CENTRE DE RECERCA MATEMATICA INSTITUT D'ESTUDIS CATALANS
Apartat 50, E-08193 Bellaterra

Separation of Complexity classes in Koiran's weak model

F. Cucker †
Universitat Pompeu Fabra
Balmes 132, Barcelona 08008
SPAIN

M. Shub ‡
IBM T. J. W. Research Center
Yorktown Heights, NY 10598
USA

S. Smale ‡
University of California
Berkeley, CA 94720
USA

1. Introduction

Very recently Pascal Koiran introduced in [10] a model of computation that comes from a modification of the cost notion of the real Turing machine of [2]. This new model —that following Koiran will be called *weak*— drops the unit cost assumption for the arithmetical operations and only allows a "moderate use of multiplication" ([11]). The main result of [10] states that when restricted to Boolean inputs the class of sets decided by these machines in polynomial time coincides with P/*poly*. As a consequence, if P = NP in the weak model then the Boolean polynomial hierarchy collapses at the second level.

In the present paper we continue the study of the computational power of the weak model. In particular, several separations between complexity classes for that model are proved, the most important one being $\mathrm{P} \neq \mathrm{NP}$. In fact, it is shown that NP_{W} (the subscript stands for "weak") strictly contains its subclass $\mathrm{NP}_{\mathrm{WD}}$ consisting of those sets that can be decided using binary guesses. Note that since $\mathrm{P}_{\mathrm{W}} \subset \mathrm{NP}_{\mathrm{W}}$ the above mentioned separation holds. The problem of whether $\mathrm{P}_{\mathrm{W}} = \mathrm{NP}_{\mathrm{WD}}$ remains open and we provide two kinds of partial answers to it. On the one hand, in section 3 and following the line of ideas of [10] we show that the above equality would imply the collapse of the polynomial hierarchy at its second level, a fact seen as unlikely in complexity theory. On the other hand, we prove in section 5 that if we restrict our attention to machines that branch only on equality tests, we can prove that the forementioned equality does not hold. This is done by showing that a well known problem (the Knapsack problem) belongs to $\mathrm{NP}_{\mathrm{WD}}$ and can not be solved in deterministic polynomial time. Finally, in section 4, we consider the alternating variation of the weak model, and we give a doubly exponential lower bound for the parallel time needed to decide problems solvable in polynomial alternating time.

2. The weak model

In the following we shall denote the direct sum $\bigoplus_1^\infty \mathbb{R}$ by $\mathbb{R}^\infty$. Also, we define the *size* $|x|$ of an element $x \in \mathbb{R}^\infty$ as the largest i such that its i^{th} coordinate x_i is different from zero. We shall denote by Σ the subset $\{0,1\} \subset \mathbb{R}$ and —following the custom in Complexity theory— by Σ^* the set of all finite strings over Σ. Note that there is a natural inclusion

† Partially supported by DGICyT PB 92/0498/C02/01, and the ESPRIT BRA Program of the EC under contracts no. 7141 and 6546, projects ALCOM II and PROMotion.

‡ Work done at the Centre de Recerca Matemàtica. Supported in part by NSF grants.

$\Sigma^* \hookrightarrow \mathbb{R}^\infty$ and that the membership of a point in $\mathbb{R}^n$ to Σ^* can be algebraically expressed by n equations of the form $X(X-1)=0$.

Also, we shall consider real Turing machines over $\mathbb{R}^\infty$ as they were defined in [2] but in a normal form that requires that every computational node performs a single arithmetic operation. This requirement does not modify the running time of the machine up to a constant factor.

Let M be a real Turing machine whose running time is bounded by $t(n)$, and let $\alpha_1, \ldots, \alpha_k$ be its real constants. For any input size n, the machine M determines an algebraic computation tree $T_{M,n}$ with depth $t(n)$. At an arithmetic node ν of this tree a value is assigned to a variable z corresponding to an arithmetical operation on some previously computed values. This value z can be expressed as $f_\nu(x_1, \ldots, x_n, \alpha_1, \ldots, \alpha_k)$ where f_ν is a rational function with rational coefficients and $(x_1, \ldots, x_n)$ is the input. These rational functions are used to define the running time in the weak model. In the next definition, we shall understand by the height of a rational number p/q its bit length i.e. $\lfloor \log(|p|+1) + \log(|q|) \rfloor$.

