

A Genetic Algorithm for Drawing Directed Acyclic Graphs

Zhou Weihua

School of Computer Science

Wuhan University

+86-027-63489238

zhouweihua.whu@gmail.com

Huang Jingwei

School of Computer Science

Wuhan University

+86-027-68773434

jwhuang@public.wh.hb.cn

ABSTRACT

Graph drawing is a standard means for visualizing the relational information. This paper presents a new genetic algorithm for drawing directed acyclic graphs, which adopts a new representation scheme of individuals, and designs genetic operators based on induced sub-graphs. As far as we know, it is the first attempt to introduce the induced sub-graphs to draw directed acyclic graphs. Experimental results and systematic comparison on various graphs prove its validity and practicality.

Categories and Subject Descriptors

I.3.3 [Computer Graphics]: Picture/Image Generation - Viewing Algorithms.

General Terms

Algorithms

Keywords

Graph drawing; genetic algorithm; induced sub-graph

1. INTRODUCTION

Graph drawing is a conventional means for visualizing the relation information, and it is widely used in VLSI design, software engineering, document management, etc. The state of the art in graph drawing is comprehensively surveyed in [1, 2].

There are many kinds of graph drawing problems, such as drawing directed graphs, drawing planar graphs, and so on. This paper concentrates on drawing directed acyclic graphs in the plane.

The most popular method to this problem is the Sugiyama heuristic [3], which contains two independent steps: (1) group the vertices into different subsets, called layering vertically, and when one edge crosses a layer, a corresponding dummy node is inserted; (2) order the nodes according to certain aesthetic constraints, called positioning horizontally. Figure 1 illustrates that a normal drawing is transformed into the Sugiyama-style drawing.

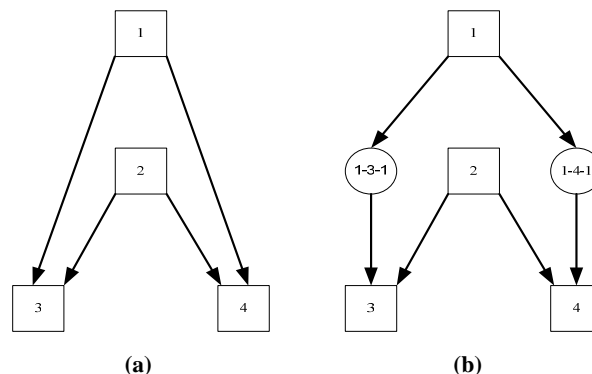


Figure 1. An example of transforming the normal drawing into the Sugiyama-style drawing. (a) Normal drawing. (b) Sugiyama-style drawing. Note that the dummy nodes are labeled with the ellipses.

To measure the “nice” drawings formally, many aesthetic constraints are proposed [3, 4, 5], and we require:

C_1 : minimizing the number of the edge crossings;

C_2 : minimizing the total area;

C_3 : minimizing the number of dummy nodes.

Unfortunately, it is NP-hard to reduce the number of edge crossings [6]. Furthermore, we expect that more aesthetics can be added according to our future requirement. Since the genetic algorithm is a good global and flexible optimizer for the nonlinear optimization problems, it is natural to apply it to draw the directed acyclic graphs [7, 8].

In this paper we present a new genetic algorithm for drawing directed acyclic graphs, which adopts a new representation scheme of individuals and designs genetic operators based on induced sub-graphs. As far as we know, it is the first attempt to introduce the induced sub-graphs into this field. Experimental results prove its validity and practicality. Also, systematic comparison is made on various graphs to evaluate the effect of different experimental parameters.

This paper is organized as follows: In Section 2 we explain the basic definitions, and the genetic graph drawing is described and discussed in Section 3. Then the experimental results and comparison are illustrated in Section 4. Finally, we present our conclusions and future work.

2. Definitions

A **directed graph** G consists of a vertex set $V(G)$ and an edge set $E(G)$ where an edge is an ordered pair of distinct vertices of G .

A **directed acyclic graph** (often short as DAG) is a directed graph containing no cycles. This means that if there is a route from the vertex u to the vertex v then there is no way back.

