A FASTER PSPACE ALGORITHM FOR DECIDING THE EXISTENTIAL THEORY OF THE REALS

James Renegar

School of Operations Research and Industrial Engineering
Cornell University, Upson Hall
Ithaca, New York 14853–7501

1. INTRODUCTION.

The decision problem for the existential theory of the reals is the problem of deciding if the set $\{x \in \mathbb{R}^n;\ P(x)\}$ is non–empty, where $P(x)$ is a predicate which is a boolean function of atomic predicates either of the form $f_i(x) \geq 0$ or $f_j(x) > 0$, the f's being real polynomials. The purpose of this paper is to present an algorithm for deciding the existential theory of the reals that simultaneously achieves the best known time and space bounds. The time bound for the algorithm is slightly better than any previous bound.

The best space bound for the existential theory of the reals is due to Canny [3], who recently developed a PSPACE algorithm for the problem. Our algorithm is similar to Canny's. However, Canny's algorithm requires exponential time even when the number of variables n is fixed, unlike the double exponential space algorithm of Collins [4], and the single exponential space algorithms of Ben–Or, Kozen and Reif [1], Grigor'ev and Vorobjov [7] and others. The time required by Canny's algorithm is exponential in the number of atomic predicates as well as in n.

Our algorithm requires only polynomial time if n is fixed, and always requires only polynomial space.

The previous best time bound for the existential theory of the reals was due to Grigor'ev and Vorobjov [7]. Although it requires exponential space, the time required by their algorithm is $L^{O(1)}(md)^{O(n^2)}$ where m is the number of atomic predicates, d is their maximal degree and L is the maximal bit length of the predicate coefficients. Our bound is $L^{O(1)}(md)^{O(n)}$.

Ben–Or, Kozen and Reif [1] presented an algorithm showing that the existential theory of the reals is in NC if n is fixed. (Their algorithm was also the first known single exponential space algorithm for the problem.) We build on their work. Consequently, if implemented in parallel, our algorithm also shows that the existential theory is in NC if n is fixed.

The above results refer to the usual Turing machine model of computation which requires the data to be rational. In contrast, one can think of an algebraic RAM requiring only real number data and allowing as operands the usual field operations $+, -, \cdot, \div$ as well as $\geq$ for branching decisions (cf. the paper by Blum, Shub and Smale in these proceedings).

Research partially supported by NSF Grant DMS–8800835.

Algorithms are often defined or guaranteed to work only if the data is rational, even when the problem is naturally stated over the reals. From a mathematical viewpoint, allowing real data is central to the decision problem for the existential theory of the reals. The original decision algorithm due to Tarski [12] certainly allows real data; so does the algorithm of Ben–Or, Kozen and Reif, that of Canny, and ours. The algorithms of Collins and of Grigor'ev and Vorobjov require rational data, the number of field operations (not just bit operations!) tending to infinity as the length of the data does, even when n, m, and d are fixed.

For real data, our algorithm establishes a new record for the number of field operations sufficient to solve the decision problem for the existential theory of the reals. The bound is $(md)^{O(n)}$.

2. PRELIMINARIES.

In the analysis, we will assume that the reader is somewhat familiar with the notion of the "u–resultant" of a system of $n+1$ homogeneous polynomials in n variables. The facts about the u–resultant that are pertinent to our approach are stated, for example, in section 2 of Renegar [11].

Again, the decision problem for the existential theory of the reals is the problem of deciding if the set $\{x \in \mathbb{R}^n;\ P(x)\}$ is non–empty, where $P(x)$ is a predicate which is a boolean function of atomic predicates either of the form $f_i(x) \geq 0$ or $f_j(x) > 0$, the f's being real polynomials. Assuming the atomic predicates are $f_1 \geq 0, \ldots, f_k \geq 0, f_{k+1} > 0, \ldots, f_m > 0$, a *sign assignment* is a vector $v \in \{-1, 0, 1\}^m$ and a *consistent* sign assignment is a sign assignment v for which there exists $x \in \mathbb{R}^n$ such that v_i and $f_i(x)$ have the same sign for all i (considering 0 itself as a sign). Given a consistent sign assignment, then it can be determined in polylog time using polynomial many processors if the points x with that sign assignment satisfy the predicate $P(x)$.

