Connectivity-Faithful Graph Drawing

Amyra Meidiana, *Seok-Hee Hong*, Yongcheng Jing

The University of Sydney

GD 2024

Contribution

- Introduce connectivity-faithful graph drawing

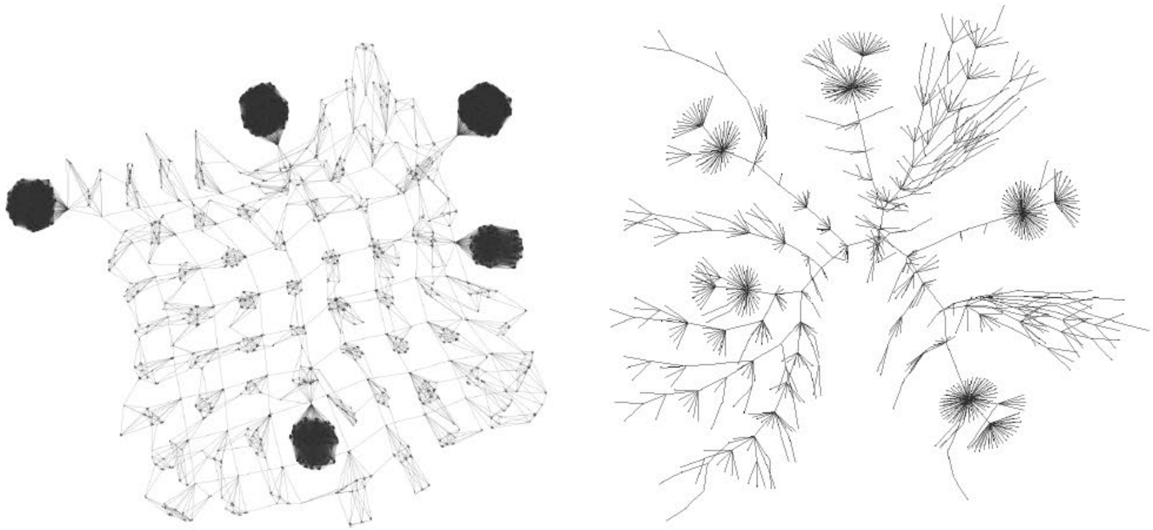
- Leverage NI (Nagamochi-Ibaraki) Algorithm

- Compute a k-connectivity preserving sparsification of a k-connected graph G
- Sampling: NI outperform SS (Spectral Sparsification)
- Drawing: NIGD computes similar quality drawing for biconnected graphs

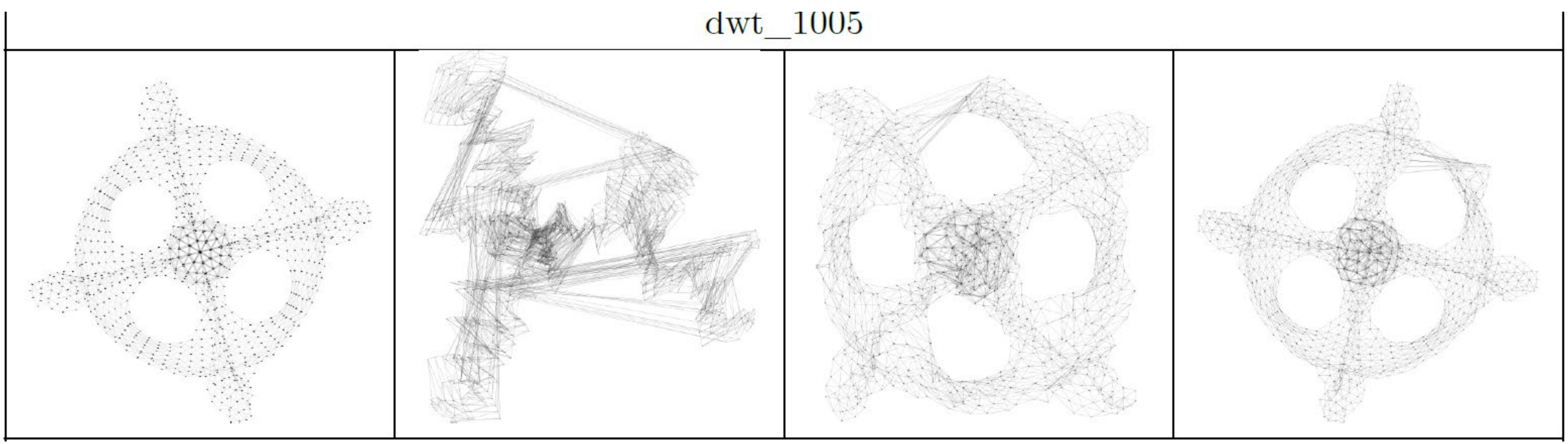
- Modification of NI Algorithm

- Preserve global k-connectivity of G and local h-connectivity of h-connected components of G
- CFNI outperforms NI: 66% improvement in sampling quality metrics and 73% improvement in proxy quality metrics.
- CFGD runs 51% faster than drawing the whole graph G with a similar quality, and outperforms NIGD, with 62% improvement in stress metrics