A **sub-graph** of a graph G is a graph G' such that

$$V(G') \subseteq V(G), E(G') \subseteq E(G).$$

A sub-graph G' of a graph G is an **induced sub-graph** if two vertices of $V(G')$ are adjacent in G if and only if they are adjacent in G .

For the drawings of directed acyclic graphs, in order to differ from the dummy node, we call the original vertex as the real node, and both the real node and the dummy node are called the node.

A depth- h induced sub-graph starting with the real node v is an induced sub-graph that consists of all the nodes that the node v is able to reach in h steps. Figure 2 shows the depth-2 induced sub-graph starting with node 1 in Figure 1(b). If $h=0$, the corresponding depth-0 induced sub-graph starting the real node v degenerates into the only real node v .

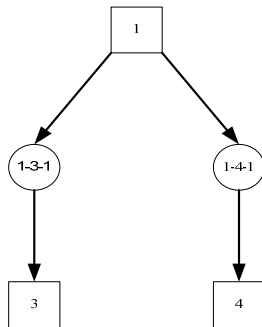


Figure 2. A depth-2 induced sub-graph.

3. Genetic Algorithm

Genetic algorithms are stochastic global search methods that have proved to be successful for many kinds of optimization problems [9].

In general, the genetic algorithms encode the possible solutions into a population of candidate individuals and try to optimize these by three basic means: selection, recombination, mutation. The basic principles are illustrated in Figure 3. The population is initialized randomly. Then, in every subsequent generation, a parent population is selected from the current population, with higher probability of selection for better individuals. From this parent population, new individuals are generated by recombining genetic information of two parents, or mutating (changing slightly) the parent. Finally, the offspring population becomes the current population and the next generation begins.

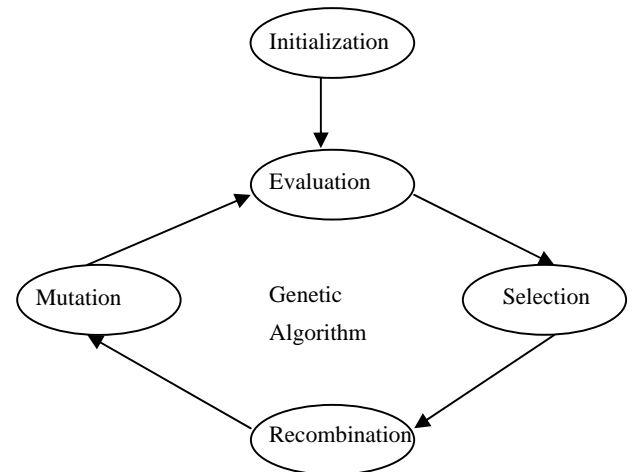


Figure 3. The basic loop of genetic algorithms.

More detail about genetic algorithms can be seen in [9], and here only the modification for drawing directed acyclic graphs is discussed, including the representation, evaluation, recombination and mutation.

3.1 Representation

The representation encodes the candidate solution into an individual, and in essence, it implements the data structure of the solution, which is the base of the whole algorithm. Utech et al. [8] propose the Edge Length Representation scheme, and here we modify their representation and adopt the combination of the array of fixed length and the linked list of variable length. Figure 4 gives an example: The information of real nodes is stored in the array, and it is easily found by the index if needed; On the other hand, since the dummy nodes are dynamically created and belong to certain edge, we store them in the linked nodes associated with the source and the target.

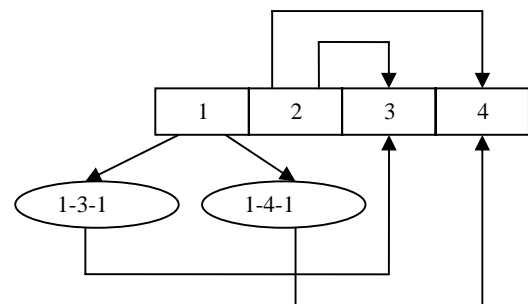


Figure 4. The representation corresponding to the drawing in Figure 1(b).

In fact, this kind of representation is designed to facilitate the operations on the induced sub-graph starting with certain real nodes. Perhaps it can be imaged as a chain-like structure when accessing the sub-graphs, which speeds up the whole algorithm significantly.

