

An Energy Model for Visual Graph Clustering

Andreas Noack

Institute of Computer Science
Brandenburg Technical University at Cottbus
PO Box 10 13 44, 03013 Cottbus, Germany
an@informatik.tu-cottbus.de

Abstract. We introduce an energy model whose minimum energy drawings reveal the clusters of the drawn graph. Here a cluster is a set of nodes with many internal edges and few edges to nodes outside the set. The drawings of the best-known force and energy models do not clearly show clusters for graphs whose diameter is small relative to the number of nodes. We formally characterize the minimum energy drawings of our energy model. This characterization shows in what sense the drawings separate clusters, and how the distance of separated clusters to the other nodes can be interpreted.

1 Introduction

Force-directed and energy-based methods are popular for creating straight-line drawings of undirected graphs. They are comparatively easy to implement, adaptable to different drawing criteria, and give satisfactory results for many graphs ([4, Chap. 10], [7]). Since the introduction of multi-scale algorithms they are even quite efficient [28,15,17,26,29].

Energy-based methods generally have two parts: an energy model, and an algorithm that searches a state with minimum total energy. In force-directed methods, the model is a force system, and an algorithm searches for an equilibrium state where the total force on each node is zero. Because force is the negative gradient of energy, this corresponds to searching a local minimum of energy. We will take the perspective of energy minimization in this paper.

Finding clusters, i.e. subsets of nodes with many internal edges and few edges to outside nodes, in graphs is an important problem in VLSI design [2], parallel computing [25], software engineering [23], and graph drawing [9]. The most popular force and energy models do not clearly isolate clusters, especially in graphs with small diameter. The first main result of this paper is an energy model whose minimum energy drawings reveal the clusters of such graphs.

The purpose of energy models is to create drawings from which a human viewer can infer properties of the drawn graph. To make the inference more efficient and its results more valid, the drawings are required to have certain properties, like small and uniform edge lengths, well-distributed nodes, or well-separated clusters. Empirical studies were performed to evaluate to what degree force and energy models fulfill such criteria (e.g. [6]), but few analytical results exist. A formal characterization of the minimum energy drawings of our energy model is the second main result of this paper.

The new energy model and its formal analysis are presented in Sect. 2. Section 3 shows example drawings and compares them with drawings of the well-known Fruchterman-Reingold force model [14]. Section 4 compares our energy model with existing force and energy models, and proposes a generalization that includes among others our energy model and the Fruchterman-Reingold model. It discusses applications, the importance of interpretability, and the benefits of theoretical analyses of energy models.

1.1 Basic Definitions

A graph $G = (V, E)$ consists of a finite set V of *nodes* and a finite set E of *edges* with $E \subseteq V^{(2)}$, where $V^{(2)}$ is the set of all subsets of V which have exactly two elements. We only consider graphs with at least two nodes. Because layouts can be computed separately for different components of a graph, we restrict ourselves to connected graphs, i.e. graphs where every pair of nodes is connected by a path.

For two nodes $u, v \in V$, the length of the shortest path connecting u and v is the *graph-theoretic distance* of u and v . The *diameter* of the graph G is the greatest graph-theoretic distance between any two nodes. A *cut* of G is a pair (V_1, V_2) of nonempty, disjoint sets of nodes with $V_1 \cup V_2 = V$.

For $V_1, V_2 \subseteq V$ and $F \subseteq V^{(2)}$ let $F[V_1] = \{\{u, v\} \in F \mid u, v \in V_1\}$ and $F[V_1, V_2] = \{\{u, v\} \in F \mid u \in V_1, v \in V_2\}$. Then $E[V_1]$ is the set of edges *in* V_1 , and $E[V_1, V_2]$ is the set of edges *between* V_1 and V_2 .

A *d-dimensional drawing* of the graph G is a vector $p = (p_v)_{v \in V}$ of node positions $p_v \in \mathbb{R}^d$. For a drawing p and two nodes $u, v \in V$ the length of the difference vector $p_v - p_u$ is called the (*Euclidean*) *distance* of u and v in p and denoted by $\|p_v - p_u\|$.

