



Skinnable graph drawing

Mountaz Hascoët

► **To cite this version:**

Mountaz Hascoët. Skinnable graph drawing. [Research Report] RR-14007, LIRMM. 2014. lirmm-00615429v2

HAL Id: lirmm-00615429

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00615429v2>

Submitted on 30 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Skinnable graph drawing

Mountaz Hascoët¹

LIRMM,
CNRS UMR 5506,
Univ. Montpellier 2,
161, rue Ada
34392 Montpellier Cedex,
France
mountaz@lirmm.fr

Abstract. Skinning is a recent concept emerging from long lasting efforts to separate look-and-feel from core functional aspects in software engineering or web design. In this paper, we propose skinnable graph drawing as a process that takes a general graph $G(V, E)$ and computes a drawing from a skin compatible with a signature of G . Skinnable graph drawing is designed to address the drawing of general graphs having no particular node labeling. Skinnable graph drawing motivations include (1) enable reuse and share of customized drawings, (2) preserve mental maps by maintaining some visual graph structures invariant and (3) facilitate the use of different conventional drawings for the same graphs to reconcile different visual perspectives on similar graph structures.

In this paper, we propose useful definitions for skinnable graph drawing and report the results of a case study based on sixteen well-known graphs as an illustration of signature computations and a proof of concept of the skinnable graph drawing approach.

1 Skins

Let $G(V, E)$ be a graph. A drawing of G is a geometric representation of G such that each element of G (vertex or edge) is drawn as a geometric shape (e.g. point, circle, polygon, curve, etc.) [3].

We call a skin for G a description of a drawing with two basic properties: (1) a skin of G can be customized, shared, and copied independently of the content of G and (2) a skin and a signature of G are sufficient to create a drawing for any graph G provided that the signature of G is compatible with the signature of the skin. In other words, a skin can be considered as a description of a drawing that can be used for a set of compatible graphs.

Consequently, collections of skins can be created once and shared amongst experts studying similar graphs. By creating skins that can be shared and reused, we hope to preserve not only an individual mental map [4] and, maybe more importantly, also a common collective mental map for most common graphs typically studied in a community.

Table 1. Computation of graph signatures for the 16 sample graphs (a)

| | |
|----------------------|-------------------------------|
| domino (diameter=3) | |
| Signature of order 1 | 2-2-4 3-3-2 |
| Signature of order 2 | 2-2-2-3-4 3-3-2-4-2 |
| Signature of order 3 | 2-2-2-3-1-2-4 3-3-2-4-2 |
| 4-sun (diameter=3) | |
| Signature of order 1 | 2-3-4 5-10-4 |
| Signature of order 2 | 2-3-4-9-4 5-10-2-4-4 |
| Signature of order 3 | 2-3-4-9-1-2-4 5-10-2-4-4 |
| 3-sun (diameter=2) | |
| Signature of order 1 | 2-3-3 4-7-3 |
| Signature of order 2 | 2-3-3-6-3 4-7-1-2-3 |
| 4-fan (diameter=2) | |
| Signature of order 1 | 2-3-2 3-5-3 5-9-1 |
| Signature of order 2 | 2-3-3-6-2 3-5-2-4-3 5-9-1 |
| claw (diameter=2) | |
| Signature of order 1 | 1-1-3 3-3-1 |
| Signature of order 2 | 1-1-2-2-3 3-3-1 |
| paw (diameter=2) | |
| Signature of order 1 | 1-1-1 2-3-2 3-4-1 |
| Signature of order 2 | 1-1-2-3-1 2-3-1-1-2 3-4-1 |
| diamond (diameter=2) | |
| Signature of order 1 | 2-3-2 3-5-2 |
| Signature of order 2 | 2-3-1-2-2 3-5-2 |
| house (diameter=2) | |
| Signature of order 1 | 2-2-2 2-3-1 3-4-2 |
| Signature of order 2 | 2-2-2-4-2 2-3-2-3-1 3-4-1-2-2 |

Furthermore, the level of customization for a skin may vary widely from automatically created skins to manually designed ones. A skin can be particularly well customized by some graphic designers for some purposes, saved and further reused by more common users or other experts not familiar with graphic design.

Finally, different perspectives on the same graph can further be embodied by different skins.

