Interfacing graphs

Thomas Kämpke
Forschungsinstitut für anwendungsorientierte Wissensverarbeitung FAW,
Helmholtzstr. 16, 89081 Ulm, Germany
email: kaempke@faw.uni-ulm.de

Abstract

Transformations of vertex sequences of a regular grid graph into paths of an arbitrary connected graph are facilitated according to various coarsening and approximation operations including minimum cost alterations and minimum cost re-routings. The sequence transformations are supposed to support issues of man-machine interaction which implies the lack of an ultimate formal design objective. Furthermore, this implies that formal methods and algorithms have to be combined in a pragmatic fashion.

For planar graphs the notion of Voronoi regions is modified to graph Voronoi regions which partition the plane according to proximity to vertices *and* edges simulatenously. The non-planar case is reduced to the planar case by adding all intersection points of vertex connections to the original vertex set and by splitting vertex connections accordingly. This allows grid point sequences to be intermediately transformed to so-called mixed or region sequences which are eventually transformed to vertex sequences by production rule-like operations.

The algorithmic preprocessing burden of partitioning and indexing the Euclidean plane via the graph Voronoi regions or approximations thereof is practically larger and typically more complicated than any of the run time computations.

Key words: graph Voronoi region, grid graph, regular expression, touch screen.

1 Introduction

The well separated fields of graph algorithms and graphical user interfaces have received considerable attention in the past. Narrowing the gap between the fields appears only to occur when the output of more or less complex graph computations or spatial relations are to be presented in an intuitive way. Here, problems related to the input of graphs esp. interactions with and selections from a formal graph structure are investigated.

Therefore, an undirected graph in the Euclidean plane is overlayed with a regular grid which is finer than the graph. This will mean that two distinct graph vertices have distinct closest grid points. A grid point is any intersection of grid lines. The interplay between the graph and the grid gives rise to a variety of questions such as how to transform a sequence of grid points into a (meaningful) path in the graph. This task resembles the classical raster vector conversion problem from computer graphics with the essential difference that "vectors" cannot be chosen arbitrarily here but have to be selected from the vertex set and the edge set of the graph.

There is no true or ultimate transformation of grid point sequences to paths. One reason is that the "correct" path may adhere to ergonomical, aesthetic, or other, non-operational criteria. Subsequent solutions of the transformation problem should be considered as proposals which may be plugged together in different manners. Also, the underlying formal constructions may be modified in manifold ways. One major modification is a problem setting that completely refers to raster graphics. This approach has not been chosen in the first place due to conceptual clarity and storage requirements.

The motivation to consider this problem stems from the increasing variety and availability of man-machine interfaces of a non-keyboard type. The grid can be thought of as being sensitive to touches by human fingers like a grid of a touch screen. The relation between grid points and an underlying screen content may be obvious for a static and "local" content where pointing serves for highlighting and selecting an icon or an option from a menu. Dynamic pointing operations serve for tasks such as moving a scroll bar, moving an icon or obtaining artistic or entertaining effects from drawing with digital ink. The latter refers to pen/tablet systems as well as to particular approaches like touchless deposition of virtual ink on a virtual 3D terrain [Co, p. 13]. Here, we relate dynamic pointing to a given structure that has some meaning concerning locations but that does not imply a-priori restrictions in direction as well as in initial and terminal touch positions. Applications of the approach include methods for input to spatial planning systems such as navigation systems and techniques for commanding motion of autonomous or partially autonomous mobile systems.

The paper is structured as follows. Section 2 introduces graph and grid notations as well as the grid-graph interplay. Section 3 presents transformations from grid point sequences to paths in the graph. The difficulty to overcome by these transformations stems from the regular neighbourhood relation of the grid and the possibly irregular neighbourhood relation of the graph being defined independent from each other. The transformations operate on two levels. The lower level utilizes geometric concepts while the upper level operates on mixed sequences of vertices and edges.

Proximity concepts for the lower level are established in a manner which is free of threshold values. This idea, originally pursued in signal analysis [KäKo1], [KäKo2], ensures robustness by not requiring to set parameters such as neighbourhood sizes. The well known concept of Voronoi regions will be modified in order to account for proximity towards vertices *and* towards connections between vertices. These regions are denoted as graph Voronoi regions. Sequence operations on the upper level can be considered as production rules for regular expressions.

Section 4 of this work focusses on computational issues. Graph Voronoi regions are approximated in vectorized as well as in rastered form. Also, the avoidance of vertex repetitions in paths is treated by constructing a surrogate graph and computing shortest paths there. A major example is presented in section 5. The reader interested in final results only may skip the intermediate sections. Section 6 concludes with a selection of future issues.

The set of real numbers will be denoted by $\mathbb{R}$ and ends of arguments will be denoted by $\diamond$. The Euclidean distance is abbreviated by $||\cdot|| = ||\cdot||_2$ and $:=$ indicates a definition. The circle with radius r around $x \in \mathbb{R}^2$ is denoted by $C_r(x) = \{y|\ ||y-x|| \leq r\}$. The first element of a sequence X is abbreviated by $fi(X)$ and the last element of that sequence is abbreviated by $la(X)$. The big-O notation is applied as usual with $O(f(n)) = \{g(n)|\ \exists C > 0$ such that for all n: $0 \leq g(n) \leq Cf(n)\}$.

2 Graphs and grids

2.1 Graphs

An undirected graph in the Euclidean plane is denoted by $G = (V, E)$ with vertex set $V \subseteq \mathbb{R}^2$. The graph is supposed to be simple meaning that there is at most one edge between any two vertices and no edge connects a vertex with itself, i.e. there are no loops. Each edge e is labeled by a non-negative length $c(e) = c_e$ which may or may not be the Euclidean distance between the end points of the edge.

Vertices joined by an edge are called adjacent. An edge is called incident with both its vertices. Two edges are incident whenever they intersect at a common vertex. A path in G is a sequence of vertices with successive vertices being adjacent. A trivial consequence from the graph being simple is that successive vertices of a path are distinct but nevertheless, repeated visits to vertices are feasible within a path. The length or cost of a path $P(v, w) = (v = v_1, \ldots, v_s = w)$ from v to w with $\{v_1, v_2\}, \ldots, \{v_{s-1}, v_s\} \in E$ is $c(P(v, w)) := \sum_{i=1}^{s-1} c(v_i, v_{i+1})$. A shortest path from v to w will be denoted by $P_0(v, w)$. The degenerate case $P_0(v, v)$ is the single vertex v.

The length of a vertex sequence that need not be a path is defined as sum of labels of succesive vertex pairs, where any pair $\{v, u\}$ of vertices is assigned the cost

$$d(v, u) := \begin{cases} c(v, u), & \text{for } \{v, u\} \in E \\ c(P_0(v, u)), & \text{for } \{v, u\} \notin E. \end{cases}$$

The length of a shortest path is not assigned persistently to the vertex pairs in order to preserve original edge labels even when they violate the triangle inequality.

Drawings of graphs assign absolute positions to vertices and edges rather than mere graphs which are (geometrical) representations of adjacency relations. The particular curve representing an edge is unessential in a graph but it can be of importance in a drawing. Whenever the mere adjacency relation of an edge between vertices v_1 and v_2 is meant, the edge is denoted by $\{v_1, v_2\}$. Whenever the drawing is alluded to, the curve which is the continuous point set connecting the two vertices is denoted by $cur(v_1, v_2)$. The distinction between a graph and one of its drawings becomes relevant in connection with proximity sets such as Voronoi regions; they make sense only in drawings. Edges and curves are symmetric in the sense that $e = \{v_1, v_2\} = \{v_2, v_1\}$ and $cur(v_1, v_2) = cur(v_2, v_1)$. In general, distinct curves are allowed to intersect at several points in the plane, but they may not have sections in common.

If not stated otherwise, all graphs are supposed to be connected. Particular focus will be given to planar graphs. These graphs admit drawings whose edges only intersect at graph vertices. Any drawing of a planar graph will have this property.

2.2 Grids

The grid consists of equidistant horizontal and vertical lines intersecting at the grid points. Distances between adjacent vertical and horizontal lines are equal resulting in the grid being regular with grid width $d > 0$. Both the numberings of vertical and horizontal lines are consecutive so that numbers of adjacent lines differ by one. The grid can be considered as a graph $Gri = (P, Li)$. For the sake of simplicity the grid is assumed to be unbounded.

Each grid point has eight neighbours. These are the grid points whose row and column indices differ at most by one from those of the grid point under consideration. The neighbourhood set of a grid point p is denoted by $N(p)$. The extended neighbourhood of a grid point includes the grid point itself, i.e. $N'(p) := N(p) \cup \{p\}$.

2.3 Grid graph relation

A graph vertex with smallest Euclidean distance towards a grid point p is denoted by $v(p)$ so that $v(p) := argmin_{v \in V} ||v - p||_2$. Tie breaking between non-unique closest graph vertices is discussed below. A grid point with smallest Euclidean distance towards graph vertex v is denoted by $p(v)$ so that $p(v) := argmin_{p \in P} ||p - v||_2$. assumed that the grid is finer than or dense compared to the graph meaning that distinct graph vertices lie apart by at least the distance of adjacent grid lines. This implies distinct graph vertices having distinct closest grid points.

The Voronoi region of some graph vertex is the set of all points with smaller distance to that vertex than to any other vertex, $V(v) := \{x \in I\!R^2 | \, ||x - v|| \leq ||x - w|| \; \forall \, w \in V - \{v\}\}$. The vertex around which a Voronoi region is formed is called germ or center of the Voronoi region. When clear from the context, Voronoi regions will be assumed to consist only of the grid points contained in the proper Voronoi regions. The density assumption of the grid graph relation implies that no grid point set of a Voronoi region is empty or contained in another.

The main problem considered here is to transform a sequence of grid points into a sequence of graph vertices under different conditions and objectives. Typically, the vertex sequence will be stipulated to be a path in the graph even if the point sequence is not a path in the grid, comp. figure 1. The difficulty of such transformations arises from the neighbourhood relations in the graph and in the grid being completely unrelated. An additional difficulty may come from the grid and the grid point sequence as well as further graph objects such as Voronoi regions being invisible in typical applications.

3 Sequence transformations

A sequence of grid points is understood to be connected if each grid point is an extended neighbour of its predecessor. Noteworthy, a connected sequence of grid points – even if free of repetitions – need not be a path in Gri since successive grid points need not be adjacent. The grid point sequence $(p^{(1)}, \ldots, p^{(N)})$ induces the sequence of closest graph vertices $\bar{v}(\bar{p}) := (v(p^{(1)}), \ldots, v(p^{(N)}))$. The induced sequence may

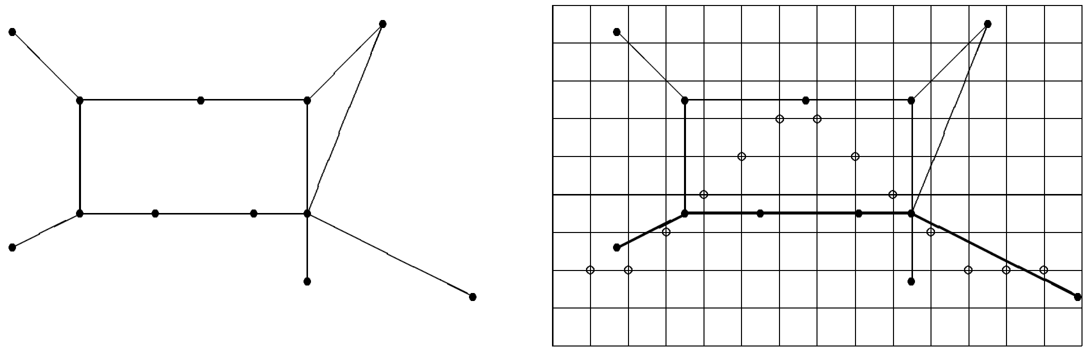

Figure 1: Left: graph G which forms the visible portion of the problem. Right: sequence of grid points (white dots) in Gri and path (with successive vertices joined by bold edges) in G.

contain successive vertices which are not distinct even if the sequence of grid points does not contain any repetition.