For a subset F of $V^{(2)}$ and a drawing p , the arithmetic mean $\text{arithmean}(F, p)$ of the distances of F is defined as

$$\text{arithmean}(F, p) = \frac{1}{|F|} \sum_{\{u,v\} \in F} \|p_v - p_u\|,$$

the geometric mean $\text{geommean}(F, p)$ of the distances of F is defined as

$$\text{geommean}(F, p) = \sqrt[|F|]{\prod_{\{u,v\} \in F} \|p_v - p_u\|},$$

and the harmonic mean $\text{harmmean}(F, p)$ of the distances of F is defined as

$$\text{harmmean}(F, p) = \frac{|F|}{\sum_{\{u,v\} \in F} \frac{1}{\|p_v - p_u\|}}.$$

2 LinLog: A Clustering Energy Model

In this section, we first define a new energy model, called LinLog, and the cut ratio as measure of the coupling of two disjoint sets of nodes. Using this measure we can formalize and prove that in minimum energy drawings of the LinLog model, clusters are clearly separated from the remaining graph, and the distance of each cluster to the remaining graph is interpretable with respect to properties of the graph (more precisely, the distance is approximately inversely proportional to the coupling).

2.1 The LinLog Energy Model

The *LinLog energy* $U_{LinLog}(p)$ of a drawing p is defined as

$$U_{LinLog}(p) = \sum_{\{u,v\} \in E} \|p_u - p_v\| - \sum_{\{u,v\} \in V^{(2)}} \ln \|p_u - p_v\|$$

The first term of the difference can be interpreted as attraction between adjacent nodes, and the second term can be interpreted as repulsion between any two (different) nodes. To avoid infinite energies we assume that different nodes have different positions. This is no serious restriction because we are interested in good drawings, i.e. drawings with low energy.

2.2 The Cut Ratio as a Measure of Coupling

Many different definitions of the term *cluster of a graph* have been proposed. Informally, we denote by cluster a set of nodes with many internal edges (high cohesion) and few edges to nodes outside the set (low coupling). Similar definitions are used in VLSI design [2], parallel computing [25], and software engineering [23].

To make more precise what we mean by a cluster, we use a formally defined measure of coupling: the cut ratio. For a cut (V_1, V_2) , the *cut ratio* is the fraction of possible edges between V_1 and V_2 which actually occur:

$$\text{cutratio}(V_1, V_2) = \frac{|E[V_1, V_2]|}{|V_1| \cdot |V_2|}$$

The problem of finding a cut with minimum cut ratio is known as the sparsest cut problem. Section 4 shortly discusses work related to this problem.

The cut ratio is normalized with respect to the number $|V_1| \cdot |V_2|$ of possible edges between V_1 and V_2 , which makes its interpretation independent of the size of V_1 and V_2 . For all cuts (V_1, V_2) of a random graph the expected value of the cut ratio is equal, while the expected number of edges crossing the cut strongly depends on the size of V_1 and V_2 .

2.3 LinLog Separates Clusters

In this subsection we show that in minimum energy drawings of the LinLog model, clusters are clearly separated from the remaining nodes, and the nodes of each cluster are close to each other. More precisely: If a set of nodes V_1 is loosely coupled to the remaining nodes $V \setminus V_1$, then the distance of V_1 to the remaining nodes is large. If V_1 is strongly coupled to $V \setminus V_1$, then V_1 is close to the nodes in $V \setminus V_1$, in particular to its adjacent nodes. This is achieved by *minimizing the ratio of the average edge length to the average distance of (all pairs of different) nodes*.

Let us illustrate this with two examples. First, consider a drawing where V_1 is close to the remaining nodes, although it is only loosely coupled to them. Loose coupling means that there are few edges and many non-edges between V_1 and the remaining nodes. Thus increasing the distance of V_1 to the remaining nodes will increase the average distance of nodes much more than the average edge length. In the opposite case, V_1 is

strongly coupled to the remaining nodes, but is drawn at a great distance from them. Then decreasing the distance, especially to the nodes to which V_1 is adjacent, will make relatively many edges and few non-edges shorter. Thus the average edge length will decrease more than the average distance of nodes.

The following theorem states that *drawings with minimal LinLog energy have a minimal ratio of the arithmetic mean of the edge lengths to the geometric mean of the node distances*. So we formalize the average node distance as the geometric mean of the node distances. The reason is our intuition that increasing the distance of two non-adjacent nodes from 10 to 11 is not as good as increasing the distance from 1 to 2. Rather, an increase from 1 to 2 (by a factor of 2) is about as good as an increase from 10 to 20 (also by a factor of 2). This also captures that different nodes should not have the same position. Thus we preferred the geometric mean instead of the arithmetic mean, but we do not claim that this is the only possible choice.