Example

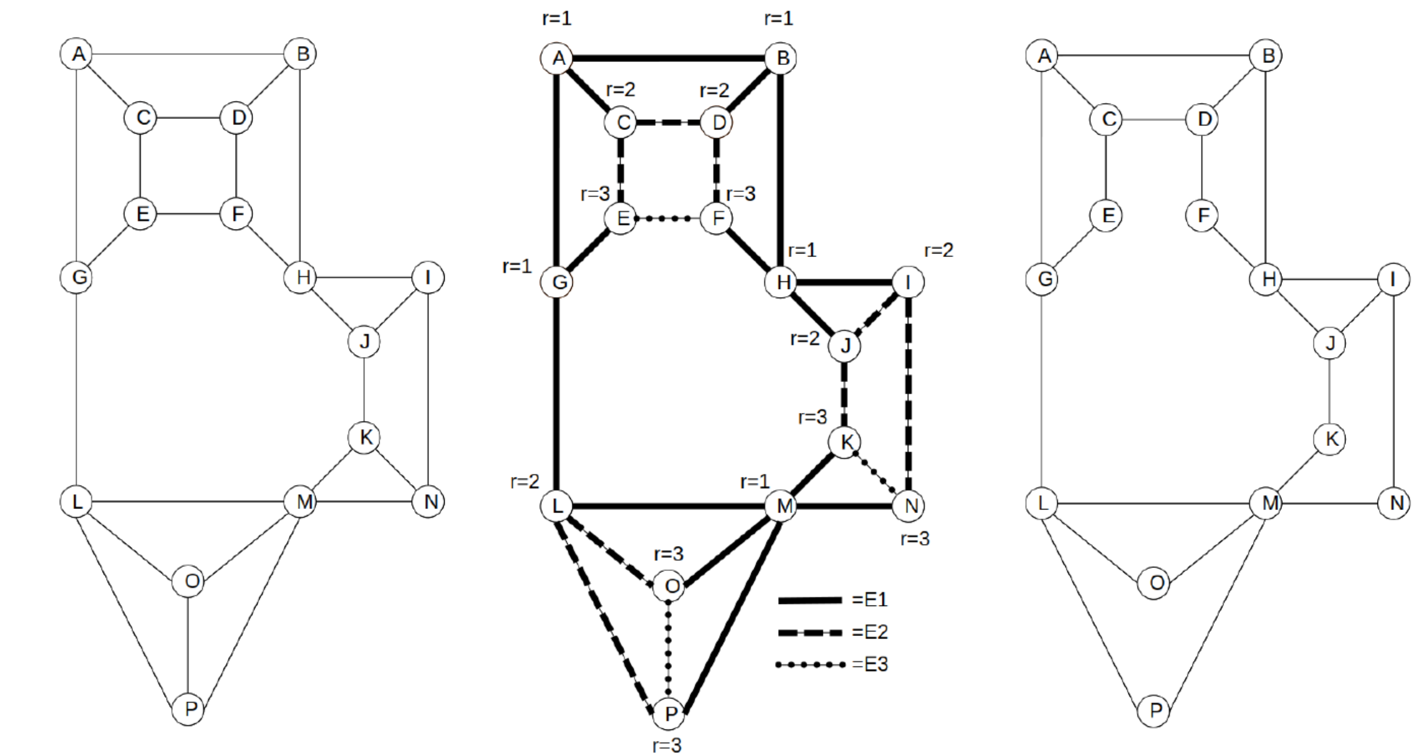


(a) G **(b)** G_1 (computed by NI) **(c)** G_3 (computed by CFNI)



NI Algorithm

Computes a k-connectivity preserving sparsification of a k-connected graph G



(a) Biconnected graph G **(b)** G after running NI **(c)** Biconnected subgraph G_k

NIGD Algorithm

NIGD

Step 1: Compute k-connected spanning subgraph $G_k = (V, E')$ of $G = (V, E)$ using NI.

Step 2: Apply a graph layout algorithm to G_k to obtain drawing D_{G_k} of G_k.

Step 3: Add all edges in $E_r = E \setminus E'$ to D_{G_k} to obtain drawing $D_{G_k+E_r}$ of G.

NI vs. SS Comparison Experiment

(a) Scale-free

G	$\|V\|$	$\|E\|$
jazz	193	2737
EVA	234	373
migrations	817	4152
oflights	1920	14458
tvcg	1925	7163
block_2000	2000	9912
CA-GrQc	2651	10480
us_powergrid	3040	4555
as19990606	3213	7937
facebook	3698	85963

(b) Mesh

G	$\|V\|$	$\|E\|$
mesh3e1	289	800
dwt_1005	1005	3808
cage8	1015	4994
bcsstk09	1083	8677
nasa1824	1824	18692
plat1919	1919	15240
sierpinski3d	2050	6144
data	2851	15093
3elt	4720	13722