It is necessary for the induced sequence to be a path that successive vertices are distinct. This property is obtained by forming the trace of any vertex sequence which indicates the changes within the sequence. Formally, the trace of sequence $(v^{(1)}, \ldots, v^{(M)})$ with $v^{(j_1)} = v^{(1)} = v^{(2)} = \ldots = v^{(j_2-1)} \neq v^{(j_2)} \ldots \neq v^{(j_M)} = v^{(M)}$ is the subsequence $tr(v^{(1)}, \ldots, v^{(M)}) := (v^{(j_1)}, \ldots, v^{(j_M)})$. An example is $tr(v_4, v_4, v_3, v_4, v_5, v_5, v_5, v_4, v_7) = (v_4, v_3, v_4, v_5, v_4, v_7)$. The trace of a vertex sequence induced by a grid point sequence still need not be a path even if the grid point sequence is a path.

Induced vertex sequences may contain ambiguities in case a grid point $p^{(i)}$ lies on the boundary of two or more Voronoi regions. Such ties have to be broken in order for a trace to be well defined. The last vertex before $p^{(i)}$ which lies in the interior of some Voronoi region serves as a tie breaker when this region is one of the ambiguous Voronoi regions. If this still leaves ambiguities, the first vertex succeeding $p^{(i)}$ which is in the interior of some Voronoi region serves as a tie breaker in the same way. If these two breaking rules still leave ambiguities, which can easily occur for non-connected grid point sequences, the tie between ambiguous Voronoi regions is broken arbitrarily.

The order in which grid points traverse Voronoi regions or modified Voronoi regions that are specified below does not affect the induced vertex sequence. This is an important though informal robustness feature of the sequence transformations which are developed here. Another feature covered by the term robustness here is concerend with the subsequent reproduction property. Traversing a path $(v^{(1)}, \ldots, v^{(M)})$ in G along the curves $cur(v^{(1)}, v^{(2)}), \ldots, cur(v^{(M-1)}, v^{(M)})$ induces a sequence of closest grid points. From this particular grid point sequence, the original path should be reproduced.

3.1 Tracing

Whenever $tr(\bar{v}(\bar{p}))$ is a path, this path is considered to be the transform of the grid point sequence $\bar{p}$. Geometrically, the trace of the induced sequence being a path means that Voronoi regions which are successively visited by the grid points belong to adjacent graph vertices.

Whenever $tr(\bar{v}(\bar{p}))$ is not a path, it is a reasonable starting point for constructing paths when $\bar{p}$ is connected. If the inducing grid point sequence does not matter, a vertex sequence $\bar{v}$ without successive repeatings, which then is the trace of an arbitrary sequence rather than of an induced sequence, is considered. This slightly generalizes assumptions and simplifies notations.

3.2 Insertions

A vertex sequence which is not a path can be transformed into a path by inserting vertices between any vertex $v^{(i)}$ and its successor with $\{v^{(i)}, v^{(i+1)}\} \notin E$. This is an immediate consequence of the graph being connected. Vertex insertions will be obtained from a shortest path between $v^{(i)}$ and $v^{(i+1)}$. Vertex insertion may adhere to constraints such as an inserted path may not use vertices from the sequence $\bar{v}$ or no two insertions may use a common vertex. The first constraint can be met by constructing a shortest path from $v^{(i)}$ to $v^{(i+1)}$ in the induced graph. This graph consists of vertices from $V - \bar{v} \cup \{v^{(i)}, v^{(i+1)}\}$ while its edge set consists of all edges from E whose both vertices belong to the restricted vertex set.

A subtle problem to which reference will be made several times in this work, is the decision whether to accept or reject a path insertion which uses vertices already in the vertex sequence. Revisiting a vertex amounts to moving back either along a partial path or by looping. Deciding on acceptance or rejection or even on a modification of the vertex sequence in case of an insertion appears to be possible only be external decision support. The issue is illustrated by figure 2, below. Under the condition of vertex $v^{(3)}$ being included in the induced vertex sequence, should the path be $(v^{(1)}, v^{(2)}, v^{(1)}, v_l, v^{(3)}, v_r, v^{(5)}, v^{(4)}, v^{(5)})$ or should it be shortcut to $(v^{(1)}, v_l, v^{(3)}, v_r, v^{(5)})$?

3.3 Deletions

In case vertex sequence $\bar{v}$ is not a path, the first pair $\{v^{(i)}, v^{(i+1)}\} \notin E$ is searched for $\{v^{(i)}, v^{(j)}\} \in E$ with minimum value $j \geq i+2$. In case such an index exists all vertices $v^{(i+1)}, \ldots, v^{(j-1)}$ are deleted from $\bar{v}$. If the new sequence does not form a path, continuing these deletion operations may eventually lead to a path but this cannot be guaranteed.

3.4 Insertions and deletions

The mixture of insertions and deletions – also known as indels from string editing [Gu] – may adhere to several objectives. A reasonable feature to decide on insertion and deletion is connectivity, Therefore, a vertex of a vertex sequence is understood to be isolated from the sequence, if the vertex is neither joined to its predecessor nor to its successor in the sequence. A vertex that is not isolated from a sequence is connected to that sequence. A vertex $v^{(i)}$ is understood to be a single isolated vertex if

$$\{v^{(i-2)}, v^{(i-1)}\} \in E,\ \{v^{(i-1)}, v^{(i)}\} \notin E,\ \{v^{(i)}, v^{(i+1)}\} \notin E,\ \text{and}\ \{v^{(i+1)}, v^{(i+2)}\} \in E.$$

Noteworthy, the last vertex before and the first vertex after the gap may be the same, i.e. $v^{(i-1)} = v^{(i+1)}$.

A vertex sequence will receive insertions so that a single isolated vertex $v^{(i)}$ becomes connected if it lies on a shortest path from $v^{(i-1)}$ to $v^{(i+1)}$. The shortest path need not be unique but one such path is selected to connect $v^{(i)}$. In case a detour from any shortest path were required to connect $v^{(i)}$ the vertex is deleted and again a shortest path is inserted from $v^{(i-1)}$ to $v^{(i+1)}$.

The trace of a vertex sequence induced by a connected grid point sequence can have a single isolated vertex as in figure 2. In this example the single isolated vertex is deleted by the previous procedure. On

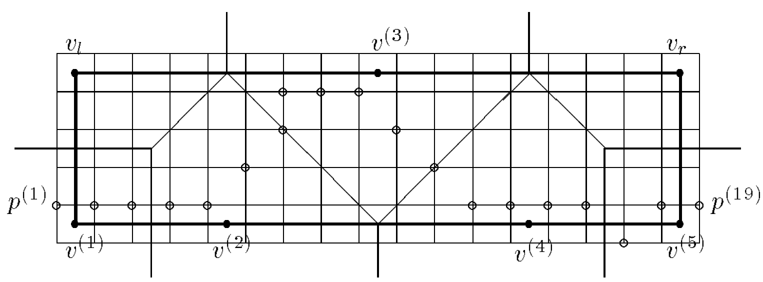

Figure 2: Planar graph with seven vertices and edges given by bold lines. The Voronoi regions are specified by their boundaries given as thin lines. These lie on perpendicular bisectors of suitable vertex pairs. To repeat, the boundaries of the Voronoi regions, the grid lines, and the grid points – either of a particular sequence or unspecified ones – are assumed to be invisible. Vertex $v^{(3)}$ is a single isolated vertex in the sequence $(v^{(1)}, \ldots, v^{(5)}) = tr(\bar{v}(\bar{p}))$ which is induced by the connected grid point sequence $\bar{p} = (p^{(1)}, \ldots, p^{(19)})$ indicated by the white dots.

the other hand, a single isolated vertex may indeed be connected to a vertex sequence which is induced by a connected grid point sequence, see figure 3.

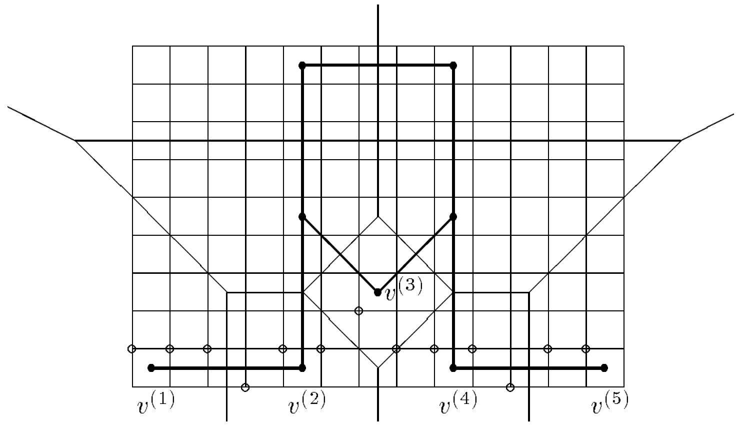

Figure 3: Vertex $v^{(3)}$ again is a single isolated vertex in the trace of the vertex sequence induced by the connected grid point sequence indicated by the white dots. As vertex $v^{(3)}$ is located on the unique shortest path from $v^{(2)}$ to $v^{(4)}$, it becomes connected to the vertex sequence.

The previous rule can be modified to connect a single isolated vertex only if a connecting shortest path does not use vertices already in the sequence or used for other insertions. However, this constraint may render path generation infeasible in certain cases like in figure 2.

3.5 Graph Voronoi regions for planar graphs

3.5.1 Simple modifications

The previous constructions are troublesome whenever a Voronoi region contains edge segments that are not incident with the germ of the Voronoi region, comp. figure 4. A grid point sequence closely following

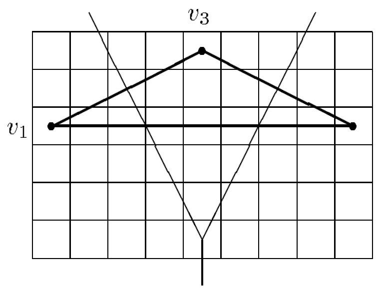

Figure 4: The Voronoi region $V(v_3)$ intersects with edge $\{v_1, v_2\}$ that is not incident with v_3.

the edge between vertices v_1 and v_2 traverses the Voronoi region $V(v_3)$. Since the edges $\{v_1, v_3\}$ and $\{v_2, v_3\}$ exist, vertex v_3 would be included in a vertex induced sequence. Marking the path (v_1, v_2) is only possible by the grid point sequence avoiding $V(v_3)$. This requires a non-intuitive detour around the annoying Voronoi region or it requires the grid point sequence to leap over that region.

Improving the separation strength or "resolution" of induced vertex sequences is addressed by shrinking the Voronoi regions. This can be achieved by several approaches one of them computing the minimum distance r between the germ and all segments of edges inside the Voronoi region that are not incident with the germ. No edge that is not incident with the germ is assumed to pass right through the germ. Even more, the distance between the germ and all non-incident edge segments is supposed to be at least twice the distance of adjacent grid lines d. The circular Voronoi region of a graph vertex v is then defined by $Vc(v) := V(v) \cap C_{r/2}(v)$. The value $r/2$ being at least the grid width ensures each circular Voronoi region to be not empty, it contains at least its germ. In figure 4, $Vc(v_3)$ contains only the four grid points forming the smallest possible grid square around v_3 while $Vc(v_1) = V(v_1)$ and $Vc(v_2) = V(v_2)$.

