A Bayesian Paradigm for Dynamic Graph Layout

Ulrik Brandes and Dorothea Wagner

Fakultät für Mathematik und Informatik
Universität Konstanz, Germany
{Ulrik.Brandes, Dorothea.Wagner}@uni-konstanz.de

Abstract. *Dynamic graph layout* refers to the layout of graphs that change over time. These changes are due to user interaction, algorithms, or other underlying processes determining the graph. Typically, users spend a noteworthy amount of time to get familiar with a layout, i.e. they build a *mental map* [ELMS91]. To retain this map at least partially, consecutive layouts of similar graphs should not differ significantly. Still, each of these layouts should adhere to constraints and criteria that have been specified to improve meaning and readability of a drawing.
In [BW97], we introduced random field models for graph layout. As a major advantage of this formulation, many different layout models can be represented uniformly by random variables. This uniformity enables us to now present a framework for dynamic layout of arbitrary random field models. Our approach is based on Bayesian decision theory and formalizes common sense procedures. Example applications of our framework are dynamic versions of two well-known layout models: Eades' spring embedder [Ead84], and Tamassia's bend-minimum orthogonal layout model for plane graphs [Tam87].

1 Introduction

A sequence of graphs that arises from repeated modification of an initial graph is called a *dynamic graph*. Dynamic graphs occur in many settings, including user interaction, software visualization, animation of graph algorithms, and graph queries. When drawing graphs dynamically, updates shall be economic in the sense that human as well as computer resources are used sparingly. With human resources we mean the effort and time a user spents in order to follow an update and regain familiarity with the drawing. Despite of the large interest in automatic graph drawing, the important extension to dynamic graphs has received little attention, yet. This may partially be due to the lack of a conceptual framework abstracting from issues inherent to specific models. Apparently, most approaches in the literature are tied to classes of admissible graphs [BP90,CDT+92,CDTT95,Nor96,PT96].

A fairly general formulation of the dynamic graph drawing problem is given by North [Nor96]. We use his formalization with minor adjustments. For each graph in the sequence, prescriptions are given for its *representation* and *rendering*, as well as a set of *layout constraints* and *readability criteria*. These define a sequence of static graph drawing problems. Since the computation of a layout and its rendering can typically be separated, we neglect everything related to imaging, and focus on the layout problem. Consecutive layouts are subject to *consistency*, *stability*, and *readability* demands. These can be seen as additional constraints (for consistency) and criteria (for stability and readability of updates), respectively. Moreover, the actual update of a drawing might consist of a number of intermediate drawings (the *physical update*), e.g. to smooth transitions or to add visual update clues. Here, we are only concerned with the *logical update*, i.e. the final layout for each graph in the sequence.

Our approach to dynamic graph drawing relies on the formulation of layout models in terms of random fields [BW97] (see [Guy94] for an introduction to random fields). In a random field model, feasible layouts are assigned probabilities reflecting their conformance to layout goals. A single random variable is sufficient to fully describe a layout model. Random fields are used in many other areas, particularly including image processing [Win95]. In the introduction of [GM85], Geman and McClure explicitly stated their hope to achieve a unification of many image processing problems through a Bayesian framework, which was pioneered in [GG84]. To say the least, random fields and the Bayesian paradigm have influenced a substantial part of the subsequent research [CJ93]. In this paper, we undertake an adaption to dynamic graph layout. We argue that dynamic graphs should be laid out according to an *a-posteriori* model, which forms a compromise between the given static, or *a-priori*, model and some stability criteria.

The main interest of this paper is the derivation of a formal concept for dynamic graph layout. The advantage of an abstract, generic framework is that it provides a common foundation for dynamic versions of arbitrary layout models. Experimental results would be interesting for special cases, but we rather want to emphasize the conceptual elegance and broad applicability of our approach.

This paper is organized as follows. In Section 2 we briefly recall the definition of a random field layout model. The Bayesian approach to dynamic graph layout is developed in Section 3, and applied to a spring model and to bend-minimum orthogonal drawings of embedded planar graphs in the succeeding sections.