3.2 Evaluation

The most significant advantage of the evaluation in a GA is that it can measure any desired aesthetic criteria on the results and

weight these criteria to suit the goals. This guarantees the flexibility of genetic algorithms since future research may reveal more accurate criteria.

In essence, the evaluation takes a role to formalize the aesthetics. As for the three pieces of rules in Section 1, computing the edge crossings C in C_1 is the most frequently discussed problems [3, 4, 8, 10, 11], and we sweep directly the edges between the neighbor layers to count it. For C_2 , The total area of the drawing is regarded as the product of the layer number L and the maximal node number W in every layer (including the dummy nodes). Then, counting dummy nodes D in C_3 is direct.

The different aesthetic constraints will be weighed next step by the group of coefficients W_1 , W_2 , and W_3 in turn. It is a common technique for the multi-objective optimization problems. For any individual p , its fitness is computed as follows:

$$\text{fitness}(p) = W_1 * C + W_2 * L * W + W_3 * D$$

3.3 Recombination

Two new offspring individuals can be created by exchanging the genetic information of different parent individuals, which is called the recombination in genetic algorithms.

The classic method is one- or two-point crossovers, which selects one or two separated points and exchanges some parts of parents [8, 9]. Take two bit strings as the example, 100100 and 011111, and if we select the third point as the beginning point to recombine, the offspring strings, 101111 and 010100 come into being. However, unfortunately, for the layouts of graphs, the exchanged information is fragmentary.

Since we expect that the exchanged genetic information is independent and meaningful, the induced sub-graph is a good choice. Firstly, we select a real node v and assign the depth of the induced sub-graph starting with v randomly, and then the corresponding sub-graphs are exchanged. Figure 5 exhibits the parent individuals and their offspring individuals after the recombination based on the depth-2 induced sub-graphs starting with the node 1. The corresponding sub-graphs are highlighted in dotted lines. Compared to the traditional point crossover, our new method decreases the computational effort of adjusting the meaningless layouts; on the other hand, it contributes to keep the good sub-layouts integral.

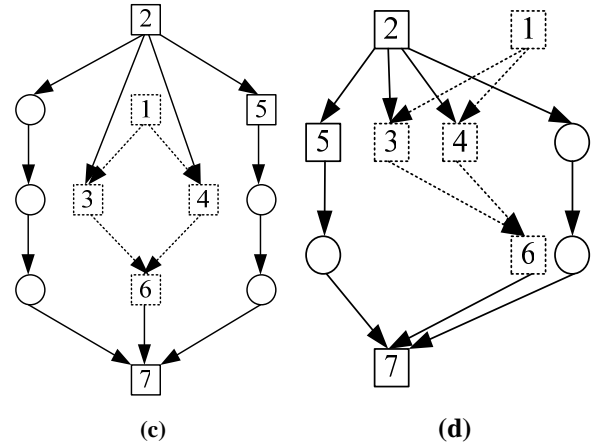
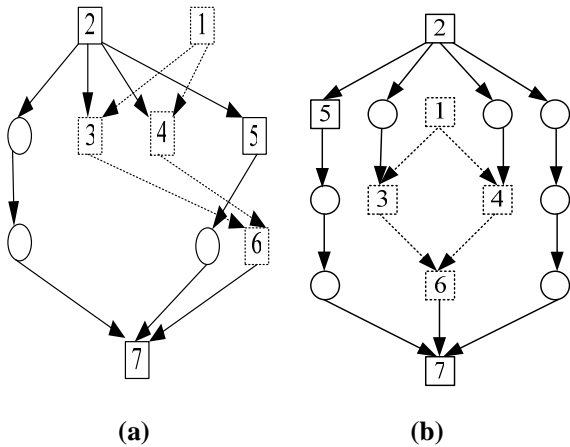


Figure 5. Recombination. (a) The parent individual A. (b) The parent individual B. (c) The offspring individual A'. (d) The offspring individual B'.

3.4 Mutation

Recombination can only find new combinations of genetic information already present. To introduce new genes into the population, mutation, i.e. the slight change of individuals, is necessary. In our algorithm, the mutation operators based on the induced sub-graphs are designed, which is different from the classic node-based operators. Two different mutation procedures are employed: the first is to “shake” the induced sub-graphs to slightly change their layers and positions as a whole. Figure 6 shows the “shake” case that the depth-3 induced sub-graph starting with node 1 in an individual is moved downwards by one layer, which eliminates the edge crossing.

