions have complexity-theoretic lives of their own that cannot be captured by studying languages, and we hope the above will provoke attention to this.

References

[AB93] M. Agrawal and S. Biswas. Polynomial isomorphism of 1-L complete sets. In *Proc. 8th Structures*, pages 75–80, 1993.

[ABI93] E. Allender, J. Balcázar, and N. Immerman. A first-order isomorphism theorem. In *Proc. 10th STACS*, volume 665 of *Lect. Notes in Comp. Sci.*, pages 163–174. Springer Verlag, 1993.

[Agr94] M. Agrawal. On the isomorphism problem for weak reductions (extended abstract). In *Proc. 9th Structures*, pages 338–355, 1994.

[BDG88] J. Balcázar, J. Díaz, and J. Gabarró. *Structural Complexity Theory.* Springer Verlag, 1988.

[BGS75] T. Baker, J. Gill, and R. Solovay. relativizations of the P=NP? question. *SIAM J. Comput.*, 4:431–442, 1975.

[BH77] L. Berman and J. Hartmanis. On isomorphisms and density of NP and other complete sets. *SIAM J. Comput.*, 6:305–321, 1977.

[Coo71] S. Cook. The complexity of theorem-proving procedures. In *Proc. 3rd STOC*, pages 151–158, 1971.

[Dew81] A.K. Dewdney. Fast Turing reductions between problems in NP. Technical report, Department of Computer Science, University of Western Ontario, London, Ontario, Canada N6A 5B9, 1981. Reports #68–#76, subtitled as chapters 1–9, the last four in 1982–1984.

[Dew82] A.K. Dewdney. Linear time transformations between combinatorial problems. *Intern. J. Comp. Math.*, 11:91–110, 1982.

[Dew89] A.K. Dewdney. Fast Turing reductions of combinatorial problems and their algorithms. In G. Bloom, R. Graham, and J. Malkevitch, editors, *Combinatorial Mathematics: Proceedings of the Third International Conference*, volume 555 of *Annals of the New York Academy of Sciences*, pages 171–180, 1989.

[FFK92] S. Fenner, L. Fortnow, and S. Kurtz. The Isomorphism Conjecture holds relative to an oracle. In *Proc. 33rd FOCS*, pages 30–39, 1992.

[FHOS93] S. Fenner, S. Homer, M. Ogiwara, and A. Selman. On using oracles that compute values. In *Proc. 10th STACS*, volume 665 of *Lect. Notes in Comp. Sci.*, pages 398–407. Springer Verlag, 1993.

[FKR89] S. Fenner, S. Kurtz, and J. Royer. Every polynomial-time 1-degree collapses iff P = PSPACE. In *Proc. 30th FOCS*, pages 624–689, 1989.

[FP74] M. Fischer and M. Paterson. String matching and other products. In R. Karp, editor, *Complexity of Computation*, volume 7 of *SIAM-AMS Proceedings*, pages 113–125. Amer. Math. Soc., 1974.

[Gan92] K. Ganesan. One-way functions and the isomorphism conjecture. In *Proc. 12th FSTTCS*, pages nnn–nnn, 1992.

[GJ79] M. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman, 1979.

[Gra93] E. Grandjean. Sorting linear time and the satisfiability problem. Technical Report 17, Laboratoire d'Informatique de l/Université de Caen, September 1993.

[Gra94] E. Grandjean. Linear time algorithms and NP-complete problems. *SIAM J. Comput.*, 23:573–597, 1994.

[GS89] Y. Gurevich and S. Shelah. Nearly-linear time. In *Proceedings, Logic at Botik '89*, volume 363 of *Lect. Notes in Comp. Sci.*, pages 108–118. Springer Verlag, 1989.

[HH91] J. Hartmanis and L. Hemachandra. One-way functions and the nonisomorphism of NP-complete sets. *Theor. Comp. Sci.*, 81:155–163, 1991.

[HS92] S. Homer and A. Selman. Oracles for structural properties: The Isomorphism Problem and public-key cryptography. *J. Comp. Sys. Sci.*, 44:287–301, 1992.

[HU79] J. Hopcroft and J. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison–Wesley, Reading, MA, 1979.

[HW89] S. Homer and J. Wang. Absolute results concerning one-way functions and their applications. *Math. Sys. Thy.*, 22:21–35, 1989.

[Joh87] D.S. Johnson. The many faces of polynomial time. *J. Alg*, 8:285–303, 1987.

[JY85] D. Joseph and P. Young. A survey of some recent results on computational complexity in weak theories of arithmetic. *Fundamenta Informatica*, 8:104–121, 1985.

[KLD86] K. Ko, T. Long, and D. Du. A note on one-way functions and polynomial-time isomorphisms. *Theor. Comp. Sci.*, 47:263–276, 1986.

[KMR88] S. Kurtz, S. Mahaney, and J. Royer. Collapsing degrees. *J. Comp. Sys. Sci.*, 37:247–268, 1988.

[KMR89] S. Kurtz, S. Mahaney, and J. Royer. The Isomorphism Conjecture fails relative to a random oracle. In *Proc. 21st STOC*, pages 157–166, 1989.

[LL76] R. Ladner and N. Lynch. Relativization of questions about log-space computability. *Math. Sys. Thy.*, 10:19–32, 1976.

[LV93] M. Li and P. Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*. Springer Verlag, 1993.

[MNT93] Y. Mansour, N. Nisan, and P. Tiwari. The computational complexity of universal hashing. *Theor. Comp. Sci.*, 107:121–133, 1993.

[MY85] S. Mahaney and P. Young. Reductions among polynomial isomorphism types. *Theor. Comp. Sci.*, 39:207–224, 1985.

[NRS94] A. Naik, K. Regan, and D. Sivakumar. Quasilinear time complexity theory. Technical Report UB-CS TR 94-21, Department of Computer Science, University at Buffalo, 1994. An earlier version appeared in the proceedings of STACS'94, LNCS 778, pp 97–108.

[Pap94] C. Papadimitriou. On the complexity of the parity argument, and other inefficient proofs of existence. *J. Comp. Sys. Sci.*, 48:498–532, 1994.

[Sch76] C. Schnorr. The network complexity and the Turing machine complexity of finite functions. *Acta Informatica*, 7:95–107, 1976.

[Sch78] C. Schnorr. Satisfiability is quasilinear complete in NQL. *J. ACM*, 25:136–145, 1978.

[Sel94] A. Selman. A taxonomy of complexity classes of functions. *J. Comp. Sys. Sci.*, 48:357–381, 1994.

[SH90] R. Stearns and H. Hunt III. Power indices and easier hard problems. *Math. Sys. Thy.*, 23:209–225, 1990.

[Wan90] J. Wang. Some remarks on polynomial time isomorphisms. In *Proc. 2nd ICCI*, volume 468 of *Lect. Notes in Comp. Sci.*, pages 144–153. Springer Verlag, 1990.

[Wan93] J. Wang. On the E-isomorphism problem. In J.-Y. Cai, editor, *Advances in Computational Complexity Theory*, pages 195–209. American Mathematical Society, 1993.

[Wan94] J. Wang. Productive functions and isomorphisms, 1994. *Math Sys. Thy.*, in press.

[WB94] J. Wang and J. Belanger. On the NP-isomorphism problem with respect to random instances, 1994. *J. Comp. Sys. Sci.*, to appear.

[Wra78] C. Wrathall. Rudimentary predicates and relative computation. *SIAM J. Comput.*, 7:194–209, 1978.

[WW86] K. Wagner and G. Wechsung. *Computational Complexity*. D. Reidel, 1986.