2 Preliminaries

A layout of a single graph $G = (V, E)$ is computed by assigning values to certain layout variables. Straight-line embeddings, for example, are completely determined by an assignment of coordinates to each vertex. More general, each element of a set $L = \{l_1, \dots, l_k\}$ of *layout elements* is assigned a value from a set of allowable *states* $\mathcal{X}_l$, $l \in L$. Clearly, L and $\mathcal{X} = \mathcal{X}^L = \mathcal{X}_{l_1} \times \cdots \times \mathcal{X}_{l_k}$ depend on the desired type of representation. Every vector $x \in \mathcal{X}$ is called a *layout*.

In a random field model [BW97], layouts $x \in \mathcal{X}$ are assigned probabilities reflecting their conformance to layout criteria. These probabilities are based on configurations of subsets of layout elements which mutually affect their states. This interaction of layout elements is modelled by an *interaction graph* $G^\eta = (L, E^\eta)$ that is obtained from a *neighborhood system* $\eta = \bigcup_{l \in L} \eta_l$, where $\eta_l \subseteq$

$L \setminus \{l\}$ is the set of layout elements for which the state assigned to l is relevant in terms of layout quality. Because of physical analogies, we assume that interactions are symmetric. In particular $l_2 \in \eta_{l_1} \Leftrightarrow l_1 \in \eta_{l_2}$ for all $l_1, l_2 \in L$, so G^η is undirected. The set of cliques in G^η is denoted by $\mathcal{C} = \mathcal{C}(\eta)$. Each clique in $\mathcal{C}$ corresponds to a set of pairwise interacting layout elements. By symmetry, the internal interactions are not visible from outside of the clique, but form a potential. We therefore define the *interaction potential* of clique $C \in \mathcal{C}$ to be any function $U_C : \mathcal{X} \to I\!R$ for which

$$x_C = y_C \quad \Rightarrow \quad U_C(x) = U_C(y)$$

holds for all $x, y \in \mathcal{X}$, where, for $x = (x_l)_{l \in L} \in \mathcal{X}$ and $C \in \mathcal{C}$, $x_C = (x_l)_{l \in C}$. Moreover, x^{y_C} denotes a vector that agrees with $y \in \mathcal{X}$ on $C \in \mathcal{C}$ and with $x \in \mathcal{X}$ on $L \setminus C$. The *energy* $U : \mathcal{X} \to I\!R$ of a layout equals the sum of all interaction potentials, i.e. $U(x) = \sum_{C \in \mathcal{C}} U_C(x)$. Motivated by results from statistical mechanics and thermodynamics, the probability of a layout x is set to

$$P(X = x) = \frac{1}{Z} e^{-U(x)},$$

where Z is a normalizing constant.[1] Clearly, these probabilities depend on the energy only. For convenience, both the random variable X and its distribution $P(X = x)$ are called a (random field) *layout model* for G. Consequently, a layout of low energy, i.e. little interaction of layout elements, is more likely than a layout of high energy. Many common layout models can be described within this framework [BW97].

3 Dynamic Layout

In this section, we introduce a generic approach to dynamically layout a (finite or infinite) sequence of graphs $G_1, G_2, \ldots$, for which (static) layout models $X_1, X_2, \ldots$ are given. The objective function of a random field layout model is its probability measure. Therefore, we would like to derive models, such that the most desirable sequence of layouts maximizes the joint probability $P(X_1 = x_1, X_2 = x_2, \ldots)$. In case the layout models are assumed to be independent, we have

$$P(X_1 = x_1, X_2 = x_2, \ldots) = \prod_i P(X_i = x_i),$$

which is maximized by maximizing each $P(X_i = x_i)$. Independent random fields correspond to a strategy that layouts each individual graph according to its own model and does not care about the user's mental map (except for, possibly, in the physical update). Since smooth transitions between consecutive layouts are pursued, dependencies among the individual models must be introduced. In the remainder of this section, we develop a Bayesian approach.

[1] For readability, we sometimes use $\exp\{x\}$ to denote e^x.

