Computing Roadmaps of Semi-algebraic Sets
(Extended Abstract)

Saugata Basu[*]
Richard Pollack[†]
Marie-Françoise Roy[‡]

November 3, 1995

Abstract

We consider a semi-algebraic set S defined by s polynomials of degree d in k variables. We present a new algorithm for computing a semi-algebraic path in S connecting two points if they happen to lie in the same connected component of S. This algorithm, which works in time $s^{k+1}d^{O(k^2)}$ improves the complexity of the fastest algorithm solving this problem known to this date.

1 Introduction

1.1 The Problem

Given a finite family of polynomials, $\mathcal{P} = \{P_1, \ldots, P_s\} \subset \mathrm{D}[X_1, \ldots, X_k]$, where D is an ordered domain contained in a real closed field R, and a semi- algebraic set S, defined by a Boolean formula with atoms of the form $P_i \sigma 0$, $\sigma \in \{<, >, =\}$, as well as a finite subset M we construct $R(S, M) \subset S$, called the *roadmap* for the pair (S, M). $R(S, M)$ is a semi-algebraic set of dimension at most one, satisfying

1. $M \subset R(S, M)$,
2. for every semi-algebraically connected component C of S, $C \cap R(S, M)$ is non-empty and semi-algebraically connected.
3. for every $x \in R$, and for every semi-algebraically connected component C' of S_x, $C' \cap R(S, M) \neq \emptyset$. (Here, and everywhere else in this paper π is the projection on the first coordinate, S_X is $S \cap \pi^{-1}(X)$, where $X \subset R$ and S_x is $S_{\{x\}}$).

If the polynomials in $\mathcal{P}$ have total degrees bounded by d, and the m points in M are defined by polynomials of degree bounded by $d^{O(k)}$, then we compute $R(S, M)$ using $s(s^k + m \log m)d^{O(k^2)}$ arithmetic operations in D. Moreover, the degrees of the polynomials defining the roadmap are

[*]Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, U.S.A.. Supported in part by NSF grant CCR-9122103

[†]Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, U.S.A.. Supported in part by NSF grant CCR-9122103, and NSA grant MDA904-92-H-3075.

[‡]IRMAR (URA CNRS 305), Université de Rennes, Campus de Beaulieu 35042 Rennes cedex FRANCE supported in part by the project ESPRIT-BRA 6846POSSO

bounded by $d^{O(k^2)}$. The roadmap allows us to decide whether two points in the set M are in the same connected component of S at no additional cost, by using standard graph traversal algorithm which runs in time linear in the number of vertices and edges of a graph. Furthermore, if these two points are in the same connected component, we obtain a semi-algebraic path joining the two points which consists of certain branches of the roadmap.

1.2 Prior Results

The problem of deciding connectivity properties of semi-algebraic sets has attracted much attention during the last ten years. The original motivation came from robot motion planning where the free space of a robot can be modeled as a semi- algebraic set, and it is important to know whether a robot can move from one configuration to another (see [12]). This is equivalent to deciding whether the two corresponding points in the configuration space are in the same connected component of the free space.

In 1987, John Canny [3] introduced the notion of a roadmap of a semi- algebraic set. Similar notions were also considered by Grigor'ev and Vorobjov [6](1992), Heintz, Roy, and Solerno [9] and Gournay and Risler [7].

Canny [3] gave an algorithm that after subsequent modifications [4] constructed a roadmap for a semi-algebraic set defined by polynomials whose sign invariant sets give a Whitney regular stratification of R^k in time $s^k(\log s)d^{O(k^4)}$. For an arbitrary semi-algebraic set he perturbs the defining polynomials and is then able to decide if two points are in the same semi- algebraically connected component in the same time. This algorithm however does not give a path joining the points. A Monte Carlo version of this algorithm runs in time $s^k(\log s)d^{O(k^2)}$.

Grigor'ev and Vorobjov [6] gave an algorithm with complexity $(sd)^{k^{O(1)}}$, for counting the number of connected components of a semi-algebraic set. Heintz, Roy, and Solerno [9] and Gournay and Risler [7] gave algorithms which compute a roadmap for any semi-algebraic set whose complexity was also $(sd)^{k^{O(1)}}$. Note that unlike the complexity of Canny's algorithm, the complexities of these algorithms are not separated into a combinatorial part (the part depending on s) and an algebraic part (the part depending on d). Since the given semi-algebraic set might have $(sd)^k$ different connected components the combinatorial complexity of Canny's algorithm is nearly optimal. On the other hand an algebraic complexity of $d^{O(k^2)}$ is unavoidable as long as the basic technique used in all known algorithms for this problem, recursion down the dimension, is kept. However, Canny's algorithm makes use of Thom's isotopy lemma for stratified sets, and as a result it requires the use of generic projections, as well as perturbations to put the input polynomials into general position in a very strong sense. In order to do this in a deterministic fashion, $O(s+k^2)$ different transcendentals are introduced, requiring the algebraic operations to be performed over an extended ring, which raises the algebraic complexity of the algorithm to $d^{O(k^4)}$.