Definition 1. The cost of any arithmetic node ν is defined to be the maximum of $\deg(f_\nu)$ and the maximum height of the coefficients of f_ν, while the cost of any other node is 1. For any $x \in \mathbb{R}^\infty$ of size n the *weak running time of M on x* is defined to be the sum of the costs of the nodes along its computational path in $T_{M,n}$. The *(weak) running time of M* is the function associating to every n the maximum over all $x \in \mathbb{R}^\infty$ of size n of the running time of M on x.

The classes $\mathrm{P_W}$ and $\mathrm{NP_W}$ of weak deterministic and nondeterministic polynomial time respectively are now defined as in [2]. Also, we define the class $\mathrm{NP_{WD}}$ of *weak digital nondeterministic polynomial time* by requiring the guesses in $\mathrm{NP_W}$ to be elements in Σ^*. This kind of nondeterminism describes the complexity of discrete search as appears for instance in the Travelling Salesman or the Knapsack problems (see [6]).

In the sequel, unless otherwise stated, all the complexity classes are in the weak model. The adjective *full* as oposed to weak will be applied to the notions as they were introduced in [2].

A first result concerning weak nondeterministic polynomial time is that it coincides with full nondeterministic polynomial time. Consequently, we derive the $\mathrm{NP_W}$-completeness of the full NP-complete problems of [2]. Let us recall that QS is the set of systems of quadratic equations having a real solution, and that 4FEAS is the set of degree 4 polynomials having a real root.

Theorem 2. *We have that* $\mathrm{NP_R} = \mathrm{NP_W}$ *where* $\mathrm{NP_R}$ *is the class of sets decided in full nondeterministic polynomial time.*

Proof: We first observe that the reductions given in [2] to reduce any problem in $\mathrm{NP_R}$ to 4FEAS, work in weak polynomial time. This can be seen either checking the weakness at the proof given in [2] or realizing that the quoted reductions does not use non integer constants and seen as a Boolean algorithm (dealing with the input x and the machine

constants as symbols) it is performed in polynomial time and thus, according with lemma 3 of [10], that it works in weak polynomial time.

Now, since 4FEAS can be trivially solved in weak nondeterministic polynomial time, we have an $\mathrm{NP_W}$ algorithm for solving all problems in $\mathrm{NP}_{\mathbf{R}}$ by composing for any $S \in \mathrm{NP}_{\mathbf{R}}$ the reduction to 4FEAS with the algorithm for solving this last problem. ∎

A side consequence of this last proof is the following result.

Theorem 3. *The sets* QS *and* 4FEAS *are* $\mathrm{NP_W}$*-complete for reductions in* $\mathrm{P_W}$. ∎

Let us introduce now a parallel computational model.

Definition 4. An *algebraic circuit* $\mathcal{C}$ over $\mathbb{R}$ is a directed acyclic graph where each node has indegree 0,1 or 2. Nodes with indegree 0 are either labeled as input or with elements of $\mathbb{R}$ (we shall call the last ones constant nodes). Nodes with indegree 2 are labeled with the arithmetic operations of $\mathbb{R}$, i.e. "+", "·", "−" and "/". Finally, nodes with indegree 1 are of a unique kind and are called sign nodes. There is a set of $m \geq 1$ nodes with outdegree 0 called output nodes. In the sequel the nodes of a circuit will be called *gates*.

To each gate we inductively associate a function of the input variables in the usual way (note that sign gates return 1 if their input is greater or equal to 0, and 0 otherwise). In particular, we shall refer to the function associated to the output gates as the function computed by the circuit.

For an arithmetic circuit $\mathcal{C}$, the *size* $s(\mathcal{C})$ of $\mathcal{C}$, is the number of gates in $\mathcal{C}$. The *depth* $d(\mathcal{C})$ of $\mathcal{C}$, is the length of the longest path from some input gate to some output gate. The *cost* of an arithmetic gate is defined as before and the cost of a path in the circuit is the sum of the costs of their gates. We define the *weak running time* of a circuit on an input x to be the maximum of the costs of their paths. The weak running time of the circuit is defined then as before.

Given an algebraic circuit $\mathcal{C}$, the *canonical encoding* of $\mathcal{C}$ is a sequence of 4-tuples of the form $(g, op, g_l, g_r) \in \mathbb{R}^4$ where g represents the gate label, op is the operation performed by the gate, g_l is the gate which provides the left input to g and g_r its right input. By convention g_l and g_r are 0 if gate g is an input gate, and g_r is 0 if gate g is a sign gate (whose input is then given by g_l) or a constant gate (the associated constant being then stored in g_l). Also, we shall suppose that the first n gates are the input gates and the last m the output gates.