Circular Voronoi regions are conceptually simple but tend to be unnecessarily restrictive since restrictions are omnidirectional. A relaxation is obtained by considering distances to offending edge segments. For an arbitrary set $A \subseteq \mathbb{R}^2$ its distance to a point $x \in \mathbb{R}^2$ with respect to a Voronoi region is understood to be

$$dist_{V(v)}(x, A) := \begin{cases} \inf_{a \in A \cap V(v)} ||x - a||, & \text{for } A \cap V(v) \neq \emptyset \\ \infty, & \text{for } A \cap V(v) = \emptyset. \end{cases}$$

The exclusive Voronoi region of a graph vertex v is defined by

$$Ve(v) := \{x \in \mathbb{R}^2 | \, x \in V(v) \text{ and } ||x - v|| \leq dist_{V(v)}(x, cur(v_1, v_2)) \; \forall \, v_1, v_2 \in V - \{v\}\}.$$

A grid point in an exclusive Voronoi region thus is at least as close to the germ than to any segment of a non-incident curve. All previous density assumptions imply the inclusion chain $Vc(v) \subseteq Ve(v) \subseteq V(v)$ where each inclusion may be proper. The set of all exclusive Voronoi regions generally does not form a cover of the grid point set. Induced vertex sequences are then understood in the obvious way where a grid point which does not lie in any exclusive Voronoi region is eliminated from considerations after possible connectivity issues of the grid point sequence are resolved.

3.5.2 Partitioned and graph Voronoi regions

A higher resolution than by exclusive Voronoi regions is offered by the formation of Voronoi regions within Voronoi regions. A Voronoi region which is traversed by at least one curve segment that is not incident with the germ of the Voronoi region is partitioned according to proximity to curve segments. All curve segments that are incident with the germ will induce one class of the partition. Each curve that is not incident with the germ will induce a separate class. Formally, classes are defined as follows for vertices and curves that intersect a Voronoi region without being incident with the region's germ.

$$\begin{aligned} V_{V(v)}(v) \;\; &:= \;\; \{x \in V(v) | \, \exists \, cur(v, v_i) \text{ such that} \\ &\qquad dist_{V(v)}(x, cur(v, v_i)) \leq dist_{V(v)}(x, cur(v_k, v_l)), \\ &\qquad \forall \, cur(v_k, v_l) \text{ traversing } V(v) \text{ and } v_k, v_l \in V - \{v\}\} \\ V_{V(v)}(cur(v_i, v_j)) \;\; &:= \;\; \{x \in V(v) | \, dist_{V(v)}(x, cur(v_i, v_j)) \leq dist_{V(v)}(x, cur(v_k, v_l)), \\ &\qquad \forall \, cur(v_k, v_l) \text{ traversing } V(v) \text{ and } \{v_k, v_l\} \neq \{v_i, v_j\}\}. \end{aligned}$$

The conditions of the second definition allow $cur(v_k, v_l)$ to be incident with v so that either $v_k = v$ or $v_l = v$.

The grid point sets which result from this partitioning or which do not have to be partitioned are called partitioned Voronoi regions. A partitioned Voronoi region of type $V(v)$ or $V_{V(v)}(v)$ is called pure (partitioned) Voronoi region, a region of type $V_{V(v)}(cur(v_i, v_j))$ is called mixed (partitioned) Voronoi region. An example is given in figure 5. An original Voronoi region $V(v)$ that does not intersect with

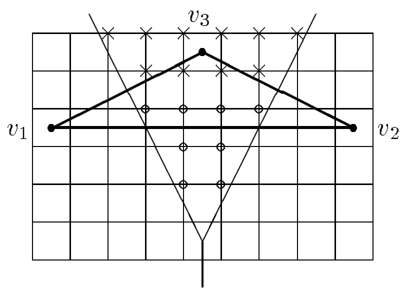

Figure 5: White dots indicate grid points in the mixed partitioned Voronoi region $V_{V(v_3)}(cur(v_1, v_2))$ and crosses denote grid points in the pure partitioned Voronoi region $V_{V(v_3)}(v_3)$ on the given section of the grid.

any curve of the form $cur(v_k, v_l)$, $v_k, v_l \in V - \{v\}$, coincides with its pure Voronoi region so that $V(v) = V_{V(v)}(v)$ and all its mixed Voronoi regions vanish meaning that $V_{V(v)}(cur(v_i, v_j)) = \emptyset$. Strictly, the ordinary Voronoi regions are covered rather than partitioned by pure and mixed Voronoi regions.

By admitting distance equality, the boundaries of modified Voronoi regions belong to at least two such regions as in the case of ordinary Voronoi regions.

The inclusion chain of modified Voronoi regions extends to $Vc(v) \subseteq Ve(v) \subseteq V_{V(v)}(v) \subseteq V(v)$ where again each inclusion may be proper. Mixed Voronoi regions are not included in this chain. Moreover, they need not even be connected sets since curves $cur(v_i, v_j)$ need not be straight lines.

So far, a Voronoi region is not partitioned whenever it contains only curve segments that are incident with the germ of the Voronoi region. Such a pure Voronoi region may contain grid points that are much closer to a curve segment outside that region than to the region's germ. This is not taken into consideration by the previous version of modified Voronoi regions. The final modification leads to what is called **graph Voronoi regions** by relaxing the traversal requirement and consequently relaxing the distance computations to the whole plane.

$$\begin{aligned} V_{V(v)}(v) &:= \{x \in V(v) \mid \exists\, cur(v, v_i) \text{ such that } dist_{\mathbb{R}^2}(x, cur(v, v_i)) \leq dist_{\mathbb{R}^2}(x, cur(v_k, v_l)), \\ &\qquad \forall\, cur(v_k, v_l) \text{ with } v_k, v_l \in V - \{v\}\} \\ V_{V(v)}(cur(v_i, v_j)) &:= \{x \in V(v) \mid dist_{\mathbb{R}^2}(x, cur(v_i, v_j)) \leq dist_{\mathbb{R}^2}(x, cur(v_k, v_l)), \\ &\qquad \forall\, cur(v_k, v_l) \text{ with } \{v_k, v_l\} \neq \{v_i, v_j\}\} \text{ for } v_i, v_j \in V - \{v\}. \end{aligned}$$

This modification leaves the pure and modified Voronoi regions of figure 5 unchanged. However, figure 6 depicts a situation which favours vertex reassignments according to the definition of graph Voronoi regions.

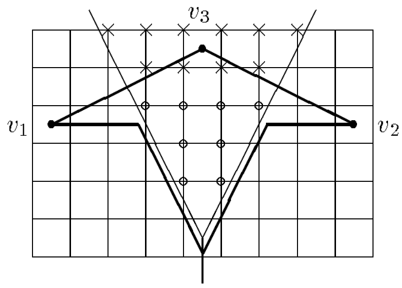

Figure 6: White dots indicate $V_{V(v_3)}(cur(v_1, v_2))$ and crosses indicate $V_{V(v_3)}(v_3)$. The first modification of Voronoi regions would leave the original region $V(v_3)$ unchanged since the curve between v_1 and v_2 does not traverse it. Formation of graph Voronoi regions leads to reassignments of vertices from $V(v_3)$ that are closer to $cur(v_1, v_2)$ than to v_3. Also, $V(v_1) = V_{V(v_1)}(v_1)$ and $V(v_2) = V_{V(v_2)}(v_2)$ since all curves are either incident with v_1 or v_2.

An immediate consequence of the definitions of pure and mixed graph Voronoi regions is that a pure Voronoi region coincides with the original (unmodified) Voronoi region if and only if all its mixed Voronoi regions are empty. Moreover, graph Voronoi regions are symmetric in the following sense. A curve that traverses an original Voronoi region without being incident with the germ affects grid points of this region in the same way as a curve that passes by. This is supposed to increase robustness of transformations of grid point sequences to paths. An ordinary Voronoi region is partitioned into its pure and at least one mixed graph Voronoi region if and only if

- either the ordinary Voronoi region $V(v)$ is traversed by a curve that is not incident with the germ v
- or if the midpoint of the shortest line segment connecting any of the curve segments $cur(v, v_i) \cap V(v)$ with a curve $cur(v_k, v_l)$, $v_k, v_l \in V - \{v\}$, lies inside $V(v)$.

A curve with non-void mixed Voronoi region is called influencial for that ordinary Voronoi region which comprises the mixed Voronoi region.

The grid point set within a pure Voronoi region is connected meaning that any two grid points of a pure Voronoi region can be reached from each other by a sequence of adjacent grid points that belong to that region. Mixed Voronoi regions need not be connected.

A grid point from a pure Voronoi region $V(v)$ or $V_{V(v)}(v)$ will induce vertex v and a grid point from a mixed Voronoi region $V_{V(v)}(cur(v_i, v_j))$ will induce the edge $\{v_i, v_j\}$. Formally, for any $p \in P$

$$ind(p) := \begin{cases} v, & \text{for } p \in V(v) \text{ or } V_{V(v)}(v) \text{ for some } v \in V \\ \{v_i, v_j\}, & \text{for } p \in V_{V(v)}(cur(v_i, v_j)) \text{ for some } v \in V \text{ and } v_i, v_j \in V - \{v\}. \end{cases}$$

Grid points from mixed Voronoi regions of the same edge induce that very edge even if the mixed Voronoi regions belong to different ordinary Voronoi regions; this means for example, that both $V_{V(v_1)}(cur(v_i, v_j))$ and $V_{V(v_2)}(cur(v_i, v_j))$ lead to $\{v_i, v_j\}$ whenever $v_i, v_j \in V - \{v_1, v_2\}$.

3.6 From mixed sequences to paths in planar graphs

3.6.1 Operations on sequences

A sequence of vertices and edges is called a mixed sequence or vertex-edge sequence. The mixed sequence induced by a grid point sequence $\bar{p} = (p^{(1)}, \ldots, p^{(N)})$ is denoted by $ind(\bar{p}) = (ind(p^{(1)}), \ldots, ind(p^{(N)}))$. The trace of a mixed sequence is understood in analogy to the trace of a vertex sequence. An example is $tr(v_7, v_7, v_7, \{v_7, v_{11}\}, \{v_7, v_{11}\}, v_{11}, v_{11}, v_{11}, \{v_{11}, v_{15}\}, \{v_{11}, v_{15}\}, v_{15}, v_{15}, v_{15}) = (v_7, \{v_7, v_{11}\}, v_{11}, \{v_{11}, v_{15}\}, v_{15})$. The latter indicates the vertex sequence (v_7, v_{11}, v_{15}) which is a path in G since $\{v_7, v_{11}\}$ and $\{v_{11}, v_{15}\}$ are obtained from mixed Voronoi regions which is possible only when the two curves and thus the edges exist in E.

For the sake of simplicity, the transition from mixed sequences to vertex sequences is first derived for special situations only under the loss of generality. A mixed sequence is said to have the path property with respect to the graph $G = (V, E)$ if successive entries of the sequence are of either of the four types

1. $v^{(i)}, v^{(i+1)}$
2. $v^{(i)}, \{v^{(i)}, v^{(i+1)}\}$
3. $\{v^{(i)}, v^{(i+1)}\}, v^{(i+1)}$
4. $\{v^{(i)}, v^{(i+1)}\}, \{v^{(i+1)}, v^{(i+2)}\}$

with $\{v^{(i)}, v^{(i+1)}\} \in E$ in all cases and $\{v^{(i+1)}, v^{(i+2)}\} \in E$ in the fourth case. The vertex sequence $\bar{v}$ of a mixed sequence consists of the proper vertices and of the vertices of incident edges successively visited by the mixed sequence. Formally, the vertex sequence of a mixed sequence with path property results from the following operations. The result $\bar{v}(\cdot)$ is obtained when no operation applies any more. To avoid trivial complications, the mixed sequence is assumed to consist of at least three elements.

$v^{(i)}, v^{(i+1)}$	$\rightarrow$	$v^{(i)}, v^{(i+1)}$.
$v^{(i)}, \{v^{(i)}, v^{(i+1)}\}$	$\rightarrow$	$v^{(i)}, v^{(i+1)}$.
$\{v^{(i)}, v^{(i+1)}\}, v^{(i+1)}$	$\rightarrow$	$v^{(i)}, v^{(i+1)}$.
$\{v^{(i)}, v^{(i+1)}\}, \{v^{(i+1)}, v^{(i+2)}\}$	$\rightarrow$	$v^{(i)}, v^{(i+1)}, v^{(i+2)}$.

