

Reviewer 1

Overall evaluation

~~This paper presents an algorithm for maximising the angle of edge crossings in a graph drawing. The authors then compare the performance of their algorithm with two others, with respect to some graph drawing aesthetic criteria, demonstrating the worth of their approach.~~

~~This is a good idea, which clearly gives improved performance.~~

- **Reviewer's Comment:** *However the approach is not novel (although it has been presented as such) - and so the paper comes across as preliminary work that needs to be rooted in a wider context in order to strengthen it.*

Response: See below.

- **Reviewer's Comment:** *The process of choosing whether to move a node ("If there exists a position among them, that does not lead to a reduction of the crossing resolution, we move the vertex to this position") is an application of a simple hill-climbing algorithm (with the same advantages and disadvantages) - this needs to be acknowledged and clarified.*

Response: Added a note stating that our approach essentially resembles a hill climbing method.

- **Reviewer's Comment:** *Or maybe the process might better be related to simulated annealing (with a temperature parameter)?*

Response: *If we make a journal version, we might consider to investigate the effect of the random movements gradually becoming smaller.*

- **Reviewer's Comment:** *The authors should consider the importance and relevance of the extensive prior work done in state-space problem solving when describing their heuristic approach. If they were to do this, then they might see the value of a composite evaluation function that would be able to take into account the other criteria discussed in section 3.1.*

Response: *This would require more extensive experiments and I would suggest to consider this approach again for a journal version.*

~~The effort taken to implement the algorithms from two other published works is acknowledged: this is a good idea (and often not easy to do). The use of standard sets of graphs (Rome and AT&T) is a very good approach in such quantitative evaluations.~~

- **Reviewer's Comment:** *Five aesthetic properties (P1-P5) are clearly defined, but never referred to again. There are only four comparative line charts in each of fig 3 and fig 4 (one of which - iterations - is not an aesthetic property). P2 and P3 do not seem to have been measured or plotted.*

Response: Removed the notation with P.

- **Reviewer's Comment:** *Despite these omissions, the evaluation process appears sound: I would have liked to have seen some examples of drawings - everything is very abstract: what did these graphs actually look like? What does a graph with a high crossing resolution look like in comparison with one with a low crossing resolution?*

Response:

- **Reviewer's Comment:** *The authors say that they made some interesting observations about the resulting drawings (e.g. "common to have adjacent edges to run almost in parallel") - some actual examples of actual graphs (before and after versions) would help elucidate these observations.*

Response:

~~In summary, this is a good idea that needs to be presented within the broader context of state space search, and with clear examples to demonstrate its success anecdotally—the quantitative evaluation is fine, but a bit dry without some concrete examples.~~

Reviewer 2

Overall evaluation

Motivated by an unsuccessful performance at last year's automated GD contest, the authors designed, implemented and experimentally evaluated heuristic algorithms for optimizing crossing, angular and total (combined) resolution. Thus, the value being optimized is the size of a smallest angle (the bigger the better) created by an edge crossing and a pair of consecutive edges in the rotation at a vertex, resp., in straight-line drawings of graphs. The authors compared the performance of their algorithm on Rome graphs [11] and AT&T graphs with the two other approaches by Argyriou et al. [5] and Huang et al. [32], both of which are force-directed unlike the heuristic proposed in the paper.

Due to its origin, the studied problem is, of course, a good fit for GD conference.

- **Reviewer's Comment:** *Although it perhaps makes more sense to choose the parameter being optimized in a more robust way, e.g., by taking also the average and variance of angle crossings into account.*

Response: *This might be an interesting direction to study if we are going to try making a journal version.*

Starting from an arbitrary straight-line drawing, the heuristic is iteratively relocating vertices in the drawing one by one while improving upon the required resolution (crossing, angular, total). The choice of the vertex in every iteration is randomized as well as the choice of its new location. In particular, a vertex is chosen uniformly at random according to the distance to the vertices in the so-called vertex pool. The vertex pool consists either of the vertices whose adjacent edge(s) are participating in a crossing with the smallest angle, or of the whole vertex set. Then several new possible locations for the picked vertex, also chosen randomly, are evaluated w.r.t. to the resolution out of which at most one is chosen. In order to escape from a local maximum, the algorithms occasionally increase the size of the vertex pool to include all the vertices.

The proposed heuristic, in spite of its simplicity, looks reasonable. According to the experimental evaluation in the case of the crossing and total resolution, the heuristic outperformed by a considerable margin the other two algorithms by Argyriou et al. and Huang et al. on the graphs in the set. In the case of the angular resolution, the proposed heuristic was considerably better than both other algorithms only on medium-size graphs in the set, but much worse than Argyriou et al. for small graphs (up to 20 vertices), and comparable to its performance for big graphs (about 100 vertices) in the set.

As for the presentation, the paper is well-written and reads smoothly, I could spot only a couple of typos. One downside is that the authors do not discuss how to choose the parameters of the heuristic (see comments below), which makes the obtained result hard to reproduce; and the description of the algorithm is also not very clear at times.