As shown in Lemma 1 of Grigor'ev [6], if we have an algorithm for determining if any sign vector is consistent, then the consistent sign vectors can be determined by $(md)^{O(n)}$ calls to that algorithm, where d is the maximum degree of the atomic predicates. The algorithm used to prove this is recursive. Once the

0272-5428/88/0000/0291$01.00 © 1988 IEEE

consistent sign vectors for $f_1, \ldots, f_j$ are known, the values 0, 1, -1 are alternately appended to these vectors and then consistency with respect to $f_1, \ldots f_{j+1}$ is checked. The $(md)^{O(n)}$ bound on calls (and hence on the number of consistent sign vectors) is proven via Milnor [10] and Heintz [8]. Alternatively, with an eye on NC, we can proceed recursively, splitting the set of atomic predicates into two disjoint subsets of approximately equal size (then splitting these subsets, etc.), determining the consistent sign assignments for the two subsets, catenating these, and from the resulting sign assignments determining the consistent sign assignments of the initial set of atomic predicates. The arguments of Lemma 1 of Grigor'ev [6] easily show that at any step during this procedure, only $(md)^{O(n)}$ sign vectors will be considered as potentially consistent sign vectors.

Henceforth, we only concern ourselves with the problem of determining if a given sign vector is consistent; that is, we are concerned with the problem of determining if a system of polynomial inequalities

$$f_1, \ldots, f_k \geq 0 \quad f_{k+1}, \ldots, f_m > 0$$

is feasible.

The other preliminary remark we need to make concerns the algorithm of Ben–Or, Kozen and Reif [1]. We will call on the BKR algorithm, as does Canny [3]. We rely on the following.

THEOREM (Ben–Or, Kozen and Reif): The BKR algorithm can be used to decide the theory of real closed fields in exponential space or exponential time. In fixed dimension, it is an NC algorithm. □

We will rely on the BKR algorithm only for $n = 2$. Hence, for our application, it is an NC algorithm. Moreover, the space required by the sequential BKR algorithm applied to determining the feasibility of a system of polynomials is polylogarithmic in their maximal degree and coefficient size.

3. AN OPTIMIZATION PROBLEM.

In this section we present a proposition whose proof motivates the algorithm through consideration of a certain optimization problem. Canny's approach also involves an optimization problem, but requires that a new variable be added for each of the constraints. The new variables are used to "regularize" the problem. By focusing on a different optimization problem that requires only a few "regularizing" variables, we reduce the required time.

Assume that we wish to determine if the following system is feasible:

$$f_1, \ldots, f_k \geq 0 \quad f_{k+1}, \ldots, f_m > 0.$$

For notational convenience we restrict attention only to systems with $\geq$. This can be achieved by replacing the strict inequalities with

$$yf_{k+1}, \ldots, yf_m \geq 1, \; y \geq 0$$

where y is an additional variable. Henceforth, assume we wish to determine if

$$f_1, \ldots, f_m \geq 0 \qquad (1)$$

is feasible.

To simplify arguments, we assume that $x = 0$ is not a feasible point for (1).

Let n denote the number of variables and assume that the degree of f_i is d_i.

In the analysis we will require the set of solutions to be bounded. To achieve this, we simply note that (1) is feasible if and only if there exists $\delta > 0$ such that

$$f_1, \ldots, f_m, \; \delta - \sum_j x_j^2 \geq 0 \qquad (2)$$

is feasible. For δ fixed, the boundedness requirement certainly holds. The parameter δ will be a variable in the algorithm.

Let d be the least even integer at least as great as $2 + \sum_i d_i$. For $\epsilon, \; \delta > 0$, we will be focusing on the function

$$g(x,\delta,\epsilon) = (\delta - \sum_j x_j^2 + \epsilon) \prod_i (f_i(x) + \epsilon) - \epsilon^{m+2} \sum_j x_j^d .$$

The following proposition was motivated by arguments on page 280 of Milnor [10]. Thanks to Mike Shub who suggested Milnor's arguments might be relevant.

PROPOSITION 1: If (2) has a solution, then some solution of (2) is the limit point of zeros $x(\epsilon_i)$ of $\nabla_x g(—,\delta,\epsilon_i)$ as $\epsilon_i \downarrow 0$. (Here, ∇_x denotes the vector of derivatives with respect to the x–coordinates.)

