Journal of Universal Computer Science, vol. 10, no. 8 (2004), 955-966
submitted: 22/3/04, accepted: 28/6/04, appeared: 28/8/04

Conflict Avoidance in Additive Order Diagrams

Bernhard Ganter
Algebra Institute
Dresden University of Technology, Germany
ganter@math.tu-dresden.de

Abstract: Additive diagrams are used by several interactive lattice layouters. We discuss a method to avoid unwanted incidences when working with additive diagrams.
Key Words: lattice diagrams, automatic diagram layout
Category: H, H.3.m

1 Introduction

Ordered sets and lattices, in particular in their interpretation as concept lattices, have become rather popular recently. One of their nice aspects is that they can be represented by means of very instructive and intuitive diagrams. There are computer programs available for making such diagrams automatically, some of which are based on standard graph drawing techniques. But their results are not fully satisfactory. Experience shows that diagrams made manually often are considerably better. It is therefore of interest to find additional strategies for automatic or semiautomatic diagram drawing. One such method is discussed here. We cannot present extensive tests of the new methods, because our implementations still are rather preliminary, but the results obtained so far are promising.

2 Additive line diagrams

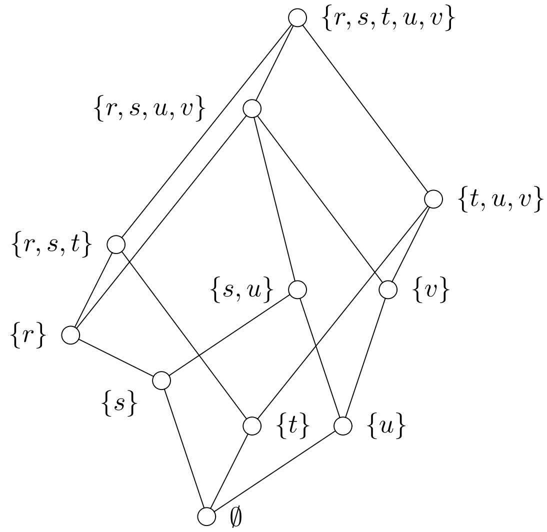
By a *line diagram* we mean a graph diagram of the simplest kind, with small circles representing the vertices and with line segments for the edges. To draw such a diagram (in the plane $\mathbb{R}^2$) we need as input a graph (V, E) and a *placement function*

$$\underline{\text{pos}}: V \to \mathbb{R}^2,$$

that associates coordinates to each vertex.

An *additive representation* of such a placement map $\underline{\text{pos}}$ consists of a set D, a map $\underline{\text{rep}}: V \to \mathfrak{P}(D)$, and a map $\underline{\text{vec}}: D \to \mathbb{R}^2$ with

$$\underline{\text{pos}}(v) = \sum_{d \in \underline{\text{rep}}(v)} \underline{\text{vec}}(d).$$



$d \in D$	$\underline{\mathrm{vec}}(d)$
r	$(-2, 1)$
s	$(-1, 3)$
t	$(1, 2)$
u	$(3, 2)$
v	$(1, 3)$

Figure 1: An example of an additive line diagram. The representing set is $D := \{r, s, t, u, v\}$, the value of the $\underline{\mathrm{rep}}$ function is displayed next to each vertex, and $\underline{\mathrm{vec}}$ is given by the table on the right hand side.

A line diagram with a given additive representation is called an *additive line diagram* [3]. An example is shown in Figure 1.

Every placement function has an additive representation, with $D := V$, $\underline{\mathrm{rep}}(v) := \{v\}$ and $\underline{\mathrm{vec}}(v) := \underline{\mathrm{pos}}(v)$ for all $v \in V$. So, in a sense, every diagram is additive. But much more interesting additive representations are those where D is small compared to V.