Theorem 1. *Let $G = (V, E)$ be a connected graph, and let p^0 be a drawing of G with minimum LinLog energy. Then p^0 is a drawing of G that minimizes $\frac{\text{arithmean}(E, p)}{\text{geommean}(V^{(2)}, p)}$.*

Proof: The basic idea is to fix the average edge length temporarily. This does not restrict generality, but only the scaling factor, and thus can be removed at the end of the proof. It allows to transform the minimization of energy into a minimization of the inverse geometric mean of the node distances.

Let the drawing p^0 be a solution of the minimization problem:

$$\text{Minimize } U_{LinLog}(p).$$

Let $c = \sum_{\{u,v\} \in E} \|p_u^0 - p_v^0\|$. Note that $c \geq 0$. Then p^0 is also a solution of

$$\text{Minimize } U_{LinLog}(p) \text{ subject to } \sum_{\{u,v\} \in E} \|p_u - p_v\| = c.$$

This is equivalent to

$$\text{Minimize } - \sum_{\{u,v\} \in V^{(2)}} \ln \|p_u - p_v\| \text{ subject to } \sum_{\{u,v\} \in E} \|p_u - p_v\| = c.$$

Because $f(x) = |v^{(2)}| \sqrt{\exp(x)}$ is a monotonic increasing function, p^0 is a solution of

$$\text{Min. } |v^{(2)}| \sqrt{\exp\left(-\sum_{\{u,v\} \in V^{(2)}} \ln \|p_u - p_v\|\right)} \text{ subj. to } \sum_{\{u,v\} \in E} \|p_u - p_v\| = c.$$

This is equivalent to

$$\text{Minimize } \frac{1}{\text{geommean}(V^{(2)}, p)} \text{ subject to } \text{arithmean}(E, p) = \frac{c}{|E|}.$$

($|E| > 0$ because we only consider connected graphs with at least two nodes.) Because $\frac{c}{|E|}$ is nonnegative, p^0 is also a solution of

$$\text{Minimize } \frac{\text{arithmean}(E, p)}{\text{geommean}(V^{(2)}, p)} \text{ subject to } \text{arithmean}(E, p) = \frac{c}{|E|}.$$

For every drawing q^0 of G that minimizes $\frac{\text{arithmetic}(E,p)}{\text{geomean}(V^{(2)},p)}$, we can construct a drawing $q^1 = \frac{c}{|E| \text{arithmetic}(E,q^0)} q^0$ with $\frac{\text{arithmetic}(E,q^1)}{\text{geomean}(V^{(2)},q^1)} = \frac{\text{arithmetic}(E,q^0)}{\text{geomean}(V^{(2)},q^0)}$ and $\text{arithmetic}(E,q^1) = \frac{c}{|E|}$. (Because $\text{geomean}(V^{(2)},q^0) > 0$, no two different nodes have the same position in q^0 , so $\text{arithmetic}(E,q^0) > 0$.) So p^0 is also a solution of

$$\text{Minimize } \frac{\text{arithmetic}(E,p)}{\text{geomean}(V^{(2)},p)}. \quad \square$$

This characterization of drawings with minimum LinLog energy is precise up to scaling. To show this, we use a lemma which states that the overall edge length in all minimum LinLog energy drawings is $\frac{(|V|-1) \cdot |V|}{2}$. So the overall edge length is independent from the number of edges and the structure of the graph. This lemma has a practical value in itself, because it helps to choose the scaling factor: For a drawing with an average edge length of 1, scale the minimum LinLog energy drawing by a factor of $\frac{2|E|}{(|V|-1) \cdot |V|}$.

Lemma 1. *Let $G = (V, E)$ be a connected graph, and let p be a drawing of G with minimum LinLog energy. Then*

$$\sum_{\{u,v\} \in E} \|p_u - p_v\| = |V^{(2)}|$$

Proof: The basic idea is to express the LinLog energy as a function of the scaling factor, where the scaling is applied to a minimum LinLog energy drawing. Then this function has a minimum at the minimum energy drawing, i.e. at a scaling factor of 1. The details of the proof are given in [24]. □

Theorem 2. *Let $G = (V, E)$ be a connected graph. Then p^0 is a drawing of G with minimum LinLog energy if and only if p^0 is a drawing that minimizes $\frac{\text{arithmetic}(E,p)}{\text{geomean}(V^{(2)},p)}$ and satisfies $\sum_{\{u,v\} \in E} \|p_u^0 - p_v^0\| = |V^{(2)}|$.*

Proof: \Rightarrow : Follows from Theorem 1 and Lemma 1.