1.3 Our work

We give a deterministic algorithm that constructs a roadmap for any semi-algebraic set in time $s^{k+1}d^{O(k^2)}$. This complexity is again nearly optimal in its combinatorial complexity and moreover it does not require the regular stratification condition, or generic projections, which helps to reduce the algebraic complexity to $d^{O(k^2)}$. Moreover, the algorithm computes a semi-algebraic path between the input points if they happen to lie in the same connected component, This solves the extended version of the problem. And lastly our proofs do not rely on the machinery of Morse Theory for stratified sets.

The rest of the paper is organized as follows. In section 2 we state a topological lemma that is crucial for our algorithm. Next, we develop some algorithmic tools that allow us to compute curves in a given algebraic set parametrized by the first co-ordinate, and moreover identify certain *distinguished slices* where it is necessary to make recursive calls in one lower dimension.

In section 3 we give an algorithm for the case where the semi-algebraic set is defined by a single polynomial equality. This also serves as a base case for our recursion (see section 4).

In section 4 we introduce the notion of the *combinatorial level* which we use in our algorithm. We give an algorithm, that given a family of polynomials $\mathcal{P}$ constructs a *uniform* roadmap which, though not a roadmap, has the property that its intersection with any set defined by a single weak sign condition on $\mathcal{P}$ is a roadmap for that set.

Lastly in section 5, we give a roadmap algorithm for an arbitrary semi- algebraic set S, defined by the family of polynomials $\mathcal{P}$. We compute a set of triples $(p_1, p_2, \gamma_{1,2})$ where p_1, p_2 are points in different connected components of two sign conditions of $\mathcal{P}$, whose union is connected, and $\gamma_{1,2}$ is a path joining p_1, p_2 lying inside S. We make our roadmap pass through the points p_i and include the paths γ_{ij}. This ensures that our roadmap remains connected even when we take unions of sign conditions. The size of this set is bounded by $O(s^k d^{O(k)})$. We next perturb the polynomials using two infinitesimals, to ensure that no $k+1$ of our polynomials have a common zero.

We then apply the algorithm of the previous section to compute the roadmap for the perturbed set. Finally we find a roadmap on the original set S by taking a suitable limit of the roadmap just constructed.

2 Preliminaries

2.1 A topological lemma

Our argument depends on perturbing the polynomials by various *infinitesimals* and to work over the field of Puiseux series in these infinitesimals. We write $R\langle\epsilon\rangle$ for the real closed field of Puiseux series in ϵ with coefficients in R [2]. The sign of an element in this field agrees with the sign of the coefficient of the lowest degree term in ϵ. This order makes ϵ positive and smaller than any positive element of R. The map eval_ϵ, maps an element of $R\langle\epsilon\rangle$ which is bounded over R, i.e. one that has no negative powers of ϵ, to its constant term.

Given a polynomial Q such that $Z(Q)$ is compact, let $\zeta \ll \frac{1}{\Omega}$ be infinitesimals, and let

$$Q_1 = (1-\zeta)Q^2 + \zeta(X_1^{2d+2} + \cdots + X_k^{2d+2} - k\Omega^{2d+2}).$$

Then, $Z(Q_1)$ is a smooth hypersurface with a finite number of critical points with respect to π (see [1]) for a proof). Following an idea in [7], the *pseudo-critical values* of Q are defined to be the first co-ordinates of the images of these critical points under the eval_ζ map.

Given a compact semi-algebraic set S defined by $P_1 \geq 0, \ldots, P_s \geq 0$, we call $v \in R$ a pseudo-critical value for S, if it is a pseudo-critical value for any of the non-empty algebraic sets, $P_{i_1}^2 + \cdots + P_{i_l}^2 = 0$, $1 \leq i_1 < i_2 \cdots < i_l \leq s$.

The following lemma, which generalizes a result in [7], allows us to characterize a certain finite number of values of the first co-ordinate so that it suffices to make recursive calls at the corresponding slices when constructing the roadmap for a basic semi-algebraic set.