The sequence of graphs to be drawn is typically assumed to be infinite when it is generated over an unknown period of time and the graphs change in an unpredictable manner. Hence, neither the next graph, nor the number of graphs to come is known at any time. Such situations occur in user interaction, network control, phone-call recording, and so on. For a finite sequence $G_1, \ldots, G_t$, the joint probability can be rewritten into conditional (transition) probabilities

$$P\left(X_1 = x_1, \ldots, X_t = x_t\right) = \prod_{i=1}^{t} P\left(X_i = x_i \mid X_{<i} = x_{<i}\right), \qquad (1)$$

where $X_{<i} = x_{<i}$ is shorthand for $X_1 = x_1, \ldots, X_{i-1} = x_{i-1}$. However, knowledge of all graphs and dependencies is still required to obtain any x_i of a sequence $x_1, \ldots, x_t$ that maximizes (1). Such knowledge is typically not provided.[2]

Throughout this paper we therefore assume that each layout of the sequence has to be computed before anything about the next graph is known, i.e. no look-ahead is available. Thus, the following formalization of the dynamic layout problem is obtained: At time $t > 1$, we are given graphs $G_1, \ldots, G_t$, static layout models $X_1, \ldots, X_t$, and layouts $x_1, \ldots, x_{t-1}$. The goal is to compute a layout $x_t \in \mathcal{X}_t$ that forms a compromise between stability and readability. Obviously, the conditional probability $P(X_t = x_t \mid X_{<t} = x_{<t})$ must reflect this notion of compromise. It is hence called the *dynamic layout model* of G_t.

Suppose, $\lambda : \mathcal{X}_t \times \mathcal{X}_t \to \{0, 1\}$ is the (imaginary) zero-one loss function of choosing x_t, when the best choice is x, i.e.

$$\lambda(x_t, x) = \begin{cases} 0 & x_t = x \\ 1 & x_t \neq x. \end{cases}$$

With this loss function, the risk $r(x_t) = \sum_{x \in \mathcal{X}_t} \lambda(x_t, x) P(X_t = x \mid X_{<t} = x_{<t}) = 1 - P(X_t = x_t \mid X_{<t} = x_{<t})$ of selecting x_t equals the average probability of error. It is minimized by choosing an x_t for which

$$P\left(X_t = x_t \mid X_{<t} = x_{<t}\right)$$

is maximized. Observe that other measures of loss yield other decision rules.

As a uniform means to obtain suitable dynamic layout models, we propose a Bayesian approach which basically provides a formalization of common sense. Note that, by Bayes' rule,

$$\begin{aligned} &\max_{x_t \in \mathcal{X}_t} P\left(X_t = x_t \mid X_{<t} = x_{<t}\right) \\ &\quad = \max_{x_t \in \mathcal{X}_t} \frac{P(X_{<t} = x_{<t} \mid X_t = x_t) \cdot P(X_t = x_t)}{P(X_{<t} = x_{<t})} \\ &\quad \propto \max_{x_t \in \mathcal{X}_t} P\left(X_{<t} = x_{<t} \mid X_t = x_t\right) \cdot P\left(X_t = x_t\right), \end{aligned}$$

where $\propto$ means "proportional to". $P(X_t = x_t)$ is easily recognized to be the static layout model for G_t. It therefore reflects the notion of readability formalized in

[2] Observe that there are applications, like animation, where the complete sequence is indeed known in advance.

X_t, and is called the *static model.* $P(X_{<t} = x_{<t} \mid X_t = x_t)$ is the *likelihood* of a sequence $x_1, \ldots, x_{t-1}$ to result in a (given) layout x_t of G_t. Consequently, it should express our notion of stability, and therefore is called the *stability model.*

In summary, we have argued how dynamic layout models $P(X_t = x_t \mid X_{<t} = x_{<t})$ can be derived from a sequence of static layout models $P(X_t = x_t)$ by introducing stability models $P(X_{<t} = x_{<t} \mid X_t = x_t)$. By the very nature of a random field formulation, joint maximization of these two components results in a compromise between those criteria describing readability and those describing stability of layouts. We have thus developed an abstract formulation of a general principle for dynamic graph layout that incorporates given static models.

The following two sections are devoted to examples providing evidence that this formulation results in a general method for uniform integration of stability criteria in a sequence of initially static layout problems.