Obviously, the vertex sequence of a mixed sequence with path property is a path. This provides for a fairly robust partially defined transformation of grid point sequences to paths

$$Tr(\bar{p}) := \begin{cases} \bar{v}(tr(ind(\bar{p}))), & \text{if } \bar{v}(tr(ind(\bar{p}))) \text{ is a path in } G \\ \text{void}, & \text{else.} \end{cases}$$

An example of the transformation is given in figure 7 where an "extremely" disconnected grid point sequence suffices to sketch a path.

The re-entry of a curve into a partitioned Voronoi region typically causes the trace of an induced mixed sequence to have repetitions. Though the transformation Tr may still be formally applicable, the resulting path is hardly meaningful, comp. figure 8.

Any extension of the current version of the transformation Tr should satisfy the following monotonicity property. Whenever, loosely speaking, more information on a path becomes available in terms of

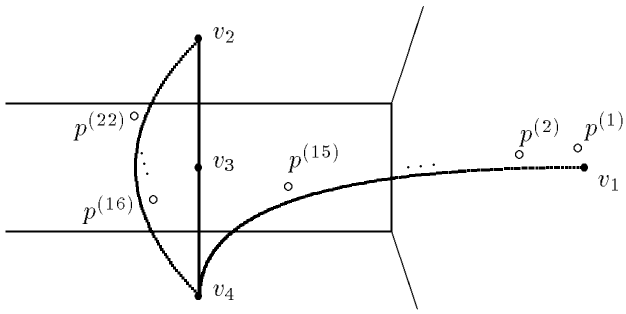

Figure 7: Grid point sequence $\bar{p} = (p^{(1)}, \ldots, p^{(22)})$ (grid omitted) with $tr(ind(\bar{p})) = (v_1, \{v_1, v_4\}, \{v_4, v_2\})$ and path $Tr(\bar{p}) = (v_1, v_4, v_2)$. This path remains unchanged when the subsequence $(p^{(16)}, \ldots, p^{(22)})$ is traversed in opposite order $(p^{(22)}, \ldots, p^{(16)})$. Noteworthy, the grid point sequence does neither have points in the Voronoi region $V(v_4)$ nor in $V(v_2)$.

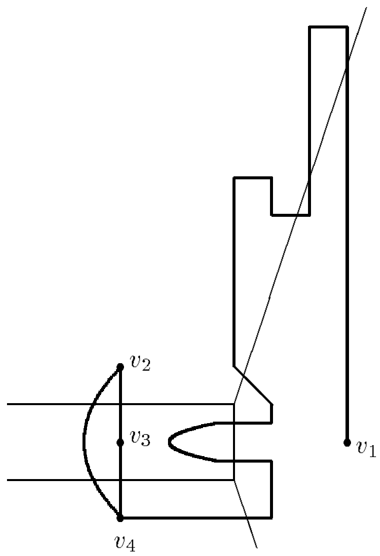

Figure 8: A grid point sequence following $cur(v_1, v_4)$ sufficiently close leads to $tr(ind(\bar{p})) = (v_1, \{v_1, v_4\}, v_1, \{v_1, v_4\}, v_1, \{v_1, v_4\}, v_1, v_4)$ with path $\bar{v} = (v_1, v_4, v_1, v_4, v_1, v_4, v_1, v_4)$.

more details on the trace of the induced mixed sequence, path identification should become more precise or less ambiguous or "less wrong". This property does not apply to the present version of transformation Tr. Suppose the grid point sequence in figure 8 leads from $V(v_1)$ into $V_{V(v_2)}(cur(v_1, v_4))$ only once and then resumes in $V_{V(v_3)}(cur(v_4, v_2))$ with $tr(ind(\bar{p})) = (v_1, \{v_1, v_4\}, \{v_4, v_2\})$. Then $Tr(\bar{p}) = \bar{v}(v_1, \{v_1, v_4\}, \{v_4, v_2\}) = (v_1, v_4, v_2)$. If the grid sequence were "more definite" with $tr(ind(\bar{p})) = (v_1, \{v_1, v_4\}, v_1, \{v_4, v_2\})$ then a situation of Tr being undefined would be encountered which is not intuitive as less information (on the trace) leads to a clear path. The grid sequence being even more definite with $tr(ind(\bar{p})) = (v_1, \{v_1, v_4\}, v_1, \{v_1, v_4\}, \{v_4, v_2\})$ results in the path $(v_1, v_4, v_1, v_4, v_2)$ which is not intuitive as well.

The monotonicity issue is dealt with by considering extended parts of the induced mixed sequence. Let therefore $A[x], A[x, y]$ etc. denote possibly empty strings made up from the bracketed terms such as $A[x] = x, x$ and $A[x, y] = y, y, x, y$ and let $A_1[x], A_1[x, y]$ etc. denote strings that consist of at least one of the bracketed terms. The set of operations leading from mixed sequences to vertex sequences is replaced by the following, more powerful set $\mathcal{O}$. The vertex sequence resulting from no further operation of $\mathcal{O}$ being applicable is again denoted by $\bar{v}(\cdot)$.

$O1.$	$(X, v^{(i)}, A[v^{(i)}, \{v^{(i)}, v^{(i+1)}\}], v^{(i+1)}, Y)$ for $la(X) \neq v^{(i)}$ and $fi(Y) \neq v^{(i+1)}$.	$\rightarrow$	$(X, v^{(i)}, v^{(i+1)}, Y)$
$O2.$	$(X, v^{(i)}, A[v^{(i)}, \{v^{(i)}, v^{(i+1)}\}], A[\{v^{(i+1)}, v^{(i+2)}\}], v^{(i+2)}, Y)$ for $la(X) \neq v^{(i)}$ and $fi(Y) \neq v^{(i+2)}$.	$\rightarrow$	$(X, v^{(i)}, v^{(i+1)}, v^{(i+2)}, Y)$
$O3.$	$(X, A_1[v^{(i)}, \{v^{(i)}, v^{(i+1)}\}], \ldots, A_1[v^{(i+k)}, \{v^{(i+k)}, v^{(i+k+1)}\}],$ $v^{(i+k+1)}, Y)$ for $la(X) \neq v^{(i)}, \{v^{(i)}, v^{(i+1)}\}$ and $fi(Y) \neq v^{(i+k+1)}, \{v^{(i+k)}, v^{(i+k+1)}\}$; $k \geq 0$.	$\rightarrow$	$(X, v^{(i)}, \ldots, v^{(i+k+1)}, Y)$
$O4.$	$(X, A_1[v^{(i)}, \{v^{(i)}, v^{(i+1)}\}], \ldots, A_1[v^{(i+k)}, \{v^{(i+k)}, v^{(i+k+1)}\}],$ $A_1[v^{(i+k+2)}, \{v^{(i+k+1)}, v^{(i+k+2)}\}], Y)$ for $la(X) \neq v^{(i)}, \{v^{(i)}, v^{(i+1)}\}$ and $fi(Y) \neq v^{(i+k+2)}$; $k \geq 0$.	$\rightarrow$	$(X, v^{(i)}, \ldots, v^{(i+k+2)}, Y)$
$O5.$	$tr(ind(\bar{p})) = (X, S_1, \ldots, S_k, Y)$ for select components $S_1, \ldots, S_k$, $k \geq 1$ v being the last vertex to which X is transformed or $X = \epsilon$ $w = fi(Y)$ or $Y = \epsilon$ and no other operation applicable; see text.	$\rightarrow$	$(X, in(S_1), out(S_1), \ldots,$ $in(S_k), out(S_k), Y)$

The prefix X and the suffix Y may be the empty string ϵ. An edge turning up without its endpoints in the trace of an induced sequence or other constellations may cause the order of vertex appearances being not unique. An example is $(v_3, \{v_4, v_7\}, v_2)$. In this case $O1$ - $O3$ are not applicable and an application of operation $O4$ leads to the following two sequences.

Variables	Settings	
$v^{(1)}$	v_3	v_3
$v^{(2)}$	v_4	v_7
$v^{(3)}$	v_7	v_4
$v^{(4)}$	v_2	v_2
$A_1[v^{(1)}, \{v^{(1)}, v^{(2)}\}]$	(v_3)	(v_3)
$A_1[v^{(2)}, \{v^{(2)}, v^{(3)}\}]$	$(\{v_4, v_7\})$	$(\{v_7, v_4\})$
$A_1[v^{(4)}, \{v^{(3)}, v^{(4)}\}]$	(v_2)	(v_2)
Sequences	(v_3, v_4, v_7, v_2)	(v_3, v_7, v_4, v_2)

In case of ambiguous numberings, a shortest among all admissable vertex sequences is selected with ties being broken arbitrarily. Sequence length is computed according to the $d(\cdot, \cdot)$ labels, see section 2.1.

In case several successive edges neither share a vertex with their predecessor nor with their successor, operations $O1$ - $O4$ may not be applicable or may result in multiple sequence ambiguities. Such cases will be resolved by select components. A select component of a mixed sequence is defined to be a $\subseteq$-maximal subsequence of successive vertices and edges such that it is either a single vertex, or a single edge, or applications of $O1$ - $O4$ lead to a unique vertex sequence. The example $(v_3, \{v_4, v_7\}, v_2)$ leads to the three select components $S_1 = (v_3)$, $S_2 = (\{v_4, v_7\})$, and $S_3 = (v_2)$.

Each select component has an entry vertex and an exit vertex which is unique in case the select component is a single vertex or leads to a unique vertex sequence. Otherwise, these vertices admit a twofold ambiguity. In the unique case, the entry vertex and the exit vertex may be identical as for a single vertex or a complete cycle. Unique entry and exit vertices are denoted by $v(in, S_i)$ and $v(out, S_i)$, the others are denoted by $v(in, S_i, 1)$, $v(in, S_i, 2)$, $v(out, S_i, 1)$, and $v(out, S_i, 2)$, where $v(in, S_i, 1) = v(out, S_i, 2)$ and $v(in, S_i, 2) = v(out, S_i, 1)$. Ambiguities are resolved by forming $P_0(v, w)$ in graphs such as in figure 9. All edges receive $d(\cdot, \cdot)$ labels with $d(v, \cdot)$ and $d(\cdot, w)$ becoming zero in case $X = \epsilon$ and $Y = \epsilon$ respectively.

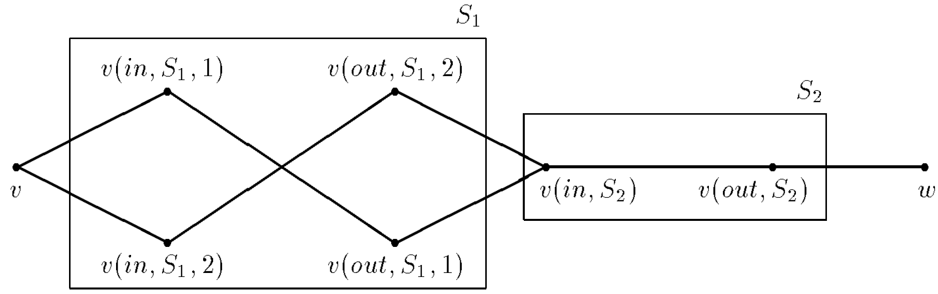

Figure 9: Substitution graph for computing a shortest path through S_1 and S_2 where S_1 has non-unique entry and exit vertices while S_2 has unique entry and exit vertices.

