

Concurrency of Line Segments in Uncertain Geometry

Peter Veelaert

Hogent, Schoonmeersstraat 52, 9000 Ghent, Belgium,
Peter.Veelaert@hogent.be

Abstract. We examine the derivation of consistent concurrency relations in uncertain geometry. This work extends previous work on parallelism and collinearity. We introduce the concept of a metadomain, which is defined as the set of parameter vectors of lines passing through two domains, where a domain is defined as the uncertainty region of the parameter vector of a line segment. The intersection graph of the metadomains is introduced as the primary tool to derive concurrency relations.

1 Introduction

We consider the concurrency of digital line segments within the framework of uncertain geometry, a geometric theory introduced to model the uncertainty of positions as well as geometric properties of objects in a digital image. In uncertain geometry points are equipped with uncertainty regions. The notion of an uncertainty region coincides with the use of a structuring element in the *discretization by dilation* scheme developed by Heijmans and Toet [1]. The definition of concurrency was introduced previously together with definitions for digital collinearity and parallelism [2].

One of the main characteristics of uncertain geometry is that properties such as parallelism, collinearity and concurrency are not necessarily consistent in the same way they are consistent in Euclidean geometry, and that somehow we must restore this consistency wherever it is needed in an application. In uncertain geometry when the line A is parallel with B , and B with C , then this does not necessarily imply that A is also parallel with C [2,3]. Thus, for the sake of consistency, after establishing a set of geometric relations between objects in uncertain geometry, we must repartition the objects, discard some relations as well as add new ones to obtain a consistent set of relations. This repartitioning can be accomplished by an optimal grouping process as was done for collinearity and parallelism in previous work [3,4,5]. As for parallelism, for example, the grouping process led to consistency by extracting cliques from an interval graph, in which each interval represented the uncertainty about the slope of a line segment.

Since concurrency concerns triples of lines, a direct approach to optimal grouping would involve the extraction of cliques from a hypergraph, where

the hyperedges represent concurrent triples [6]. Here, to simplify the grouping process, we reduce the conditions imposed on concurrency, similar to the way collinearity was replaced by the concept of weak collinearity to simplify the extraction of collinear groups of line segments [5].

What we propose in this paper is one approach to geometric reasoning when the position and parameter vectors are unprecise or uncertain. Also in robotics, mechanical design and computer vision, there is a need to deal with uncertainty. The models proposed there include the use of finite precision arithmetic [7], the use of probability density functions [8,9], the use of tolerance zones for mechanical parts [10], and significance measures for geometric relations [11]. However, these methods have not yet been integrated into a larger mathematical framework, what this work is aimed at.

In Section 2 we briefly sketch how we derive concurrency relations. In Section 3 we give more details on the computation of a metadomain. Section 4 examines the extraction of consistent relations. We conclude the paper in Section 4.

2 Concurrency and Line Transversals

During discretization the precise knowledge about the position of geometric objects is lost. We model this uncertainty by an *uncertainty region* that we associate with each point. The discretization process that coincides naturally with this notion of uncertainty is the *discretization by dilation* scheme developed by Heijmans and Toet [1]. The structuring element A_p used in this scheme coincides with our notion of an uncertainty region associated with a point p .

Furthermore, to keep the complexity of the computations acceptable, as model for the uncertainty of the position of a digital point $p = (x, y)$, we often use a very simple uncertainty region, i.e, the vertical line segment $C_p(\tau)$, which comprises all points $(x, b) \in \mathbb{R}^2$ that satisfy $y - \tau/2 \leq b < y + \tau/2$. Here τ is a positive real number, which controls the uncertainty. Also to simplify the exposition, we restrict ourselves to the concurrency of straight lines of the form $y = \alpha x + \beta$, where the slope α satisfies $-1 < \alpha < 1$. We assume that each set S contains at least two points with distinct x -coordinates. We shall also not discuss how to deal with lines of slope $|\alpha| \geq 1$, which can be done along the same lines as discussed in [2,4] for parallelism.

Definition 1. A finite digital set $S \subset Z^2$ is *digitally straight* if there is a Euclidean straight line that cuts all uncertainty segments $C_p(\tau)$, $p \in S$. We call such a set also a *digital line segment*.