PROOF: In the proof we implicitly assume that $\epsilon^{m+2}\delta^{d/2} < \epsilon^{m+1}$.

Since any solution to (2) satisfies $\sum_j x_j^d \leq \delta^{d/2}$, it must also satisfy $g(—,\delta,\epsilon) > 0$. So the solutions of (2) are contained in the connected components of solutions of $g(—,\delta,\epsilon) > 0$. Also note that if x satisfies $f_i(x) = -\epsilon$ for some i, or $\delta - \sum_j x_j^2 = -\epsilon$, then $g(x,\delta,\epsilon) < 0$. (Here we are using the assumed infeasibility of 0.) Since any solution of (2) is a solution of

$$f_1, \ldots, f_m, \; \delta - \sum_j x_j^2 \geq -\epsilon \qquad (3)$$

it follows that any solution component of $g(-,\delta,\epsilon) > 0$ that contains a solution of (2) is itself contained in a solution component of (3). Since any solution component of (3) is bounded, it follows that any solution component of $g(-,\delta,\epsilon) > 0$ that contains a solution of (2) is bounded, and hence contains a local maximum of $g(-,\delta,\epsilon)$. Moreover, since the solution components of (3) tend to the solution components of (2) as $\epsilon \downarrow 0$, it now follows that if (2) has a solution, then some solution of (2) will be the limit point of a sequence $x(\epsilon_i)$, $\epsilon_i \downarrow 0$, where $x(\epsilon_i)$ is a local maximum of $g(-,\delta,\epsilon_i)$. The proposition follows. □

As we shall see, there are only finitely many values of ϵ for which $\nabla_x g(-,\delta,\epsilon)$ has infinitely many zeros. Unfortunately, the construction of g makes it likely that $\epsilon = 0$ is one of these values. Hence, the analysis as $\epsilon \downarrow 0$ is a little tricky.

4. ANALYSIS VIA THE U–RESULTANT.

Let $G(---,\delta,\epsilon): \mathbb{R}^{n+1} \to \mathbb{R}^n$ denote the homogenization of $\nabla_x g(-,\delta,\epsilon)$. The zero set of $G(-,\delta,\epsilon)$ is a union of lines through the origin. Moreover, $x \in \mathbb{R}^n$ is the limit of zeros of $\nabla_x g(-,\delta,\epsilon)$ as $\epsilon \downarrow 0$ if and only if the line $\{(x,t);\ t \in \mathbb{R}\}$ is the limit of zeros of $G(-,\delta,\epsilon)$ as $\epsilon \downarrow 0$. This observation and the previous proposition give the following.

Proposition 2: If (2) has a solution, then some solution x of (2) is such that the line $\{(x,t);\ t \in \mathbb{R}\}$ is the limit of zeros of $G(-,\delta,\epsilon)$ as $\epsilon \downarrow 0$. □

For $u \in \mathbb{R}^{n+1}$, let $R(u,\delta,\epsilon)$ be the u–resultant of $G(-,\delta,\epsilon)$. This is a real polynomial in the variables u, δ, and ϵ. Hence, assuming δ fixed, $R(u,\delta,\epsilon)$ is identically zero for all values of ϵ or for only finitely many values of ϵ. However, in our application it is not identically zero for all values of ϵ because (i) for $\epsilon \geq 1$, $R(u,\delta,\epsilon)$ is a greater than unity multiple of the u–resultant of $\frac{1}{\epsilon^{m+2}} G(-,\delta,\epsilon)$; (ii) the u–resultant of $\frac{1}{\epsilon^{m+2}} G(-,\delta,\epsilon)$ tends to the u–resultant of the system $(x_1, \ldots, x_{n+1}) \to (-dx_1^{d-1}, \ldots, -dx_n^{d-1})$ as $\epsilon \to \infty$; and (iii) the u–resultant of the latter system is not identically zero because the zero set of the latter system, considered as a system over $\mathbb{C}^{n+1}$, is the union of finitely many complex lines.

Consider $R(u,\delta,\epsilon)$ in powers of ϵ, that is,

$$R(u,\delta,\epsilon) = C_k(u,\delta)\epsilon^k + \sum_{i>k} C_i(u,\delta)\epsilon^i$$

where $C_k(u,\delta) \not\equiv 0$. The next proposition is very similar to results of Canny [2] and is proven by the same techniques.