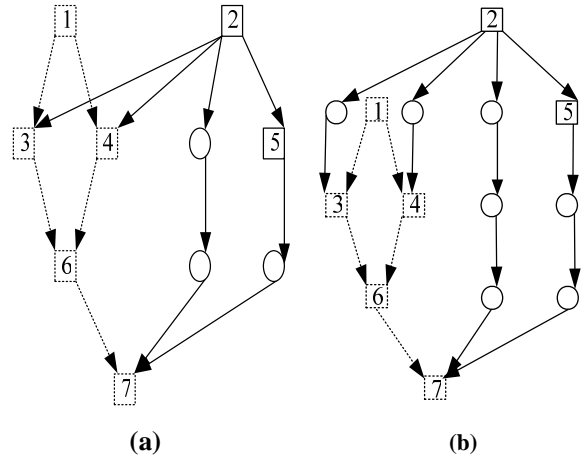


Figure 6. “Shake” mutation. (a) Original individual. (b) The offspring individual.

The second procedure is the same as the BC-Walk method proposed by Utech et al. [8], which is also a variant of the popular method to reduce the edge crossings [3, 4]. In practice, it set the position of selected real node to the arithmetic mean of the positions of its successors or predecessors. Please see [8] for more details.

4. Experimental Results

We have implemented our algorithm in Java and tested a large variety of directed acyclic graphs. The iteration number, i.e. the total generation number is appointed as the termination condition of the algorithm. Some relevant parameters are listed in Table 1:

Table 1. The parameters of our experiments.

name	value	meaning
generation_num	1000	The total generation number

individual_num	50	The total number of individuals
pc	0.7	The probability of recombination
pm	0.2	The probability of mutation

Table 2 lists a summary of test results on various DAGs. Three groups of coefficients for the aesthetic constraints are assigned. We can conclude that the coefficients of different aesthetic constraints influence the resulting layouts significantly. For some certain graphs, even the edge crossings can be eliminated.

Table 2. Fitness of various graphs and different coefficients.

Graph	Size		Aesthetic Results of Different Coefficients											
			(0.4, 0.3, 0.3)				(0.7, 0.2, 0.1)				(1.0, 0.0, 0.0)			
	V	E	C1	C2	C3	fitness	C1	C2	C3	fitness	C1	C2	C3	fitness
DAG_7 ¹	7	10	1	16	3	6.1	1	16	3	4.2	0	25	7	0.0
DAG_15 ²	15	17	2	32	6	12.2	0	42	11	9.3	0	48	18	0.0
ecosystem*	15	26	13	60	17	28.3	9	66	22	21.7	5	96	21	5.0
klayer ⁺	18	24	5	20	0	8.0	5	20	0	7.5	3	30	8	3.0
sketchsys*	30	37	21	70	0	29.4	21	70	0	28.7	21	70	0	21.0
DAG_30 ³	30	41	20	84	23	40.1	19	112	31	38.8	17	228	51	17.0
tree5	31	30	0	80	0	24.0	0	80	0	16.0	0	80	0	0.0
UNIX ⁺	41	49	7	121	24	46.3	4	121	24	29.4	2	121	24	2.0

¹ is taken from Figure 3 in [8].

^{2,3} are taken from Figure 4 in [8].

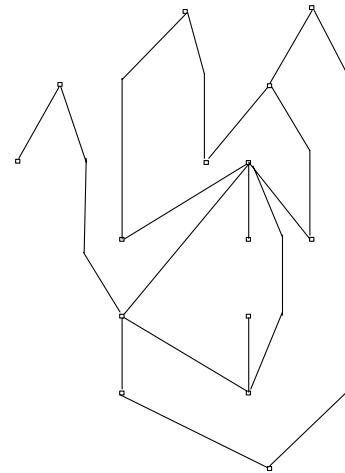
* are taken from Fig. 1 in [4].

⁺ are taken from the graphs of GDHints [12].