Definition 2. Let S_1, \dots, S_n be a finite collection of digital line segments. We define the following properties:

- The sets $\{S_i : i = 1, \dots, n\}$ are called *digitally collinear* if there exists a common Euclidean straight line A that cuts the uncertainty segments of all the sets S_i ;

- The sets in $\{S_i : i = 1, \dots, n\}$ are called *digitally parallel* if there exist n Euclidean straight lines A_1, \dots, A_n that are parallel and such that for $i = 1, \dots, n$ the line A_i cuts the uncertainty segments of the set S_i ;
- The sets in $\{S_i : i = 1, \dots, n\}$ are called *digitally concurrent* if there exist n Euclidean straight lines A_1, \dots, A_n that meet in a common point and such that for $i = 1, \dots, n$ the line A_i cuts the uncertainty segments of the set S_i ;

In the remainder the notions of collinearity, parallelism and concurrency are used in the sense of Definition 2, unless we specify explicitly that we are using the Euclidean definition. When the sets S_1, \dots, S_n are digitally concurrent we denote this as $\text{conc}(S_1, \dots, S_n)$. The parameter vectors of the Euclidean lines passing through the uncertainty regions of a digital line segment, define a new kind of uncertainty region, also called the domain of the line segment.

Definition 3. Let S be a finite digital set that contains at least two points with distinct x -coordinates, and let τ be the uncertainty parameter. Then the *domain* of S , denoted as $\text{dom}_x S$, is the set of all parameter vectors $(\alpha, \beta) \in \mathbb{R}^2$ that satisfy the following system of inequalities:

$$-\tau/2 \leq \alpha x_i + \beta - y_i \leq \tau/2, \quad (x_i, y_i) \in S. \tag{1}$$

Note that a domain is defined here as a convex, closed polygon. The properties of Definition 2 can be reformulated in terms of domains [2]. Two digital line segments are collinear when their domains intersect. Two line segments are parallel when the intervals that result from the projection of the domains upon the α -axis have a non-empty intersection. Finally, three line segments are concurrent if there is a straight Euclidean line passing through their domains.

Attaching uncertainty regions to points and parameter vectors leads to additional levels of uncertainty. In fact, the introduction of domains can proceed indefinitely. Thus, we may consider the uncertainty of the parameter vector of a Euclidean line passing through two or more domains, and in this way construct a metadomain. One of the strong points of the use of uncertainty regions is that on each level the uncertainty depends in a direct way on the uncertainties introduced at the first level, that is, on positions of image points.

Given a collection of digital line segments and their domains, we shall perform the following steps to find out whether the line segments are concurrent or not:

1. compute the metadomains for pairs of domains;
2. construct an intersection graph of the metadomains;
3. find a consistent grouping on the vertices of the intersection graph;
4. compute the metadomains for the groups.

We illustrate these steps further in Fig. 1. In Fig. 1(a) we are given a collection of digital line segments which have been extracted from a digital image by use of the Ransac method [12]. The domains of these segments are shown in Fig. 1(b), where each domain corresponds to one line segment in Fig. 1(a). Large line segments correspond to small domains, because there is less uncertainty regarding their slope and height. Note, for example, that the domain of line segment G is