\Leftarrow : Similar to the proof of Theorem 1. The details are given in [24]. □

2.4 LinLog Creates Interpretable Distances

In this subsection we show that in minimum energy drawings of the LinLog model, the distance of each cluster to the remaining nodes is interpretable. More precisely, the distance approximates the inverse coupling of the cluster to remaining nodes.

The following theorem states that *wherever one cuts a one-dimensional minimum LinLog energy drawing into two nonempty parts, the harmonic mean of the distances between the nodes in the two parts equals the inverse cut ratio*. So we formalized the distance of two sets of nodes as the harmonic mean of the distances of their members. The harmonic mean corresponds better to our intuition than the arithmetic or geometric mean because it weights small distances higher than large distances. If the distances within the two sets of nodes are much smaller than the distances between the two sets, then the harmonic, the geometric and the arithmetic mean are roughly equal.

Theorem 3. *Let $G = (V, E)$ be a connected graph, and let p be a one-dimensional drawing of G with minimal LinLog energy. Let (V_1, V_2) be a cut of G such that the nodes in V_1 have smaller positions than the nodes in V_2 (i.e. $\forall v_1 \in V_1 \forall v_2 \in V_2 : p_{v_1} < p_{v_2}$). Then*

$$\text{harmmean}(V^{(2)}[V_1, V_2], p) = \frac{1}{\text{cutratio}(V_1, V_2)}.$$

Proof: The basic idea is to express the LinLog energy as a function of the relative positions of the two sets of nodes, and to exploit that the minimum energy drawing p is a minimum of this function.

The LinLog energy of the drawing p is:

$$\sum_{\{u,v\} \in E} |p_u - p_v| - \sum_{\{u,v\} \in V^{(2)}} \ln |p_u - p_v|$$

(To simplify the notation, we consider the positions as scalars here.) If some $d \in \mathbb{R}$ is added to the coordinates of all nodes in V_1 , such that the condition that the nodes of V_1 have smaller coordinates than the nodes of V_2 is still satisfied (i.e. $d < \min\{p_v \mid v \in V_2\} - \max\{p_v \mid v \in V_1\}$), the LinLog energy is

$$\begin{aligned} U(d) &= \sum_{\{u,v\} \in E[V_1] \cup E[V_2]} |p_u - p_v| - \sum_{\{u,v\} \in V_1^{(2)} \cup V_2^{(2)}} \ln |p_u - p_v| \\ &+ \sum_{\{u,v\} \in E[V_1, V_2]} (|p_u - p_v| + d) - \sum_{\{u,v\} \in V^{(2)}[V_1, V_2]} \ln(|p_u - p_v| + d) \end{aligned}$$

Because p is a drawing with minimal LinLog energy, this function has a global minimum at $d = 0$, so $U'(0) = 0$.

$$0 = U'(0) = |E[V_1, V_2]| - \sum_{\{u,v\} \in V^{(2)}[V_1, V_2]} \frac{1}{|p_u - p_v|}$$

Inserting the harmonic mean $\text{harmmean}(V^{(2)}[V_1, V_2], p)$ of the distances between V_1 and V_2 we get

$$\begin{aligned} 0 &= |E[V_1, V_2]| - \frac{|V_1| \cdot |V_2|}{\text{harmmean}(V^{(2)}[V_1, V_2], p)} \\ \text{harmmean}(V^{(2)}[V_1, V_2], p) &= \frac{|V_1| \cdot |V_2|}{|E[V_1, V_2]|} = \frac{1}{\text{cutratio}(V_1, V_2)} \quad \square \end{aligned}$$

The generalization of the theorem to higher-dimensional drawings is: *For each hyperplane that cuts a minimum LinLog energy drawing into two nonempty parts, the harmonic mean of the distances between the corresponding sets of nodes equals their inverse cut ratio.* But it is not true. The reason is, loosely stated, that in two or more dimensions the directions of the edges between the two sets of nodes can be different from the directions of the non-edges. So when one set is moved, the average change of the edge lengths is different from the average change of the non-edge lengths. However, this difference is small in most practical cases. In particular, the one-dimensional case is a good approximation when the distance of a cluster to the remaining nodes is large.

3 Examples

Figures 1, 2 and 3 show drawings of pseudo-random graphs with clusters for the LinLog energy model and the well-known Fruchterman-Reingold force model ([14], further discussed in Sect. 4). In all figures, the LinLog model (left drawings) reveals the clusters more clearly than the Fruchterman-Reingold model (right drawings).