4 Dynamic Spring Layout

The *spring embedder* [Ead84] is one of the most well-known layout models for straight-line representations. A graph is modelled by a physical system of rings corresponding to the vertices, and springs corresponding to the edges. Rings of adjacent vertices are joined by springs, whereas rings of non-adjacent vertices are repelling. The spring embedder then aims to produce a layout that corresponds to a stable configuration of the system.

The layout elements of a random field formulation of a spring embedder for a graph $G = (V, E)$ simply are the vertices, i.e. $L = V$. Sets of admissible states $\mathcal{X}_v$, $v \in V$, are locations in two or three-dimensional space. In this section, we use attractive and repelling forces between vertices as introduced in the spring embedder variant of [DH96]. Each pair of vertices $u \neq v \in V$ is assigned an interaction potential

$$U_{\{u,v\}}(x) = \begin{cases} \frac{c_1}{d(x_u,x_v)^2} + c_2 \cdot d(x_u, x_v)^2 & \text{if } \{u,v\} \in E \\ \frac{c_1}{d(x_u,x_v)^2} & \text{otherwise} \end{cases}$$

such that the static layout model for G is

$$P(X = x) = \frac{1}{Z} e^{-\sum\limits_{u \neq v \in V} U_{\{u,v\}}(x)}$$

with constant parameters $c_1, c_2 > 0$ and normalizing constant Z. $d(x_u, x_v)$ denotes the Euclidean distance of locations x_u and x_v. While c_1 controls the strength of repelling forces between each pair of vertices, c_2 can be used to define the ideal edge length.[3]

The above spring model is now extended to dynamic graphs. For stability, we use notions of change in vertex locations ("anchoring"), relative locations ("stiffening"), and an accumulated version of the latter. These result in three

[3] Potentials $\frac{c_1}{d(x_u,x_v)^2} + c_2 \cdot d(x_u, x_v)^2$ are minimized for $d(x_u, x_v) = \sqrt[4]{\frac{c_1}{c_2}}$.

different dynamic models for G_t. In each case we obtain a new random field model that compromises between readability and stability as proposed in the preceding section.

Anchoring. Let us assume stability is demanded only with respect to consecutive layouts. To simplify notation, we use a number of abbreviations. Let $X = X_t$ be the static spring model for G_t, and $Y = X_{t-1}$ be the static spring model for G_{t-1}, $t > 1$. Furthermore, let V_X, V_Y, and $V_{X,Y}$ denote the vertices of G_t, the vertices of G_{t-1}, and their intersection, respectively. U^X and U^Y denote the energy function with respect to the edge set of G_t and G_{t-1}, respectively, and so on. According to the above dependency assumption, the dynamic model satisfies the Markov property

$$\begin{aligned} P\left(X_t = x_t \mid X_{<t} = x_{<t}\right) &= P\left(X_t = x_t \mid X_{t-1} = x_{t-1}\right) \\ &= P\left(X = x \mid Y = y\right), \end{aligned}$$

such that our formal expression of stability reduces to $P(Y = y \mid X = x)$. Since a straight-line embedding is completely determined by locations assigned to vertices, a very natural criterion of stability is the absence of excessive movement of vertices between consecutive layouts. More formally, let the likelihood of $y \in \mathcal{Y}$ leading to $x \in \mathcal{X}$ be measured by independent two-dimensional Gaussian distributions with mean in the conditioning location x_v of each $v \in V_{X,Y}$:

$$P\left(Y = y \mid X = x\right) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\sum_{v \in V_{X,Y}} \frac{\|y_v - x_v\|^2}{2\sigma^2}}$$

where σ is a constant controlling the amplitude of deviation. Since this distribution is symmetric in x and y, the formula can also be read in a more intuitive way: The new location x_v is distributed normally around the current location y_v. Since $\|y_v - x_v\|^2 = d(x_v, y_v)^2$, there is an obvious correspondence to the attracting forces of [DH96].