When an additive diagram is to be modified, a natural strategy is to keep the $\underline{\mathrm{rep}}$ function fixed and to manipulate only the $\underline{\mathrm{vec}}$ mapping. If only a single value, say $\underline{\mathrm{vec}}(d)$, is changed by adding a vector $\mathbf{a}$, then the diagram changes as follows: all vertices v with $d \in \underline{\mathrm{rep}}(v)$ are shifted by $\mathbf{a}$, all other vertices remain fixed, see Figure 2.

3 Order diagrams

Ordered sets, lattices (i.e., ordered sets with infima and suprema), and more generally directed acyclic digraphs can be represented by "upward drawings" (also called Hasse diagrams), where the direction of an edge is indicated by an increasing y-coordinate. A random placement function for such a diagram may be not admissible for two reasons:

"conflicts": vertices colliding with other vertices or with edges they are not

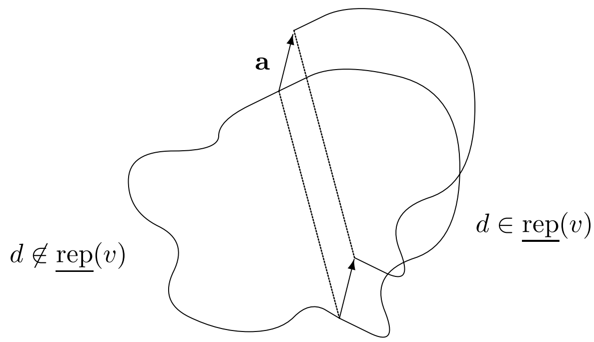


Figure 2: Changing $\underline{\text{vec}}(d)$ corresponds to shifting those vertices v with $d \in \underline{\text{rep}}(v)$.

incident with,

"inversions": edges that should be upward but are not.

Additive diagrams offer an easy way to avoid inversions. If the mapping $\underline{\text{rep}}$ is chosen to be order preserving[1], then it suffices to use for $\underline{\text{vec}}(d)$ only values with a positive y-coordinate. The resulting diagram then is free of inversions.

For example, let $D := J(L)$ be the set of all join-irreducible elements of a finite lattice L. Since every lattice element is the join of join-irreducible elements,

$$\underline{\text{rep}}(v) := \{j \in J(L) \mid j \leq v\}$$

defines an order embedding of L into $\mathfrak{P}(J(L))$. Figures 1 and 3 gives a small example. The join-irreducible elements are those with a unique lower neighbor. The edge connecting this lower neighbor with the respective element d represents the vector $\underline{\text{vec}}(d)$. The position of an arbitrary lattice element is determined by the sum of such vectors below it. Since the vectors $\underline{\text{vec}}(d)$ are directly visible in the diagram, they can easily be modified in an interactive layout program: Join-irreducible elements may be dragged, and the placement function is modified accordingly. The effect on the diagram is that with a join-irreducible element, the order filter it generates is moved. It turns out that such restricted modifications often lead to better results than unrestricted ones. This technique was successfully used by several popular implementations, for example by the ANACONDA lattice layouter of NAVICON AG[2]. In practice the method is usually

[1] That is, if $v \leq w$ always implies $\underline{\text{rep}}(v) \leq \underline{\text{rep}}(w)$. Such a mapping is also called a *set representation.*

[2] www.navicon.de

dualized and meet-irreducible elements are used instead, because this is more intuitive. Confer [3] for examples and further references.