(c) GION

G	$\|V\|$	$\|E\|$
2_gion	1159	6424
5_gion	1748	13957
6_gion	1785	20459
7_gion	3010	41757
8_gion	4924	52502
4_gion	5953	186279
1_gion	5452	118404
3_gion	7885	427406

(d) Black-hole

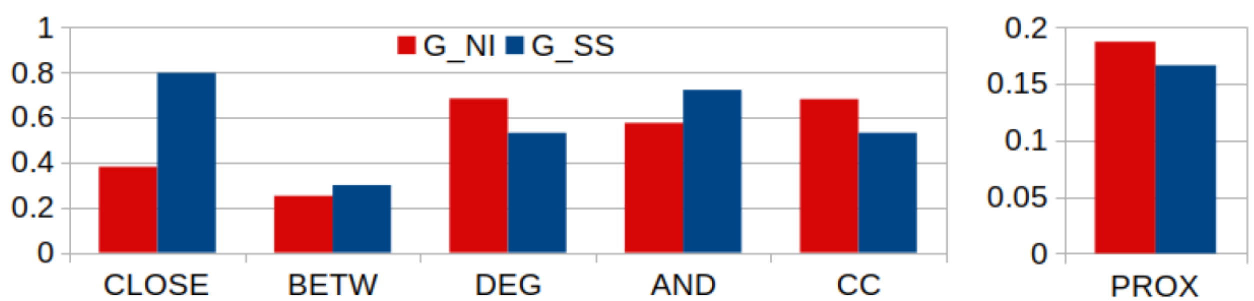
G	$\|V\|$	$\|E\|$
G443	285	2009
Cycle759	377	4790
G462	733	62509
Cycle907	823	14995
Cycle896	1031	22638
G500	1080	17636
G887	4784	38135

Quality Metrics Comparison

Sampling quality metrics: quality of samples

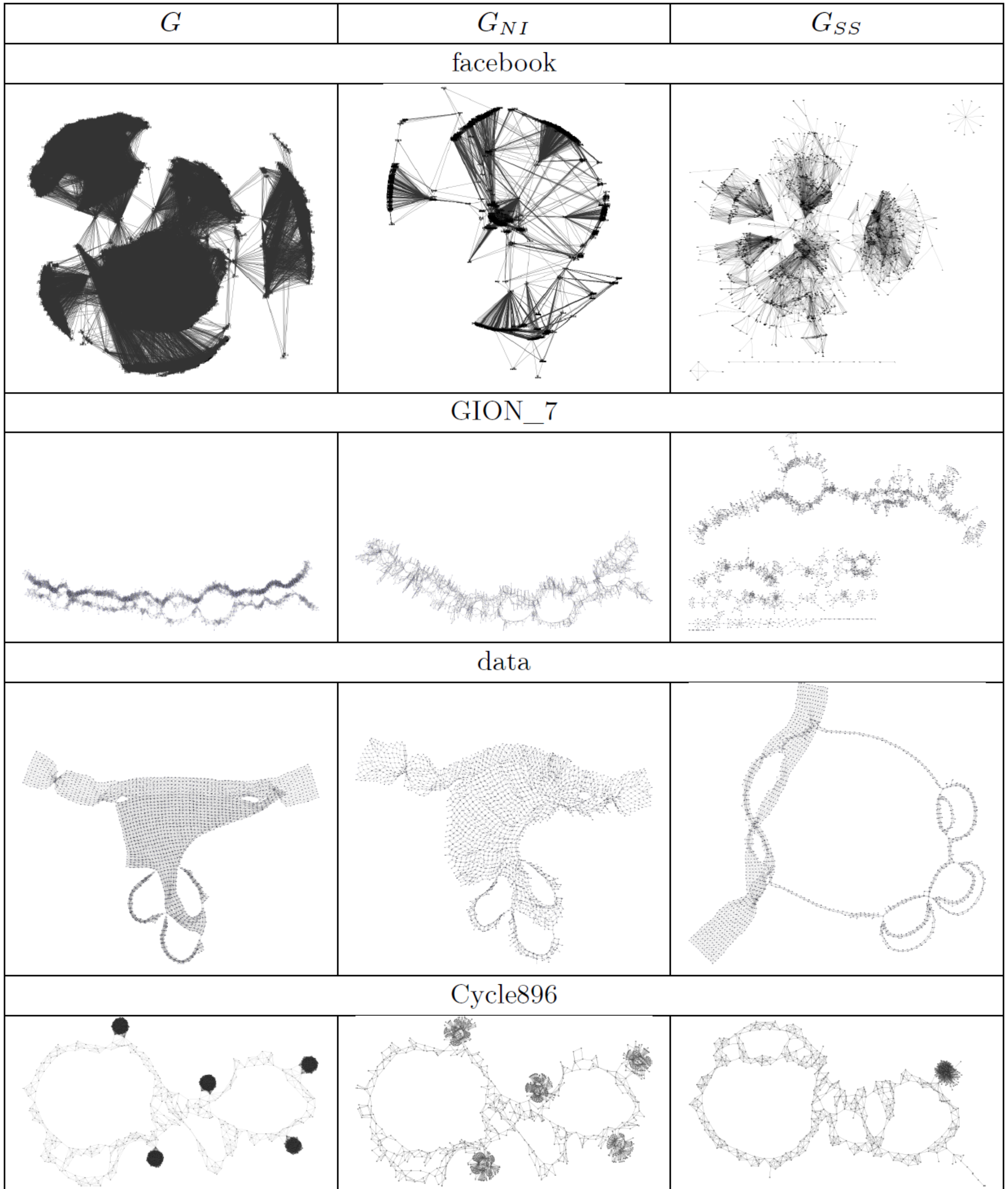
- Closeness centrality
- Betweenness centrality
- Degree Correlation
- Average Neighbor Degree
- Clustering Coefficient