3.6.2 Transformations

Whenever a vertex sequence results from the operations of $\mathcal{O}$ applied to a mixed sequence, the vertex sequence has no immediate repetition. The transformation based on the extended definition of $\bar{v}$ formally is again given by

$$Tr(\bar{p}) := \begin{cases} \bar{v}(tr(ind(\bar{p}))), & \text{if } \bar{v}(tr(ind(\bar{p}))) \text{ is a path in } G \\ \text{void}, & \text{else.} \end{cases}$$

The trace $(v_1, \{v_1, v_4\}, v_1, \{v_1, v_4\}, v_1, \{v_1, v_4\}, v_1, v_4)$ of the induced mixed sequence from figure 8 is now transformed to (v_1, v_4) by one application of $O1$ with $X = Y = \epsilon$, $v^{(1)} = v_1$, $v^{(2)} = v_4$, and $A[v_1, \{v_1, v_4\}] = (\{v_1, v_4\}, v_1, \{v_1, v_4\}, v_1, \{v_1, v_4\}, v_1)$.

The path (v_1, v_4, v_2) results for $tr(ind(\bar{p})) = (v_1, \{v_1, v_4\}, \{v_4, v_2\})$ by one application of $O4$ with $X = Y = \epsilon$, $A_1[v_1, \{v_1, v_4\}] = (v_1, \{v_1, v_4\})$, and $A_1[v_2, \{v_4, v_2\}] = \{v_4, v_2\}$. If the grid sequence becomes more definite with $tr(ind(\bar{p})) = (v_1, \{v_1, v_4\}, v_1, \{v_4, v_2\})$ then again a single application of operation $O4$ now with $A_1[v_1, \{v_1, v_4\}] = (v_1, \{v_1, v_4\}, v_1)$ and again $X = Y = \epsilon$ and $A_1[v_2, \{v_4, v_2\}] = \{v_4, v_2\}$ leads to the same path thus preserving monotonicity.

Noteworthy, whenever the trace of an induced sequence alternates between two vertices such as (v_4, v_2, v_4, v_2), no operation of $\mathcal{O}$ applies and thus the sequence is left unchanged by $\bar{v}(\cdot)$. It is thus possible to state deliberate vertex repetitions in paths by suitable grid point sequences. Also, the reproduction property of transformation Tr is ensured under the following separation condition for curves. Any two curves are assumed to be apart by at least the triple grid width $3d$ with the exception of pure Voronoi regions where they may come closer in case they are incident with the germ of the Voronoi region. Then $Tr(\bar{p}(\bar{v})) = \bar{v}$, where the path $\bar{v}$ induces the sequence $\bar{p}(\bar{v})$ of closest grid points by following the curves of $\bar{v}$, comp. the robustness discussion at the beginning of section 3.

3.6.3 Complete transformations

In case $Tr(\bar{p})$ is void, the vertex sequence $\bar{v}(tr(ind(\bar{p})))$ can be extended to a path by inserting shortest paths between any successive vertices that are not adjacent in the graph G. Also, single isolated vertices may be deleted as stated in the section on insertions and deletions. Numerous variations of the previous transformation and combinations of transformation elements such as tracing, forming yet other modifications of Voronoi regions, insertions and deletions are possible.

A reasonable decision on inserting and deleting vertices appears to be based on the connectivity of the grid point sequence. If the grid point sequence is disconnected, deletions are forbidden. The reason is that a disconnection of the grid point sequence may result from deliberate jumps to sections of the graph that must be visited by the path. If the grid point sequence is connected, single isolated vertices will be deleted if they do not lie on a shortest connecting path in G, otherwise they will be connected. This is motivated by possible overshoots of the grid point sequence that unintentionally reach into some pure or mixed Voronoi region, comp. figure 2. The complete procedure is formally stated as follows.

A

1. Input grid point sequence $\bar{p}$ with vertex sequence $Tr(\bar{p}) = \bar{v}(tr(ind(\bar{p}))) = (v^{(1)}, \ldots, v^{(M)}) = \bar{v}$.

2. If $Tr(\bar{p})$ is a path, no operations are performed.

   If $\bar{v}$ is not a path and

   if $\bar{p}$ is disconnected then any $v^{(i)}, v^{(i+1)}$ with $\{v^{(i)}, v^{(i+1)}\} \notin E$ are connected by a shortest path $P_0(v^{(i)}, v^{(i+1)})$ to result in a new path $\bar{v}$; else
   insertions of $P_0(v^{(i)}, v^{(i+1)})$ and $P_0(v^{(i+1)}, v^{(i+2)})$ are replaced by $P_0(v^{(i)}, v^{(i+2)})$ (potential deletion of $v^{(i+1)}$) in case $v^{(i+1)}$ is a single isolated vertex in the original vertex sequence $\bar{v}$, i.e. in case $\{v^{(i-1)}, v^{(i)}\} \in E$, $\{v^{(i)}, v^{(i+1)}\} \notin E$, $\{v^{(i+1)}, v^{(i+1)}\} \notin E$, and $\{v^{(i+2)}, v^{(i+3)}\} \in E$. This results in a new path $\bar{v}$.

3. Output path $\bar{v}$.

Insertions by this algorithm may lead to revisiting vertices. The algorithm applied to the problem of figure 2 results in the path $(v^{(1)}, v^{(2)}, v^{(4)}, v^{(5)})$; the single isolated vertex $v^{(3)}$ is deleted since the grid point sequence is connected and the single isolated vertex does not lie on a shortest path from $v^{(2)}$ to $v^{(4)}$. In case of a potential deletion of some vertex which means that in case the vertex is singly isolated and the grid point sequence is connected, the deletion need not occur. An example is given in figure 3, where the single isolated vertex $v^{(i+1)} = v^{(3)}$ lies on the unique shortest path $P_0(v^{(2)}, v^{(4)})$ and thus becomes included in the overall path.

The look ahead ability of transformation Tr based on operation $O2$ and on operation $O4$ as illustrated by figure 7 may in certain situations lead to vertex repetitions. Whenever these are unintended, they can be suppressed by short cuts. Let therefore v_0 be a vertex that repeatedly appears in path $\bar{v} = (v^{(1)}, \ldots, v^{(i)}, \ldots, v^{(j)}, \ldots, v^{(M)})$ with $v^{(i)} = v^{(j)} = v_0$. Then the path $\bar{v}$ can be shortened to the path $(v^{(1)}, \ldots, v^{(j)}, \ldots, v^{(M)})$ by deleting the vertices $v^{(i)}, \ldots, v^{(j-1)}$. The shortest path that can be obtained from $\bar{v}$ by repeated applications of this shortening operation is the shortest $\bar{v}$-path also called best shortened path. The best shortening is always a path from $v^{(1)}$ to $v^{(M)}$ whose length may exceed that of a shortest path from $v^{(1)}$ to $v^{(M)}$. The best shortening cannot be computed by applying shortening operations in a greedy manner in case partial paths containing vertex repetitions overlap. The computation of best shortenings is discussed in section 4.

Whenever connectivity of the grid point sequence should not be made a criterion of the path construction, potential deletion of a single isolated vertex can still be considered meaningful. This results in a simplified version of the previous algorithm.

A'

1. Input grid point sequence $\bar{p}$ with vertex sequence $Tr(\bar{p}) = \bar{v}(tr(ind(\bar{p}))) = (v^{(1)}, \ldots, v^{(M)}) = \bar{v}$.

2. If $Tr(\bar{p})$ is a path, no operations are performed.

   If $\bar{v}$ is not a path then any $v^{(i)}, v^{(i+1)}$ with $\{v^{(i)}, v^{(i+1)}\} \notin E$ are connected by $P_0(v^{(i)}, v^{(i+1)})$ to result in a new path $\bar{v}$ where insertions of $P_0(v^{(i)}, v^{(i+1)})$ and $P_0(v^{(i+1)}, v^{(i+2)})$ are replaced by $P_0(v^{(i)}, v^{(i+2)})$ (potential deletion of $v^{(i+1)}$) in case $v^{(i+1)}$ is a single isolated vertex in the original vertex sequence $\bar{v}$.

3. Output path $\bar{v}$.

To summarize, the two favourite paths induced by a grid point sequence are the ones resulting either from algorithm **A** or from algorithm **A'** and the best shortenings thereof.

3.7 Further aspects

A mixed sequence can be considered as a regular expression, see for example [Le, p. 29f], over the alphabet $V \cup E$. Only regular expressions without immediate repetitions emerge as traces of mixed sequences. The set of these regular expressions and the set of those resulting from applying the operations of $\mathcal{O}$ obviously fall into the class of recursive languages. It is unclear whether they also fall into the smaller class of context sensitive languages. The difficulty of ensuring the latter property stems from the fact that operations of $\mathcal{O}$ may shorten strings, comp. [HoUl].

3.8 Graph Voronoi regions for non-planar graphs

Two curves that are incident with a common vertex may intersect in the Voronoi region of that vertex without apparantly warranting to change the transformation Tr or any of its components, comp. figure 10. In general, changes are required. One reason is that even pure Voronoi regions need not be connected in

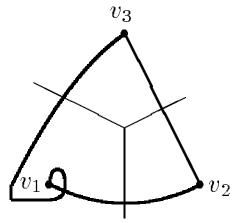

Figure 10: Non-planar situation with original Voronoi regions (boundaries indicated by thin lines) which differ from the partitioned Voronoi regions as for example $V(v_1)$ contains a non-vanishing mixed Voronoi region $V_{V(v_1)}(cur(v_2, v_3))$. This can be seen from the intersection of the three boundaries being closer to $cur(v_2, v_3)$ than to $cur(v_1, v_3)$ and $cur(v_1, v_2)$. However, the transformation Tr of grid point sequences to paths is applicable as in planar situations.

the non-planar case. Formally, the appropriate changes will affect the underlying graph rather than the defining conditions of pure and mixed Voronoi regions.

All curve intersection points that do no belong to the vertex set V are collected in the finite set W. Any curve that passes through at least one such intersection point is split into several curves between adjacent points of the vertex set $V \cup W$. This results in the associated extended graph being planar. Graph Voronoi regions of a non-planar graph are then defined as graph Voronoi regions of the associated extended graph.

3.9 From mixed sequences to paths in non-planar graphs

The transformation of a grid point sequence to the extended graph may result in a path using intersection points from W. In order to obtain a path in the original graph, the path in the extended graph is assumed to begin and terminate at a vertex from the original graph. This condition must be satisfied by some "external" mechanism. Then, any partial path $(v^{(i)}, w^{(i+1)}, \ldots, w^{(j-1)}, v^{(j)})$ with $v^{(i)}, v^{(j)} \in V$ and $w^{(i+1)}, \ldots, w^{(j-1)} \in W$ is subject to the following replacement.

$$(v^{(i)}, w^{(i+1)}, \ldots, w^{(j-1)}, v^{(j)}) \rightarrow \begin{cases} (v^{(i)}, v^{(j)}), & \text{if } \{v^{(i)}, v^{(j)}\} \in E \text{ and} \\ & w^{(i+1)}, \ldots, w^{(j-1)} \in cur(v^{(i)}, v^{(j)}), \\ P_0(v^{(i)}, v^{(j)}), & \text{else.} \end{cases}$$

Whenever immediate vertex repetitions should be left over, these are eliminated. Such repetitions can occur when successive partial paths with vertices in W meet in a common vertex of the original graph such as in $v^{(i)}$ or $v^{(j)}$.

Replacements of partial paths either by eliminating all intersection vertices or by taking the shortest path between successive vertices from the original graph – as both appear in the foregoing replacement operation – may lead to different results even if the successive vertices are adjacent in the original graph. Such a situation can occur in case the edge labels of the original graph violate the triangle inequality. The proposed replacement strategy favours edge replacement without forming shortest paths in order to reproduce edges also for labels that violate the triangle inequality.

Either the path resulting from all replacements or the best shortening of that path is finally accepted as path in the original graph resulting from a grid point sequence.