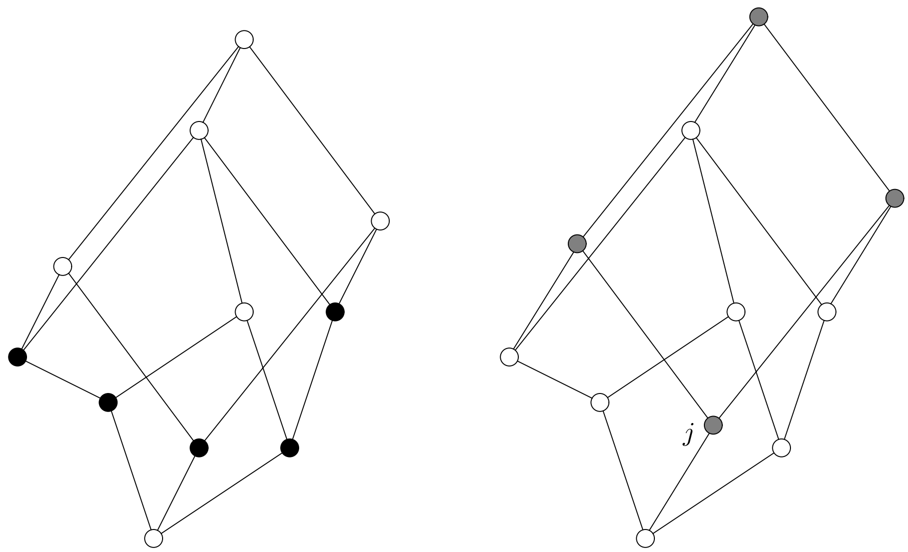


Figure 3: Left: Additive lattice diagram in which the join-irreducible elements (black circles) are used for the set representation. Right: The same lattice after a slight shift of the join-irreducible element j and the filter generated by it (shaded vertices).

4 Conflict charts

The additive method avoids inversions in order diagrams, but contributes nothing to the problem of possible conflicts. We discuss this issue next, assuming an additive diagram[3]. Again we concentrate on a single $d \in D$ and describe the conflicts that can occur when $\underline{\text{vec}}(d)$ is changed by adding a vector $\mathbf{a}$. All values of $\mathbf{a}$ that lead to a conflict, i.e., to an unwanted collision in the line diagram, will be displayed in a *conflict chart* for d, which we will introduce next.

Let v_1 be a vertex of the graph (V, E) and let $\{v_2, v_3\}$ be an edge not containing v_1. Let $p_i := \underline{\text{pos}}(v_i)$, $i \in \{1, 2, 3\}$, be the points where v_1, v_2, v_3 are drawn by a given placement map $\underline{\text{pos}}$. The edge $\{v_2, v_3\}$ is then represented by

[3] This condition is not essential.

the undirected line segment $\overline{p_2,\ p_3}$. Finally, let ε_i be a flag indicating if d is contained in $\underline{\text{rep}}(v_i)$, thus

$$\varepsilon_i := \begin{cases} 0 & \text{if } d \notin \underline{\text{rep}}(v_i), \\ 1 & \text{if } d \in \underline{\text{rep}}(v_i). \end{cases}$$

A conflict arises when

$$p_1 + \varepsilon_1 \mathbf{a} \quad \in \quad \overline{p_2 + \varepsilon_2 \mathbf{a},\ p_3 + \varepsilon_3 \mathbf{a}}.$$

There are eight possible choices for $(\varepsilon_1, \varepsilon_2, \varepsilon_3)$ to consider. But clearly $(\varepsilon_1, \varepsilon_2, \varepsilon_3) = (0,0,0)$ and $(\varepsilon_1, \varepsilon_2, \varepsilon_3) = (1,1,1)$ cannot cause any new conflict. Moreover, we may w.l.o.g. assume $\varepsilon_2 \leq \varepsilon_3$. This leaves four possible incidences between a point and a line segment:

1. $p_1 \in \overline{p_2 + \mathbf{a},\ p_3 + \mathbf{a}}$,
2. $p_1 \in \overline{p_2,\ p_3 + \mathbf{a}}$,
3. $p_1 + \mathbf{a} \in \overline{p_2,\ p_3}$,
4. $p_1 + \mathbf{a} \in \overline{p_2,\ p_3 + \mathbf{a}}$.

Solving these for $\mathbf{a}$, we get the cases displayed in Table 1.