Figure 1 shows a pseudo-random graph with eight clusters of 100 nodes each. The probability of an edge $\{u, v\}$ is 0.16 if u and v belong to the same cluster and 0.02 otherwise. (All edges are chosen independently.) Overall, the graph has 6341 intra-cluster edges and 5625 inter-cluster edges. The eight clusters are clearly separated in the drawing for the LinLog model, but the borders of the clusters look fuzzy and some nodes do not seem to belong to any cluster. This is to be expected of a random graph: Some nodes have a small degree, and hence drift to the border of the drawing. Other nodes are equally connected to two clusters, and are drawn between these clusters.

Figure 2 shows a graph with four clusters, each having four subclusters of 50 nodes. The subclusters are cliques (edge probability 1), the edge probability between nodes of different subclusters of the same cluster is 0.32, and the edge probability between nodes of different clusters is 0.16. The drawing for the LinLog model clearly shows the hierarchical structure. With 19600 intra-subcluster edges and 57346 inter-subcluster edges, separating the (sub)clusters seems more difficult than in the first graph. Nevertheless, Fig. 2 looks more orderly than Fig. 1, because every node is guaranteed to be adjacent to all other nodes of its subcluster.

The graph of Fig. 3 is a pseudo-random graph with one central cluster of 200 nodes and three “satellite” clusters of 100 nodes each, called cluster A , B and C . The probability of an edge $\{u, v\}$ is

- 0.12 if u and v belong to the same cluster,
- 0.04 if u belongs to the central cluster and v belongs to cluster A ,
- 0.02 if u belongs to the central cluster and v belongs to cluster B ,
- 0.01 if u belongs to the central cluster and v belongs to cluster C , and
- 0 otherwise.

The first thought might be that the distance from the central cluster to cluster A should be half the distance from the central cluster to cluster B , which again should be half the distance from the central cluster to cluster C . But the actual central-to- B distance in the drawing for the LinLog energy model is greater than twice the central-to- A distance. This is because for cluster B , the central cluster and cluster A effectively form one big cluster of 300 nodes, yielding an effective edge probability of only about 0.013. On the other hand, cluster B has little influence on the central-to- A distance because it is relatively far away from both. So the LinLog model did not only separate different clusters, it also produced interpretable distances between the clusters.

4 Discussion

Related Work: Force and Energy Models. In Sect. 2.4 we have shown that in drawings with minimum LinLog energy, the distance of a cluster to the remaining graph is approximately inversely proportional to the coupling. For many of the best-known force

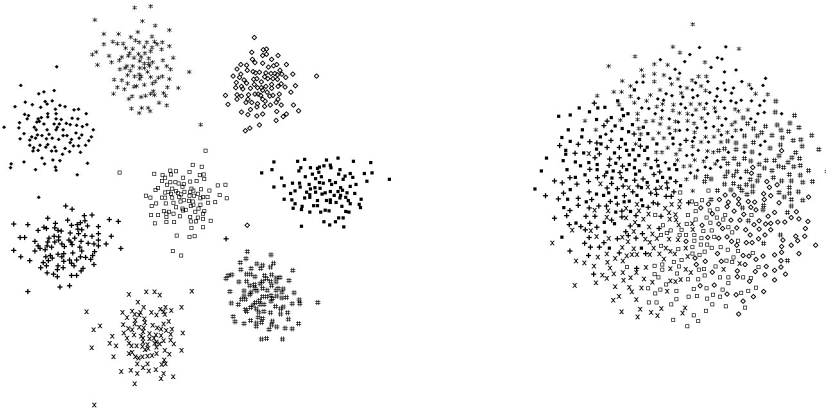


Fig. 1. Pseudo-random graph with intra-cluster edge probability 0.16 and inter-cluster edge probability 0.02; left: LinLog model, right: Fruchterman-Reingold model

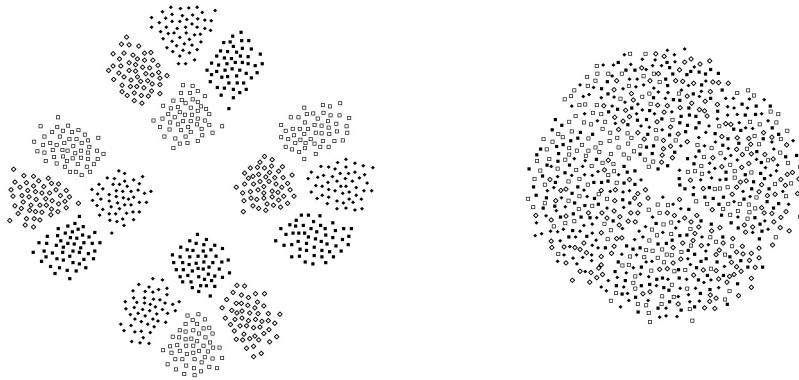


Fig. 2. Pseudo-random graph with hierarchical clusters; left: LinLog model, right: Fruchterman-Reingold model