Proxy quality metrics: faithfulnesss of shapes in drawings of samples



(a) Sampling quality metrics **(b)** Proxy quality

Figure 8 Average sampling quality metrics (lower is better) and proxy quality metrics (higher is better) for NI and SS. NI obtains better metrics especially on connectivity-related quality metrics CLOSE and BETW, as well as on AND and proxy quality metrics.


G
G_{NI}
G_{SS}
facebook
GION_7
data
Cycle896

NIGD Experiment: Quality Metrics

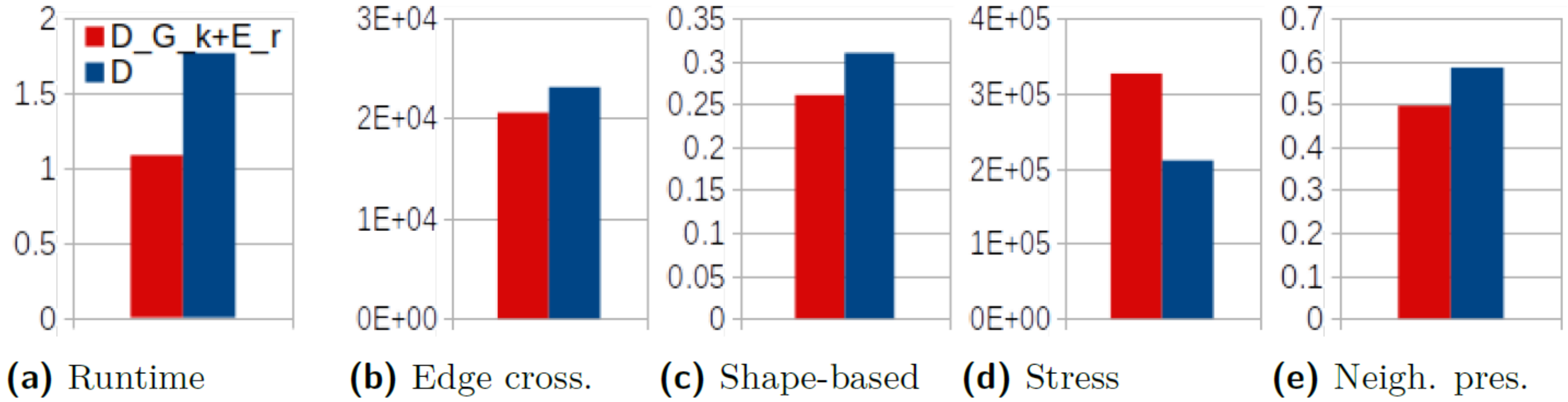
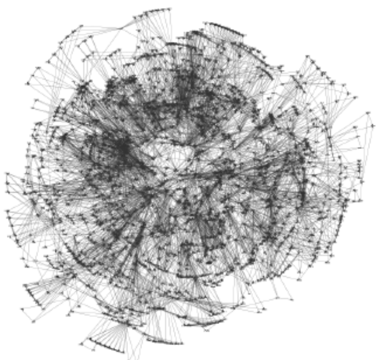
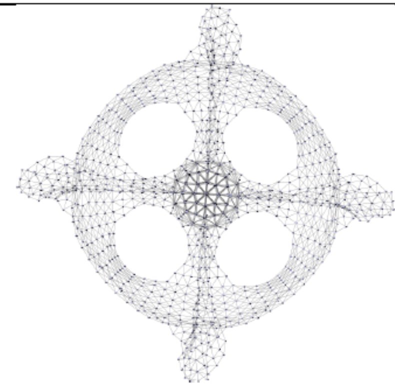
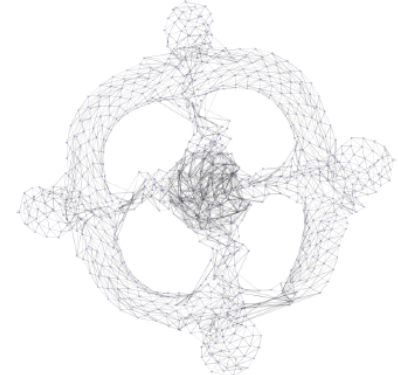
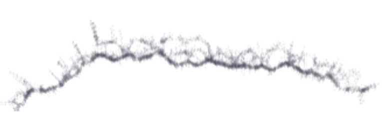
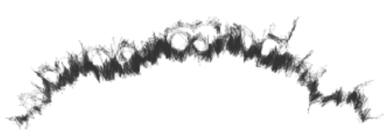
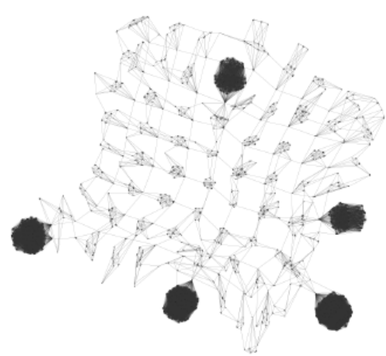
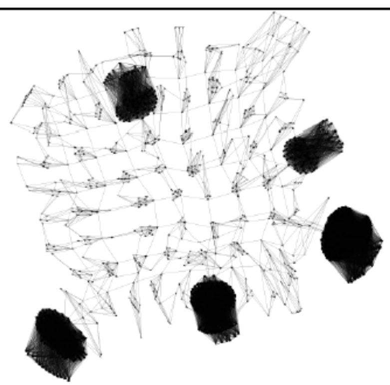