Lemma 1 *Let the sets $Z(P_i), 1 \leq i \leq s$ be compact and C be a non-empty semi- algebraically connected component of $S_{[a,b]}$. If $v \in (a,b)$ is the only pseudo-critical value for S in $[a,b]$, or if $v \in [a,b]$ and there are no pseudo-critical values for S in $[a,b]$, then the set C_v is non-empty and semi-algebraically connected.*

The proof appears in the full version of the paper.

2.2 Curve Segments Subroutine

The input is a polynomial $Q \in \mathrm{D}[Y, X_1, \ldots, X_k]$, whose degree is at most d and a set of m points, say $M \subset Z(Q)$, such that each point of M is defined by polynomials of degree at most $d^{O(k)}$. We assume that $Z(Q)$ is contained in the ball of radius Ω centered at the origin The output contains a partition of the Y axis into points, called *distinguished values*, and open intervals. The distinguished values (and hence the intervals) are ordered. Among the distinguished values are the first coordinates of the points in M. Over each distinguished value the subroutine also outputs a finite set of points, called *distinguished points*, and over each interval a finite set of *curve segments*: i.e. continuous semi-algebraic curves parameterized by the interval along the Y axis. The sets of distinguished points and curve segments are contained in $Z(Q)$. The set of distinguished values and the sets of distinguished points and curve segments satisfy the following properties.

P_1: Let $[a, b]$ be an interval, and C be a non-empty semi-algebraically connected component of $\pi^{-1}([a, b]) \cap Z(Q)$. If $v \in (a, b)$ is the only distinguished value in $[a, b]$, or if $v \in [a, b]$ and there are no distinguished values in the interval $[a, b]$, then the set $\pi^{-1}(v) \cap C$ is non-empty and semi-algebraically connected.

P_2: For every $y \in R$ the set of distinguished points and curve segments output intersect every semi-algebraically connected component of $Z(Q) \cap \pi^{-1}(y)$.

P_3: For each curve segment output over an interval with endpoint a given distinguished value, there exists a distinguished point over this distinguished value which belongs to the closure of the curve segment.

The adjacency relations between curve segments and distinguished points are computed in the subroutine.

The subroutine makes use of the parametrized cell representatives subroutine and the univariate representation subroutine in [1]. The parametrized cell representatives subroutine allows one to construct curves in $Z(Q)$ satisfying P_2, and using the univariate representation subroutine we compute the pseudo- critical values of Q. This ensures property P_1 by lemma 1.

The cost of computing the curves and the pseudo-critical values is bounded by $d^{O(k)}$ (see [1] for details). However, there is the additional cost of sorting the distinguished values computed, and this adds a factor of $m \log m$. The total cost is bounded by $(m \log m) d^{O(k^2)}$.

3 The roadmap algorithm for an algebraic set

The input is a polynomial $Q \in \mathrm{D}[X_1, \ldots, X_k]$, whose total degree is at most d and a set of m points, say M, such that each point in M is defined by polynomials of degrees not greater than $d^{O(k)}$. The output is a roadmap $R(Z(Q), M)$.

So that we can use the curve segments subroutine, we first introduce a new variable X_{k+1} and a new infinitely large variable Ω and replace Q by the polynomial $Q^2 + (X_1^2 + \cdots + X_{k+1}^2 - \Omega^2)^2$.

Using the curve segments subroutine taking X_1 as parameter, we obtain a partition of the X_1 axis and sets of semi- algebraic continuous curves parametrized along the X_1 axis, with properties P_1, P_2 and P_3.

At every distinguished value v, make a recursive call to the algorithm in $R^{k-1} = \pi^{-1}(v)$, with the following input: the polynomial, $Q(v, X_2, \ldots, X_k)$, and the set of points which are the points of M and the distinguished points belonging to the fiber $\pi^{-1}(v)$. Note that the polynomial $Q(v, X_2, \ldots, X_k)$, has its coefficients in the ordered ring $D[v]$.

We then project all the curves segments and distinguished points obtained onto their first k coordinates, and let $\Omega \to \infty$ to obtain the roadmap over all of R^k.

3.1 Correctness and Complexity

The correctness of the algorithm follows from the correctness of the curve segments subroutine. The cost of a call to the curve segments subroutine is $(m \log m)d^{O(k)}$ arithmetic operations in the ring D.

We make a recursive call at $d^{O(k)}$ distinguished values. Note that in a recursive call at depth ℓ the computations are performed over a ring $D[v_1, \ldots, v_\ell]$, and a single arithmetic operation in $D[v_1, \ldots, v_\ell]$ costs $d^{O(k\ell)}$ operations in D. Thus, the total number of arithmetic operations made in the ring D during the algorithm is bounded by $(m \log m)d^{O(k^2)}$.