4 Computational issues

The case of all curves being straight line segments is investigated for computing the graph Voronoi regions. Curved and bended connections such as given in figures 8 and 6 are hence ruled out. Still, connections completely outside but close to an ordinary Voronoi region may affect its partition into graph Voronoi regions. Mixed Voronoi regions can then be formed by intersecting ordinary Voronoi regions with Voronoi regions of straight line segments as stated below. The Voronoi region of a curve that may or may not be a straight line is defined by

$$V(\{v_i, v_j\}) := \{x \in \mathbb{R}^2 | \; dist_{\mathbb{R}^2}(x, cur(v_i, v_j)) \leq dist_{\mathbb{R}^2}(x, cur(v_k, v_l)) \; \forall \{v_k, v_l\} \in E - \{\{v_i, v_j\}\}\}.$$

A mixed graph Voronoi region is given as the intersection of an ordinary Voronoi region with the Voronoi region of an edge meaning that for $v_i, v_j \in V - \{v\}$

$$V_{V(v)}(cur(v_i, v_j)) = V(v) \cap V(\{v_i, v_j\}).$$

Noteworthy, even under the condition of all curves being straight, $cur(v_i, v_j)$ lying completely outside $V(v)$ may have a non-void mixed region $V_{V(v)}(cur(v_i, v_j))$. A pure graph Voronoi region is the remainder of an ordinary Voronoi region when all mixed graph Voronoi regions are subtracted:

$$\begin{aligned} V_{V(v)}(v) &= V(v) - \bigcup_{\{v_i, v_j\} \in E, v_i, v_j \in V - \{v\}} V_{V(v)}(cur(v_i, v_j)) \\ &= V(v) - \bigcup_{\{v_i, v_j\} \in E, v_i, v_j \in V - \{v\}} V(v) \cap V(\{v_i, v_j\}). \end{aligned}$$

The number of ordinary Voronoi regions trivially is n and the number of edge Voronoi regions in a planar graph with n vertices is $O(n)$ since the number of edges is bounded by $3n - 6$. However, the number of graph Voronoi regions is $O(n^2)$ with the quadratic growth rate being attained in examples such as in figure 11.

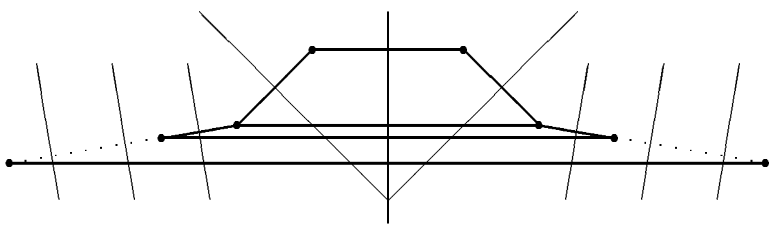

Figure 11: Planar graph with n vertices, n even, and $0 + 2 + 4 + \ldots + (n-2) = 1/4\, n(n-2)$ mixed graph Voronoi regions. Counting follows the horizontal edges from the top.

Ordinary Voronoi regions can be computed in several ways. For an overview sketching $O(n^2 \log n)$, $O(n^2)$, and $O(n \log n)$ algorithms see [O'R]. The Voronoi region of a finite collection of line segments can also be computed in $O(n \log n)$ [Y]. A typical situation of the latter is depicted in figure 12, where the angular bisector of two line segments is smoothly joined with the perpendicular bisectors of the lines' outer endpoints by parabolas. All line segments shrinking to single points results in the ordinary Voronoi diagram with boundaries of Voronoi regions lying on perpendicular bisectors only; angular bisectors and parabolas shrink until disappearance. On the other hand, in case of the line segments forming the boundary of a convex polygon, the parabolas and perpendicular bisectors vanish inside the polygon where the Voronoi regions are bounded by angular bisectors only, comp. [O'R, p. 180]. In this case however, angular bisectors of line segments that do not intersect on the polygon's boundary but whose extensions intersect outside the polygon may turn up.

Combining an algorithm for ordinary Voronoi regions with one for Voronoi regions of lines allows to compute the graph Voronoi regions by forming suitable intersections and unions. However, difficulties such as arising from forming intersections of regions bounded by parabolas and the definite conceptual complexity of algorithms such as Yap's [Y] warrant approximations. Any approximation of a pure or mixed graph Voronoi region should fulfill the subsequent individual preservation conditions

$$\begin{array}{rclcll} cur(v_i, v_j) \cap V(v) & \subseteq & App(V_{V(v)}(cur(v_i, v_j))) & \subseteq & V(v) & \forall \{v_i, v_j\} \in E \text{ with } v_i, v_j \in V - \{v\} \\ cur(v, v_j) \cap V(v) & \subseteq & App(V_{V(v)}(v)) & \subseteq & V(v) & \forall \{v, v_j\} \in E, \end{array}$$

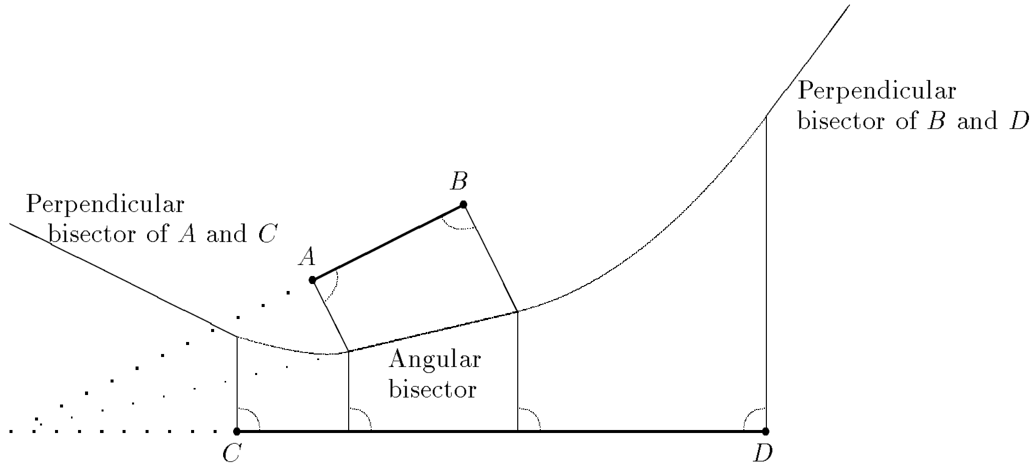

Figure 12: Voronoi region of two non-parallel straight line segments AB and CD that do not intersect. The small dotted curves indicate 90^0 angles. The Voronoi region of the segment AB is convex in this case.

where $App(\cdot)$ denotes a region approximation. In addition, all approximated graph Voronoi regions belonging to the same ordinary Voronoi region should satisfy the joint preservation condition

$$App(V_{V(v)}(v)) \cup \bigcup_{\{v_i,v_j\}\in E, v_i,v_j\in V-\{v\}} App(V_{V(v)}(cur(v_i,v_j))) = V(v).$$

4.1 Graph Voronoi regions and line approximations

Approximations of the general situation can be formed by omitting the parabolas. These can either be cut short by joining the endpoints with a straight line segment, see figure 13 (left), or by using the two intersections of "adjacent" angular and perpendicular bisectors, see figure 13 (right). The latter curve appears to have an advantage over the former by ensuring a larger minimum distance to the endpoints of the given line segments. This feature is bought at the minor expense of additional support vertices.

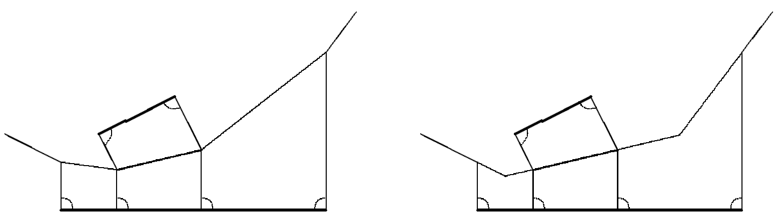

Figure 13: Short cuts of the parabolas (left) and detours to the convex side of the parabolas (right) for the situation of figure 12.

An even further simplified approximation of graph Voronoi regions is given by angular bisector Voronoi regions, where region boundaries are formed by angular bisectors only such as sketched in figure 14. For an edge set that forms a simple polygon possibly comprising other simple polygons the angular bisector regions can be computed with moderate conceptual complexity in $O(n^2 \log n)$, see [ClOlSt].

4.2 Graph Voronoi regions and point approximations

The approach to approximate line Voronoi regions by ordinary Voronoi regions [Su] can be formulated for graph Voronoi regions so that all preservation conditions are satisfied. Whenever an ordinary Voronoi

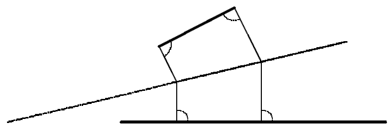

Figure 14: Further simplifications by using angular bisectors only.

region is partitioned into pure and mixed graph Voronoi regions, the edge segments within the ordinary Voronoi regions will be provided with particular points of sufficient density. These points will serve as additional germs. Their Voronoi regions in the original graph, rather than the Voronoi regions formed by these germs alone, are united whenenver the germs lie on the same edge segment. Intersecting these unions with the original Voronoi regions results in approximations of the graph Voronoi regions, comp. figure 15.

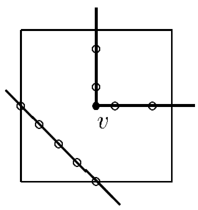

Figure 15: Ordinary Voronoi region of vertex v with one influencial traversing edge and nine additional germs (white dots).

The density of additional germs is expressed by the maximum distance along each edge where it is taken to be the distance between neighbouring additional germs. For a suitable selection, the minimum distance between any two edge segments traversing an ordinary Voronoi region but not being incident with the germ is computed. Also, the minimum distance between any two of the non-incident and the incident lines is computed. Planarity of the graph ensures that both minima are positive and the smaller of the minima is chosen as a density parameter.

Each of the line segments receives additional germs so that neighbouring germs on the same line are no further apart than twice the density parameter. So, each point on any of the lines is closer to an additional germ on the same line segment than to a germ on any other line.

All edges that are influencial for but completely outside the ordianry Voronoi region are provided with additional germs as well. The maximum distance between neighbouring germs on each of these influencial edges is at most the distance between any of the influencial edges and all segments edges that are incident with the germ. Since none of the influencial edges need to traverse the ordinary Voronoi region, the criterion for placing additonal germs there must be independent from the density parameter for influencial edges that traverse the Voronoi region.

An influencial edge as well as an edge incident with the germ may receive as few as a single additional germ. All in all, this ensures all the preservation conditions to hold. Though computing distances between line segments is easy, this approach may generate a practically unmanageable number of additional germs and unnecessarily complicated boundaries of region approximations.

4.3 Spatial indexing

The ordinary Voronoi diagram is well known to be preprocessable so that the computation of a Voronoi region that contains some query point can be executed in $O(\log n)$ time. The technique of monotone separators [O'R, p. 286 ff] can be applied to angular bisector regions to compute the approximatively closest edge to the query point in $O(\log^2 n)$ time. The latter is based on so-called monotonicity of angular bisector partitions.

Thus, computing the mixed induced sequence of a grid point sequence of length N based on the previous region approximations requires time $O(N \log^2 n)$.

4.4 Direct computations

A simple brute force computation relies on the observation that for each grid point the nearest vertex and the nearest edge can both be computed in $O(n)$ time; planarity of the graph ensures that it has at most $3n-6$ edges. These computations do not require any preprocessing. Whenever the nearest line is incident with the nearest vertex, the grid point lies in the pure Voronoi region of that vertex. Whenever the nearest line is not incident with the nearest vertex, the grid point lies in the mixed Voronoi region of that line with respect to the nearest vertex. Computing the mixed induced sequence of a grid point sequence of legth N then requires time $O(Nn)$.