2 Graph Signature

Let $G(V, E)$ be a graph. We introduce the *context coefficient* of order p of a node $u \in V$ denoted by $c_p(u)$ as a form of node invariant [1]. The context coefficient can be considered as a way to characterise the neighborhood of a node at various scales or orders. Note that clustering coefficients are another node invariant frequently used to measure the neighborhood connexity of a given node. However, clustering coefficient is limited not only to its direct neighborhood, but also to a ratio, the ratio of the number of effective edges over the number of

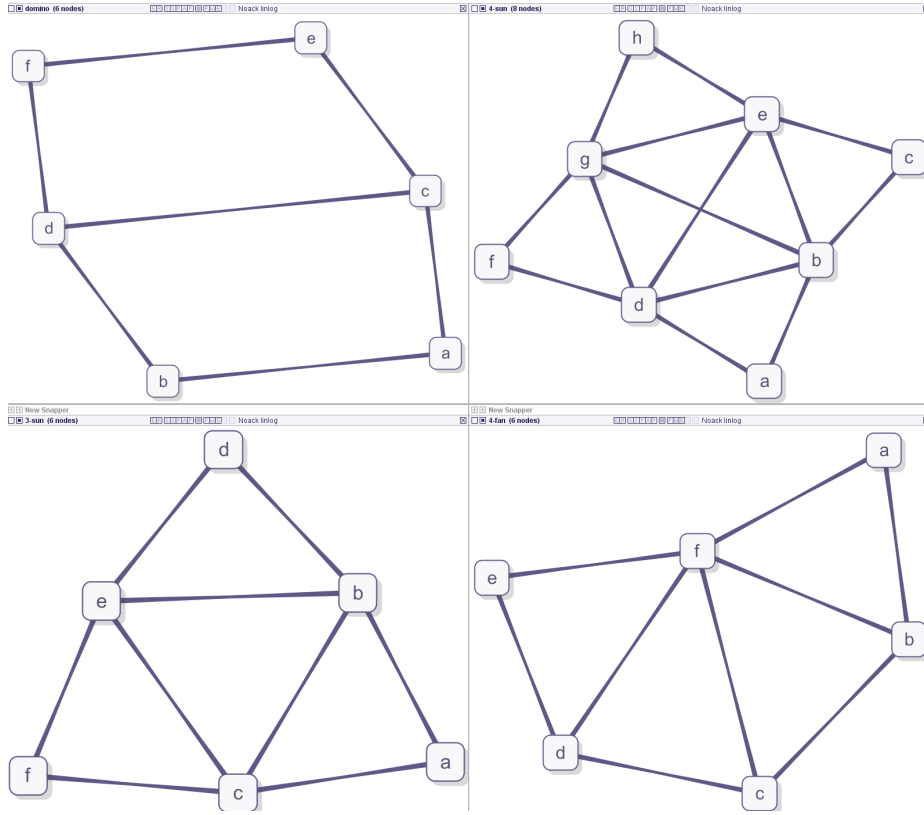


Fig. 1. From left to right, top to bottom: domino, 4-sun, 3-sun, 4-fan graphs

possible edges. Therefore, contrary to context coefficients, clustering coefficients have limited discriminating power and are not sufficient to achieve skinnable graph drawing like many other node invariants.

Context coefficient is suitable for skinnable graph drawing because (1) it has sufficient discriminating power to fully capture a given node structural role in a given graph and (2) it can be computed at various scales, possibly enabling possible lighter or deeper computations depending on the situation.

In order to propose a general formal definition of the context coefficient, we start by recalling our definitions of neighborhoods of depth p for a given node u .

Given a graph $G(V, E)$, a graph distance $d(u, v)$ defined by, for example, the length of the shortest path between u and v , the neighborhood of u of depth p is denoted by $N(u, p)$ and defined as:

$$\forall p > 0, N(u, p) = \{v \in V | d(v, u) = p\}$$

and

$$\forall p > 0, E(u, p) = \{(v, w) \in E | (v \in V', w \in V'' \cup V') \vee (v \in V'' \cup V', w \in V')\}$$