$(\varepsilon_1, \varepsilon_2, \varepsilon_3)$	$q :=$	$r :=$	conflict if	for some λ with
$(0,1,1)$	$p_1 - p_2$	$p_1 - p_3$	$\mathbf{a} = q + \lambda(r - q)$	$0 \leq \lambda \leq 1$
$(0,0,1)$	$p_2 - p_3$	$p_1 - p_3$	$\mathbf{a} = q + \lambda(r - q)$	$1 \leq \lambda$
$(1,0,0)$	$p_2 - p_1$	$p_3 - p_1$	$\mathbf{a} = q + \lambda(r - q)$	$0 \leq \lambda \leq 1$
$(1,0,1)$	$p_2 - p_3$	$p_2 - p_1$	$\mathbf{a} = q + \lambda(r - q)$	$1 \leq \lambda$

Table 1: These conflicts can occur.

The values of $\mathbf{a}$ for which a conflict occurs are the “conflict lines” (actually, line segments and rays) for the chosen $d \in D$. Together, they form the conflict chart of d. Again, this is only the simplest version. There are other aspects that may be included in the conflict chart, for example the actual size of the circles that represent the vertices.

It is convenient to display the conflict chart when a diagram is being manipulated. Figure 4 (left) shows such a situation: a diagram with a join-irreducible element selected, superimposed with a rectangular part of the conflict chart for that element. It is apparent that in the upper left half of the diagram there is a large “conflict free” zone. Thus moving the selected element into this zone will not cause new conflicts. This is demonstrated in the right part of Figure 4. The new diagram is slightly better, even after shrinking it to the height of the previous diagram.

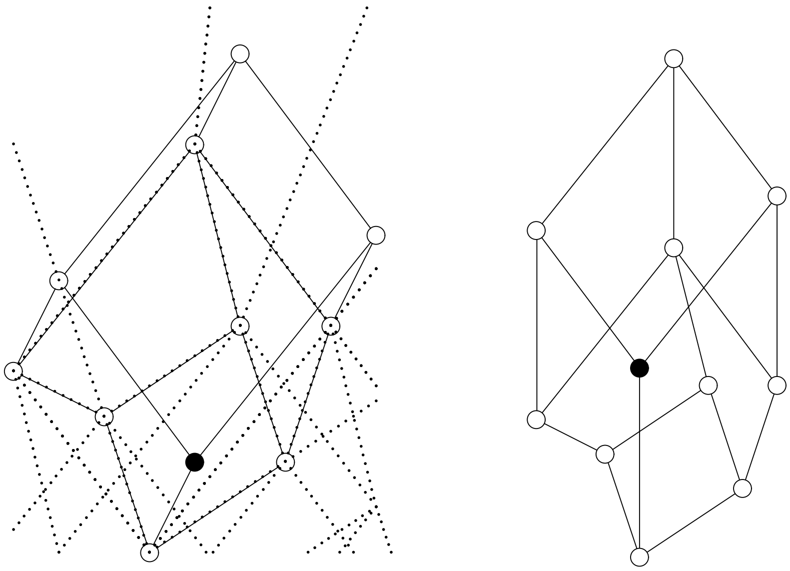


Figure 4: Left: An additive diagram (solid lines) superimposed with the conflict chart for the blackened join-irreducible element (dotted lines). Right: A slightly better diagram obtained by first moving the blackened element into a conflict free zone and then adjusting size.

5 Modified Voronoi diagrams

For a given conflict chart and an arbitrary point p of the plane let the *conflict line distance* of p be the shortest distance of p to a conflict line. Figure 4 suggests an automated method for improving additive diagrams: Move the selected join irreducible lattice element to a point with largest possible conflict line distance (within given limits, say). With other words: look for a largest circle that does not intersect any conflict line, and move the irreducible to its midpoint.

This can easily be realized graphically by coloring the plane with one color, say, grey, such that the color intensity expresses small conflict line distance. Then dark areas contain positions close to conflicts, while white areas are safe. In an interactive layout program, such color charts can be useful because they very intuitively "give recommendations" how to avoid conflicts.