Figure 9 Average runtime and quality metrics obtained by drawing $D_{G_k+E_r}$ by NIGD and drawing D directly from G. NIGD runs significantly faster than drawing G directly, at 30% faster, with 11% better edge crossing and only 15% lower shape-based metrics and neighborhood preservation, however with on average 55% higher stress.

NIGD Experiment

<table>
<tr><th>D</th><th>$D_{G_k+E_r}$</th><th>D</th><th>$D_{G_k+E_r}$</th></tr>
<tr><td colspan="2">tvcg</td><td colspan="2">dwt_1005</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="2">GION_8</td><td colspan="2">G500</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
</table>

CFNI Algorithm

- divide-and-conquer algorithm
- computes a sparsification, preserving the global k-connectivity of a graph G and the local h-connectivity of h-connected components of G
- Example: 1-connected graph G

- G1: computed by NI
- G2: run NI for each biconnected component
- G3: run NI for each triconnected components

CFGD Algorithm

Algorithm CFGD

Step 1: Compute subgraph $G_{CFNI} = (V, E'_h)$ preserving global k-connectivity and local h-connectivity of k-connected graph G using CFNI.

Step 2: Compute a drawing $D_{G_{CFNI}}$ of G_{CFNI} using a graph drawing algorithm.

Step 3: Add all edges in $E_{r_h} = E \setminus E'_h$ to $D_{G_{CFNI}}$ to obtain a drawing D of G.

CFNI vs. NI Comparison Experiment

(a) Scale-free

G	$\|V\|$	$\|E\|$
soc_h	2000	16097
block_2000	2000	3992
oflights	2905	15645
tvcg	3213	10140
facebook	4039	88234
CA-GrQc	4158	13422
EVA	4475	4652
us_powergrid	4941	6594
as19990606	5188	9930
migrations	6025	9378
lastfm_asia	7624	27806

(b) Mesh

G	$\|V\|$	$\|E\|$
dwt_1005	1005	4813
cage8	1015	4994
bcsstk09	1083	8677
nasa1824	1824	18692
plat1919	1919	15240
sierpinski3d	2050	6144
data	2851	15093
3elt	4720	13722

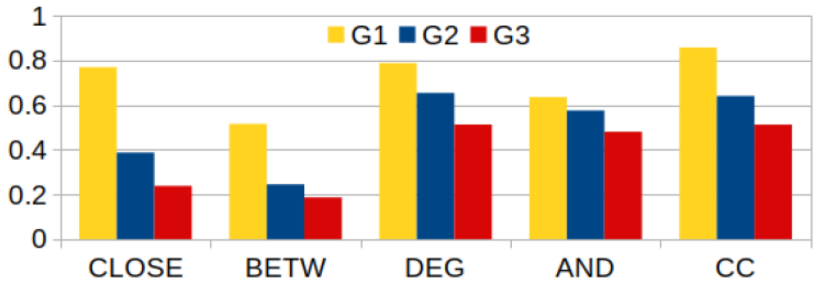
(c) GION

G	$\|V\|$	$\|E\|$
2_gion	1159	6424
5_gion	1748	13957
6_gion	1785	20459
7_gion	3010	41757
8_gion	4924	52502
4_gion	5953	186279
1_gion	5452	118404
3_gion	7885	427406

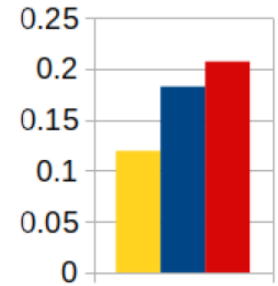
(d) Black-hole

G	$\|V\|$	$\|E\|$
G443	285	2009
Cycle759	377	4790
G462	733	62509
Cycle907	823	14995
Cycle896	1031	22638
G500	1080	17636
G887	4784	38135