4 A Uniform Algorithm

4.1 Combinatorial Level

Given a polynomial Q and a set of polynomials $\mathcal{P} = \{P_1, \ldots, P_s\}$ we define the *combinatorial level* of the system $(Q, \mathcal{P})$ to be the minimum number ℓ (if it exists) satisfying the following :

1. no more than ℓ of the polynomials in $\mathcal{P}$ have a real zero in common with Q, and

2. any real zero common to ℓ polynomials of $\mathcal{P}$ and Q is isolated.

Note that if the combinatorial level of $(Q, \mathcal{P})$ is bounded by ℓ, the the combinatorial level of $(Q^2 + P_1^2, \{P_2, \ldots, P_s\})$ is bounded by $\ell - 1$. We say that a family of polynomials $\mathcal{P} = \{P_1, \ldots, P_s\}$ is in *general position* on $Z(Q)$ if no $k+1$ polynomials of this family have a real zero in common with Q and if the only real zeroes common to k polynomials of the family and Q are isolated. Thus, if the set of polynomials $P_1, \ldots, P_s$ is in general position over $Z(Q)$, then the combinatorial level of the system $(Q, \mathcal{P})$ is at most k.

4.2 Uniform Roadmap Construction

In subsection 4.3 we describe an algorithm that takes as input a polynomial Q for which $Z(Q)$ is compact and a set of polynomials $\mathcal{P} = \{P_1 \ldots, P_s\}$, where the combinatorial level of the system $(Q, \mathcal{P})$ is at most ℓ, and the degrees of the polynomials are bounded by d, and a finite set of points $M \subset Z(Q)$, with $|M| = O(s^\ell d^{O(k)})$.

The algorithm outputs a one-dimensional semi-algebraic set, $R(Q, \mathcal{P}, M)$, which is a union of open curve segments and points satisfying the following two properties:

(a) The signs of the polynomials, $Q, P_1, \ldots, P_s$ are constant on each curve segment,

(b) The intersection of this set with any basic closed semi- algebraic set S_σ, defined by a sign condition $\sigma = (0, \sigma_1 \ldots \sigma_s)$, i.e. $Q = 0, P_1\sigma_1 0, \ldots, P_s\sigma_s 0$, where $\sigma_i \in \{\geq, \leq\}$, (we call such a σ a *weak* sign condition on $\mathcal{P}$) is a roadmap $R(S_\sigma, M \cap S_\sigma)$.

The complexity of the algorithm is $s^{\ell+1}d^{O(k^2)}$.

4.3 Computing Uniform Roadmaps

4.3.1 The Basic Idea Behind the Algorithm

Consider a weak sign condition, say σ, of the polynomials $\mathcal{P}$, and let S_σ be the set defined by σ. Consider a sweep hyperplane orthogonal to the X_1 axis, defined by $X_1 = t$, and look at the semi-algebraically connected components $C_1(t), \ldots, C_l(t)$, of $S \cap (X_1 \leq t)$, that is the intersection of S with the left half-space defined by the sweep hyperplane. The key fact is that, as the hyperplane sweeps from left to right, if two of these connected components, $C_i(t), C_j(t)$, *join together* and become one connected component at some point, then there exists an algebraic set $Z(P_{i_1}, \ldots, P_{i_m})$, such that a semi- algebraically connected component of the intersection of this algebraic set with S, belongs to this single connected component. Moreover, the X_1 co-ordinate of the sweep hyperplane at this point is a pseudo-critical value for $Z(P_{i_1}, \ldots, P_{i_j})$.

The algorithm constructs curves in the connected components, $C_i(t)$, such that the to the left of the sweep hyperplane the curves satisfy the roadmap conditions for $C_i(t)$. When two connected components join together to form a single connected component, the partial roadmaps of these two components computed so far, get linked by means of a recursive call to the algorithm. The recursive call computes a roadmap in the slice, restricted to the algebraic set mentioned above.

4.3.2 Finger Subroutine

We first describe a subroutine which, given a polynomial Q, a set of polynomials $\mathcal{P} = \{P_1, \ldots, P_s\}$ and a point p in $Z(Q)$, constructs a finite number of continuous semi-algebraic curves starting at p so that every semi-algebraically connected component of every realizable sign condition of $\mathcal{P}$ in $Z(Q)$ in a sufficiently small neighborhood of p, contains one of these curves without the point p.

The subroutine takes as input a polynomial Q and a family of polynomials $\mathcal{P} = \{P_1, \ldots, P_s\}$ with combinatorial level of the system $(Q, \mathcal{P})$ bounded by ℓ and a point $p \in Z(Q)$ which is defined by polynomials of degree at most $d^{O(k)}$.