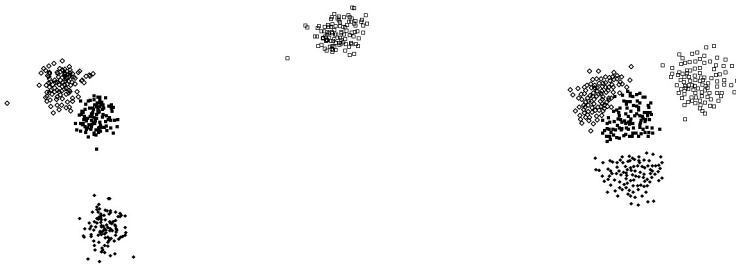


Fig. 3. Pseudo-random "satellite" graph; left: LinLog model, right: Fruchterman-Reingold model

and energy models there is only a weak dependency of distance and coupling, and thus these energy models do not isolate clusters well. (See [24] for a more detailed discussion of these models.)

The models of Eades [12], Fruchterman and Reingold [14], and Davidson and Harel [10] strongly enforce uniform edge lengths. However, separating clusters requires some long edges. Section 3 contains example drawings of the Fruchterman-Reingold model. In multidimensional scaling [20] and the model of Kamada and Kawai [18], the distances between nodes in the drawing are determined by their graph-theoretic distances. But the graph-theoretic distances and the coupling are only weakly related. The energy model of Hall ([16], recently used e.g. in [8,19]) places some non-adjacent nodes and loosely coupled subgraphs very close or at the same position.

All mentioned energy models easily generalize to graphs with weighted edges. Given appropriate edge weights, the models create drawings which reveal clusters. But this means putting clusters in (in the form of edge weights) to get clusters out. Other approaches apply force-directed methods to draw graphs with a *given* hierarchical cluster structure [30,13]. But finding clusters or appropriate edge weights is difficult, because most variants of the graph clustering problem are \mathcal{NP} -hard.

In the LinLog energy model, the attraction and repulsion between the nodes reveal the clusters. It does not require knowledge about clusters as input, but provides it as output. Classical force and energy models and the LinLog model complement each other, as discussed in the third subsection.

Related Work: Clustering by Minimizing Distance Ratios. For a given graph, the cut with the minimum cut ratio is called the sparsest cut. Computing sparsest cuts is \mathcal{NP} -hard. In [22] and [3] approximation algorithms are given that compute sparse cuts from minima of $\frac{\text{arithmetican}(E,p)}{\text{arithmetican}(V^{(2)},p)}$ (with respect to appropriate metrics). So it is not surprising that minimizing the similar ratio $\frac{\text{arithmetican}(E,p)}{\text{geomean}(V^{(2)},p)}$ reveals clusters. The proof that minima of the LinLog energy are also minima of the latter ratio (Theorem 2) is a main result of this paper, because it enables the application of algorithms from force-directed graph drawing to minimize this ratio. Lacking space for a detailed discussion, we can only note that the results concerning sparsest cut approximations from [22] and [3] cannot be directly transferred to minima of the LinLog model, and that the minima used in these approximations are generally less interpretable to humans than LinLog minima.

Generalization: The r -PolyLog Energy Models. The class of energy models r -PolyLog contains two energy models that were already discussed: The 1-PolyLog model is the LinLog model introduced in Sect. 2, and the 3-PolyLog model is equivalent to the Fruchterman-Reingold model [14] discussed in the first subsection. (The Fruchterman-Reingold model is usually expressed as force model, but we give the energy version for uniformity.) For all $r \in \mathbb{R}$ with $r \geq 1$, the r -PolyLog energy $U_{r\text{-PolyLog}}(p)$ of a drawing p is defined as

$$U_{r\text{-PolyLog}}(p) = \sum_{\{u,v\} \in E} \frac{1}{r} \|p_u - p_v\|^r - \sum_{\{u,v\} \in V^{(2)}} \ln \|p_u - p_v\|$$

The class contains energy models which isolate clusters ($r = 1$), energy models which enforce uniform edge lengths ($r \rightarrow \infty$), and many compromises between both extremes ($1 < r < \infty$). Thus it contains suitable energy models for many kinds of undirected graphs from the small worlds described in the next subsection to the graphs addressed by classical force-directed graph drawing.