Quality Metrics Comparison

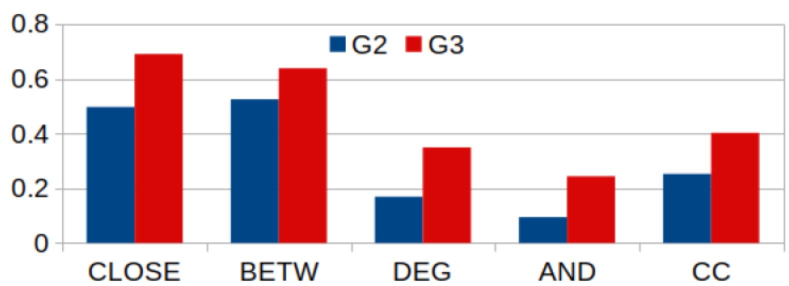


(a) Sampling quality metrics

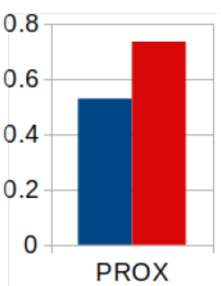


(b) Prox. qual.

Figure 3 Average sampling (lower = better) and proxy quality metrics (higher = better) for G_1, G_2, and G_3. On average, G_3 obtains significantly better metrics than G_1 (i.e., NI), especially on connectivity-related metrics CLOSE and BETW at 66% better on average.

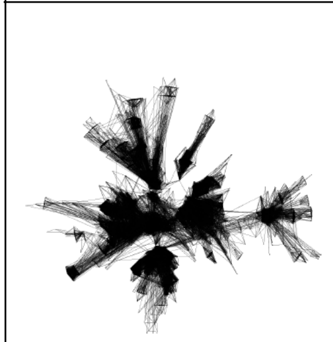
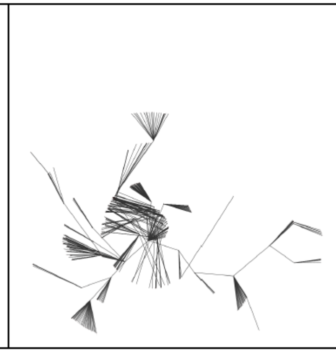
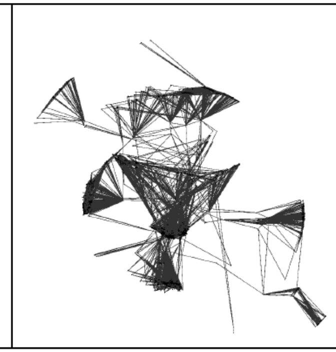
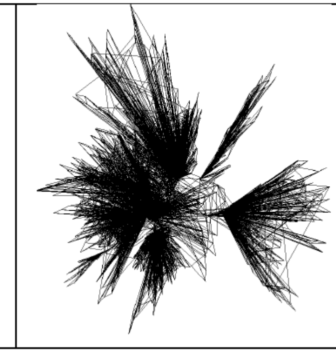
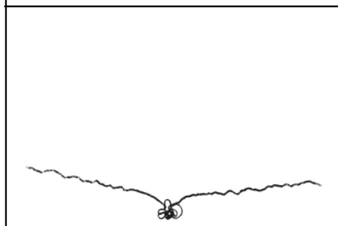
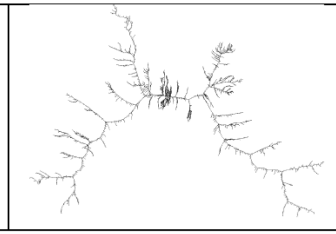
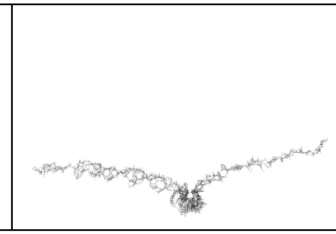
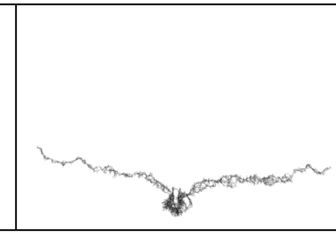
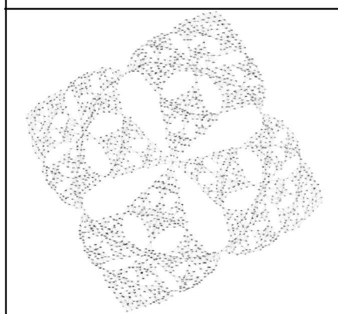
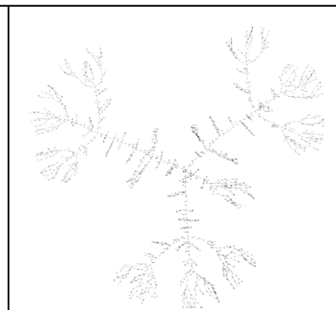
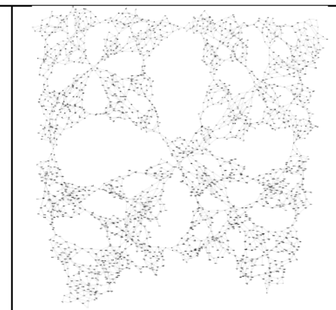
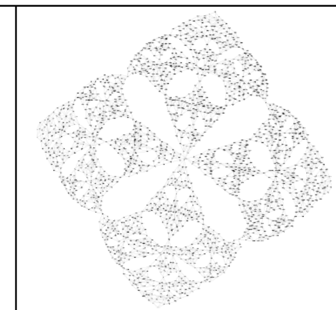
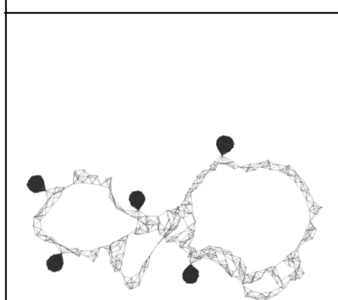
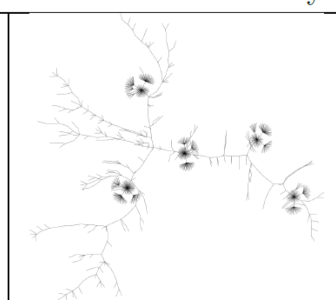
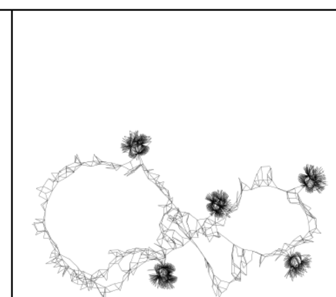
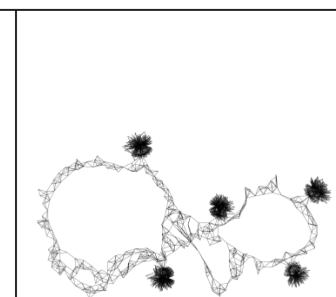