Computationally, a largest circle that fits into (the interior of) a conflict chart can be determined using its *Voronoi diagram* (defined as follows: a point p of the plane belongs to the Voronoi diagram boundary of a conflict chart if its conflict line distance is the distance of p to more than one conflict line). It is evident that the center of a circle touching three conflict lines is a branching point ("vertex") of the Voronoi diagram boundary. The algorithmic problems of efficiently constructing the Voronoi diagram and its vertices are well studied, see de Berg et al [1] for an introduction. In our setting however, efficiency is not the main problem anyway, since the input size is small and discretization is possible.

A more important problem is that our notion of "conflict line distance" needs to be revisited, for two reasons:

1. We have suggested to move the selected element to a point of "largest possible conflict line distance". However, there is no such point, because conflict charts of finite graphs consist of finitely many conflict lines only. Therefore there will always be points of arbitrary large conflict line distance "far out". This corresponds to the trivial fact that all non-zero distances always can be enlarged simply by magnifying the diagram as a whole.

2. The conflict line distance does not always represent the distance to a conflict. This is explained in Figure 5 by means of an example: On the left hand side, we see a diagram of a very small graph having three vertices u, v and w and only one edge $\{v, w\}$. p_1, p_2, and p_3 shall denote the positions of u, v, w. For simplicity we assume $p_3 = (0, 0)$.

   Let $d \in D$ be some element of the representation set and assume that $d \in \underline{\text{rep}}(w)$, but $d \notin \underline{\text{rep}}(u)$, $d \notin \underline{\text{rep}}(v)$. If we change $\underline{\text{vec}}(d)$ by, say, $\mathbf{a}$, then the position of the vertex w will be moved to $p_3 + \mathbf{a}$, which is equal to $\mathbf{a}$. The other two vertices remain where they are. The only possibility for a conflict with this choice of d is to move w in such a way that the edge $\{v, w\}$ goes through u. Therefore this conflict chart has only one conflict line. It is a ray starting at p_1 in the direction away from p_2.

   Now suppose that we change $\underline{\text{vec}}(d)$ by a vector $\mathbf{a}$ as indicated on the right hand side. The conflict line distance for this choice of $\mathbf{a}$ is δ_2. The distance δ_1 between p_1 and the edge is shorter.

Both problems can simultaneously be solved by modifying our notion of conflict distance. A first modification is suggested by Figure 5: we redefine the distance to be δ_1 instead of δ_2. This has to be done in all instances of the cases (2) and (4) from Section 3, that is, for all conflict rays. The modified distance can always be expressed as the Euclidian distance of a point to a line segment, see Table 2. The following proposition describes the outcome.

Proposition 1 *Let v_1 be a vertex and let $\{v_2, v_3\}$ be an edge of (V, E), $v_1 \notin \{v_2, v_3\}$. Let an additive diagram of (V, E) be given by $\underline{\text{rep}} : V \to \mathfrak{P}(D)$ and $\underline{\text{vec}} : D \to \mathbb{R}^2$, and let $p_i := \underline{\text{pos}}(v_i)$ for $i \in \{1, 2, 3\}$. Moreover, let*

$$p_i^0 := \begin{cases} \underline{\text{pos}}(v_i) & \text{if } d \notin \underline{\text{rep}}(v_i) \\ \underline{\text{pos}}(v_i) - \underline{\text{vec}}(d) & \text{if } d \in \underline{\text{rep}}(v_i) \end{cases}$$

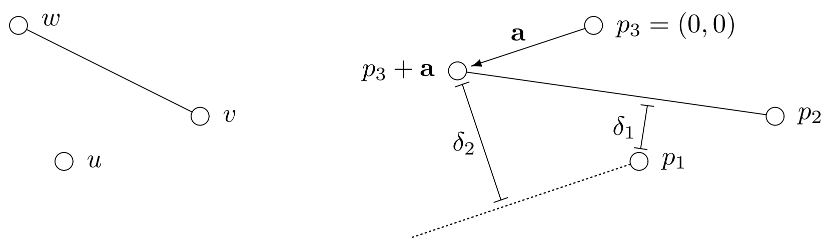


Figure 5: The distance δ_1 between p_1 and the edge $\overline{p_2,\ p_3+\mathbf{a}}$ is different from the distance δ_2 between $\mathbf{a}$ and the dotted conflict line.

be the placement for $\underline{\text{vec}}(d) = o$.