Application: Drawing Clustered Small-World Graphs. According to Albert and Barabási [1], three concepts are prominent in contemporary thinking about complex networks: small worlds, clustering, and degree distributions. In small-world graphs, the nodes have small graph-theoretic distances, relative to the number of nodes. For clustered small-world graphs with hundreds or thousands of nodes, drawings with uniform edge lengths or Euclidean distances proportional to graph-theoretic distances do not reveal the structure of the graph. That is why most well-known force and energy models are not suitable for drawing such graphs.

The LinLog model can reveal the clusters in small worlds. We applied it for drawing call graphs and similar models of object-oriented programs [21]. Further examples of clustered small-world graphs include models of computer networks, the World Wide Web, and social networks [27,1].

Interpretability of Drawings. Drawings of graphs are useful because we can infer properties of the graph from properties of the drawing. For valid inferences, we need precise statements which properties of the drawing correspond to which properties of the graph. For minimum energy drawings of the LinLog model, Theorem 1 states how the Euclidean distance of nodes corresponds to the adjacency of nodes, and Theorem 3 states how the Euclidean distance of sets of nodes corresponds to their coupling.

Empirical studies have shown that human viewers indeed attribute semantics to the placement of nodes, even if they are not told that such a correspondence exists. In a study of Dengler and Cowan, the most important semantic attribution is that observers view graph drawings hierarchically, that they separate them into interconnected sub-graphs [11]. Blythe et al. concluded from their study in the context of social network analysis, that the Euclidean distance of nodes has a significant effect on the viewers' assignment of nodes to groups [5].

The Role of Theory in the Development and Evaluation of Energy Models. To date, force and energy models have been evaluated mainly empirically. Complementing this empirical evaluation with theoretical evaluation (as done in Sects. 2.3 and 2.4) is desirable for two reasons. First, while empirical studies can only examine a limited number of graphs, theory can prove results for large or infinite classes of graphs. Second, theory can explain why an approach does or does not work for a given class of graphs.

Unlike the strategy of presentation in this paper, the development process of the LinLog model was not first guessing and then evaluating it. Rather, we gradually developed the energy model from the informal requirement of having isolated clusters with interpretable distances. In this process, theoretical considerations were used for the stepwise reduction of the space of candidate energy models. (See [24] for details.)

5 Conclusion

The LinLog energy model was introduced. It was shown that in its minimum energy drawings, clusters are clearly separated from the remaining nodes and their distance to the remaining nodes is approximately the inverse of the coupling. The LinLog model complements well-known force and energy models like the Fruchterman-Reingold model which do not separate the clusters of graphs with small diameter well. The class of energy models r -PolyLog was proposed that includes suitable energy models for many graphs from clustered graphs with small diameter to the graphs addressed by classical force-directed graph drawing.

In the future, we will apply and extend our theoretical tools to develop special-purpose energy models for interpretable visualizations of software systems. Another direction of future work is the theoretical analysis of existing force and energy models. Initial results can be found in [24].

An important problem that was not addressed in this paper is the development of algorithms that reliably and efficiently find good energy minima. We have shown that we can draw valid inferences from minimum energy drawings, but interpreting drawings with high energy (e.g. most random drawings) in the same way can give invalid results. Different kinds of graphs not only require different energy models, but may also require different minimization algorithms. For example, some multi-scale algorithms are not expected to find good energy minima for dense graphs or graphs with small diameter [15, 17,29].

References

1. R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–97, 2002.
2. C. J. Alpert and A. B. Kahng. Recent directions in netlist partitioning: A survey. *Integration, the VLSI Journal*, 19(1-2):1–81, 1995.
3. Y. Aumann and Y. Rabani. An $O(\log k)$ approximate min-cut max-flow theorem and approximation algorithm. *SIAM Journal on Computing*, 27(1):291–301, 1998.
4. G. D. Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, 1999.
5. J. Blythe, C. McGrath, and D. Krackhardt. The effect of graph layout on inference from social network data. In F.-J. Brandenburg, editor, *Proc. Symposium on Graph Drawing (GD 1995)*, LNCS 1027, pages 40–51, 1996. Springer-Verlag.
6. F.-J. Brandenburg, M. Himsolt, and C. Rohrer. An experimental comparison of force-directed and randomized graph drawing algorithms. In F.-J. Brandenburg, editor, *Proc. Symposium on Graph Drawing (GD 1995)*, LNCS 1027, pages 76–87, 1996. Springer-Verlag.
7. U. Brandes. Drawing on physical analogies. In M. Kaufmann and D. Wagner, editors, *Drawing Graphs: Methods and Models*, LNCS 2025, pages 71–86. Springer-Verlag, 2001.
8. U. Brandes and S. Cornelsen. Visual ranking of link structures. In F. Dehne, J.-R. Sack, and R. Tamassia, editors, *Proc. 7th International Workshop on Algorithms and Data Structures (WADS 2001)*, LNCS 2125, pages 222–233, 2001. Springer-Verlag.
9. R. Brockenaue and S. Cornelsen. Drawing clusters and hierarchies. In M. Kaufmann and D. Wagner, editors, *Drawing Graphs: Methods and Models*, LNCS 2025. Springer-Verlag, 2001.