Combining the models for stability and readability in a dynamic model for G_t yields

$$P\left(X = x \mid Y = y\right) = \frac{1}{Z_{X|Y}} \exp\left\{-\sum_{v \in V_{X,Y}} \frac{d(y_v, x_v)^2}{2\sigma^2} - \sum_{u \neq v \in V_X} U^X_{\{u,v\}}(x)\right\}$$

with the appropriate normalizing constant $Z_{X|Y}$. Clearly, this is also a random field, and even correspondent to a force directed placement model, yet with additional forces attracting vertices to their previous locations in y. It is interesting to note that by choosing different functions for attracting and repelling forces one obtains the *anchored spring model* introduced by Lyons in the context of (static) graphs with geographic semantics (preferred locations for vertices) [Lyo92].

Stiffening. Assume now, the criterion of stability between two consecutive layouts y and x is relative instead of absolute location, or *structure* instead of *position.* In other words, consecutive layouts should retain pairwise distances rather than single vertex locations.

Let $x \in \mathcal{X}$ be the conditioning layout of the stability model. We consider a layout y of G_{t-1} likely to lead to layout $x \in \mathcal{X}$, if the forces of X are already apparent in y. Therefore, the likelihood of y, given x, is set to

$$P\left(Y = y \mid X = x\right) = \frac{1}{Z_{Y|X}} \exp\left\{ - \sum_{u \neq v \in V_Y} U^Y_{\{u,v\}}(y) - \sum_{\substack{u \neq v: \\ v \in V_{X,Y}, v \in V_Y}} U^Y_{\{u,v\}}(y^{\hat{x}}) \right\}$$

where $\hat{x} = x_{V_{X,Y}}$ and $Z_{Y|X}$ is a normalizing constant. In terms of forces, the stability model states that each vertex contained in both G_{t-1} and G_t contributes twice, once from its location in y, and once from its subsequent location in x. That is, y is a layout with shadow forces of vertices that remain in the graph, excerted from their location in the next layout. Combined with the static layout model X, this notion of stability yields the dynamic model

$$\begin{aligned}
&P\left(X = x \mid Y = y\right) \\
&\quad \propto P\left(Y = y \mid X = x\right) \cdot P\left(X = x\right) \\
&\quad = \tfrac{1}{Z_{Y|X}} \exp\left\{ - \sum_{u \neq v \in V_Y} U^Y_{\{u,v\}}(y) - \sum_{\substack{u \neq v: \\ u \in V_{X,Y}, v \in V_Y}} U^Y_{\{u,v\}}(y^{\hat{x}}) \right\} \\
&\qquad \cdot \tfrac{1}{Z_X} \exp\left\{ - \sum_{u \neq v \in V_X} U^X_{\{u,v\}}(x) \right\} \\
&\quad = \tfrac{1}{Z_{X|Y}} \exp\left\{ - \sum_{\substack{u \neq v: \\ u \in V_{X,Y}, v \in V_Y}} U^Y_{\{u,v\}}(y^{\hat{x}}) - \sum_{u \neq v \in V_X} U^X_{\{u,v\}}(x) \right\}
\end{aligned}$$

with the obvious normalizing constants and $\hat{x} = x_{V_{X,Y}}$. For the last equality, note that the energy function of a random field is unique up to an additive constant. Just like the anchored dynamic model, the above represents a force directed placement model. First note that vertices $u \in V_X$ contribute with their location in $x \in \mathcal{X}$, while vertices $v \in V_Y \setminus V_X$ contribute with their location in $y \in \mathcal{Y}$. Each pair of vertices $u, v \in V_{X,Y}$ contributes twice to the energy of the dynamic model, once according to $U^Y_{\{u,v\}}$, and once according to $U^X_{\{u,v\}}$. Using the spring analogy, the spring connecting u and v is stiffened, i.e. its length remains unchanged, while its strength is increased. Pairs of vertices with exactly one vertex in $V_{X,Y}$ contribute to the energy in the usual way, such that the layout remembers deleted vertices. Unchanged parts of the graph are connected by a stiffer structure than new or altered ones.

Cumulative stiffening. The stiffening stability model is extended to long range dependencies quite easily. Without going into details, we note that pairwise interactions may be accumulated over the sequence of graphs. The longer a relation existed, the less should its displacement be likely to change. Evaluation of the resulting energy function is not quite as complex as it may seem, since time stamps for vertices and edges may be used to store the multiplicity of their contribution.