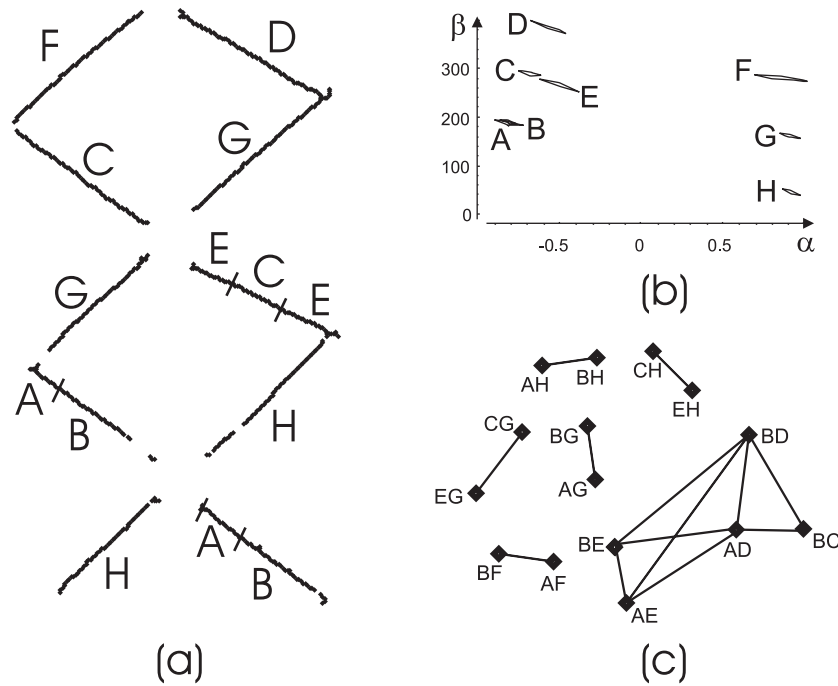


Fig. 1. (a) Scanned image of handmade line drawing, (b) domains, and (c) intersection graph.

much smaller than the domain of F , because G actually consists of two parts as shown in Fig. 1(a). Because the line segments were drawn by hand, to be able to extract at least some geometric relations, we have chosen a relatively large value for the uncertainty parameter used to compute the domains, i.e., $\tau = 5$. Once they have been computed, the domains can serve as new uncertainty regions, for which we want to find line transversals, as line transversals determine concurrency relations. Therefore, for each pair of non-parallel digital line segments we derive a metadomain formed by the parameter vectors of those Euclidean lines that cut their domains.

Definition 4. Let A and B be two digital line segments with domains $\text{dom}_x A$ and $\text{dom}_x B$. In addition assume that A and B are not-parallel, that is, $\text{dom}_x A$ and $\text{dom}_x B$ can be separated by a Euclidean line parallel to the β -axis. The *metadomain* consists of all parameter vectors (p, q) such that the Euclidean line

$$\beta - \alpha p - q = 0 \tag{2}$$

cuts $\text{dom}_x A$ as well as $\text{dom}_x B$ in the $\alpha\beta$ -plane.

The points of a metadomain can be identified with the points in the original image space. In fact, since the Euclidean line (2) cuts both domains, it contains

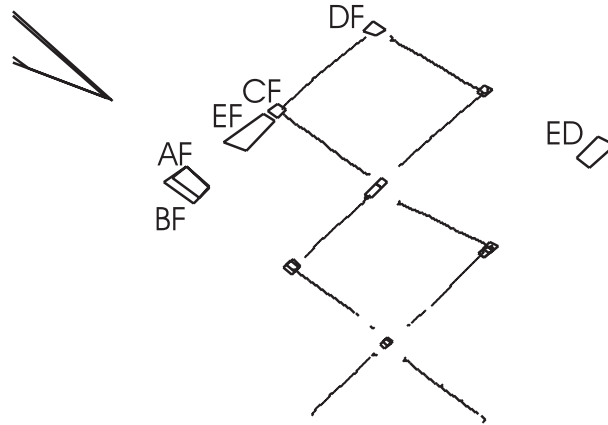


Fig. 2. Metadomains for pairs of lines.

two points (α_1, β_1) and (α_2, β_2) such that $(x, y) = (p, q)$ lies on a line $\beta_1 - \alpha_1 x - y = 0$ which cuts all uncertainty regions of points in A , as well as on the line $\beta_2 - \alpha_2 x - y = 0$ which cuts all uncertainty regions of B . In this sense the metadomain is the uncertainty region of the intersection point (p, q) of a Euclidean line passing through the uncertainty regions of A and a Euclidean line passing through the uncertainty regions of B .

Therefore, the metadomains of the domains in Fig. 1(b) can be superimposed on the original image, as done in Fig. 2. Next, we construct the intersection graph of the metadomains, as shown in Fig. 1(c). Each metadomain is represented by a vertex. When two metadomains intersect, they are joined by an edge. Thus the existence of an edge refers to the fact that there is a Euclidean line cutting either three or four domains. The precise number of domains cut depends on whether the two domain pairs have one domain in common or not. For example, since the vertices AG and BG are adjacent, there must be a straight line cutting the domains A , B and G .

