Convex Approximation by Spherical Patches

Kevin Buchin[*‡]   Simon Plantinga [†‡]   Günter Rote[*‡]   Astrid Sturm[*‡]   Gert Vegter[†‡]

Abstract

Given points in convex position in three dimensions, we want to find an approximating convex surface consisting of spherical patches, such that all points are within some specified tolerance bound ϵ of the approximating surface. We describe a greedy algorithm which constructs an approximating surface whose spherical patches are associated to the faces of an inscribed polytope. We show that deciding whether an approximation with not more than a given number of spherical patches exists is NP-hard.

1 Introduction

Problem Statement. We are given a set P of n three dimensional points in convex position. We want to find a convex approximating surface S that consists of *spherical patches*. A spherical patch is part of the boundary of a sphere. There are two quality criteria that we want to optimize: (a) the approximation error, which is defined as the maximum distance from a point of P to S; and (b) the number of patches.

Motivation. Our motivation for studying this problem is based on open problems in polytope approximation as well as on practical considerations. Surface reconstruction and surface simplification is an important area of computer graphics and geometric modeling [1]. One goal is to approximate complex objects by simpler shapes. A lot of research has been done in the field of approximation of three dimensional point sets with polytopes with surfaces of higher order [3].

A first natural step to higher order approximation is the approximation with spheres or spherical patches. Since polyhedral facets can be seen as spherical patches with infinite radius, spherical patch approximation generalizes polytope approximation.

We initiate the study of this problem by considering convex surfaces only, for simplicity. The results might nevertheless be interesting for real data sets, e.g. data sets from imaging procedures such as MRT. A wide range of objects scanned for data consist of *parts* that are convex, and thus our results remain valid at least on a local scale.

The complexity of the approximation problem is related to open problems in polytope approximation, in particular to the complexity of the minimum facet polytope approximation. We hope to use our new methods from the NP-hardness proof of the more generalized problem to solve the complexity question of the minimum facet polytope approximation.

Results and Techniques. We present an algorithm for solving the approximation problem with a specified error bound ε. It is based on a triangulated inscribed polytope which is the convex hull of a subset of the input points, and on which the spherical patches are built. This polytope is successively refined in a greedy manner. We attempt to extend the well-known Douglas-Peuker algorithm for polygonal line approximations of curves to our setting.

Proofs are omitted due to space constraints. A full version is available[1].

2 Approximation of a convex point set by Spherical Patches

Outline The optimization problem we are considering is the *Approximation by Spherical Patches problem* (ASP): the approximation of a convex point set with a number g of spherical patches resulting in a convex surface with all points within some specified tolerance to the surface. We construct a point set defining an instance of the ASP with zero tolerance such that - in the satisfiable case - a minimal solution of the approximation problem corresponds to a truth assignment in the NP-hard grid-3-SAT problem. This point set is lifted to a paraboloid and extended with additional points. We describe the minimal solution in the satisfiable case and prove that more patches than g are needed in the non-satisfiable case for the ASP.

Grid-3-Satisfiability 3-SAT statements consist of a Boolean conjunction of clauses, where each clause consists of a disjunction of three boolean variables, each of which may be negated. Such a statement can be represented by a bipartite graph, where variables

---

*Institut für Informatik, Freie Universität Berlin, {`buchin, rote, sturm`}`@inf.fu-berlin.de`

†University of Groningen Department of Mathematics and Computing Science, {`simon, gert`}`@cs.rug.nl`

---

[1]`http://page.mi.fu-berlin.de/ sturm/Spheres.pdf`

and clauses are represented by vertices. Each clause vertex is connected to its three variable vertices by an edge marked $+$ or $-$ depending on whether this variable occurs negated in that clause. The 3SAT problem is NP hard even if the variable-clause graph of a formula of length n in 3-conjunctive normal form can be embedded on a $c \cdot n^2$ grid with c some constant [2].