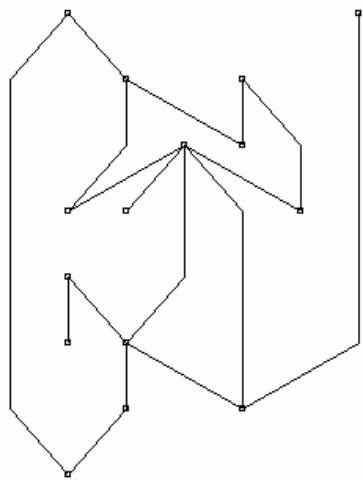
Figure 7 compares the drawings of [8] and our results corresponding to two groups of distinct coefficients. All the drawings eliminate the edge crossings; however, different results of other aesthetics influence the ultimate evaluation.

Figure 8 shows the drawing of a full binary tree. It performs quite satisfactorily, and both the hierarchical structure and the branches are conveyed apparently.

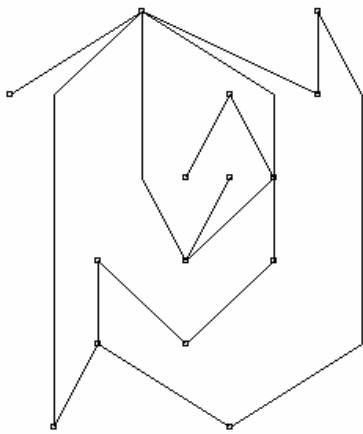
The example drawings of two real-world graphs in Figure 9 show the simple application of our algorithm. We exhibit the hierarchical structure of a predator-prey ecosystem in Figure 9(a) clearly, and obviously, it is more convincing than giving the abstract description about the predator-prey relation. Figure 9(b) reveals the version history of the UNIX system.



(a)



(b)



(c)

Figure 7. The DAG₁₅ graph. (a) Drawing from [8]. (b) Drawing corresponding to the coefficients (1.0, 0.0, 0.0). (c) Drawing corresponding to the coefficients (0.7, 0.2, 0.1).

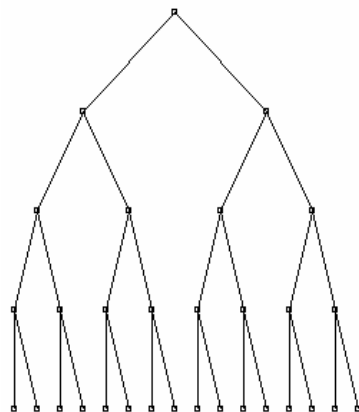
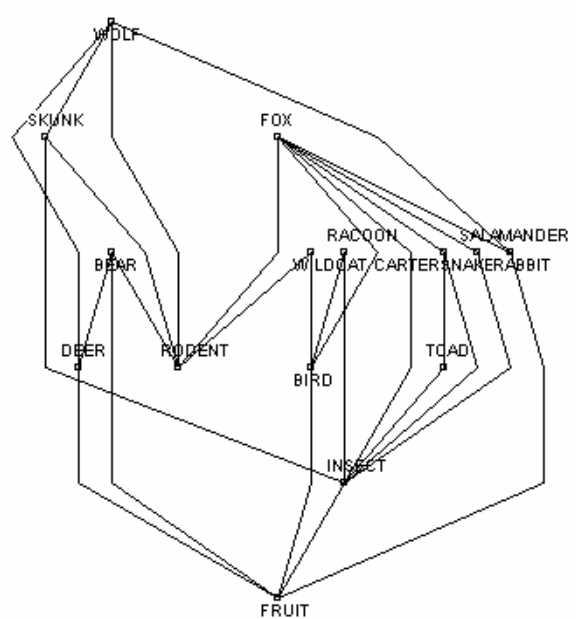
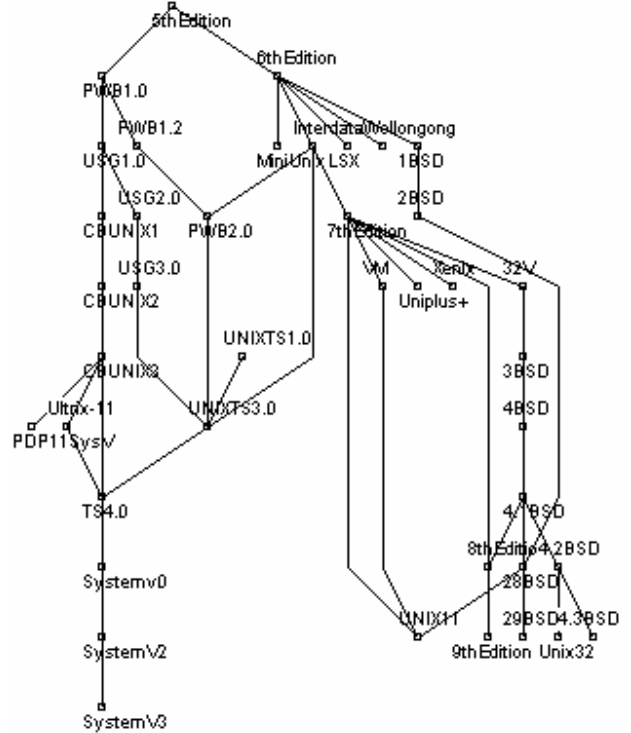