Proposition 3: If δ is fixed so that $C_k(u,\delta)$ does not vanish identically in u, then $C_k(u,\delta)$ factors linearly

$$C_k(u,\delta) = \prod_{i=1}^{D} (\xi^{(i)}(\delta) \cdot u)$$

where the $\xi^{(i)}(\delta)$ are complex vectors, $D = (d-1)^n$ and we define $\xi^{(i)}(\delta) \cdot u := \sum_j \xi_j^{(i)}(\delta)\, u_j$. The subset of $\mathbb{C}^{n+1}$

$$\bigcup_i \{\omega\, \xi^{(i)}(\delta);\ \omega \in \mathbb{C}\}$$

then consists precisely of the limits of zeros of $G(-,\delta,\epsilon)$ (considered as a system over $\mathbb{C}^{n+1}$) as $\epsilon \downarrow 0$. Moreover, if

$$x \in \{\omega\, \xi^{(i)}(\delta);\ \omega \in \mathbb{C}\}$$

and x is a real vector, then it may be assumed that $\xi^{(i)}(\delta)$ is a real vector.

Sketch Of Proof: The proof of the proposition rests primarily on the fact that if the coefficients of a homogeneous polynomial over $\mathbb{C}^{n+1}$ vary continuously, then so do its zeros as long as the polynomial does not vanish identically. (This can be proven by reduction to the single variable non–homogeneous case.) By this fact, the zeros of $\frac{1}{\epsilon^k} R(-,\delta,\epsilon)$ vary continuously in ϵ for ϵ in an open neighborhood of 0 if we define $\frac{1}{0^k} R(-,\delta,0) := C_k(-,\delta)$. Since $R(-,\delta,\epsilon)$ factors linearly

$$R(u,\delta,\epsilon) = \prod_{i=1}^{D} (\xi^{(i)}(\delta,\epsilon) \cdot u)$$

for all sufficiently small $\epsilon \neq 0$, the first statement of the proposition follows. The second statement then follows from the fact that the zero set of $G(-,\delta,\epsilon)$ is precisely the set

$$\bigcup_i \{\omega\, \xi^{(i)}(\delta,\epsilon);\ \omega \in \mathbb{C}\}\ .$$

The third statement is now easily proven using the fact that the coefficients of $C_k(-,\delta)$ are real. □

5. THE ALGORITHM.

The algorithm is motivated through the following observations.

Let $F_1, \ldots, F_{m+1}$ be the homogenizations with respect to x of $f_1, \ldots, f_m$, $\delta - \sum_j x_j^2$. We write $F_{m+1}(x,\delta)$ to indicate that F_{m+1} is a polynomial in δ as well as in x. Then the original system (1) is feasible if and only if the system in x and δ

$$F_1, \ldots, F_{m+1},\ x_{n+1} > 0,\ \delta > 0 \qquad (4)$$

is feasible. Moreover, defining $\xi^{(i)}(\delta) = 0$ for the finitely many values of δ for which $C_k(u,\delta)$ vanishes identically in u, from propositions 2 and 3 we see that if (1) is feasible then there is some δ and some i such that either $x = \xi^{(i)}(\delta)$ and δ, or $x = -\xi^{(i)}(\delta)$ and δ, is a solution to (4).

Let

$$S_1 = \{(1, i, i^2, \ldots, i^n);\ 1 \leq i \leq nD + 1\}$$

$$S_2 = \{(1, i, i^2, \ldots, i^n);\ 1 \leq i \leq nD(D-1)/2 + 1\}$$

LEMMA: Assume δ is fixed so that $C_k(u,\delta)$ does not vanish identically in u. Then for some $\alpha \in S_1$, $\beta \in S_2$, both of the following are true for each real $\xi^{(i)}(\delta)$.

(i) The line $\{t\alpha + \beta; t \in \mathbb{R}\}$ intersects $\{u \in \mathbb{R}^{n+1}; \xi^{(i)}(\delta) \cdot u = 0\}$ in exactly one point t'.

(ii) If t' (as in (i)) satisfies $t'\alpha + \beta \in \{u \in \mathbb{R}^{n+1}; \xi^{(j)}(\delta) \cdot u = 0\}$, then $\xi^{(j)}(\delta)$ is a real multiple of $\xi^{(i)}(\delta)$.