Let $\{\mathcal{C}_n\}_{n\in\mathbf{N}}$ be a family of circuits. We shall say that the family is P_{W}*–uniform* if there exists a real Turing machine M that generates the i^{th} coordinate of the encoding of $\mathcal{C}_n$ with input $(i, \overbrace{1, \ldots, 1}^{n-1})$ in weak polynomial time in n. We shall say that the family is EXP_{W}*–uniform* when there is a real Turing machine M as above but working in time weak exponential in n. We now define $\mathrm{PAR_W}$ to be the class of sets S such that there is a $\mathrm{P_W}$–uniform family of circuits $\{\mathcal{C}_n\}$ having size exponential in n and weak polynomial running time such that the circuit $\mathcal{C}_n$ computes the characteristic function of the set of

elements in S with size n. The class $\mathrm{PEXP}_{\mathrm{W}}$ of sets decided in weak exponential parallel time is defined in an analogous manner.

The next proposition is a weak model version of the main theorem in [4].

Proposition 5. *Let $f_n \in \mathbb{R}[X_1, \ldots, X_n]$ be a family of irreducible polynomials such that for all n the zero set $\mathcal{Z}(f_n)$ is a variety of dimension $n-1$ and $\deg(f_n) \geq d(n)$. Then, any family of circuits deciding the set $S = \{x \in \mathbb{R}^\infty \mid f_{|x|}(x) = 0\}$ has a weak running time greater than $d(n)$.*

Proof: Let us assume that there exists a family of circuits $\mathcal{C}_n$ having running time bounded by $r(n)$ and deciding S.

For each n we consider the size N of $\mathcal{C}_n$ and we call "configuration" any point in $\mathbb{R}^N$ and "initial configuration" the point $(x_1, \ldots, x_n, \overbrace{0, \ldots, 0}^{N-n})$.

At each step of the computation we modify some of the coordinates of the current configuration replacing them by the result of operating (via one of $(+, -, *, /)$) on two other coordinates. Those modifications can depend on Boolean conditions of the form

$$Q_i(x_1, \ldots, x_n) \geq 0$$

where $Q_i(X_1, \ldots, X_n)$ is a rational function (whose coefficients depend on the output of previous sign gates, and therefore on the actual input $x_1, \ldots, x_n$) and $Q_i(x_1, \ldots, x_n)$ is the content of coordinate i in $\mathbb{R}^N$.

At the end of the computation the N^{th} coordinate of the final configuration will be 0 or 1 according with the truth of a large (but finite) system of the form

$$\bigvee_{j=1}^{r} \Big(\bigwedge_{i=1}^{s_j} Q_{j,i}(X_1, \ldots, X_n) \leq 0 \wedge \bigwedge_{i=s_j+1}^{t_j} Q_{i,j}(X_1, \ldots, X_n) < 0 \Big)$$

where the degrees of the numerator and denominator of the $Q_{i,j}$ are bounded by $r(n)$.

By expressing the sign of a quotient in terms of the signs of numerator and denominator we can replace the rational functions by polynomials with the same bound for the degrees. Also, expressing an inequality like

$$F(X_1, \ldots, X_n) \geq 0$$

as the disjunction

$$F(X_1, \ldots, X_n) = 0 \vee F(X_1, \ldots, X_n) > 0$$

and then distributing, we can describe S as a union of sets given by systems of polynomial inequalities of the form

$$\bigwedge_{i=1}^{s} F_i(X_1, \ldots, X_n) = 0 \wedge \bigwedge_{j=1}^{t} G_j(X_1, \ldots, X_n) > 0$$

Now, since the zero set $\mathcal{Z}(f_n)$ has dimension $n-1$, one of those sets must contain a subset of dimension $n-1$. Since the set described by the G_j's is open, it must be non–empty, and then it defines an open subset of $\mathbb{R}^n$. But our zero set has dimension $n-1$, and therefore we must have $s > 0$.