Figure 8. Example drawing of a full binary tree with the depth 5.



(a)



(b)

Figure 9. Example drawings of two real-world graphs. (a) The predator-prey ecosystem. (b) Unix System.

5. Conclusions and Future Work

We have presented a new genetic algorithm for drawing directed acyclic graphs, which introduces a new representation scheme of individuals, and designs genetic operators based on induced sub-

graphs. Experimental results on various graphs prove its validity and practicality.

Also, our experiments have suggested the influence of different coefficients about the aesthetic constraints. Increasing the corresponding weights can improve the resulting quality on certain aesthetic requirements, which makes use of the advantage of genetic algorithms.

However, the disadvantage of comparatively long running time weakens the utility of the genetic algorithms, so our algorithm should be put into use in the situation which puts more emphasis on the drawing quality. Also, we are planning to improve the efficiency of our algorithm. For example, in our algorithm, it require $O(|E|^2)$ time to compute the edge crossings, which is quite time-consuming; a faster heuristic is proposed in [11], and we expect to reduce the running time by employing this new method.

REFERENCES

- [1] Battista, G. D., Eades, P., Tamassia, R. and Tollis, I.G. 1999. Graph Drawing: Algorithms for the Visualization of Graphs. Prentice-Hall.
- [2] Kaufmann, M. and Wagner, D. (Eds.). 2001. Drawing Graphs: Methods and Models. LNCS 2025, Springer Verlag.
- [3] Sugiyama, K., Tagawa, S. and Toda, M. 1981. Methods for Visual Understanding of Hierarchical Systems. IEEE Transactions on Systems, Man and Cybernetics, 11 (2): 109 - 125.
- [4] Eades, P. and Sugiyama, K. 1990. How to Draw a Directed Graph. Journal of Information Processing, 13 (4):424-437.
- [5] Rosete, A. and Ochoa, A. 1998. Genetic Graph Drawing. Proceeding of the 13th International Conference of Applications of Artificial Intelligence in Engineering, Galway, 37-41.
- [6] Garey, M.R. and Johnson, S.J. 1983. Crossing Number is NP-hard. SIAM Journal on Algebraic and Discrete Methods, 4: 312-316.
- [7] Groves, L. J., Michalewicz, Z., Elia, P.V. and Janikow, C.Z. 1990. Genetic Algorithms for Drawing Directed Graphs. Proceedings of the Fifth International Symposium on Methodologies for Intelligent Systems, Elsevier North-Holland, 268-276.
- [8] Utech, J., Branke, J., Schmeck, H. and Eades, P. 1998. An Evolutionary Algorithm for Drawing Directed Graphs. Proceedings of the International Conference on Imaging Science, Systems and Technology, GSREA Press, 154-160.
- [9] Zhengzhi, W. and Tao B. 2000. Evolutionary Computation. Changsha, China: National University of Defense Technology Press.
- [10] Eades, P. and Kelly, D. 1986. Heuristics for Reducing Crossing in 2-Layered Networks. ARS Combinatoria, 21-A: 89-98.
- [11] Barth, W. and Mutzel, P. 2004. Simple and Efficient Bilayer Cross Counting. Journal of Graph Algorithms and Application, 8(2):179-194.
- [12] Nascimento H. A. D. do. 2003. User Hints for Optimization Processes. Doctoral Thesis. University of Sydney.