It outputs a set of continuous curves starting at p with the property that every semi-algebraically connected component of non-empty sign conditions of $\mathcal{P}$ in $(Z(Q) \cap B_p(r))_{(-\infty,\pi(p))}$for some sufficiently small r contains one of these curves without the endpoint p.

Let the set of polynomials (possibly empty) that are zero at p be be $\mathcal{P}_p = \{P_j | j \in J\}$, and $B_p(\epsilon)$ be a ball of radius ϵ and center p, where ϵ is an infinitesimal. Using the sample points subroutine [1] with the polynomials defining $B_p(\epsilon)_{(-\infty,\pi(p))}$ along with the polynomials $\mathcal{P}_p$ as input, we find a point $p_i(\epsilon)$ in every semi- algebraically connected component of every non-empty sign conditions of these polynomials in $B_p(\epsilon)_{(-\infty,\pi(p))}$.

Moreover, $\mathrm{eval}_\epsilon(p_i) = p$. For a small enough t_0, replacing ϵ by t where $0 \le t \le t_0$ gives for each $p_i(\epsilon)$, continuous curves $p_i(t)$, which join p to points in every semi- algebraically connected component of the non-empty sign conditions of $\mathcal{P}$ intersected by $(Z(Q) \cap B_p(\epsilon))_{(-\infty,\pi(p))}$.

The complexity of this subroutine is easily seen to be the cost of calling the sample points subroutine, with $O(\ell)$ polynomials, in k variables, $X_1, \ldots, X_k$.

Since at most ℓ polynomials can be zero at p and using the complexity bound of the sample points subroutine in [1], we conclude that the complexity of the finger subroutine is $\binom{O(\ell)}{k} d^{O(k)}$.

4.3.3 The Uniform Roadmap Algorithm

Note that, if the combinatorial level of the system, $(Q, \mathcal{P})$ is 0 then we can use the algorithm for the algebraic case, since on every semi-algebraically connected component of $Z(Q)$ the signs of the polynomials in $\mathcal{P}$ are fixed and are not zero. Moreover, this algorithm has complexity $sd^{O(k^2)}$.

Also, if $k = 1$ then we can sort the roots of the univariate polynomials $Q, P_1, \ldots, P_s$ and output a description of the one dimensional set as the roadmap. If the size of the input set of points is bounded by $d^{O(1)}$, then the algorithm has a time complexity of $(s\log s) d^{O(1)}$.

The remaining cases are described below.

We use the curve segment subroutine to compute a set of curves and distinguished values for the sets $Z(Q)$ and $Z(Q_J)$ for each $J \subset \{1, 2, \ldots, s\}$ of cardinality at most $\ell - 1$ where

$$Q_J = Q^2 + \sum_{j \in J} P_j^2.$$

Denote the set of curves constructed on $Z(Q_J)$ by $R(Q_J)$. We store the set of distinguished values and points computed by the subroutine and call the end points of these segments the set of *end points.*

Intersect each of these curve segments with the zero set of each polynomial in $\mathcal{P}$. Note that the intersection of a curve segment with the zero set of a polynomial, is either the segment itself, or a finite set of points (possibly empty). We check which case holds by substituting the univariate representation of the curve into each polynomial in $\mathcal{P}$ and checking whether or not the resulting univariate polynomial vanishes identically or not.

If the intersection is the curve segment itself, we ignore this intersection. Otherwise, the pointsof intersection yield a partition of the curve segment. We add these points of intersection to our list of end points.

The signs of the polynomials of $\mathcal{P}$ do not change on each segment of this partition. Moreover, we can maintain a data structure that stores the sign vector of the set of polynomials $\mathcal{P}$ on each curve segment and point computed above.

We add the points in M to the set of end points. The X_1 co- ordinates of end points are called *distinguished values.* A *distinguished slice* is $\pi^{-1}(v)$ for a distinguished value v. Note that a distinguished value originates either from a call to the curve segments subroutine for a particular algebraic set $Z(Q_J)$, or as the X_1 co-ordinate of a point of intersection of a curve computed by the curve segment subroutine on a particular algebraic set $Z(Q_J)$, or as the X_1 co-ordinate of a point in M. The algebraic set *associated* to v is $Z(Q_J)$ in the first two cases, and $Z(Q_\emptyset) = Z(Q)$ in the last case.

For every distinguished value v, compute the set M_J, of points of intersection of the curves constructed on $Z(Q_J)$ in the previous step and the slice $\pi^{-1}(v)$. Call the algorithm recursively to construct a uniform roadmap, $R(Q_J, \mathcal{P}, M_J)$, restricted to the slice $\pi^{-1}(v)$.