Finally, all the polynomials F_i, $i = 1, \ldots, s$, vanish on that $n-1$–dimensional subset of the variety. But, since the variety is irreducible, this implies that every F_i must vanish on the whole variety. Using the fact that the ideal (f_n) is the definition ideal of $\mathcal{Z}(f_n)$ (see [3] théorème 4.5.1) we conclude that all the F_i are multiples of f_n and thus, that their degrees are greater than $d(n)$. Since these degrees are a lower bound of the running time of the circuit $\mathcal{C}_n$ we deduce the proposition's statement. ∎

Theorem 6.
a) $\mathrm{P}_{\mathbf{R}} \not\subset \mathrm{PAR}_{\mathrm{W}}$.
b) $\mathrm{NP}_{\mathrm{W}} \not\subset \mathrm{PAR}_{\mathrm{W}}$.

Proof: *a)* Let us consider the set $S = \{x \in \mathbb{R}^\infty \mid x_1^{2^n} = x_2 \text{ where } n = |x|\}$. Note that for each n the subset of elements of S having size n is an irreducible variety of dimension $n-1$. Thus, S can not be decided in weak polynomial parallel time because of the preceding proposition. On the other hand, it clearly belongs to $\mathrm{P}_{\mathbf{R}}$.

b) Trivial since $\mathrm{P}_{\mathbf{R}} \subseteq \mathrm{NP}_{\mathbf{R}} = \mathrm{NP}_{\mathrm{W}}$. Note that it can also be shown observing that the following sentence

$$\exists y_1 \exists y_2 \ldots \exists y_{n-1}\ x_1^2 = y_1 \ \wedge\ y_1^2 = y_2 \ \wedge\ \ldots \ \wedge\ y_{n-1}^2 = x_2$$

is equivalent to $x_1^{2^n} = x_2$ and can be checked in weak nondeterministic polynomial time. ∎

Corollary 7. *The inclusion* $\mathrm{NP}_{\mathrm{WD}} \subset \mathrm{NP}_{\mathrm{W}}$ *is strict.*

Proof: Trivial since $\mathrm{NP}_{\mathrm{WD}} \subseteq \mathrm{PAR}_{\mathrm{W}}$ (the parallel machine just tests the exponential number of possible guesses independently). ∎

Corollary 8. *The problems* QS *and* 4FEAS *cannot be solved in weak polynomial time even allowing parallelism or digital nondeterminism.* ∎

Theorem 6 provides a result quite unusual in complexity theory since either in the Boolean setting or in the full real setting the class NP is included in its corresponding PAR. We can prove however that in the weak model nondeterministic polynomial time can be solved in deterministic exponential time.

Lemma 9. *If a set* $S \subset \mathbb{R}^\infty$ *can be decided in (full) parallel time* $t(n)$ *then it can be decided in weak deterministic time* $2^{O(t(n))}$.

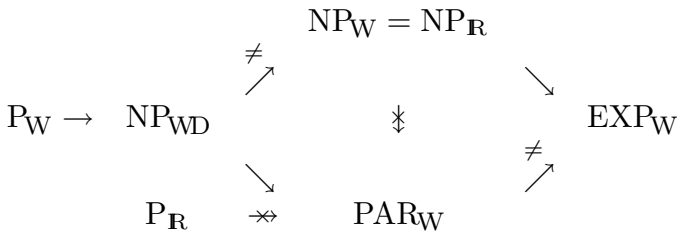
Proof: The weak machine simply simulates the parallel one. This takes full time $2^{O(t(n))}$ and, since the degree and coefficient length of the rational functions associated to the circuits are bounded by $2^{t(n)}$ the result follows. ∎

Proposition 10. $\mathrm{NP_W} \subseteq \mathrm{EXP_W}$.

Proof: In part I of [14] it is shown that 4FEAS can be solved in parallel polynomial time in the full model. Thus, it can be solved in exponential time in the weak one. ∎

Corollary 11. *The inclusion* $\mathrm{PAR_W} \subset \mathrm{EXP_W}$ *is strict.* ∎

The preceding results can be summarized in the following picture



where an arrow $\rightarrow$ means inclusion, an arrow $\stackrel{\neq}{\rightarrow}$ means strict inclusion and a crossed arrow $\nrightarrow$ means that the inclusion between the corresponding complexity classes does not hold.

Theorem 2 asserts that the classes $\mathrm{NP_W}$ and $\mathrm{NP_{\mathbf{R}}}$ coincide. This is not necessarilly the case for their subclasses $\mathrm{NPC_W}$ and $\mathrm{NPC_{\mathbf{R}}}$ of complete problems, since in the first case the reductions considered is in $\mathrm{P_W}$ and in the second in $\mathrm{P_{\mathbf{R}}}$. In fact, it is trivial that $\mathrm{NPC_W} \subseteq \mathrm{NPC_{\mathbf{R}}}$. The converse, however, seems less trivial to prove according to the consequences that it has.