Most algorithms in force directed placement are designed to find a local minimum of the aesthetic cost function, and hence of the energy function in their corresponding random field model. In dynamic graph layout this is no drawback, but an advantage, since locations in y are suitable initial values for locations of vertices $v \in V_{X,Y}$. If consecutive graphs do not differ significantly, these algorithms should therefore be able to quickly find a satisfactory local optimum close to the initial layout.

5 Dynamic Orthogonal Layout

In this section, the Bayesian paradigm is applied to dynamic orthogonal drawings of 4-planar graphs with fixed planar embbeding. A planar graph is called 4-planar, if no vertex has degree larger than 4, and an embedding is given by a cyclic ordering of the edges incident to each vertex. In an orthogonal drawing, vertices are placed on grid points, while edges are drawn along the grid lines. Edges may overlap at grid points only.

Tamassia describes an algorithm minimizing the number of bends among all planar orthogonal drawings that preserve the embedding [Tam87]. We review briefly his transformation to a minimum cost flow problem, and state it in terms of a random field model. The static model is then combined with a reasonable criterion for stability. Finally, the resulting dynamic model is re-stated in terms of a minimum cost flow problem.

Orthogonal representation. Given a planar graph $G = (V, E)$ with maximum vertex degree 4 and a planar embedding, the drawing is determined from an *orthogonal representation* $H(G)$ of circular lists $H_f = [(e_1, s_1, a_1), \ldots, (e_r, s_r, a_r)]$ for each face f of G. Each tupel (e_i, s_i, a_i) of a list H_f consists of an edge e_i, a string $s_i \in \{0,1\}^*$, and an integer $a_i \in \{90, 180, 270, 360\}$, such that $e_1, \ldots, e_r$ is a counterclockwise (clockwise, if f is the outer face) traversal of the edges incident to f (note that some edges may appear twice in this traversal), 0's and 1's in s_i represent 90 and 270 degree bends on the right side of e_i, respectively, and a_i is the angle between e_i and its succeeding edge (H_f is cyclic).

Flow network. In [Tam87], a one-to-one correspondence of $H(G)$ to a flow of specified value in some network $N(G) = (U, A, s, t, cap, cost)$ is shown.[4] The vertex set U of $N(G)$ consists of the vertices V of G, an additional vertex for each face of G, and new source and target vertices s and t, respectively. Let $d_G(v)$ denote the number of edges incident to vertex v, and $d_G(f)$ denote the number of edges in the circular list of face f. Arcs are introduced

[4] It should always be clear from context, whether t is used to denote an index, or the target vertex of a network, respectively.

– from s to each $v \in V$ (capacity $4 - d_G(v)$ and zero cost),
– from each vertex to its incident faces (infinite capacity, zero cost),
– from s to each internal face with less than 4 edges (capacity $4 - d_G(f)$, zero cost),
– from each face to its neighboring faces (infinite capacity, unit cost),
– from a face to itself, if some edge appears twice in the circular list (infinite capacity, unit cost),
– from each internal face with more than 4 boundary edges to t (capacity $d_G(f) - 4$, zero cost),
– and from the external face to t (capacity $d_G(f) + 4$, zero cost).

Then, each unit of flow represents an angle of 90 degrees, such that the cost of an (s,t)–flow of value $\sum_{(s,u)\in A} cap(s,u)$ equals the number of bends in the corresponding layout.

Random field model. Since a drawing is determined from an orthogonal representation which is obtained from a minimum cost flow in $N(G)$, the layout elements of the above representation are the arcs of $N(G)$, i.e. $L = A$. Admissible values for $a \in L$ are the integers $x_a \in \mathcal{X}_a = \{0, \ldots, cap(a)\}$ satisfying the capacity constraint. Since the number of bends is the only criterion of readability, the energy of a layout equals its total cost

$$U(x) = \sum_{a\in A} cost(a) \cdot x_a .$$

Note that the random field model does not incorporate flow constraints. It is therefore a relaxation of the original formulation. The relaxation is used to derive a relaxed dynamic model, on which the flow constraints are then imposed again.