Modifying the grid The first step of the reduction requires a refinement of the grid, vertices correspond to facets and edges correspond to rectilinear paths on the grid. Further we disperse the grid cells by a small constant factor δ which creates free space between the cells. Depending on the label of the edge in the variable clause graph, we change the number of facets in the path on the grid corresponding to the edge. A negatively labeled edge is represented by a path with an odd number of cells and a positively labeled edge corresponds to a path with an even number of cells. To achieve this correspondence we need sufficiently many cells on a straight path. The inclusion of an additional cell is done by reducing the size of the cells in a straight segment of the path and fitting in another cell of this size. Next we delete all grid cells corresponding to clauses (the clauses will be represented later by a single point). We also drop the lower right vertex of each grid cell.
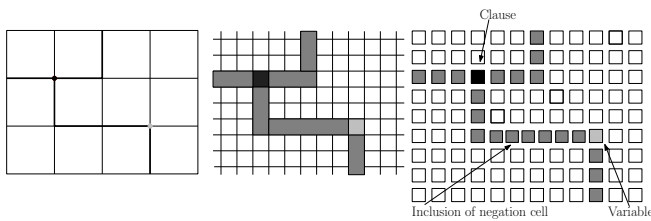

Figure 1: refinement of the grid

Lifting to a paraboloid The next step is a lifting of the point triples of the grid cells onto a very flat paraboloid. The distance between vertices in one grid cell is set to one. For a $c \cdot (n \times n)$ grid we pick a paraboloid of the form $z = \lambda \cdot (x^2 + y^2)$. The parameter λ has to be chosen in such a way that for two neighboring point triples the disks D_i and D_j corresponding to the circles C_i and C_j intersect. This guarantees the existence of valid spherical patches. For a lifting of a δ dispersed $c \cdot (n \times n)$ grid this leads to an upper bound on λ of $\frac{\sqrt{\left(1-1/\sqrt{2}\right)^2 - \delta^2}}{\delta^2 + 2\delta(c+\delta)n}$. For this, δ has to be chosen less than $1 - 1/\sqrt{2}$. For our explicit construction we choose $\delta := 1/10$ and $\lambda := 1/(10m)$ with $m = c(1+\delta)n$ a bound on the width and length of the dispersed grid (see Theorem 4).

Fill points After lifting the point set we place one point into each triangular face defined by point triples

corresponding to grid cell vertices of cells which did not belong to the 3SAT. We refer to these four points as a set of fill-points.

Lemma 1 *Each set of fill points induces exactly one spherical patch and all sets cannot be covered with less than one patch per set.*

Wire The wire corresponds to edges in the variable-clause graph. An edge in the variable-clause graph corresponds to a set of the lifted point triples. Each point triple P_i defines a circle C_i. These circles do not lie on the paraboloid, but (since λ is small) lie close to the lifted circumcircles of the base squares (which are ellipses). The supporting plane of each circle splits the space into an inner half space containing the convex hull of the original point set and an outer half space. Each circle C_i defines a *family of spheres*, i.e. set of all spheres induced by the circle C_i. Candidates for valid patches are only spheres with centers in the inner half space. Furthermore adjacent spherical patches need to intersect properly. The intersection of the outer half space with the circle C_j of an adjacent patch has to lie outside the spherical patch, i.e., the spherical patch should pass below the circular arcs of its neighbors.

We build a wire out of consecutive spherical patches to propagate information from the variables to the clauses. The main idea of the reduction is to place points on the intersection of consecutive spherical patches in the wire. These points narrow down the choice from a family of spheres to only two spheres for each patch a flat or bulbous patch. Furthermore the points force alternating spherical patches in the wire.

The approximating surface is constructed by taking the inner upper hull of the patch intersection. This is the surface of the intersection of the patch defining spheres. We need to guarantee that the additional points on the flat and bulbous patches will lie on the approximating surface (see Lemma 3). We place four points, F_k, on the intersection circle of consecutive flat patches and four points, B_{k_i}, on the intersection of bulbous patches. The points F_{k_i} lie on the circular arc which is in between the intersection of the defining circles and the points B_{k_i} lie on the circular arc outside the intersection (see Figure 2).

For the wire gadget we need to place a point (approximately) on the intersection of two neighboring "bulbous" spheres. If at least one of the two bulbous spheres is chosen the point must lie on the inner hull of the construction. In the following we formulate conditions under which a point lies on the inner hull. Then we pick such a point and prove that the conditions hold.