PROOF: Each subset of $n+1$ vectors from S_1 is linearly independent. Hence, at least one $\alpha \in S_1$ does not lie in $\bigcup_j \{u \in \mathbb{R}^{n+1}, \xi^{(j)}(\delta) \cdot u = 0\}$. Then for any β', property (i) is satisfied by α and β', and if $\xi^{(j)}(\delta)$ is complex, $\{t\alpha + \beta'; t \in \mathbb{R}\}$ intersects $\{u \in \mathbb{R}^{n+1}; \xi^{(j)}(\delta) \cdot u = 0\}$ in at most one point.

Let $H_1 \subset \mathbb{R}^{n+1}$ denote the set of points contained in at least two distinct hyperplanes $\{u \in \mathbb{C}^{n+1}; \xi^{(j)}(\delta) \cdot u = 0\}$. Then H_1 is contained in the union of at most $D(D-1)/2$ linear subspaces of dimension $n-1$. Hence

$$H_2 = \{\beta' \in \mathbb{R}^{n+1}, \alpha t + \beta' \in H_1 \text{ for some } t \in \mathbb{R}\}$$

is contained in the union of at most $D(D-1)/2$ linear subspaces of dimension n. Since every subset of $n+1$ vectors from S_2 is linearly independent, at least one $\beta \in H_2$ is not in S_2. □

Using

$$C_k(u,\delta) = \prod_{i=1}^{D} (\xi^{(i)}(\delta) \cdot u)$$

it follows from the lemma and the product rule for differentiation that for each δ there exists $\alpha \in S_1$, $\beta \in S_2$ such that for each real $\xi^{(i)}(\delta)$ there exists $1 < p \le D$ and $t' \in \mathbb{R}$ satisfying

$$\xi^{(i)}_j(\delta) = \frac{\partial^p}{\partial s \partial^{p-1} t} \Big|_{\substack{s=0 \\ t=t'}} C_k(\alpha t + se_j + \beta, \delta)$$

where e_j is the j^{th} unit vector. The value t' is the value at which $\alpha t + \beta$ intersects $\{u \in \mathbb{R}^{n+1}; \xi^{(i)}(\delta) \cdot u = 0\}$, and p is the multiplicity of t' as a zero of $C_k(\alpha t + \beta, \delta)$.

For each $\alpha \in S_1$, $\beta \in S_2$, and each $1 \le p \le D$, defining $x_j(t,\delta)$ to be the polynomial

$$x_j(t,\delta) = \frac{\partial^p}{\partial s \partial^{p-1} t} \Big|_{s=0} C_k(t\alpha + se_j + \beta, \delta),$$

the algorithm is simply to apply the BKR algorithm to the two variable system

$$F_1(x(t,\delta)), \ldots, F_{m+1}(x(t,\delta), \delta) \ge 0$$

$$x_{n+1}(t,\delta) > 0, \quad \delta > 0$$

and to the same system with $x(t,\delta)$ replaced by $-x(t,\delta)$.

Combining the observations of this section, one of these two systems is feasible for some $\alpha \in S_1$, $\beta \in S_2$ and $1 \le p \le D$ if and only if the original system (1) is feasible.

All that remains to be discussed is the computation of the polynomials $x_j(t,\delta)$. An essentially identical computation is required by Canny.

For this we rely, as does Canny, on the result of Macaulay [9] that states that the u–resultant for systems of homogeneous polynomials is a polynomial equal to a quotient $\det A(u)/\det M$ of determinants, where A is a particular matrix whose coefficients are polynomials in u and the coefficients of the homogeneous systems, and M is a particular matrix whose coefficients are polynomials in the coefficients of the systems. For a discussion of the particular matrices, see, for example, section 2 of Renegar [11].

Since we are concerned with $R(u,\epsilon,\delta)$, the coefficients of $A(u) = A(u,\epsilon,\delta)$ depend on those of (1) along with ϵ and δ. By simply relying on the definition of $A(u,\epsilon,\delta)$ it can be easily shown that $\lim_{\epsilon \to \infty} A(u,\epsilon,\delta) \ne 0$. Hence, since $R(u,\epsilon,\delta)M(\epsilon,\delta) = A(u,\epsilon,\delta)$, $M = M(\epsilon,\delta)$ cannot vanish identically. Expanding in powers of ϵ