The complexity can be reduced whenever for each grid point the nearest vertex and the nearest line and the so induced mixed element are computed prior to obtaining the grid sequence. If preprocessing allows to store the results in an array, the hash function of that array admits access in constant time. Then computing the mixed induced sequence of the grid point sequence $\bar{p}$ requires time $O(N)$ only. However, extra storage and preprocessing of order $O(n)$ is required for each grid cell.

Effort for the latter preprocessing can be expected to be reduced by undersampling for ordinary Voronoi diagrams. The computational load tends to be lowered in a pragmatic sense rather than in the sense of worst or average case complexity. Convexity of Voronoi regions suggests to consider the grid points in a particular sequence since two grid points with identical closest graph vertex induce all grid points in between to belong to the same Voronoi region. This motivates a two step procedure with four times undersampling. In the first step, the grid point set is undersampled by considering each other row and each other column there. Columns are selected with an off-set equalling the grid width as shown in figure 16 (left). Closest graph vertices are computed in a brute force manner for all points of the undersampled set. In the second step, closest graph vertices are computed for all remaining grid points. If two neighbours of

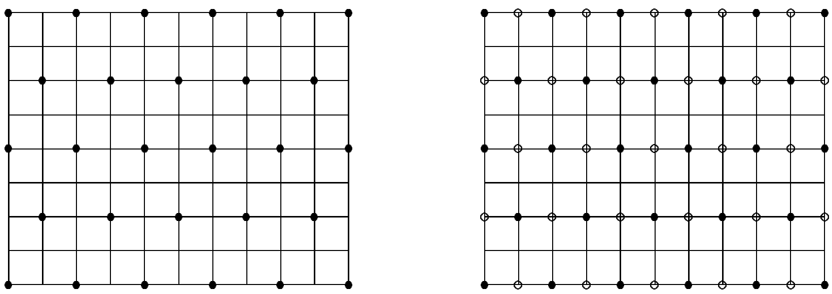

Figure 16: Grid with four times undersampled point set indicated by black dots (left) and fill-in of each other row indicated by white dots (right). Boundary effects are not shown.

such a grid point have the same closest vertex, this vertex is assigned to the grid point under consideration. If two neighbours have different closest graph vertices, a brute force computation is performed. This fills in each other row completely, see figure 16 (right). Finally, the computations are applied to fill in the empty rows. In principle, any other than the given pattern of undersampling is applicable.

Computing the edge Voronoi regions by undersampling will result in a rough approximation only since edge Voronoi regions need not be convex. All direct computations stated here do not require the graph to be rastered to the grid Gri or any other grid.

4.5 Best shortenings

The computation of best shortenings goes along the computation of shortest paths in directed graphs. The line graph of directed edges or arcs of a path is therefore enriched by an arc between each successive repetition of a vertex. These additional arcs represent short cuts in the given path so they are directed according to increasing vertex labels in the sense of the original sequence; this avoids cycles. Consequently, short cutting arcs are assigned cost 0. All other arcs carry the original cost assignments. Within this shortening graph, repeatedly appearing vertices should receive different labels according to their multiplicity as they are accounted for as distinct, comp. figure 17. This label refinement is omitted to avoid notational overload.

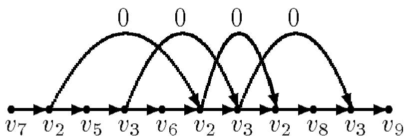

Figure 17: The shortening graph of path $\bar{v} = (v^{(1)}, \ldots, v^{(11)}) = (v_7, v_2, v_5, v_3, v_6, v_2, v_3, v_2, v_8, v_3, v_9)$ is the directed line graph of the path enriched by the four arcs joining successive appearances of the vertices v_2 and v_3 respectively. The best shortening is a shortest path from v_7 to v_9 in this graph.

A shortest $\bar{v}$-path or best shortening is a shortest path from the first to the last vertex in the shortening graph. Computations can be facilitated by dynamic programming since the shortening graph is free of cycles and by the Dijkstra algorithm since all labels are non-negative.

For the construction of the shortening graph, it is insufficient to place arcs only between the first and the last appearance of a vertex. This can be seen from the example of figure 17 in case the cost of each of the two arcs (v_5, v_3) and (v_8, v_3) exceeds the cost sum over all other arcs. Any path from v_7 to v_9 now can avoid either of the expensive edges but not both of them if only the two shortcuts from the first to the last appearance of v_2 and v_3 respectively were permitted.

5 Major example

Figures 18 through 21 show a planar graph with 12 vertices and a sequence of about 200 grid points being transformed to a path by algorithm **A'**. Edges are labeled by their Euclidean lengths. The graph Voronoi regions shown in figure 20 are computed by the direct approach of section 4.4. The final path has four vertices without any repetition.

In the second example for the same graph, shown in figures 22 through 24, the resulting path has nine vertices. The overshoot of the grid point sequence in the center region of the graph is so large that four vertices whose pure Voronoi regions are not traversed by the grid point sequence are inserted. These vertices are third, fifth, sixth, and seventh in the final path. The reason for the insertions is that the grid point sequence traverses the small mixed Voronoi region of the edge between the second and third vertex and the mixed Voronoi region between the seventh and eigth vertex in the final path. Thus, the fourth vertex is not isolated and is hence not eliminated.

6 Further issues

The decision on allowing or cutting short vertex repetitions should primarily be based on outside information which is to be provided in each particular case or by convention in general. If none of these is available, a subtle descision criterion can be based on the precision with which the path is specified. Whenever the trace of an induced mixed sequence contains a subsequence which consists of all vertices that lead to a cycle in the graph, this cycle is accepted; the subsequence is allowed to contain edges. In all other cases meaning in all cases in which vertex insertions are required due to the grid point sequence being inexact, best shortenings are taken.

The identification of transformational features like accepting or rejecting vertex repetitions should ideally be based on prototypical examples offering all options under consideration. A mere selection should suffice for identification. Such adaptation or learning examples remain to be designed.

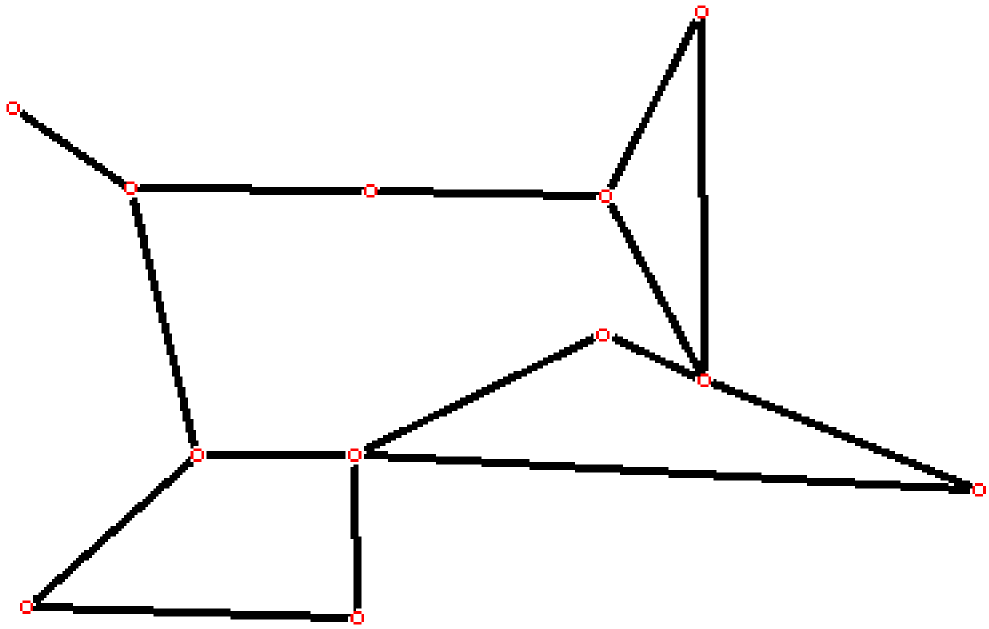

Figure 18: Planar graph with 12 vertices.

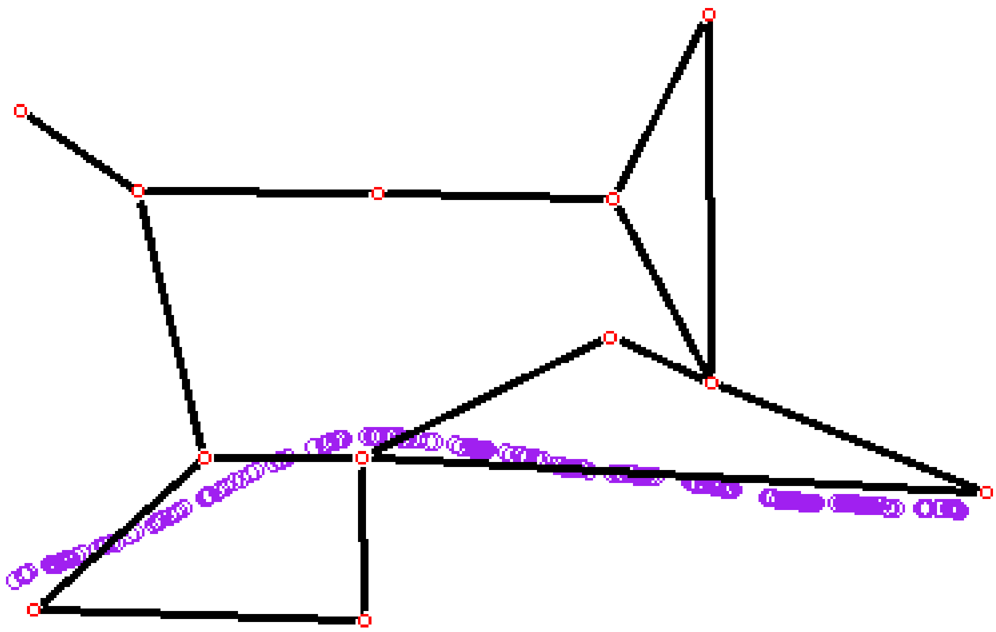

Figure 19: Same graph with grid point sequence edited by a touch screen.

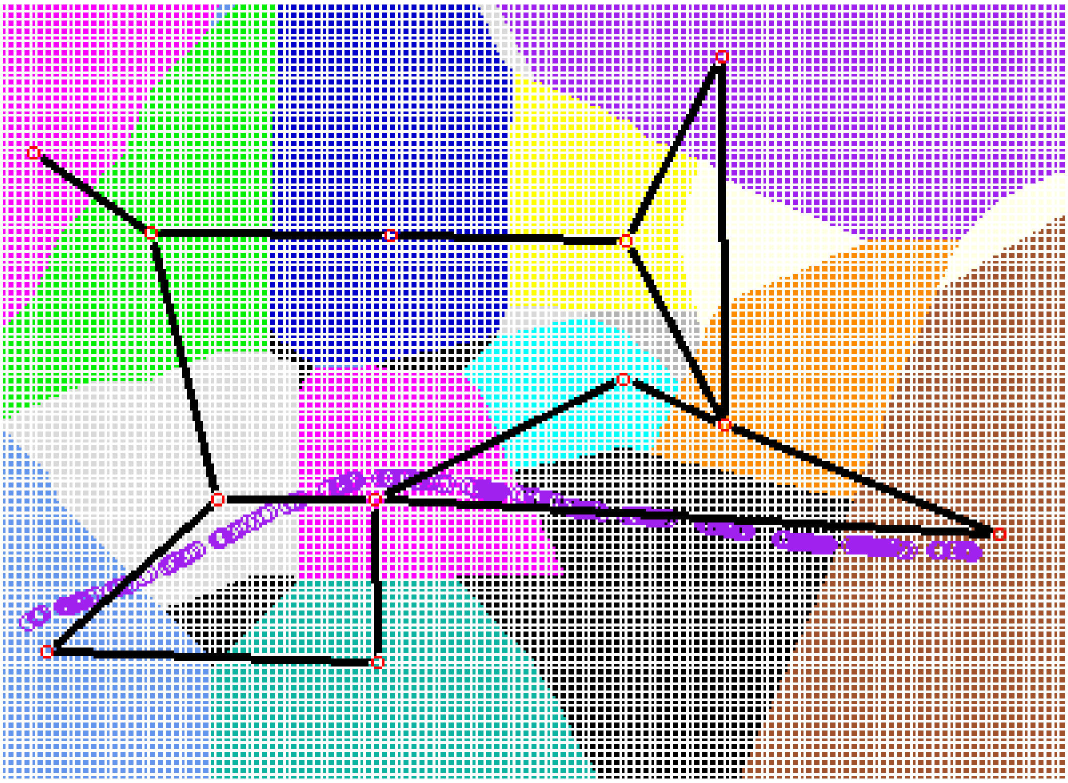

Figure 20: Graph with grid point sequence and graph Voronoi regions.