(a) Sampling metrics

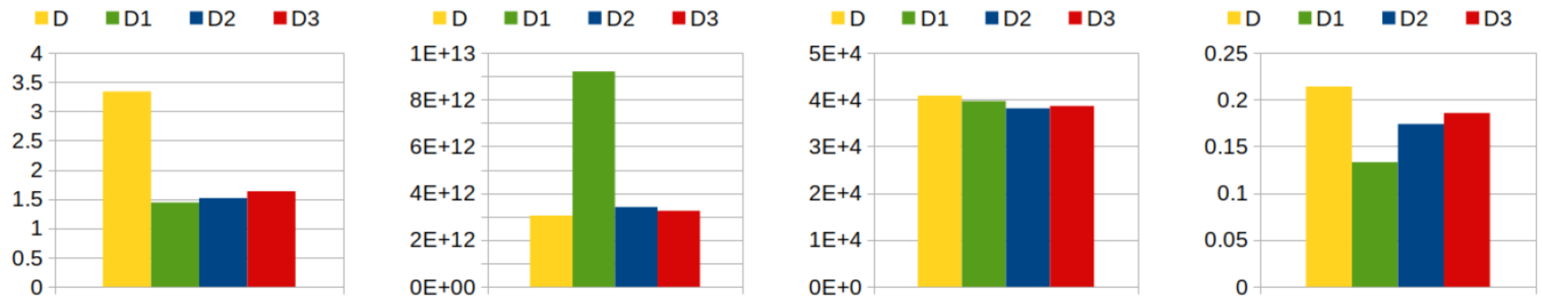


(b) Proxy metrics

Figure 4 Average improvements by G_2 and G_3 (computed by CFNI) over G_1 (computed by NI). CFNI obtains improvement over NI on all metrics, most significantly on connectivity-related sampling quality metrics CLOSE and BETW.