Then the point-edge distance between p_1 *and* $\overline{p_2,\ p_3}$ *is given in Table 2, provided it depends on the choice of* $\mathbf{a} := \underline{\text{vec}}(d)$.

Proof In the first case we have $p_1 = p_1^0$, $p_2 = p_2^0 + \mathbf{a}$, $p_3 = p_3^0 + \mathbf{a}$. Then

$$\begin{aligned}\text{dist}(p_1, \overline{p_2,\ p_3}) &= \text{dist}(p_1^0, \overline{p_2^0 + \mathbf{a},\ p_3^0 + \mathbf{a}})\\ &= \text{dist}(p_1^0 - p_1^0 - \mathbf{a}, \overline{p_2^0 + \mathbf{a} - p_1^0 - \mathbf{a},\ p_3^0 + \mathbf{a} - p_1^0 - \mathbf{a}})\\ &= \text{dist}(-\mathbf{a}, \overline{p_2^0 - p_1^0,\ p_3^0 - p_1^0})\\ &= \text{dist}(\mathbf{a}, \overline{p_1^0 - p_2^0,\ p_1^0 - p_3^0}).\end{aligned}$$

The other cases are similar. □

Taking the minimum of this value over all appropriate v_1, v_2, v_3, we obtain, for arbitrary $\mathbf{a} \in \mathbb{R}^2$, the *conflict distance* of $\mathbf{a}$. Note that, although the difference to the previous definition seems to be small, the change is remarkable. The conflict distance is, except for trivial cases[4], bounded and never exceeds the *diameter*, i.e., the maximal distance between two vertices of the diagram. This is true because $\text{dist}(q, \overline{\mathbf{a},\ r})$ is bounded above by $\text{dist}(q, r)$.

Although the conflict distance is bounded above, we propose a second modification to compensate for the increase in distance that is caused by plainly magnifying the diagram. This can be done by dividing the conflict distance of any given point $\mathbf{a}$ by the diagram growth that is caused by letting $\underline{\text{vec}}(d) := \mathbf{a}$. There is no canonical definition of a diagram's size, but a natural choice is the

[4] when there are no conflict rays at all.

$d \in \underline{\text{rep}}(v_1)$	$d \in \underline{\text{rep}}(v_2)$	$d \in \underline{\text{rep}}(v_3)$	$q :=$	$r :=$	distance to conflict
$-$	$\times$	$\times$	$p_1^0 - p_2^0$	$p_1^0 - p_3^0$	$\text{dist}(\mathbf{a}, \overline{q,\ r})$
$-$	$-$	$\times$	$p_1^0 - p_3^0$	$p_2^0 - p_3^0$	$\text{dist}(q, \overline{\mathbf{a},\ r})$
$\times$	$-$	$-$	$p_2^0 - p_1^0$	$p_3^0 - p_1^0$	$\text{dist}(\mathbf{a}, \overline{q,\ r})$
$\times$	$-$	$\times$	$p_2^0 - p_1^0$	$p_2^0 - p_3^0$	$\text{dist}(q, \overline{\mathbf{a},\ r})$

Table 2: Edge-vertex distance between a vertex at p_1 and an edge at $\overline{p_2,\ p_3}$ caused by letting $\underline{\text{vec}}(d) := \mathbf{a}$. W.l.o.g. only the cases with $d \in \underline{\text{rep}}(v_2) \Rightarrow d \in \underline{\text{rep}}(v_3)$ are listed. p_i^0 denotes the respective position for $\underline{\text{vec}}(d) = 0$.

diameter. Then

$$\text{growth}(\mathbf{a}) := \frac{\text{diameter when } \underline{\text{vec}}(d) = \mathbf{a}}{\text{diameter when } \underline{\text{vec}}(d) = 0},$$

assuming the denominator to be non-zero.