Deriving concurrency relations directly from the intersection relations of the metadomains would almost always lead to inconsistency. Instead, as will be explained later, we perform a consistent grouping of the vertices of the intersection graph of the metadomains. We shall prove that we can obtain consistency when we extract cliques from the graph such that two cliques do not have a common neighbor. Next, for each clique we compute the common intersection of the metadomains that correspond to the vertices of the clique. These intersections are shown in Fig. 3. They form new uncertainty regions, representing the uncertainty regarding the position of the intersection point of three or more concurrent lines. For example, the region ABF corresponds to the uncertainty of the intersection point of the three Euclidean lines passing through the uncertainty regions of A , B and F . Each clique gives rise to a concurrency relation, pro-

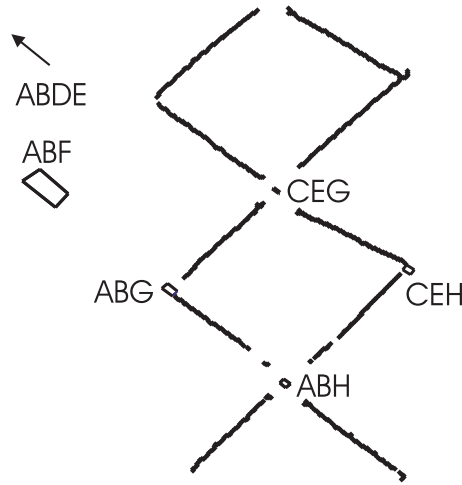


Fig. 3. Uncertainty regions for intersection points.

vided the intersection of the metadomains that are involved is non-empty. The cliques of the graph in Fig. 1(c) yield the following candidates for the concurrency relations: $\text{conc}(A, B, F)$, $\text{conc}(A, B, G)$, $\text{conc}(A, B, H)$, $\text{conc}(A, B, D, E)$, $\text{conc}(C, E, H)$ and $\text{conc}(C, E, G)$.

Finally, to improve consistency we may discard illegal cliques that contain two or more parallel lines. Since C and E are parallel, but not collinear, the only relations that remain are $\text{conc}(A, B, F)$, $\text{conc}(A, B, G)$, $\text{conc}(A, B, H)$. They can coexist since A and B are in fact collinear.

3 Derivation of Convex Metadomains from Domain Pairs

Since the construction of the intersection graph of the metadomains is greatly simplified when the metadomains are convex, an important question is the following: Given two disjoint convex domains, when is their metadomain also convex?

To examine this, given two domains A and B we introduce the lines and halflines shown in Fig. 4(a). Here V denotes a Euclidean line of minimal slope cutting A as well as B . Similarly, we let W denote a transversal of maximal slope. In addition, we choose a point $a_u \in (V \cap A)$ common to the line V and the set A , as well as a point $b_l \in (V \cap B)$. Let A_u denote the halfline starting at a_u parallel to the y -axis and extending in the direction of increasing y -values. Similarly we introduce the points b_u, a_l for the transversal of maximum slope W , and the halflines A_l, B_u and B_l , as shown in Fig. 4. We then have the following result.

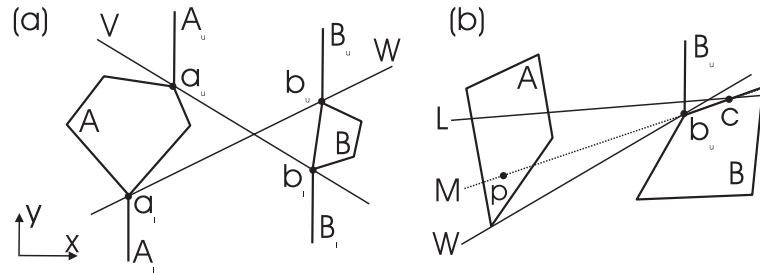


Fig. 4. A non-vertical line that does not cut A_u, A_l, B_u, B_l , must cut A and B .

Lemma 1. *Let A and B be two domains that are convex polygons, and let A_l, A_u, B_l and B_u be defined as in Figure 4. If there is a transversal cutting either B_l or B_u then one of the supporting lines of B cuts A .*

Proof. Without loss of generality we suppose there is a transversal L of A and B that cuts the halfline B_u and that does not pass through any of the vertices of B , as shown in Fig. 4. We choose an arbitrary point $p \in A \cap L$, and we define c as the point $c \in L \cap B$ such that the Euclidean line segment cp has only c in common with B . Let M denote the supporting line of B that passes through c . Since B is convex, the slope of M must be smaller than the slope of W , but larger than the slope of L . Hence M cuts B_u as well as the domain A .