$$\det A(u,\epsilon,\delta) = B_h(u,\delta)\epsilon^h + \sum_{i>h} B_i(u,\delta)\epsilon^i,$$

$$\det M(\epsilon,\delta) = N_\ell(\delta)\epsilon^\ell + \sum_{i>\ell} N_i(\delta)\epsilon^i,$$

where $B_h(u,\delta)$, $N_\ell(\delta) \ne 0$, it follows that $C_k(u,\delta) = B_h(u,\delta)/N_\ell(\delta)$. Thus, we can replace $C_k(t\alpha + se_j + \beta, \delta)$ by $B_h(t\alpha + se_j + \beta, \delta)$ in the algorithm. The four variable polynomial $\det A(t\alpha + se_j + \beta, \delta, \epsilon)$ can be computed in PSPACE using Csanky [5].

The computations required for the algorithm are very similar to those required by Canny – in fact, they are less involved. The major difference is that because Canny introduces a variable for each constraint, his A(u) matrix has size exponential in m as well as in n. For n fixed, the size of our A(u) matrix is $(md)^{O(n)}$.

In closing we remark that although the algorithm does not construct an (approximate) solution for the problem, it apparently can be altered to do so, but at the cost of requiring exponential (in n) space over the rationals. Roughly, the procedure is this. For some $\alpha \in S_1$, $\beta \in S_2$, $1 \le p \le D$, $\delta > 0$ and some zero t' of $C_k(t\alpha + \beta, \delta)$, $x(t',\delta)$ as defined above is a solution to

(4), assuming a solution exists. Using Vorobjov [13], δ can be taken to be $2^{L^{O(1)}(md)^{O(n)}}$ where L is the maximal bit length of the polynomial coefficients. One first approximates the zeros t' of $C_k(t\alpha + \beta, \delta)$, then easily obtains approximations to the vectors $x(t',\delta)$. Checking these approximations for an appropriate relaxation of (4) (the ≥ 0 will be replaced by $> -2^{-L^{O(1)}(md)^{O(n)}}$) one chooses any of these approximations satisfying the relaxation – if none of the approximations satisfy the relaxation, the original problem is infeasible. The chosen approximation is then easily converted into an approximation of a solution for the original system (1). Some of the details that would be required in attempting to make this argument rigorous can be found in Grigor'ev and Vorobjov [7].

6. REFERENCES.

[1] M. Ben–Or, D. Kozen, and J. Reif, "The complexity of elementary algebra and geometry," *Journal of Computer and System Science* 32 (2) (1986), 251–264.

[2] J. Canny, "Generalized characteristic polynomials," preprint, Department of Computer Science, University of California, Berkeley (1988).

[3] J. Canny, "Some algebraic and geometric computations in PSPACE," *Proceedings of the 20th Annual ACM Symposium on the Theory of Computing* (1988).

[4] G.E. Collins, "Quantifier elimination for real closed fields by cylindrical algebraic decomposition," *Lecture Notes in Computer Science* 33, Springer–Verlag, New York, (1975).

[5] L. Csanky, "Fast parallel matrix inversion algorithms," *SIAM Journal on Computation* 5 (4) (1976), 618–623.

[6] D.Y. Grigor'ev, "Complexity of deciding Tarski algebra," *Journal of Symbolic Computation* (1988).

[7] D.Y. Grigor'ev and N.N. Vorobjov, "Solving systems of polynomial inequalities in subexponential time," *Journal of Symbolic Computation* (1988).

[8] J. Heintz, "Definability and fast quantifier elimination in algebraically closed fields," *Theory of Computer Science* 24 (1983), 239–278.

[9] F.S. Macaulay, "Some formulae in elimination," *Proceedings of the London Mathematical Society* 1 (35) (1902), 3–27.

[10] J. Milnor, "On the Betti numbers of real varieties," *Proceedings of the American Mathematical Society* 15 (2) (1964), 275–280.

[11] J. Renegar, "On the worst–case arithmetic complexity of approximating zeros of systems of polynomials," to appear in *SIAM Journal on Computing.*

[12] A. Tarski, *A Decision Method for Elementary Algebra and Geometry*, University of California Press, Berkeley (1951).

[13] N.N. Vorobjov, "Bounds of real roots of a system of algebraic equations," *Notes of Scientific Seminars of Leningrad Department of Mathematics' Steklov Institute* 137 (1984), 7–19.