- **Reviewer's Comment:** *I would also suggest to the authors to make their code publicly available.*

Response: Added a footnote to the github repo.

Due to the aforementioned lack of clarity I propose weak accept.

-
- **Reviewer's Comment:** *l.103: The result of Dujmovic et al. looks like folklore. Split the edge set of the drawn graph into π/α parts each of which must be planar. If this is indeed correct, you can perhaps mention it.*

Response:

- **Reviewer's Comment:** *l.141: "... weighted random selection procedure ..." Could you be more specific?*

Response: Changed to probabilistic.

- **Reviewer's Comment:** *l.143: Which distance do you have in mind? Graph theoretical, Euclidean, Manhattan?*

Response: Added the following comment: (that is the shortest path to any vertex in the vertex-pool has length i)

- **Reviewer's Comment:** *l.143-145: "If the vertex pool contains ONLY critical vertices ... Otherwise, THE VERTEX POOL CONTAINS ALL THE VERTICES and each vertex can be chosen with the same probability."*

Response: Changed.

- **Reviewer's Comment:** *l.155: " Next, we describeD ... "(remove D)*

Response: Removed.

- **Reviewer's Comment:** *l.158: How do you choose ρ ?*

Response: Added the following comment: For our algorithm, we chose $\delta_{max} = \frac{1}{2} \max\{w, h\}$ (where w and h are the width and the height of the initial drawing, respectively), $\delta_{min} = \frac{1}{100} \delta_{max}$ and $\rho = 50$.¹

- **Reviewer's Comment:** *l.165: You should discuss the choice of the parameters δ_{min} and δ_{max} .*

Response: See above.

- **Reviewer's Comment:** *l.190: "... becomes wider containing all the vertices" What do you mean? That the vertex pool becomes wider unless it is not containing all the vertices already, or that you would include all the vertices into the pool.*

Response: Changed "containing" to "by including".

- **Reviewer's Comment:** *l.193: How big is a medium-size graph?*

Response: Made a note that we consider large to be more than 100 vertices.

¹In order to further decrease the time needed for an iteration, we only computed a candidate position for 10 of the 50 rays.

- **Reviewer's Comment:** *l.207: naïve*

Response: Done.

- **Reviewer's Comment:** *l.235: Similarly*

Response: Done.

- **Reviewer's Comment:** *l.394: " ... helped IN significantly ... " (remove IN)*

Response: Removed.

Reviewer 3

Overall evaluation

This paper presents a novel heuristic algorithm for maximizing the crossing resolution of a graph layout, which is defined to be the maximum angle of all the minimum angles formed by any two crossing edges. The idea here is to randomly displace the end vertices of the crossing edges that define the crossing resolution of the entire graph layout for exploring the better solutions that increase the crossing resolution. The algorithm first collects critical edges that define the crossing resolution. It then seeks the better positions of their end vertices by randomly moving it along a set of rays that emanate from their original positions, where the angles of the rays from the x-axis are obtained by uniformly dividing 2π radians (360 degrees). The proposed algorithm was tested through the comparison with conventional algorithms based on variants of the force directed algorithm. The experiments were conducted exhaustively, and the associated results are convincing. Overall, the paper has been well written and easy to understand. Interesting variants of the proposed formulation were discussed, as well as possible problems including unwanted traps by local minima and computational complexity issues. Thus, the paper is ready for presentation in the conference after an appropriate revision is carried out.

The authors are encouraged to address technical concerns as follows:

- **Reviewer's Comment:** *Improve the explanation at the end of the section "Complexity issues" on page 6.*

The last paragraph on page 6 describes the computational complexity of the proposed algorithm for computing edge crossings. This is basically acceptable, while still a bit hard to understand. I believe that the description can be further improved by inserting some visual figures.

Response: Refined the text a bit. No new figure.

- **Reviewer's Comment:** *Categorization of aesthetic properties for evaluating the present algorithm*

Section 4 describes that the present algorithm was compared with existing ones in terms of five aesthetic properties: (P1) crossing resolution, (P2) total resolution, (P3) angular resolution, (P4) aspect ratio, and (P5) number of crossings. However, the first three were employed as the measure for adjusting the algorithms, while the last two were evaluated in the respective experiments. Thus, just listing these five properties looks unnatural. The authors are encouraged to categorize important evaluation properties appropriately from the beginning.

Response: Pointed out that P4 and P5 are additionally measured.

- **Reviewer's Comment:** *Aspect ratio control in the proposed approach*

I understand that the present algorithm may significantly distort the aspect ratio of the drawing areas if we do not impose any specific constraints over the displacement of graph vertices. However, just restricting the positions of vertices within the permitted range of the aspect ratio is quite naive and may be excessively restrictive. The authors are encouraged to discuss possible strategies to effectively control the aspect ratio of the drawing area in the context of the proposed algorithm.

Response: *I guess that goes in the direction of composite evaluation function again (see comment of reviewer 1).*