Note, that the combinatorial level of this system is $\ell - |J|$, and the number of variables is $k - 1$.

For each distinguished point p and the corresponding distinguished slice, we use the finger subroutine to construct curves joining p to points in every semi-algebraically connected component of every realizable sign condition of the family of polynomials $\mathcal{P}$ intersected with, $(Z(Q) \cap (B_p(r_p) \setminus \{p\}))_{(-\infty, \pi(p))}$, for some small enough r_p. Let the other endpoints of these curves be $p_1, \ldots, p_j$.

For each $p_i, 1 \leq i \leq j$, in the previous step we compute a roadmap, $\Gamma(p_i)$, for the algebraic set $Z(Q)$ in the slice which passes through p_i as well as the finite set of points, $I(p_i) = R(Q)_{\pi(p_i))}$.

We then compute the intersections of the points and curve segments defining $\Gamma(p_i)$ with $Z(P_i), 1 \leq i \leq s$, and retain the maximal semi- algebraically connected component of $\Gamma(p_i)$ which contains p_i and over which the signs of the polynomials in $\mathcal{P}$ are constant. We call this set $\Gamma'(p_i)$.

For each p_i in the previous step, either $\Gamma'(p_i) \cap I(p_i) \neq \emptyset$, or $\Gamma'(p_i)$ intersects $Z(P)$, at a point q_i, for some $P \in \mathcal{P}$.

In the former case, continue with the next p_i in the list. In the latter case, repeat Step 4 with p_i replaced by q_i, and staying on the algebraic set $P = Q = 0$ instead of $Q = 0$

In other words, we compute a roadmap, $\Gamma(q_i)$, of $P = Q = 0$ passing through q_i and the set of points $I(q_i)$. We retain the semi-algebraically connected component, $\Gamma'(q_i)$ of $\Gamma(q_i)$ which contains q_i and over which the signs of the polynomials in $\mathcal{P}$ are constant.

Again, either $\Gamma'(q_i) \cap I(q_i)$ is non-empty, or $\Gamma'(q_i)$ intersects $Z(P_1)$ for some polynomial $P_1 \in \mathcal{P} \setminus P$, and so on. Since the combinatorial level of $(Q, \mathcal{P})$ is bounded by ℓ this procedure is bound to terminate, after at most ℓ iterations. When it terminates, we have connected the point p_i by a series of curves in the same weak sign condition as p_i, lying in the slice, to some other point of our partially computed roadmap in this weak sign condition.

Next, we connect each point $p \in M$, with the part of the roadmap lying in the weak sign condition satisfied by p. To do this we repeat the computations of the previous step with p_i replaced by p.

We output all the curves, and endpoints computed above with every curve labelled by the sign condition it satisfies.

4.4 Proof of Correctness

We only give the idea behind the proof. The complete proof appears in the full version. We prove that the set $R(Q, \mathcal{P}, M)$ computed by the algorithm satisfies properties (a) and (b) of section 4.2.

Obviously property (a) of section 4.2 is satisfied by $R(Q, \mathcal{P}, M)$. Let $v_1, \ldots, v_n$ be the set of distinguished values computed by the algorithm.

We fix a weak sign condition σ and prove that $R(Q, \mathcal{P}, M) \cap S_\sigma$ is a roadmap for the pair $(S_\sigma, M \cap S_\sigma)$. This fact follows by induction on i and the following two lemmas.

Lemma 2 *For $1 \leq i \leq n$, if $R(Q, \mathcal{P}, M)_{(-\infty, v_i]}$ is a roadmap for the pair $(S_{(-\infty, v_i]}, M_{(-\infty, v_i]})$ then $R(Q, \mathcal{P}, M)_{(-\infty, v_{i+1})}$ is a roadmap for the pair $(S_{(-\infty, v_{i+1})}, M_{(-\infty, v_{i+1})})$.*

Lemma 3 *For $1 \leq i \leq n$, if $R(Q, \mathcal{P}, M)_{(-\infty, v_i)}$ is a roadmap for the pair $(S_{(-\infty, v_i)}, M_{(-\infty, v_i)})$ then $R(Q, \mathcal{P}, M)_{(-\infty, v_i]}$ is a roadmap for the pair $(S_{(-\infty, v_i]}, M_{(-\infty, v_i]})$.*

4.4.1 Complexity

When $\ell = 0$, it follows from the analysis of the algebraic case, that the complexity is $(s + m \log m) d^{O(k^2)}$. When $k = 1$, the complexity is $(s \log s + m \log m) d^{O(1)}$.