Since the radii of the bulbous spheres have been chosen in such a way that they only come close to
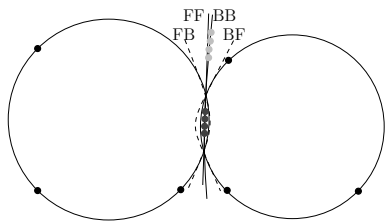
27

Figure 2: Placing of the points on the circular arc of the FF and BB patch intersection. The light gray points are on BB.

the grid polytope at the face by which it is defined, it suffices to consider the local configuration. The *grid polytope* is the polytope obtained by lifting all the grid vertices restricted to the region of the 3SAT construction. Thus, for a point to lie on the inner hull the following conditions are sufficient:

1. The point lies above (i.e. on the same side as the grid polytope) both of the planes defined by the two triples of points.

2. The point lies below (i.e. on the other side than the grid polytope) the face of the grid polytope between the two faces defining the spheres.

3. The projection of the point lies within (possibly on the boundary of) the face of the grid polytope between the two faces defining the spheres.

For two neighboring triples of points there are two points p_1 and p_2, one of each triple, neighboring in the grid. Let e be the plane orthogonal to the z-axis through these two points. For the intersection point of the plane e with the two spheres we can prove that the conditions above hold.

For a flat patch we choose the sphere with center at infinity, the plane defined by the point triple. For a bulbous patch we request that all points of the grid polytope except the point triple defining the patch lie inside the "bulbous" ball. This leads to a set of constraints on the radius of the ball by considering the radii of the balls defined by the point triple and a set of possible fourth points. These constraints can be fulfilled by a radius linear in n. To guarantee further that all points of the grid polytope except the point triple have distance of at least $1/n^2$ to the surface of the ball the radius can be chosen quadratic in n. We choose for all bulbous patches the same radius.

Variable A variable is a point triple which is handled as a wire point set. Choosing the flat patch corresponds to a 0 assignment and the bulbous patch to a 1 assignment. Choosing the flat patch will result in a covering of all flat points in the free space around the variable point triple, therefore all consecutive wire patches will propagate the same information

- all wires starting from this variable will start with a bulbous patch. The case of picking a bulbous patch for a variable point triple is symmetric.

Clause Before the lifting a clause corresponds to a grid cell in the plane which is connected to three wires (from three variables). The vertices of this grid cell are not lifted. In the lifted point set the clause corresponds to a single point. This point is placed in the free space between the three wire point triples and is the intersection point of the bulbous patches of these point sets.

Theorem 2 *There exists a true assignment for the grid 3SAT instance if and only if the lifted point set with all additional points can be approximated with s spherical patches. For a $c(n \times n)$ grid, with g clauses and t cells included for negation, $s = c(n^2) + t - g$*

Lemma 3 *All point sets F_{k_i} and B_{k_i} are on the approximating surface.*

Theorem 4 *For a SAT instance on a $c(n \times n)$ grid let P be the set of points in convex position constructed as above with $\delta := 1/10$, $\lambda := 1/(10m)$, $m := \sqrt{2}(1 + \delta)cn$, and the common radius of the bulbous spheres $r := 10m$. Let P' equal P with the exception that the points on the bulbous-bulbous sphere intersections might be displaced by $\epsilon := 1/m^2$. For sufficiently large m the following holds: If the SAT instance is feasible, then there is a surface with g patches such that all points have distance at most ϵ to a patch. If the SAT instance is infeasible, then for every surface with at most g patches there is at least one point which has distance more than 100ϵ from all patches.*

3 Greedy algorithm

In this section we present a construction of curved surfaces based on inscribed polytopes, resulting in a convex surface consisting of spherical patches. The inscribed polytope approach makes our construction suitable for various incremental algorithms. We start with a minimal inscribed polytope consisting of a tetrahedron, and incrementally extend this polytope until the corresponding surface is a valid approximation.

Constructing a curved surface In order to produce a valid surface, we require that spherical caps pass through triples of input points ensuring that adjacent caps intersect properly. The approximating surface is generated by a convex triangulation, in particular the convex hull of a subset of the input points. The triangles of this hull are called *supertriangles*. Our

goal is to inflate this polytope by replacing its faces with curved, spherical patches.