For a given additive diagram and a fixed $d \in D$ we define the *conflict avoidance parameter* of an arbitrary point $\mathbf{a}$ to be the conflict distance of $\mathbf{a}$ divided by growth($\mathbf{a}$). We speak of a *variable edge-vertex distance* for those edge-vertex pairs where the distance depends on the choice of $\underline{\text{rep}}(d)$.

Theorem 1 *Except for trivial cases, there are points* $\mathbf{a}$ *of maximal conflict avoidance. These are the points where the minimal variable edge-vertex distance, relative to the diameter, is maximal.*

Proof The second sentence simply rephrases the definition of the conflict avoidance parameter: it is the conflict distance divided by the growth. That is, it is equal to the minimal variable distance, times the diameter δ_0 for $\underline{\text{rep}}(d) = 0$ (a positive constant), divided by the diameter $\delta_{\mathbf{a}}$ for $\underline{\text{rep}}(d) = \mathbf{a}$.

The growth function is bounded below by some positive constant[5]. Therefore the avoidance is bounded above. $\delta_{\mathbf{a}}$ is at least $|a| - \delta_0$. Thus the avoidance parameter goes to zero whenever $|a|$ goes to infinity, and takes its maximal values on some compact neighborhood of the origin. □

How to find the points of maximal conflict avoidance? This could be done with a "modified Voronoi diagram", defined as above but with the conflict avoidance parameter replacing the conflict line distance. Such diagrams are no longer

[5] We have excluded trivial cases.

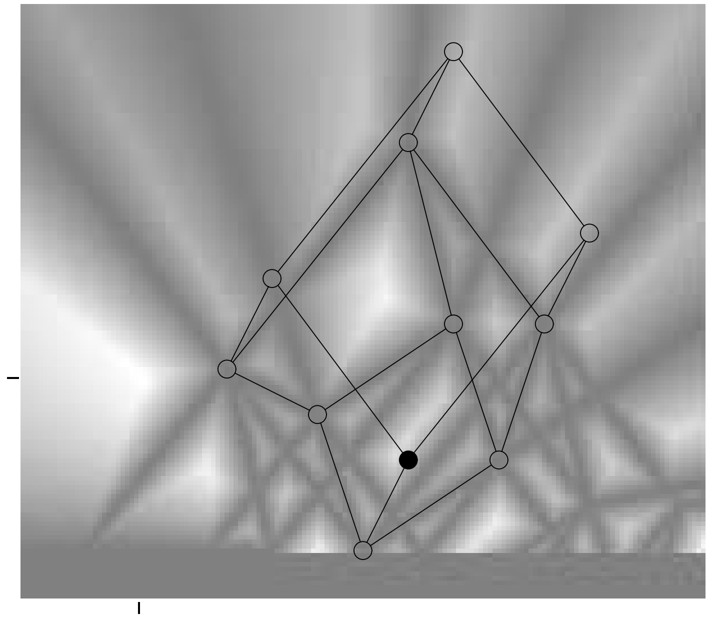


Figure 6: Low conflict avoidance values correspond to darker grey. A spot of highest conflict avoidance can be found on the left. Its coordinates are marked at the boundary. Only points with positive y-coordinate were considered, in order to avoid inversions.

linear and can be quite complicated. It is not yet clear how to handle them algorithmically. Note that already in our simple example there are one hundred conflict lines (only a few of them intersecting the window printed in Figure 4).