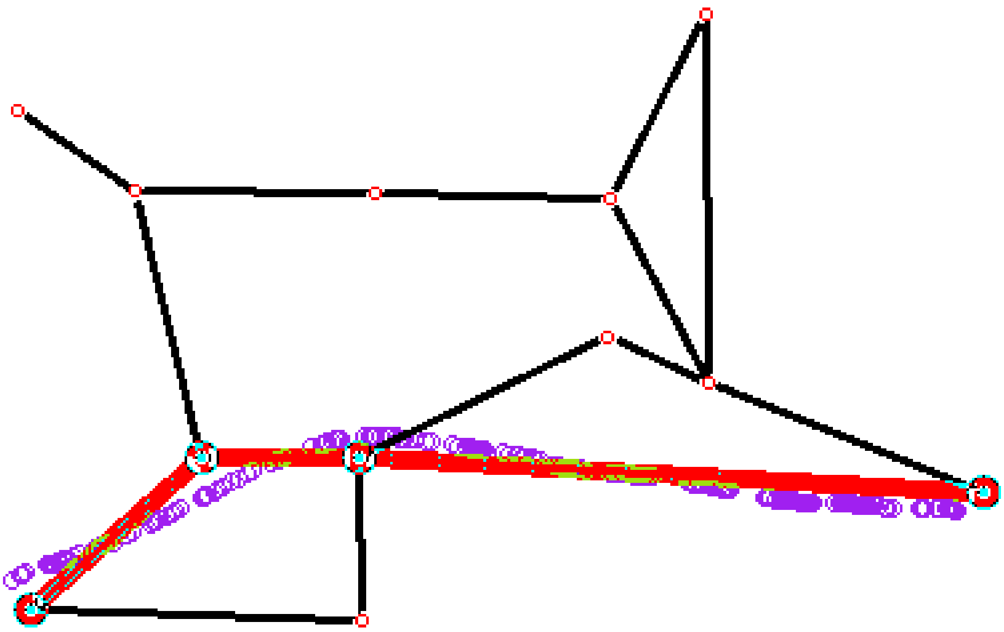

Figure 21: Graph with grid point sequence transformed to a path indicated by bold edges.

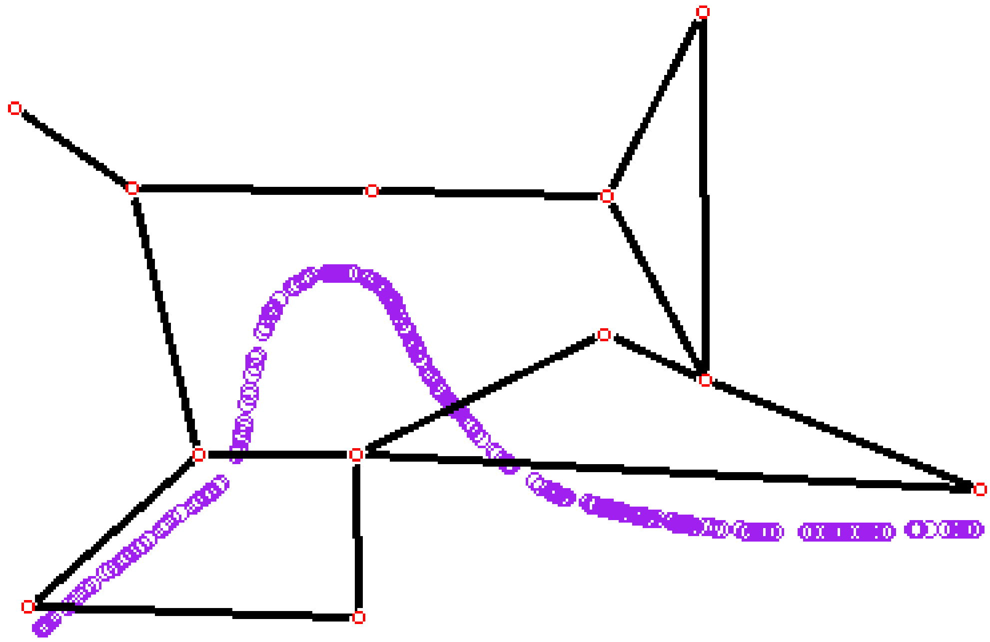

Figure 22: Graph with other grid point sequence.

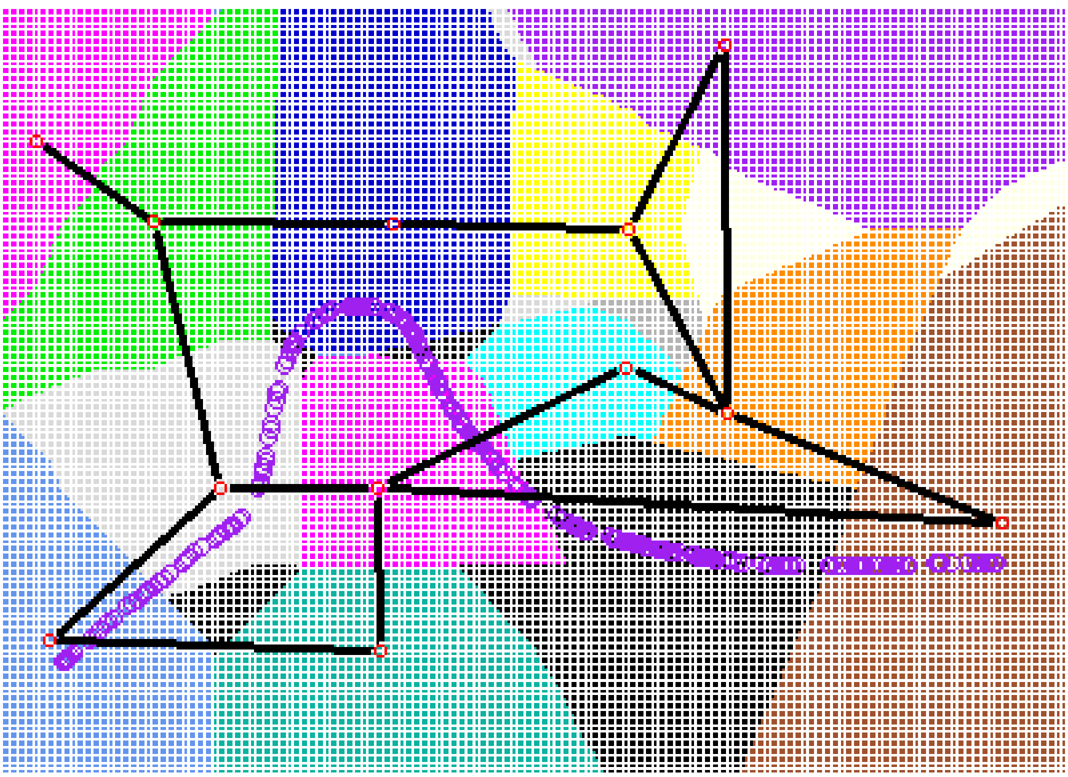

Figure 23: Graph with grid point sequence and graph Voronoi regions.

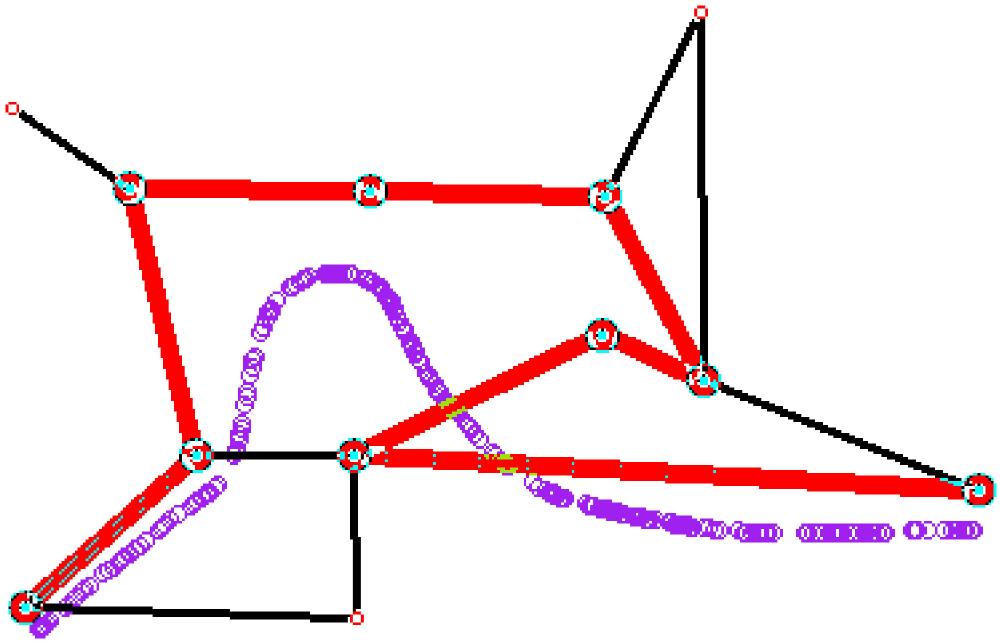

Figure 24: Graph with grid point sequence transformed to a path.

Acknowledgement
Direct computations of graph Voronoi regions and transformations from grid point sequences to paths were implemented by Matthias Strobel (FAW Ulm). Input was obtained from an elo 151R IntelliTouch 15-inch touch screen, geometric data handling was organized within the LEDA system, and the algorithms were written in C++.

References

[ClOlSt] Cloppet, F., Olivia, J.-M., Stamon, G., "Angular bisector network, a simplified generalized Voronoi diagram: applications to processing complex intersections in biomedical images", IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 2000, p. 120-128.

[Co] Cohen, P. et al., "Multimodal interaction for 2D and 3D environments", Computer Graphics and Applications, July/August 1999, p. 10-13.

[Gu] Gusfield, D., "Algorithms on strings, trees, and sequences: computer science and computational biology", Cambridge University Press, Cambridge, 1997.

[HoUl] Hopcroft, J.E., Ullman, J., "Introduction to automata theory, languages and computation", Addison-Wesley, New York, 1979.

[KäKo1] Kämpke, T., Kober, R., "Discrete signal quantization", Pattern Recognition 32, 1999, p. 619-634.

[KäKo2] Kämpke, T., Kober, R., "Nonparametric image segmentation", Pattern Analysis and Applications 1, 1998, p. 145-154.

[Le] Leeuwen, J.v. (ed.), "Handbook of theoretical computer science: algorithms and complexity", vol. A, Elsevier, Amsterdam, 1990.

[O'R] O'Rourke, J., "Computational Gemetry in C", 2nd ed., Cambridge University Press, Cambridge, 1998.

[Su] Sugihara, K., "Approximations of generalized Voronoi diagrams by ordinary Voronoi diagrams", Computer Vision and Graphic Image Processing 55, 1993, p. 522-531.

[Y] Yap, C.K., "An $O(n * \log n)$ algorithm for Voronoi diagrams of a set of simple curve segments", Discrete and Computational Geometry 2, 1987, p. 365-393.