Proposition 1. *Let A and B be two domains that are convex polygons. If the supporting lines of one domain do not cut the other domain then their metadomain is a convex polygon.*

Proof. Let $a_l = (p_{al}, q_{al}), \dots, b_u = (p_{bu}, q_{bu})$ be the coordinates of the endpoints. By Lemma 1 if no supporting line of either A or B cuts the other set, then there is no transversal line cutting one or more of the halflines A_l, A_u, B_l or B_u . On the other hand, any line that cuts none of these four halflines is a transversal of A and B . Therefore the metadomain in the uv parameter space is determined by the four inequalities

$$\begin{cases} v - q_{al} - up_{al} > 0 \\ v - q_{bl} - up_{bl} > 0 \\ v - q_{au} - up_{au} < 0 \\ v - q_{bu} - up_{bu} < 0. \end{cases} \tag{3}$$

Hence the domain is convex.

Metadomains are not always convex. It is easy to construct a counterexample as follows. By the identification of the metadomain with the uncertainty region of an intersection point it follows that the metadomain can also be considered as the intersection of two collinearity regions as defined in [2]. Furthermore,

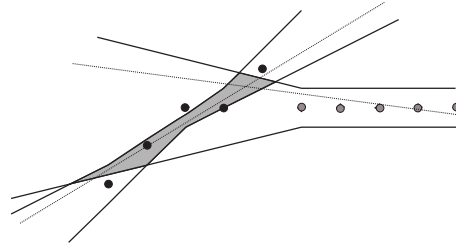


Fig. 5. Non-convex uncertainty region for the intersection point of two lines.

since collinearity regions are not necessarily convex, it is not difficult to find two collinearity regions whose common intersection is not convex, as shown in Fig. 5. In [2] a collinearity region of a digital set S is defined as the set consisting of those points p in \mathbb{R}^2 for which there is at least one Euclidean line passing through p as well as through the uncertainty regions associated with the points of S . In Fig. 5 the intersection of the two collinearity regions is shown as a gray, non-convex region.

Since metadomains are not necessarily convex, to construct the intersection graph of Fig. 1(c), we replace the metadomain by its convex hull, which is done as follows. First, we compute the parameter vector of each line that passes through any vertex of the first domain and any vertex of the second domain, next we take the convex hull of the parameter vectors. This approach overestimates uncertainty. When the conditions of Proposition 1 are met, however, which for real lines is often the case, then the metadomain coincides with its convex hull without any overestimation of uncertainty.

4 Consistency and Grouping

From the metadomains we construct an intersection graph, in which two vertices are joined by an edge when the two metadomains have a non-empty intersection. Next, the connectivity of the intersection graph is used to find larger groups of metadomains that have a common non-empty intersection.

The basic idea is that when two vertices are joined by an edge then this points to the existence of a line crossing three or four domains, and therefore also points to a concurrency relation for digital line segments, i.e., the line segments whose domains have given rise to the metadomains that correspond with the two vertices. The extension of this idea is to extract cliques from the intersection graph, since the vertices of a large clique will often correspond to a large collection of metadomains that have a common non-empty intersection, and thus indirectly it will correspond to a concurrency relation that involves a large number of line segments. There is no guarantee, however, that the metadomains of clique always have a non-empty intersection. Fig. 6(a) shows an exception, where we have six line segments, and the metadomains AB , CD and EF which are

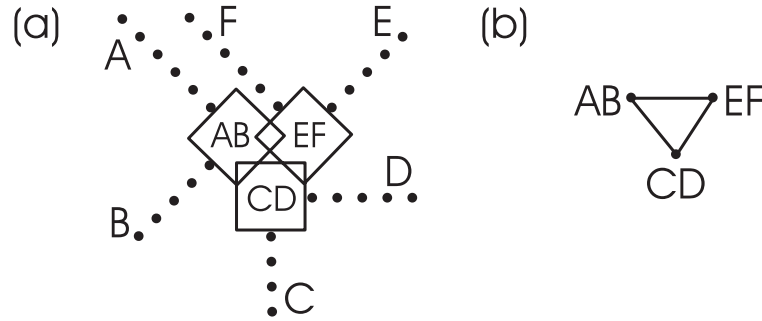


Fig. 6. Clique whose metadomains have no common intersection.