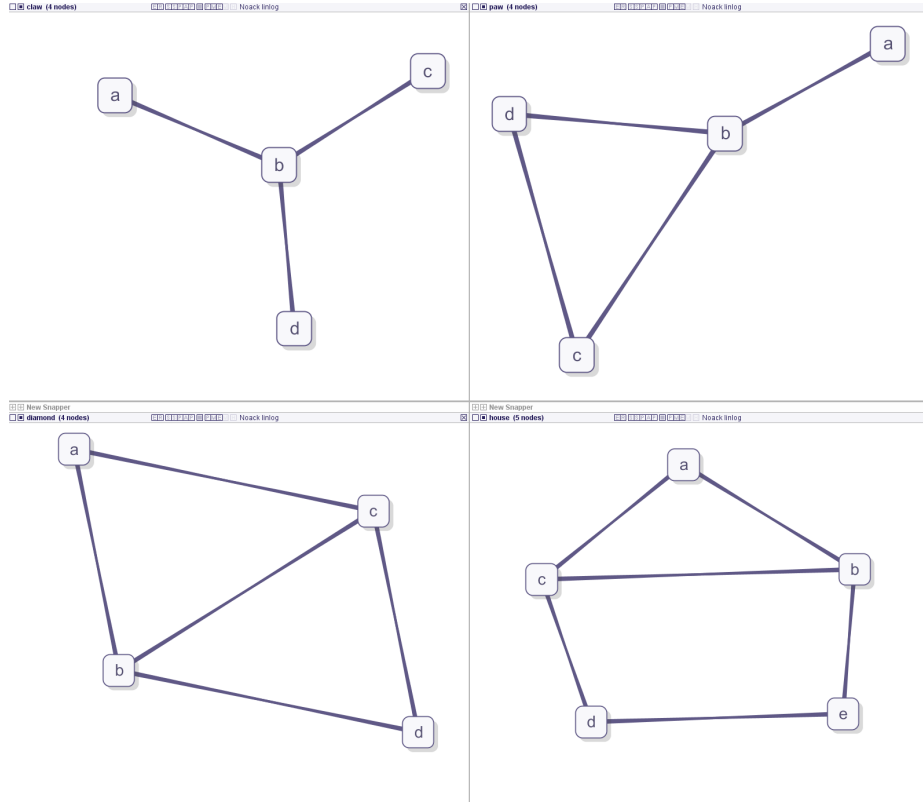


Fig. 2. From left to right, top to bottom: claw, paw, diamond, house

where

$$V' = N(u, p), V'' = N(u, p-1)$$

Based on these definitions, we denote by $c_p(u)$, the context coefficient of order p of a node u . $c_p(u)$ is defined as the couple made from the number of nodes and the number of edges in the neighborhood of u at depth p or at the frontier between the neighborhood of u at depth p and the neighborhood of u at depth $p-1$. In other terms,

$$c_p(u) = (|N(u, p)|, |E(u, p)|) \quad (1)$$

A stamping of order p for G is further defined as a function ξ_p such that:

$$\begin{aligned} \xi_p: V &\rightarrow (\mathbb{N} \times \mathbb{N})^p \\ u &\mapsto (c_1(u), \dots, c_p(u)) \end{aligned} \quad (2)$$

Finally, the signature of order p of G is the ordered list of tuples $(\xi_p(u); n)$ where n is the number of nodes in G for which the stamping of order p is the same,