Theorem 12. *If* $\mathrm{NPC_W} = \mathrm{NPC_{\mathbf{R}}}$ *then* $\mathrm{P_{\mathbf{R}}} \neq \mathrm{NP_{\mathbf{R}}}$.

Proof: Let us suppose that $\mathrm{P_{\mathbf{R}}} = \mathrm{NP_{\mathbf{R}}}$. Then we have that $\mathrm{NP_{\mathbf{R}}} = \mathrm{NPC_{\mathbf{R}}}$ and in particular, that $\mathrm{NP_{WD}} \subset \mathrm{NPC_{\mathbf{R}}}$. By hypothesis this entails that $\mathrm{NP_{WD}} \subseteq \mathrm{NPC_W}$ and thus that $\mathrm{NP_{WD}} = \mathrm{NP_W}$, contradicting corollary 7. ∎

3. Weak machines and Boolean complexity classes

Definition 13. Given a class $\mathcal{C}$ of subsets of $\mathbb{R}^\infty$, we shall call its *Boolean part* the class of subsets of Σ^* obtained by considering for any $S \in \mathcal{C}$ its subset of elements belonging to Σ^*.

One of the main results in [10] states that the Boolean part of P_W is P/*poly*. This result was used then to show that if $\mathrm{P}_\mathrm{W} = \mathrm{NP}_\mathrm{W}$ then the polynomial hierarchy of Meyer and Stockmeyer collapses, a consequence that now becomes meaningless since we know that $\mathrm{P}_\mathrm{W} \neq \mathrm{NP}_\mathrm{W}$. The main ideas of the paper (and the techniques used there) remain however very interesting since, as we shall see, they can still be fruitfully used. We begin by recalling the main technical tool obtained in [10].

Theorem 14. *Let $S \subset \mathbb{R}^k$ be a semialgebraic set defined by a system*

$$P_i(X_1, \ldots, X_k) > 0, \qquad i = 1, \ldots, N$$

with $P_i \in \mathbb{Z}[X_1, \ldots, X_k]$, and let D be the maximum degree of the P_i's and H the maximum height of their coefficients. If $S \neq \emptyset$, there exists a rational point $x \in \mathbb{Q}^k$ belonging to S and having height bounded by aHD^b with a and b depending only on k.

Theorem 15. *The Boolean part of* PAR_W *is* PSPACE/*poly.*

Proof: Let $S \in \mathrm{PAR}_\mathrm{W}$ and let us consider its subset $\tilde{S}$ of elements in Σ^*. We shall see that $\tilde{S}$ belongs to PSPACE/*poly*.

Since S belongs to PAR_W, there is a family of algebraic circuits $\{\mathcal{C}_n\}$ having weak running time bounded by a polynomial $q(n)$. Moreover there is a real Turing machine M that given (n, i) produces the i^{th} gate of $\mathcal{C}_n$ within a time that we can suppose to be also bounded by $q(n)$. Let the size of $\mathcal{C}_n$ be bounded by $s(n) = 2^{n^q}$ and $\alpha_1, \ldots, \alpha_k$ be the constants of M.

With the exception of the constant value for the constant gates, all the remaining values computed by M are positive integers with polynomial height. Without loss of generality we will suppose that M first produces a base two representation of these numbers and then —without using the real constants $\alpha_1, \ldots, \alpha_k$— computes the corresponding integers. This property allows us to suppose that the integer value returned at the end of a computational path only depend on the path itself and not on the constants $\alpha_1, \ldots, \alpha_k$.

On the other hand the constant gates depend on $\alpha_1, \ldots, \alpha_k$ and their associated constant $\gamma_{i,n}$ can then be expressed as $r_{n,i}(\alpha_1, \ldots, \alpha_k)$ where $r_{n,i}$ is a rational function having polynomial degree and coefficient heights since the weak running time of M is polynomially bounded. Also, let

$$\begin{cases} h_{n,i,j}(\alpha_1, \ldots, \alpha_k) \geq 0 \\ r_{n,i,j}(\alpha_1, \ldots, \alpha_k) < 0 \quad j = 1, \ldots, h_n \end{cases}$$

the rational functions that determine the computation path followed by M on input (n,i) and let $\Upsilon_{n,i}$ be the system of inequations resulting by replacing the $\alpha_1,\ldots,\alpha_k$ by the indeterminates $X_1,\ldots,X_k$.