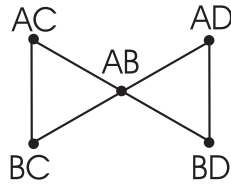


Fig. 7. Minimum clique covering leads to inconsistencies.

shown here as the uncertainty regions of the six line segments. Fig. 6(b) shows a subgraph of the intersection graph induced by the vertices AB , CD , EF , which form a clique. Nonetheless, the intersection of the metadomains is empty. Such cases are rare, however. In fact, they can only occur when the union of the three metadomains forms a topological torus. In this work, whenever such a situation occurred we solved this problem by taking the convex hull of the hole of the torus as the uncertainty region of the intersection point.

A second problem is that most groupings, and therefore cliques also, lead to inconsistent geometric relations. This is illustrated in Fig. 7. A minimum clique covering algorithm may propose, for example, the cliques $AC - AB - BC$ and $AD - BD$ and thus the concurrent relations $\text{conc}(A, B, C)$, $\text{conc}(A, B, D)$. Since these two triples share two line segments (A and B), we should also have $\text{conc}(A, B, C, D)$, which is not the case here, because some edges are missing, and therefore some of the metadomains that are involved do not have a common non-empty intersection.

Informally we can define a set of relations between digital line segments as consistent if the set contains no relations that cannot occur in Euclidean geometry. A further requirement would be that for sufficiently small values of the uncertainty parameter τ the relations found in uncertain geometry should coincide with Euclidean geometry. Note, however, that consistency is violated already at the lowest level, since parallelism and collinearity are not transitive in

uncertain geometry. It follows that we can only restore consistency upto a certain degree. For example, even when we restore the usual properties of parallelism, collinearity and concurrency, more involved properties such as Pappus' Theorem may still not hold. As for concurrency, in this work we restrict ourselves to the removal of the most obvious inconsistencies:

$$(A \parallel B) \wedge (A \neq B) \wedge \text{conc}(A, B, C), \quad (4)$$

$$(A \not\parallel B) \wedge (A \not\parallel C) \wedge (B \not\parallel C) \wedge \text{conc}(A, B, C) \wedge \text{conc}(B, C, D) \wedge \neg \text{conc}(A, B, D). \quad (5)$$

To remove inconsistencies of the form (4) or (5) we shall use the following result, which gives a sufficient, but not necessary, condition.

Lemma 2. *Let G be the intersection graph of the metadomains of a collection of digital line segments A, B, \dots . Let S be a set of disjoint cliques of G such that the pathlength between two vertices belonging to distinct cliques in S is always larger than two. If in the set of concurrency relations implied by the cliques in S we have two relations of the form $\text{conc}(A, B, C, \dots)$ and $\text{conc}(A, B, P, \dots)$, then we have $A \parallel B$.*

Proof. Suppose we have $\text{conc}(A, B, C, \dots)$ and $\text{conc}(A, B, P, \dots)$. Assume that $A \not\parallel B$. Then AB appears as a vertex in the intersection graph. Since there is a clique whose vertices imply the relation $\text{conc}(A, B, C, \dots)$, either AB must belong to this clique, or there must be vertices AX and BY in the clique, where X and Y can even be identical. Since AX and BY are adjacent, there is a common transversal cutting the four domains A, X, B and Y . Hence, both AX and BY must also be adjacent to AB . Similarly, in the clique generating the relation $\text{conc}(A, B, P, \dots)$, there must be vertices AU and BV that are also adjacent to AB . In this case both cliques would have a common neighbor. Since we exclude such cliques, it follows that the assumption $A \not\parallel B$ cannot be true. In fact, two relations of the form $\text{conc}(A, B, C, \dots)$ and $\text{conc}(A, B, P, \dots)$ can only arise when AB is not a vertex of the graph, or in other words when $A \parallel B$. In particular, A may even be collinear with B .