The total number of calls to the curve segments subroutine is $\sum_{1 \leq j < \ell} \binom{s}{j}$, and each call costs $d^{O(k)}$. Thus, the total cost of the calls to the curve segments subroutine is bounded by $s^{\ell - 1} d^{O(k)}$ arithmetic operations in D.

The cost of computing the intersection of the curves computed with the zero sets of the polynomials in $\mathcal{P}$ is bounded by by $s^\ell d^{O(k)}$.

The total cost of the calls to the finger subroutine and linking is bounded by $s^{\ell+1} d^{O(k)}$.

We next count the recursive calls. For each j, $0 \leq j < \ell$ we make $\binom{O(s)}{j} d^{O(k)}$ recursive calls, to the algorithm with the system having combinatorial level $\ell - j$ and geometric dimension $k - 1$.

Let $m = s^\ell d^{O(k)}$, and let let $T(s, d, \ell, k)$ denote the number of arithmetic operations needed for the problem with these parameters. Since at any depth of the recursion the cost of a single arithmetic operations is bounded by $d^{O(k^2)}$ arithmetic operations in D, we ignore the fact that the ring changes as we go down in the recursion. Thus, we have the following recurrence,

$$T(s, d, \ell, k) \leq \sum_{0 \leq j < \ell} \binom{O(s)}{j} d^{O(k)} T(s, d, \ell - j, k - 1) + O(s^{\ell+1} d^{O(k)}), \ell > 0, k > 1,$$

$$T(s, d, 0, k) = s d^{O(k^2)}, k > 1,$$

$$T(s, d, \ell, 1) = s \log s d^{O(1)}.$$

This recurrence solves to $T(s, d, \ell, k) = s^{\ell+1} d^{O(k^2)}$.

It follows immediately that the total cost is still bounded by $s^{\ell+1} d^{O(k^2)}$.

5 The Algorithm for the General Case

5.1 Computing a set of connecting triples

The above algorithm gives a roadmap for a set defined by a single weak sign condition on the family of polynomials $\mathcal{P}$. In order to compute roadmaps for arbitrary semi-algebraic sets defined by the polynomials in $\mathcal{P}$ we first compute a set V of *connecting triples* $(p_1, p_2, \gamma_{1,2})$, where p_1, p_2 are points, and $\gamma_{1,2}$ is a semi-algebraic path joining, p_1 to p_2. The set V has the following property: For any two semi-algebraically connected components, C_1 and C_2, of sets S_{σ_1} and S_{σ_2}, where σ_1 and σ_2 are sign conditions on $\mathcal{P}$, such that $\bar{C}_1 \cap C_2 \neq \emptyset$, there exists $(p_1, p_2, \gamma_{1,2}) \in V$, such that $p_1 \in C_1$, $p_2 \in C_2$ and $\gamma_{1,2} \setminus \{p_2\} \in C_2$.

Thus, if C_1 and C_2 are two semi-algebraically connected components of two distinct sign conditions which are included in S and whose union is semi-algebraically connected then there exists $(p_1, p_2, \gamma_{1,2}) \in V$ such that $\gamma_{1,2}$ connects the point $p_1 \in C_1$ with the point $p_2 \in C_2$ by a semi-algebraic path lying in $C_1 \cup C_2$.

In order to compute the set$V$ we introduce two infintesimals $\delta' \ll \delta$, and consider the set $\mathcal{P}' = \cup_{P \in \mathcal{P}} \{P, P \pm \delta', P \pm \delta\}$.

We call the sample points subroutine with the family $\mathcal{P}'$ as input, and construct a set of points in $(R\langle\delta, \delta'\rangle)^k$ which intersects every semi- algebraically connected component of the family $\mathcal{P}'$.

For each point p_1 constructed above, we construct the point $p_2 = \mathrm{eval}_{\delta'}(p_1)$. The point p_1 is represented as a k-univariate representation (u, σ) with coefficients in $D[\delta, \delta']$. Replacing, δ' in u by a new variable t, and letting t vary over the interval $[0, \delta']$ we get a semi-algebraic path $\gamma_{1,2}$ joining p_1 to p_2. We include the triple $(p_1, p_2, \gamma_{1,2})$ in the set V.

We are finally in a position to describe the algorithm in the general case. We first compute the set of connecting triples V. We then make a perturbation of the polynomials in $\mathcal{P}$ as follows.