10. R. Davidson and D. Harel. Drawing graphs nicely using simulated annealing. *ACM Transactions on Graphics*, 15(4):301–331, 1996.
11. E. Dengler and W. Cowan. Human perception of laid-out graphs. In S. H. Whitesides, editor, *Proc. 6th International Symposium on Graph Drawing (GD 1998)*, LNCS 1547, pages 441–443, 1998. Springer-Verlag.
12. P. Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.
13. P. Eades and M. L. Huang. Navigating clustered graphs using force-directed methods. *Journal of Graph Algorithms and Applications*, 4(3):157–181, 2000.
14. T. M. J. Fruchterman and E. M. Reingold. Graph drawing by force-directed placement. *Software – Practice and Experience*, 21(11):1129–1164, 1991.
15. P. Gajer, M. T. Goodrich, and S. G. Kobourov. A multi-dimensional approach to force-directed layouts of large graphs. In J. Marks, editor, *Proc. 8th International Symposium on Graph Drawing (GD 2000)*, LNCS 1984, pages 211–221, 2001. Springer-Verlag.
16. K. M. Hall. An r-dimensional quadratic placement algorithm. *Management Science*, 17(3):219–229, 1970.
17. D. Harel and Y. Koren. A fast multi-scale method for drawing large graphs. In J. Marks, editor, *Proc. 8th International Symposium on Graph Drawing (GD 2000)*, LNCS 1984, pages 183–196, 2001. Springer-Verlag.
18. T. Kamada and S. Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31(1):7–15, 1989.
19. Y. Koren, L. Carmel, and D. Harel. Ace: A fast multiscale eigenvector computation for drawing huge graphs. In *Proc. IEEE Symposium on Information Visualization (InfoVis 2002)*, pages 137–144, 2002.
20. J. B. Kruskal and M. Wish. *Multidimensional Scaling*. Sage Publications, 1978.
21. C. Lewerentz and A. Noack. CrocoCosmos – 3D visualization of large object-oriented programs. In *Graph Drawing Software*, pages 279–297. Springer-Verlag, 2003.
22. N. Linial, E. London, and Y. Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, 1995.
23. S. Mancoridis, B. S. Mitchell, C. Rorres, Y. Chen, and E. R. Gansner. Using automatic clustering to produce high-level system organizations of source code. In *Proc. 6th IEEE International Workshop on Program Understanding (IWPC 1998)*, pages 45–52, 1998.
24. A. Noack. Energy models for drawing clustered small-world graphs. Technical Report 07/03, Institute of Computer Science, Brandenburg University of Technology at Cottbus, 2003.
25. A. Pothen. Graph partitioning algorithms with applications to scientific computing. In D. E. Keyes, A. Sameh, and V. Venkatakrisnan, editors, *Parallel Numerical Algorithms*, pages 323–368. Kluwer, 1997.
26. A. J. Quigley and P. Eades. FADE: Graph drawing, clustering, and visual abstraction. In J. Marks, editor, *Proc. 8th International Symposium on Graph Drawing (GD 2000)*, LNCS 1984, pages 197–210, 2001. Springer-Verlag.
27. S. H. Strogatz. Exploring complex networks. *Nature*, 410:268–276, 2001.
28. Daniel Tunkelang. *A Numerical Optimization Approach to General Graph Drawing*. PhD thesis, School of Computer Science, Carnegie Mellon University, 1999.
29. C. Walshaw. A multilevel algorithm for force-directed graph drawing. In J. Marks, editor, *Proc. 8th International Symposium on Graph Drawing (GD 2000)*, LNCS 1984, pages 171–182, 2001. Springer-Verlag.
30. X. Wang and I. Miyamoto. Generating customized layouts. In F.-J. Brandenburg, editor, *Proc. Symposium on Graph Drawing (GD 1995)*, LNCS 1027, pages 504–515, 1996. Springer-Verlag.