To find consistent concurrency relations, we use a greedy algorithm which is based on Lemma 2. First, we extract a maximum clique and remove all its neighbors as well as the neighbors of these neighbors from the intersection graph of the metadomains. By proceeding with the next maximum clique in the graph, we construct thus a list of cliques such that no two cliques share a common neighbor. Next, from each clique we derive the implied concurrency relation. For example, a clique with vertices AB, CD, AE implies $\text{conc}(A, B, C, D, E)$. The result is shown in Fig. 3, where we draw the uncertainty regions for the intersection points of a number of concurrency relations, i.e., the region ABH for the relation $\text{conc}(A, B, H)$. Finally, we remove those concurrency relations which involve one or more pairs of digital line segments that are parallel but not collinear. Since C and E are parallel, only $\text{conc}(A, B, F)$, $\text{conc}(A, B, G)$, $\text{conc}(A, B, H)$ are left as valid concurrency relations.

With the above algorithm we can avoid inconsistencies of the form (4) as well as (5). Suppose, for example, that we have found the concurrency relations $\text{conc}(A, B, C)$ and $\text{conc}(A, B, D)$. Because the extracted cliques satisfy the conditions of Lemma 2, it follows that $A \parallel B$, which excludes an inconsistency of the form (5). In addition, if A is not collinear with B , then both concurrency relations will be discarded in the final step of the algorithm where we examine parallel pairs. This excludes an inconsistency of the form (4). If A and B are collinear, however, then the concurrency relations will not be discarded because they can coexist without any contradiction.

5 Concluding Remarks

We presented a method for the derivation of concurrency relations in uncertain geometry. The removal of inconsistencies is a major part in this approach. Without doubt, there is still no unique decisive procedure that guarantees the removal of all possible geometric inconsistencies. In this work we have proposed one approach, which gives satisfactory results on real images, but still has some shortcomings. In particular, the following important questions have been barely examined. How can we obtain consistency when we consider distinct types of relations? For example, if A is collinear with D while A , B and C are concurrent, then consistency requires that B , C and D must also be concurrent. Upto now the consistency of properties such as parallelism, collinearity and concurrency have only been examined separately. Furthermore, in real applications we must combine the extraction of linear structure with other properties such as proximity and symmetry. Also this question has not been sufficiently examined yet. Finally, the uncertainty is controlled by the uncertainty parameter τ . When analyzing relations in a digital image, we often derive geometric relations for different values of τ . It is a natural requirement that these relations should behave in a consistent way when we increase or decrease τ .

References

1. H. Heijmans and A. Toet, "Morphological sampling," *CVGIP: Image Understanding*, vol. 54, pp. 384–400, 1991.
2. P. Veelaert, "Geometric constructions in the digital plane," *J. Math. Imaging and Vision*, vol. 11, pp. 99–118, 1999.
3. P. Veelaert, Parallel line grouping based on interval graphs, *Proc. of DGCI 2000*, vol. 1953 of Lecture Notes in Computer Science, pp. 530–541. Uppsala, Sweden: Springer, 2000.
4. P. Veelaert, "Graph-theoretical properties of parallelism in the digital plane," submitted.
5. P. Veelaert, "Collinearity and weak collinearity in the digital plane," *Digital and Image Geometry*, vol. 2243 of Lecture Notes in Computer Science, pp. 434–447, Springer, 2001.
6. P. Veelaert, "Line grouping based on uncertainty modeling of parallelism and collinearity," in *Proceedings of SPIE's Conference on Vision Geometry IX*, (San Diego), pp. 36–45, SPIE, 2000.

7. V. J. Milenkovic, "Verifiable implementations of geometric algorithms using finite precision arithmetic," in *Geometric Reasoning* (Kapur and Mundy, eds.), pp. 377–401, Cambridge: MIT Press, 1989.
8. H. F. Durrant-Whyte, "Uncertain geometry in robotics," *IEEE Trans. Robotics Automat.*, pp. 23–31, 1988.
9. H. F. Durrant-Whyte, "Uncertain geometry," in *Geometric Reasoning* (Kapur and Mundy, eds.), pp. 447–481, Cambridge: MIT Press, 1989.
10. A. Fleming, "Geometric relationships between toleranced features," in *Geometric Reasoning* (Kapur and Mundy, eds.), pp. 403–412, Cambridge: MIT Press, 1989.
11. D. Lowe, "3-d object recognition from single 2-d images," *Artificial Intelligence*, vol. 31, pp. 355–395, 1987.
12. M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *CACM*, vol. 24, pp. 381–395, 1981.