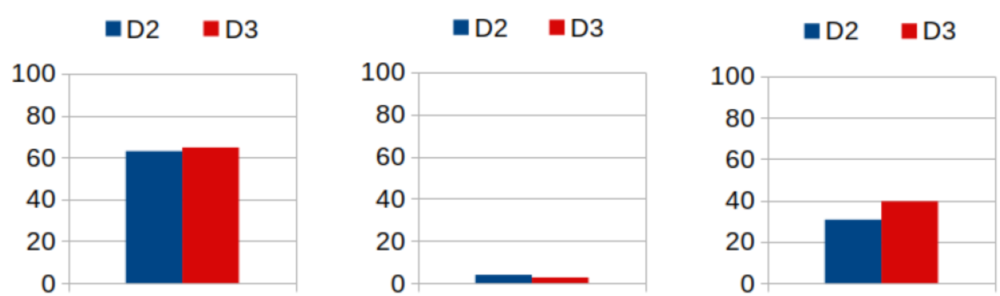
G	G_1	G_2	G_3
facebook			
			
GION_1			
			
sierpinski3d			
			
Cycle896			
			

CFGD vs. NIGD Experiment: Quality Metrics



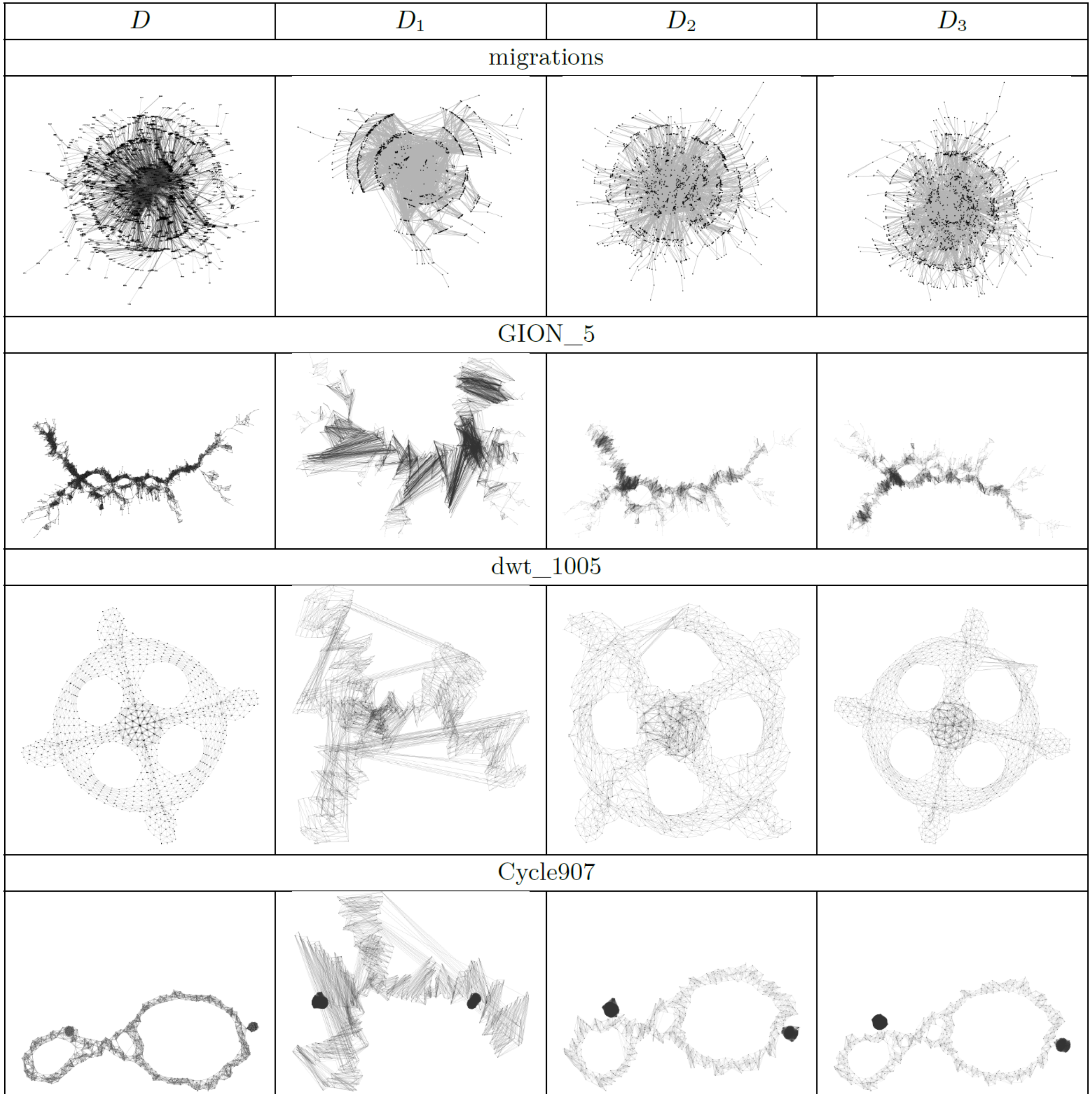
(a) Runtime **(b)** Stress **(c)** Edge crossing **(d)** Shape-based

Figure 5 Average runtime and quality metrics (lower is better for stress and edge crossing, and higher is better for shape-based) of computing D_1, D_2, and D_3 compared to computing D directly on G. CFGD (D_2 and D_3) obtains significant runtime improvements over computing D directly on G, while obtaining significantly lower stress than D_1 and similar metrics to D.



(a) Stress **(b)** Edge crossing **(c)** Shape-based

Figure 6 Average improvements (in %) in quality metrics computed by D_2 and D_3 over D_1, i.e., improvement of CFGD over a naive application of NI to graph drawing. CFGD obtains improvements on all quality metrics, with the largest improvement on stress at over 63%.


D
D_1
D_2
D_3
migrations
GION_5
dwt_1005
Cycle907

Conclusion

- Introduce connectivity-faithful graph drawing

- Leverage NI (Nagamochi-Ibaraki) Algorithm

- NI outperform SS (Spectral Sparsification), esp. Closeness metrics
- NIGD runs fast with similar quality drawing for biconnected graphs

- Improvement of NI Algorithm

- CFNI outperforms NI
- CFGD runs fast with similar quality drawing and outperforms NIGD in stress metrics