We have only first experiences with an automatic search. Bernhard Schmidt [4] has developed an algorithm that effectively finds the points of maximal conflict avoidance. His proof is based on the Lipschitz continuity of the parameter. His implementation, however, was experimental and not fast enough for practical use. Christian Zschalig [5] has combined these ideas with a standard technique in Graph Drawing [2] and has used the conflict avoidance parameter as a force in a Force Directed Placement Algorithm. The results are promising, but his implementation is also not yet ready for general use.

Color maps are easier to use. An example is shown in Figure 6. For the growth function we have used height instead of diameter, because this is easier to compute. Other variations of the growth function may be introduced to reward integer coordinates, few slopes, etc.

Figure 6 suggests dragging the blackened point to the white spot on the left. This does in fact result in a diagram with a slightly better minimum distance. It is shown in Figure 7. No better minimum distance can be achieved in an additive diagram of this height without moving other points. The new diagram is wider than the previous ones, but its diameter is only slightly larger.

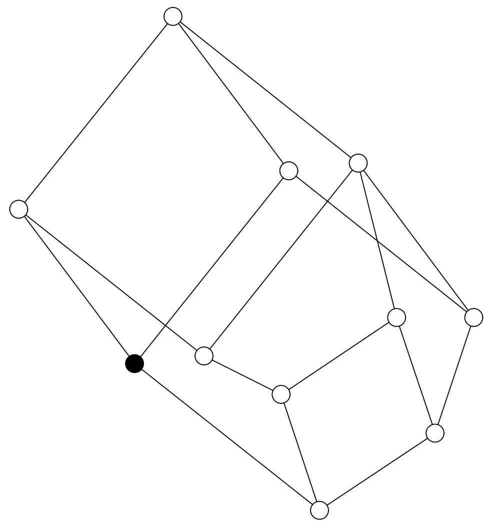


Figure 7: Positioning the marked join-irreducible at the position suggested by Figure 6 does lead to an additive diagram in which the shortest nonzero distance between an edge and a vertex is larger than in Figure 4, even after height adjustment.

6 A concluding remark

The reader should be warned that, although the technique presented here may be useful, it does not by itself provide a way to good lattice diagrams. The automated lattice drawing problem still appears to be difficult, although some progress has been made. Figure 8 shows a reasonably good diagram, as it would be intuitively be drawn by hand. Repeated application of the methods described above does not necessarily converge to such a “nice” diagram, unless supported by further considerations. This is subject of ongoing research.

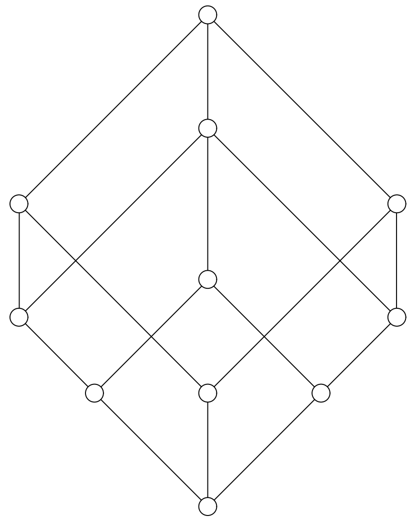


Figure 8: A better additive diagram for the lattice in Figures 1, 3, 4, and 7.

References

1. M. de Berg, M. Kreveld, M. Overmars, and O. Schwarzkopf: *Computational Geometry — Algorithms and Applications.* Springer, 1999.
2. G. Di Battista, P. Eades, R. Tamassia, and I.G. Tollis: *Graph Drawing.* Prentice Hall, 1999.
3. B. Ganter, R. Wille: *Formal Concept Analysis — Mathematical Foundations.* Springer, 1999.
4. B. Schmidt: *Ein Optimierngsalgorithmus für additive Liniendiagramme.* Diplomarbeit, Dresden University, 2002.
5. C. Zschalig: *Ein Force Directed Placement Algorithmus aum Zeichnen von Liniendiagrammen.* Diplomarbeit, Dresden University, 2002.