We replace the set $\mathcal{P}$ by the following set, $\mathcal{P}^*$, of $4s$ polynomials:

$$\mathcal{P}^* = \cup_{i=1,\dots,s} \{(1-\epsilon')P_i - \epsilon H_{4i-3}, (1-\epsilon')P_i + \epsilon H_{4i-2},$$
$$(1-\epsilon')P_i - \epsilon'\epsilon H_{4i-1}, (1-\epsilon')P_i + \epsilon'\epsilon H_{4i}\},$$

where $H_i = (1 + \sum_{1 \le j \le k} i^j x_j^{d'})$ and d' is an even number greater than the degree of any P_i.

We have the following two lemmas which show that the family $\mathcal{P}^*$ is in general position, and that there exists a correspondence between the cells of $\mathcal{P}$ and $\mathcal{P}^*$. Very similar lemmas occur in [1].

Lemma 4 *The combinatorial level of the system $(0, \mathcal{P}^*)$ is at most k.*

Lemma 5 *Let σ be a realizable strict sign condition on the set of polynomials $\mathcal{P}$ given by $P_1 = \ldots = P_\ell = 0, P_{\ell+1} > 0, \ldots, P_s > 0$, and let C be a semi- algebraically connected component of σ. Let, σ' be the weak sign condition on $\mathcal{P}^*$ given by,*

$$-\epsilon\epsilon' H_{4i} \le (1-\epsilon')P_i \le \epsilon'\epsilon H_{4i-1}, 1 \le i \le \ell,$$
$$(1-\epsilon')P_i \ge \epsilon H_{4i-3}.$$

Then, there exists a unique semi-algebraically connected component C' of the sign condition σ', such that $\mathrm{eval}_{\epsilon'}(C') \subset C_{R\langle\epsilon\rangle}$.

Moreover, if $x \in R^k$ is in C then $x \in C'$.

We now use the algorithm for constructing a uniform roadmap with input $(0, \mathcal{P}^*, M^*)$, where M^* is the set of points obtained by adding the points appearing in the triples of V to M. We add the curves $\gamma_{1,2}$ occurring in the triples of V to the output of the uniform roadmap algorithm. We then compute the image of the roadmap constructed above under the $\mathrm{eval}_{\epsilon'}$ map, and retain only those portions which are in the given set S.

5.2 Proof of Correctness and Complexity

The proof of correctness follows from the proof of correctness of the uniform algorithm and lemmas 4 and 5.

The combinatorial level of the system $(0, \mathcal{P}^*)$ is bounded by k. Using the complexity bound of the uniform algorithm and noting that the cost of computing the set of connecting triples is $s^{k+1}d^{O(k)}$, we see that the complexity is bounded by $s^{k+1}d^{O(k^2)}$, provided that $m = s^k d^{O(k)}$. For larger m a more careful analysis yields a complexity of $(s^k + m \log m)sd^{O(k^2)}$.

References

[1] S. Basu, R. Pollack, M.-F. Roy *On the combinatorial and algebraic complexity of Quantifier Elimination*, *Proc. 35th IEEE Symp. on Foundations of Computer Science*, 632-642, (1994), (Journal version to appear in JACM).

[2] J. Bochnak, M. Coste, M.-F. Roy Géométrie algébrique réelle. Springer-Verlag (1987).

[3] J. Canny, The Complexity of Robot Motion Planning, MIT Press (1987).

[4] J. Canny, *Computing road maps in general semi-algebraic sets*, The Computer Journal, 36: 504–514, (1993).

[5] G. E. Collins, *Quantifier elimination for real closed fields by cylindrical algebraic decomposition.* Springer Lecture Notes in Computer Science 33, 515-532.

[6] D. Grigor'ev, N. Vorobjov *Counting connected components of a semi-algebraic set in subexponential time.* Comput. Complexity, 2 :133-186 (1992).

[7] L. Gournay, J. J. Risler *Construction of roadmaps of semi-algebraic sets.* Technical Report, DMI, Ecole Normale Superieure, Paris, France (1992).

[8] R. M. Hardt *Semi-algebraic Local Triviality in Semi-algebraic Mappings.* American Journal of Math. 102, 1980, 291-302.

[9] J. Heintz, M.-F. Roy, P. Solernò *Single exponential path finding in semi-algebraic sets II : The general case.* Algebraic geometry and its applications. C. L. Bajaj editor, Springer Verlag (1993) 467-481.

[10] J. Milnor *Morse Theory*, Annals of Mathematical Studies, Princeton University Press.

[11] J. Renegar *On the computational complexity and geometry of the first order theory of the reals*, J. of Symbolic Computation, (1992).

[12] J. Schwartz, M. Sharir *On the 'piano movers' problem II. General techniques for computing topological properties of real algebraic manifolds.* Advances in Applied Mathematics 4, 298-351.