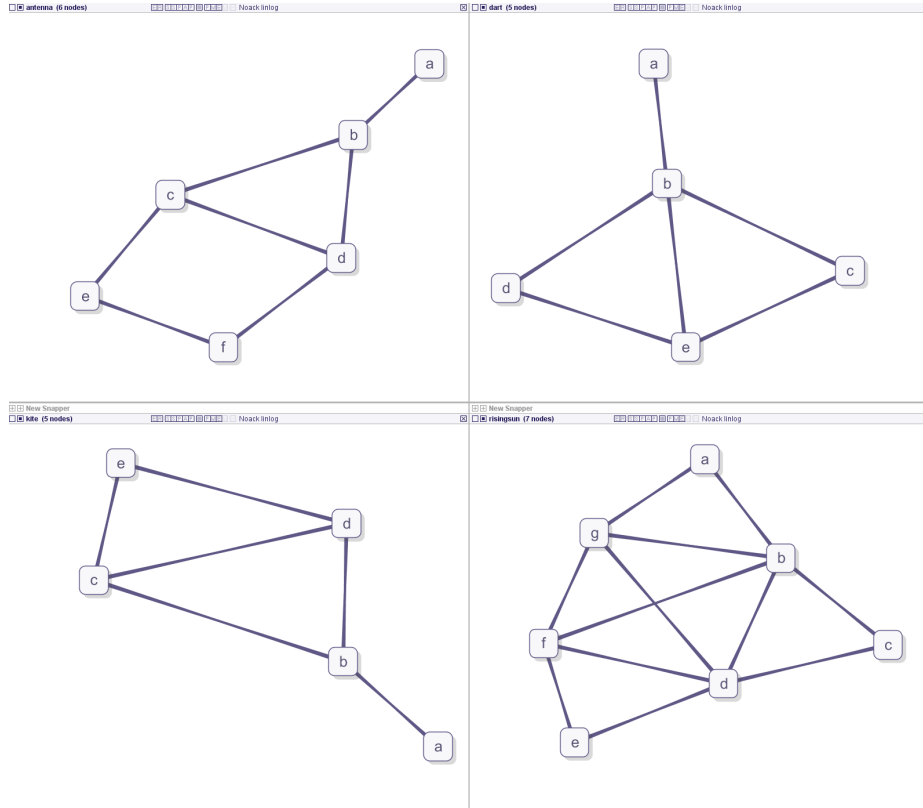


Fig. 3. From left to right, top to bottom: antenna, dart, kite, rising sun

and the order of the tuples in the signature is based on $\xi_p(u)$ values comparing $c_p(u)$ from 0 to p .

3 Skinnable graph drawing

A general algorithm for skinnable graph drawing can be derived from the concepts previously defined.

Skin elements are identified by context coefficient based stampings so that the mapping of graph elements to their corresponding skin graphical elements is straightforward.

In the case of graphs where several nodes in the graph have the same stamping, the choice of which node corresponds to which skin graphical element is random since these nodes have same structural roles.

By construction, two isomorphic graphs have the same signature. Therefore, two isomorphic graphs can use the same skins and be drawn with the same drawings. This is one of the main contributions of this work.

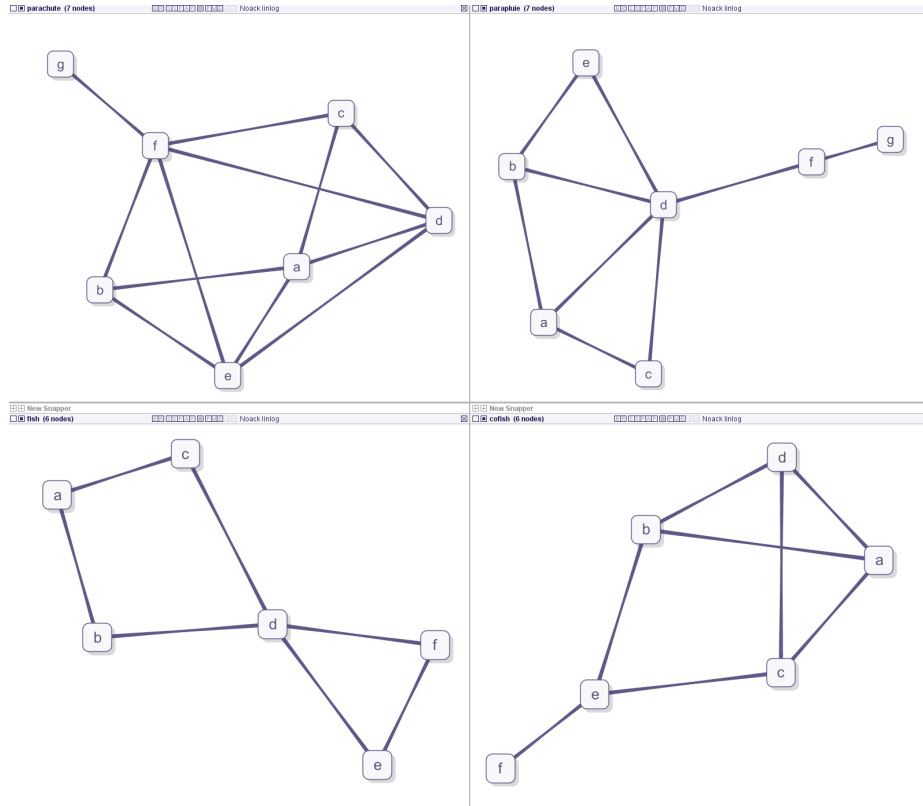


Fig. 4. From left to right, top to bottom: parachute, parapluiie, fish, co-fish

Note that, the question of whether two graphs with the same signatures are isomorphic is left to readers interested in the long lasting isomorphism debate [1].