Dynamic model. To simplify presentation, we make a few reasonable assumptions on the dynamic layout problem. Let the layout of G_t be independent of $x_1, \ldots, x_{t-2}$. Moreover, let G_{t-1} and G_t be connected, where G_t is obtained from G_{t-1} by insertion or deletion of a single edge. Since both graphs are connected, insertion or deletion of an edge may also require insertion or deletion of one vertex, respectively.

Now, we are given networks $N(G_{t-1}) = (U_{t-1}, A_{t-1}, s, t, cap_{t-1}, cost)$ and $N(G_t) = (U_t, A_t, s, t, cap_t, cost)$, and a layout x_{t-1} corresponding to a flow of value $\sum_{(s,u)\in A_{t-1}} cap_{t-1}(s,u)$ in $N(G_{t-1})$. Since the criterion of readability is the number of bends, the most natural criterion for stability between consecutive layouts is the difference in the number of bends on edges present in both layouts,

$$\sum_{a\in A_t\cap A_{t-1}} cost(a) \cdot |(x_{t-1})_a - (x_t)_a| .$$

This criterion is taken to be the energy function of a random field model for stability. The dynamic model becomes

$$
\begin{aligned}
&P\left(X_t = x_t \mid X_{t-1} = x_{t-1}\right) \\
&\quad \propto P\left(X_{t-1} = x_{t-1} \mid X_t = x_t\right) \cdot P\left(X_t = x_t\right) \\
&\quad = \tfrac{1}{Z_{t-1|t}} \exp\left\{ - \sum_{a \in A_t \cap A_{t-1}} cost(a) \cdot \left|(x_{t-1})_a - (x_t)_a\right| \right\} \\
&\qquad \cdot \tfrac{1}{Z_t} \exp\left\{ - \sum_{a \in A_t} cost(a) \cdot (x_t)_a \right\} \\
&\quad = \tfrac{1}{Z_{t|t-1}} \exp\left\{ - \sum_{a \in A_t \cap A_{t-1}} cost(a) \cdot \left|(x_{t-1})_a - (x_t)_a\right| - \sum_{a \in A_t} cost(a) \cdot (x_t)_a \right\},
\end{aligned}
$$

such that reducing the number of bends of a remaining edge is just as costly as bends on a new edge, whereas new bends on remaining edges are counted twice.

Implementation. What is probably most appealing about this model is that it can be re-translated into a minimum cost flow problem. Due to space limitations we do not go into details, but note that inserting or deleting an edge in G_{t-1} results in a graph G_t with an associated network $N(G_t)$ that differs from $N(G_{t-1})$ by a small number of vertices and arcs. Let x_{t-1} be a feasible flow of $N(G_{t-1})$ with value $\sum_{(s,u) \in A_{t-1}} cap_{t-1}(s,u)$. Moreover, let x_{t-1} have the property that, for each $(u,v) \in A_t \cap A_{t-1}$, at most one of (u,v) and (v,u) does carry positive flow (which is certainly true for a minimum cost flow). Then, minimization of the energy function of the dynamic model can be performed by solving a new minimum cost flow problem in a modified network $N'(G_t)$. For each arc $(u,v) \in A_t$, for which $(v,u) \notin A_t$, we insert a new arc (v,u) with zero capacity and $cost(v,u) = cost(u,v)$. Now, costs and capacities are modified according to x_{t-1}, where $(x_{t-1})_{(v,u)}$ equals 0, if $(v,u) \notin A_{t-1}$. For each arc (u,v), $\{u,v\} \cap \{s,t\} = \emptyset$, in the symmetric hull of $A_t \cap A_{t-1}$, let $cap'_t(u,v) = cap_t(u,v) - (x_{t-1})_{(u,v)} + (x_{t-1})_{(v,u)}$. Furthermore, let $cost'(u,v) = cost(u,v) + cost(v,u)$. Because of the flow network's special structure, these changes reflect the additional cost of changing flow x_{t-1}. A small number of minimum cost augmenting flow computations in the modified network yields a flow that can be used to transform x_{t-1} into a feasible flow x_t of $N(G_t)$ that is optimal with respect to the dynamic model. Moreover, x_t also satisfies the above property that there are no circulations of length two. It can hence be used in a dynamic model for G_{t+1}. Details will be given in a more general presentation of a Bayesian approach for dynamic orthogonal layouts with few bends, which is in preparation.