For any n, and for any element $u\in\Sigma^n$, the computation done by $\mathcal{C}_n$ on input u can be described by a set of inequations

$$\Xi_{n,u}=\begin{cases} f_{n,u,i}(\gamma_{n,1},\ldots,\gamma_{n,i_n})\geq 0\\ g_{n,u,i}(\gamma_{n,1},\ldots,\gamma_{n,i_n})<0\end{cases}$$

where the $f_{n,u,i}$ and the $g_{n,u,i}$ are rational functions having polynomial degree and coefficient heights (because of the polynomial weak running time of $\mathcal{C}_n$) and the third subindex runs over the sign gates of $\mathcal{C}_n$.

Let us replace in $\Xi_{n,u}$ each ocurrence of a $\gamma_{n,i}$ by its correspondent rational function $r_{n,i}(X_1,\ldots,X_k)$ and let $\zeta_{n,u}$ be the resulting system of inequations. If we now define

$$\Xi_n=(\cup_{i\leq s(n)}\Upsilon_{n,i})\cup(\cup_{u\in\Sigma^n}\zeta_{n,u})$$

we obtain a system of inequations that has the real solution $(\alpha_1,\ldots,\alpha_k)$.

In order to apply the preceding theorem to ensure the existence of a small rational solution we use Koiran's trick to get rid of the equalities (see section 5.2. of [10]). We obtain then a new system $\tilde{\Xi}_n$ having only strict inequalities and such that if a point $r=(r_1,\ldots,r_k)$ is a solution of the system the machine M^r obtained by replacing α_i by r_i produces a circuit $\mathcal{C}_n^r$ whose outcome for any $u\in\Sigma^n$ is the same as that of $\mathcal{C}_n$.

We can now deduce the existence of a point $r=(r_1,\ldots,r_k)\in\mathbb{Q}^k$ satisfying the system Ξ_n such that each component has height polynomial in n. Therefore the computations done by M^r over binary inputs can be carried out by a Turing machine in polynomial time (see lemma 3 of [10]). On the other hand the circuit $\mathcal{C}_n^r$ can be readily transformed into a Boolean circuit having polynomial depth.

¿From the classical equivalence between parallel polynomial time and PSPACE in the Boolean setting, we deduce that $\tilde{S}$ belongs to PSPACE/*poly*.

On the other hand, and using the same equivalence, one trivially shows that any set in PSPACE/*poly* can be accepted by a $\mathrm{P_W}$–uniform family of circuits in weak parallel time.

■

Theorem 16. *i) The Boolean part of* $\mathrm{NP_{WD}}$ *is* NP*/poly.*
ii) The Boolean part of $\mathrm{NP_{WD}}\cap\text{co-}\mathrm{NP_{WD}}$ *is* $(\mathrm{NP}\cap\text{co-NP})$*/poly.*

Proof: They are done in a similar manner as the preceding one. ■

Some consequences follow from the preceding theorems. In order to state them, let us recall that we denote by PH the polynomial hierarchy of Meyer and Stockmeyer and by Σ_k^{P} its k^{th} level for any $k\in\mathbb{N}$ (see [13] and ch. 8 of [1]).

Corollary 17. *i) If* $\mathrm{P_W} = \mathrm{NP_{WD}}$ *then the polynomial hierarchy collapses at its second level.*

ii) If $\mathrm{NP_{WD}} = \mathrm{PAR_W}$ *then* $\mathrm{PSPACE} = \Sigma_2^{\mathrm{P}}$.

Proof: If $\mathrm{P_W} = \mathrm{NP_{WD}}$ then we have that $\mathrm{P}/poly = \mathrm{NP}/poly$, from where we deduce (see [9] th. 6.1) the first statement.