4 Case Study

In order to better illustrate the concept of skinnable graph drawing, we have conducted a case study on a set of 16 small typical graphs suitable for skinnable drawing. The Figures 1 to 4 show drawings of these graphs computed from the linlog algorithm from Noack [5] and the side-by-side views were built using Donatien, an open source software designed to visually compare and match general graphs [2].

The Tables 1 and 2 summarize the uncompressed signatures for each graph displayed in the Figures 1, 2, 3, 4 at the various possible orders. Note that for any graph, the maximum order of the signature is its diameter.

At the first order, the graph antenna —Table 2 and Figure 3, has 1 node with a context coefficient of 1-1, 2 nodes with a context coefficient of 2-2 and 3 nodes

Table 2. Computation of graph signatures for the 16 sample graphs (b)

| | |
|------------------------|---|
| antenna (diameter=3) | |
| Signature of order 1 | 1-1-1 2-2-2 3-4-3 |
| Signature of order 2 | 1-1-2-3-1 2-2-2-4-2 3-4-2-3-3 |
| Signature of order 3 | 1-1-2-3-2-3-1 2-2-2-4-1-1-2 3-4-2-3-3 |
| dart (diameter=2) | |
| Signature of order 1 | 1-1-1 2-3-2 3-5-1 4-6-1 |
| Signature of order 2 | 1-1-3-5-1 2-3-2-3-2 3-5-1-1-1 4-6-1 |
| kite (diameter=3) | |
| Signature of order 1 | 1-1-1 2-3-1 3-4-1 3-5-2 |
| Signature of order 2 | 1-1-2-3-1 2-3-1-2-1 3-4-1-2-1 3-5-1-1-2 |
| Signature of order 3 | 1-1-2-3-1-2-1 2-3-1-2-1-1-1 3-4-1-2-1 3-5-1-1-2 |
| risingsun (diameter=3) | |
| Signature of order 1 | 2-3-3 4-8-2 5-10-2 |
| Signature of order 2 | 2-3-3-7-2 2-3-4-9-1 4-8-2-4-2 5-10-1-2-2 |
| Signature of order 3 | 2-3-3-7-1-2-2 2-3-4-9-1 4-8-2-4-2 5-10-1-2-2 |
| parachute (diameter=3) | |
| Signature of order 1 | 1-1-1 3-5-2 4-7-1 4-8-2 5-8-1 |
| Signature of order 2 | 1-1-4-7-1 3-5-3-7-2 4-7-1-4-1 4-8-2-4-2 5-8-1-4-1 |
| Signature of order 3 | 1-1-4-7-1-4-1 3-5-3-7-2 4-7-1-4-1-1-1 4-8-2-4-2 5-8-1-4-1 |
| parapluie (diameter=3) | |
| Signature of order 1 | 1-1-1 2-2-1 2-3-2 3-5-2 5-8-1 |
| Signature of order 2 | 1-1-1-1-1 2-2-4-7-1 2-3-3-5-2 3-5-2-3-2 5-8-1-1-1 |
| Signature of order 3 | 1-1-1-1-4-7-1 2-2-4-7-1 2-3-3-5-1-1-2 3-5-2-3-1-1-2 5-8-1-1-1 |
| fish (diameter=3) | |
| Signature of order 1 | 2-2-3 2-3-2 4-5-1 |
| Signature of order 2 | 2-2-1-2-1 2-2-3-5-2 2-3-2-2-2 4-5-1-2-1 |
| Signature of order 3 | 2-2-1-2-2-3-1 2-2-3-5-2 2-3-2-2-1-2-2 4-5-1-2-1 |
| cofish (diameter=3) | |
| Signature of order 1 | 1-1-1 3-3-1 3-4-2 3-5-2 |
| Signature of order 2 | 1-1-2-2-1 3-3-2-5-1 3-4-2-4-2 3-5-1-2-2 |
| Signature of order 3 | 1-1-2-2-2-5-1 3-3-2-5-1 3-4-2-4-2 3-5-1-2-1-1-2 |

with a context coefficient of 3-4. Developing the context coefficient at higher orders is not necessary to discriminate the structural aspects of an antenna as can be seen from the resulting higher order signatures contrary to other graphs where computing higher orders of signature reveals discriminating information.