First we construct the spherical caps. The supporting plane of each supertriangle splits space into an inner halfspace containing the convex hull, and an outer halfspace. For each supertriangle we construct a spherical cap by first taking a sphere through its vertices with its center in the inner halfspace. Then we take the intersection of this sphere with the outer halfspace. The intersection of the outer halfspace with the circumcircle of an adjacent supertriangle has to lie outside the spherical cap (see Figure 3), i.e., the spherical cap should pass below the circular arcs of its three neighbors.
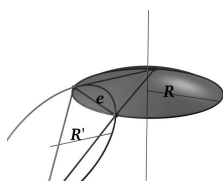

Figure 3: Supertriangle with spherical cap.

For each neighboring supertriangle, the circumcircle and dihedral angle give a lower bound on the radius of the spherical cap, to ensure that the cap is flat enough to pass below that circumcircle. Taking the maximum over the three adjacent supertriangles results in a single lower bound for the cap radius. The centre of the spherical cap now has to lie on a halfline perpendicular to the supertriangle. The approximating surface consists of the inner hull of the union of these spherical patches.

Lemma 5 *If neighboring spherical caps intersect properly, the inner hull of the union of caps forms a convex surface.*

Incremental construction We can now construct a curved convex surface from a subset S of the input points P. The convex hull of S generates a surface as long as the patch radii are large enough, to ensure proper intersection.

For an incremental approach, we initialize S to the four extremal points of the point set P, in the directions of the normals of a regular tetrahedron. Respecting the lower bound on the radii, we try to choose cap radii such that the caps are closer than ϵ to the remaining input points. If this is not possible we add more input points to S.

A supertriangle is *valid* if there exists a corresponding spherical cap with radius larger than its lower bound, such that all points inside the outer halfspace of the supertriangle are closer than ϵ to this spherical cap.

If all supertriangles of the inscribed polytope are valid, all input points lie closer than ϵ to the union

of caps. However, it is still possible that they are not ϵ-close to the inner hull of these caps, especially if adjacent supertriangles have a large dihedral angle or if some supertriangles are obtuse. We therefore have to test whether the input points that are ϵ-close to a cap but not to the patch, are ϵ-close to their nearest patch.

Testing supertriangles for validity results in more bounds for the patch radius. The centre of the spherical cap has to lie on the *centre line* of the supertriangle, which is the line passing through the circumcentre and is perpendicular to that triangle. If an input point inside the corresponding halfspace needs to be ϵ-close to the spherical cap, this condition gives an interval of valid cap centres on the centre line. If the intersection of all of these intervals together with the half-line is nonempty, the supertriangle is valid. Since the lower bound for proper intersection of caps corresponds to the surface being convex, we expect the intersection of these intervals to lie within that valid half-line.

First we test all supertriangles for validity. If there are invalid supertriangles, we add an input vertex to S and update the convex hull incrementally. We then test the validity for the newly constructed supertriangles and for previously invalid neighbors of new supertriangles. This way we gradually refine the approximating curved surface without having to revalidate the entire structure.

For an invalid supertriangle we choose the outlying input point corresponding to the smallest radius of the spherical cap. As we increase the number of spherical caps, we increase the radius. This is motivated by the fact that we want flatter patches as the dihedral angles between supertriangles decrease. The incremental algorithm now moves gradually from a curved surface consisting of four patches, to the entire convex hull of P.

The extra test for points close to caps but not to the corresponding patch can also reveal points that are further than ϵ from the approximating surface. These points are also added to S.

Lemma 6 *The greedy algorithm terminates.*

References

[1] T. K. Dey. Curve and surface reconstruction. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry, CRC Press, 1997, 2004*, volume 2. 2004.

[2] M. Godau. On the difficulty of embedding planar graphs with inaccuracies. In R. Tamassia and I. G. Tollis, editors, *Graph Drawing*, volume 894 of *LNCS*, pages 254–261. Springer, 1994.

[3] N. Kruithof and G. Vegter. Envelope surfaces. In N. Amenta and O. Cheong, editors, *Symposium on Computational Geometry*, pages 411–420. ACM, 2006.