6 Discussion

We have presented a general principle for dynamic graph layout that exploits the uniformity of random field modelling. The logical update of a layout is formalized by means of a stochastic estimator composed of the static layout model and an additional stability model. If the stability model is also a random field, it is easy to see that It is possible to bias the inherent compromise between stability and readability by multiplication of a constant factor to the respective energy functions.

Our approach is neither specific to a certain layout model, nor to a set of update operations. Its underlying formalism is based on common sense, yet sound, mature and widely applied. In particular, it cleanly separates the modelling of readability and stability demands. Moreover, it does not imply the use of particular algorithms, even though, in general, algorithms used for static models require only slight modification to apply to the dynamic case as well. Moreover, the flexibility of random fields allows the easy experimentation with different stability models within the same environment. There is no need to implement a new algorithm every time the model is modified. It is still recommended, though, after a suitable model has been identified, since a general optimization procedure is almost certainly inefficient.

For particular layout models, experimental results are needed to identify suitable stability models. It will be interesting to see, how different criteria compare (and perform) in practice. However, the formal value of the framework lies in its generality, simplicity and uniformity. It should not be judged by some good or bad usage.

References

[BP90] Karl-Friedrich Böhringer and Frances Newbery Paulisch. Using constraints to achieve stability in automatic graph layout algorithms. In *CHI'90 Proceedings*, pages 43–51. ACM, The Association for Computing Machinery, New York, 1990.

[BW97] Ulrik Brandes and Dorothea Wagner. Random field models for graph layout. Konstanzer Schriften in Mathematik und Informatik 33, Universität Konstanz, 1997.

[CDT+92] Robert F. Cohen, Giuseppe Di Battista, Roberto Tamassia, Ioannis G. Tollis, and P. Bertolazzi. A Framework for Dynamic Graph Drawing. In *Proc. of 8th Annual Computational Geometry, 6/92, Berlin, Germany*, pages 261–270. ACM, The Association for Computing Machinery, New York, 1992.

[CDTT95] Robert F. Cohen, Giuseppe Di Battista, Roberto Tamassia, and Ioannis G. Tollis. Dynamic graph drawings: Trees, series-parallel digraphs, and planar *st*-digraphs. *SIAM J. Comput.*, 24(5):970–1001, 1995.

[CJ93] R. Chellappa and A.K. Jain. *Markov Random Fields: Theory and Applications*. Academic Press, 1993.

[DH96] Ron Davidson and David Harel. Drawing graphs nicely using simulated annealing. *ACM Transactions on Graphics*, 15(4):301–331, 1996.

[Ead84] Peter Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.

[ELMS91] Peter Eades, W. Lai, Kazuo Misue, and Kozo Sugiyama. Preserving the mental map of a diagram. *Proceedings of Compugraphics*, 9:24–33, 1991.

[GG84] Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[GM85] Stuart Geman and Donald E. McClure. Bayesian image analysis: an application to single photon emission tomography. *Proc. American Statistical Association, Statistical Computing Section*, pages 12–18, 1985.

[Guy94] Xavier Guyon. *Random Fields on a Network*. Springer, 1994.

[Lyo92] Kelly A. Lyons. Cluster busting in anchored graph drawing. In *Proceedings of the '92 CAS Conference/CASCON'92, Toronto, 1992*, pages 7–17, 1992.

[Nor96] Stephen North. Incremental Layout in DynaDAG. *Proceedings of GD'95*, pages 409–418. Springer-Verlag, Lecture Notes in Computer Science, vol. 1027, 1996.

[PT96] Achilleas Papakostas and Ioannis G. Tollis. Issues in Interactive Orthogonal Graph Drawing. *Proceedings of GD'95*, pages 419–430. Springer-Verlag, Lecture Notes in Computer Science, vol. 1027, 1996.

[Tam87] Roberto Tamassia. On embedding a graph in the grid with the minimum number of bends. *SIAM J. Comput.*, 16(3):421–444, 1987.

[Win95] Gerhard Winkler. *Image Analysis, Random Fields and Dynamic Monte Carlo Methods*, volume 27 of *Applications of Mathematics*. Springer, 1995.