The case of the graph rising sun —Figure 3 and signatures in Table 2, is a good example of a case where higher signature orders provide additional information. At the order 1 the rising sun signature exhibits 3 different types of node neighborhood structure with respective context coefficient of 2-3, 4-8 and 5-10. But at the order 2, the rising sun signature exhibits four types of nodes with respective context coefficients of order 2 : 2-3-3-7, 2-3-4-9, 4-8-2-4, and 5-10-1-2. This suggests that at the first order, the nodes with labels c, a and e may be confused. Only the higher order signature makes it possible to further discrimi-

Table 3. GraphML-like co-fish graph description (left) and skin definition (right)

| | |
|---|---|
| <pre> <graph id='Graph 14'> <node id='a' x="0" y="1"/> <node id='b' x="1" y="2"/> <node id='c' x="1" y="0"/> <node id='d' x="0.5" y="1"/> <node id='e' x="1.5" y="1"/> <node id='f' x="2" y="1"/> <edge source='a' target='b'/> <edge source='a' target='c'/> <edge source='a' target='d'/> <edge source='d' target='b'/> <edge source='d' target='c'/> <edge source='b' target='e'/> <edge source='c' target='e'/> <edge source='e' target='f'/> </graph> </pre> | <pre> <skin signature='1-1-2-2-2-5-1 3-3-2-5-1 3-4-2-4-2 3-5-1-2-1-1-2 '> <point name='3-5-1-2-1-1-1' x="0.5" y="1"/> <point name='3-4-2-4-1' x="1" y="2"/> <point name='3-4-2-4-2' x="1" y="0"/> <point name='3-5-1-2-1-1-2' x="0" y="1"/> <point name='3-3-2-5-1' x="1.5" y="1" /> <point name='1-1-2-2-2-5-1' x="2" y="1"/> <line from='3-5-1-2-1-1-1' to='3-4-2-4-1'/> <line from='3-5-1-2-1-1-1' to='3-4-2-4-2'/> <line from='3-5-1-2-1-1-1' to='3-5-1-2-1-1-2'/> <line from='3-5-1-2-1-1-2' to='3-4-2-4-1'/> <line from='3-5-1-2-1-1-2' to='3-4-2-4-2'/> <line from='3-4-2-4-1' to='3-3-2-5-1'/> <line from='3-4-2-4-2' to='3-3-2-5-1' /> <line from='3-3-2-5-1' to='1-1-2-2-2-5-1'/> </pre> |
|---|---|

nate the different roles played by c compared to e and a as can be seen from the Figure 3.

Once the signatures have been computed, and skins have been defined, drawings can be made for any compatible graph.

To illustrate this further step, let's take the example of a co-fish graph - see Fig. 5 and 4 for three possible drawings of the cofish graph, and Table 2 for its signature at various orders.

Finally, a skin for the co-fish graph is defined in Table 3 (right side).

This skin definition, uses a node labeling made from context coefficient based stamping defined previously, and layout is described by providing explicit positions (namely attributes x and y) for each node. Even though other skin definitions are possible, for the illustration purpose, the skin representation used in Table 3 is well-suited because it is both straightforward and sufficient to describe the conventional drawing of the co-fish of Figure 5 (left side). This definition further satisfies the requirements of skin introduced in the first section of this paper.

Given this skin, and a co-fish graph such as the co-fish graph described in the Table 3 using a GraphML based format, the conventional drawing of Figure 5 (left side) is produced automatically a result of a straightforward matching of nodes with same context coefficient based stampings.