For the second statement we use that if $\mathrm{NP_{WD}} = \mathrm{PAR_W}$ then, since $\mathrm{PAR_W}$ is closed under complements, we must indeed have that $(\mathrm{NP_{WD}} \cap \text{co-}\mathrm{NP_{WD}}) = \mathrm{PAR_W}$. This entails on the one hand that $\mathrm{PSPACE}/poly \subseteq \mathrm{NP}/poly$ and thus, because of a slightly modified version of [9] th. 4.2 (that can be found in [15] Cor. 4.29) that $\mathrm{PSPACE} = \Sigma_3^{\mathrm{P}}$. But on the other hand our assumption implies that $\mathrm{NP} \subseteq (\mathrm{NP} \cap \text{co-NP})/poly$ and thus, because of [8] 4.9, that $\mathrm{PH} = \Sigma_2^{\mathrm{P}}$. From both equalities we conclude the desired result. ∎

4. The power of alternation

A common computational resource in Complexity Theory is alternation. It constitutes a strengthening of nondeterminism in the sense that the machine can now alternate existential —i.e. nondeterministic— guesses with universal ones. It is then no surprising that the complete problems for polynomial alternating time generalize the NP–complete problems in a very precise way. Thus, while in the Boolean setting the classical NP–complete problem —SAT— can be seen as the decision of the existential theory of Boolean logic, the most well known complete problem for polynomial alternating time turns out to be the decision of the unrestricted Boolean logic. A similar situation holds in the real setting (see [5]).

In this section we shall see that there is a doubly exponential lower bound in the parallel time needed to solve some problems solvable in polynomial alternating time.

Definition 18. We shall say that a set S is accepted in *Polynomial Alternating Time* if there exists a polynomial p and a machine M such that for every $y \in \mathbb{R}^\infty$, $y \in S$ iff

$$\exists x_1 \forall z_1 \ldots \exists x_{p(|y|)} \forall z_{p(|y|)}\ M \text{ accepts } (y, x_1, z_1, \ldots, x_{p(|y|)}, z_{p(|y|)}) \text{ in time } p(|y|)$$

and we shall denote this fact by $S \in \mathrm{PAT_W}$.

A variation of an argument already used in [16] and in [7] toghether with proposition 5 allows us to prove the following result.

Theorem 19. $\mathrm{PAT}_{\mathrm{W}} \not\subset \mathrm{PEXP}_{\mathrm{W}}$.

Proof: Let us consider the set $S = \{x \in \mathbb{R}^\infty \mid x_1^{2^{2^n}} = x_2 \text{ where } n = |x|\}$. Because of proposition 5 this set is not in $\mathrm{PEXP}_{\mathrm{W}}$. On the other hand, for any $n \in \mathbb{N}$ we consider the formula $\Phi_n(t, z)$ inductively defined as follows

$$\Phi_0(t, z) \equiv z = t^2$$
$$\Phi_m(t, z) \equiv \exists y \forall v \forall w[(t = v \wedge w = y) \vee (v = y \wedge w = z) \Rightarrow \Phi_{m-1}(w, v)]$$

When expanded, $\Phi_n(t, z)$ is a formula whose length is polynomial (in fact linear) in n and logically equivalent to

$$z = t^{2^{2^n}}$$

Thus, S can be accepted in polynomial alternating time by a machine that with input $x_1, \ldots, x_n$ checks the validity of $\Phi_n(x_1, x_2)$. ∎

Note that the above theorem gives a doubly exponential lower bound on the weak parallel time needed to decide the elementary theory of the reals.

5. The unordered case and the Knapsack problem

In [12] it is shown that nondeterministic polynomial time is strictly more powerful than deterministic polynomial time for real Turing machines that only perform scalar multiplications and only branch on equalities. In [11] this result is improved by showing that the Knapsack problem can be solved in nondeterministic polynomial time but not in deterministic polynomial time by these kind of machines. In this section we further extend this last result to weak machines with the same kind of branching.

In the rest of this section all the real Turing machines branch according with tests of the kind $x = 0$. Let us recall from [2] that the real Knapsack problem is defined to be the set

$$\mathrm{KP} = \{x \in \mathbb{R}^\infty \mid \exists u_1, \ldots, u_n \in \Sigma \text{ s.t. } \sum_{i=1}^{n} u_i x_i = 1 \text{ where } n = |x|\}$$

Theorem 20. *The Knapsack problem cannot be solved in weak polynomial time.*

Proof: Let M be a machine solving KP in time $t(n)$. For any n we consider the polynomial

$$H(X_1, \ldots, X_n) = \prod_{(b_1, \ldots, b_n) \in \Sigma^n} (b_1 X_1 + \ldots + b_n X_n - 1)$$

that has degree 2^n. Clearly, for any $(x_1, \ldots, x_n)$ we have that $(x_1, \ldots, x_n) \in \mathrm{KP}$ iff $H(x_1, \ldots, x_n) = 0$.