The co-fish example is also a good example to recall that isomorphic graphs might have many different node labeling. For example, in the co-fish graph defined in Table 3, the nodes respectively labeled a and d have interchangeable structural roles. The labeling convention can further be used to display a at the left of b like in the Figure 5 (left) or symmetrically display d at the left of a like in the Figure 5 (right). Similarly, c and b also have interchangeable structural

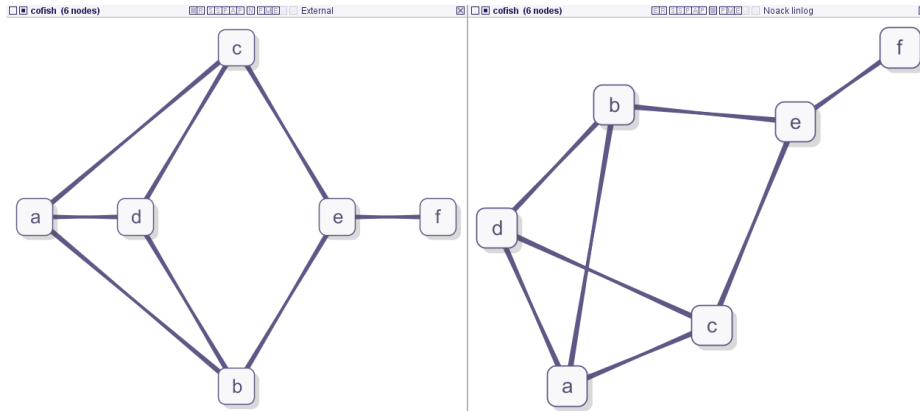


Fig. 5. Co-fish graph layout (a) with a skin (left) (b) with a force-based computation [5] (right)

roles and can be displayed respectively top, bottom Figure 5 (left) or bottom, top Figure 5 (right).

The choice of the labeling strategy in the drawing can be important or not depending on the application. For simplification reasons and because the main original contribution of skinnable graph drawing is with the preservation of visual graph structural properties in the drawing, representing labeling strategies has been considered out of the scope of this paper and left for future work.

5 Conclusion

In this paper, we have proposed skinnable graph drawing as complementary approach to previous graph drawing approaches. Skinnable graph drawing is based on the definition at various orders of context coefficients, node stamping, graph signatures to come up with a drawing of a given graph. We have reported the results of a case study based on sixteen well-known graphs as an illustration of signature computations and a proof of concept of the skinnable graph drawing approach.

The first asset of skinnable graph drawing is to enable the re-use of a drawing computed once for a given graph for all isomorphic graphs, to begin with the graph itself. Therefore enabling (1) invariant drawings for a given graph (2) saving of previous drawing customization time and efforts, (3) saving of computation time, and (4) mental map preservation.

The second advantage of this approach derives from the first and consists in facilitating the construction of visual standards. This is useful to facilitate the construction of collective mental map shared by a community of users. This has been one recurrent concern in graph drawing since the creation of the field and is particularly useful for sharing analyses on various complex graphs. It is also very useful in the context of visual comparison and visual matching of graphs [2].

The third result is to facilitate the identification of multiple drawing conventions to satisfy different visual expectations corresponding to various communities drawing conventions and purposes. By making it possible to identify drawing conventions for a given graph, we hope to facilitate the construction and switching of multiple drawings where different visual perspectives are useful for the analysis, comparison or matching of graphs.

Future work includes the design of a general notation for skin definition. Future work also include various aspects of signature optimization both in terms of computation, representation or compression. Finally, future work also include the analysis of skin compatibilities, possibly identifying skin compatibility for classes of graphs.

6 Acknowledgments

Thanks to the anonymous reviewers of Graph Drawing conference who have provided comments useful to improve the quality of this paper.

References

1. Fortin, S.: : The graph isomorphism problem. Tech. rep., University of Alberta (1996)
2. Hascoët, M., Dragicevic, P.: Interactive graph matching and visual comparison of graphs and clustered graphs. In: AVI. pp. 522–529 (2012)
3. Jünger, M., Mutzel, P.: Graph drawing software. Springer (2004)
4. Misue, K., Eades, P., Lai, W., Sugiyama, K.: Layout adjustment and the mental map. *Journal of Visual Languages And Computing* 6(2), 183–210 (1995)
5. Noack, A.: An energy model for visual graph clustering. In: Liotta, G. (ed.) *Graph Drawing, Lecture Notes in Computer Science*, vol. 2912, pp. 425–436. Springer Berlin Heidelberg (2004)