Now, for any n we consider the algebraic computation tree $T_{M,n}$ and its canonical path, which is obtained by answering $\neq$ at all the branching nodes ν. Moreover, let us consider

the rational function $f_\nu(X_1, \ldots, X_n, \alpha_1, \ldots, \alpha_k)$ in the variables $X_1, \ldots, X_n$ associated to each one of them and let F be their product.

The set of points following the canonical path is a n dimensional subset of $\mathbb{R}^n$. Thus, since the set of points in $\mathbb{R}^n$ satisfying KP has dimension $n-1$, we deduce that its corresponding leaf must be labeled `REJECT`. Thus, if a point $x \in \mathbb{R}^n$ is in KP then we have that $F(x) = 0$ i.e. the rational function F vanishes on the zero set of H. But this implies that the degree of F must be bigger than the degree of H, and from the weakness of M we deduce that $t(n)^2 \geq 2^n$. ∎

If we denote by $\mathrm{P}^{=}_{\mathrm{W}}$ and $\mathrm{NP}^{=}_{\mathrm{WD}}$ the classes of weak deterministic and digital nondeterministic polynomial time for real Turing machines that branch on equalities, our last result separates $\mathrm{P}^{=}_{\mathrm{W}}$ from $\mathrm{NP}^{=}_{\mathrm{WD}}$ since KP is certainly in $\mathrm{NP}^{=}_{\mathrm{WD}}$.

Corollary 21. $\mathrm{P}^{=}_{\mathrm{W}} \neq \mathrm{NP}^{=}_{\mathrm{WD}}$. ∎

It is an open problem whether this separation holds for machines with arbitrary branching.

References.

[1] J.L. Balcázar, J. Díaz and J. Gabarró; *Structural Complexity I.* EATCS Monographs of Theoretical Computer Science, n. 11, Springer Verlag, 1988.

[2] L. Blum, M. Shub and S. Smale; "On a theory of computation and complexity over the real numbers: *NP*-completeness, recursive functions and universal machines". *Bulletin of the Amer. Math. Soc..*, vol.**21**, n.**1**, pp.1-46, 1989.

[3] J. Bochnak, M. Coste and M.-F. Roy; *Géométrie algébrique réelle.* Ergebnisse der Math., **12**, Springer Verlag, 1987.

[4] F. Cucker; "$\mathrm{P}_{\mathbb{R}} \neq \mathrm{NC}_{\mathbb{R}}$", *J. of Complexity*, **8**, pp.230-238, 1992.

[5] F. Cucker; "On the complexity of quantifier eleimination: the structural approach", *The Computer Journal*, **36**, pp. 400-408, 1993.

[6] F. Cucker and M. Matamala; "On Digital Nondeterminism". Preprint, 1993.

[7] J.H. Davenport and J. Heintz; "Real Quantifier Elimination is Doubly Exponential", *J. of Symb. Comp.*, **5**, pp. 29–35, 1988.

[8] J. Kämper; "Nonuniform proof systems: a new framework to describe nonuniform and probabilistic classes", *Theoretical Computer Science*, **85**, pp. 305-331, 1991. 1988.

[9] R. M. Karp and R. J. Lipton; "Turing machines that take advice", *Enseign. Math.*, **28**, pp. 191-209, 1982. 1988.

[10] P. Koiran; "A weak version of the Blum, Shub & Smale model", *34*$^{\mathrm{th}}$ *Found. of Comp. Sc*, pp.486-495, 1993.

[11] P. Koiran; "Computing over the reals with addition and order", Preprint, 1993.

[12] K. Meer; "A note on a P $\neq$ NP result for a restricted class of real machines", *J. of Complexity*, **8**, pp.451-453, 1992.

[13] A. Meyer and L. Stockmeyer; "The equivalence problem for regular expressions with squaring requires exponential time", *13*$^{\text{th}}$ *Symp. on Switching and Automata Theory*, pp. 125-129, 1973.

[14] J. Renegar; "On the computational complexity and geometry of the first order theory of the reals", parts I, II and III. *J. of Symb. Comp.*, **13**, pp.255-352, 1992.

[15] U. Schöning; *Complexity and Structure*, Lect. Notes. in Computer Science, n. 211, Springer Verlag, 1986.

[16] L. Stockmeyer and A. Meyer; "Word problems requiring exponential time", *5*$^{\text{th}}$ *Symp. on Theory of Computing*, pp. 1-9, 1973.