Heuristics for *t*-admissibility with complex network approach

Universidade Federal Fluminense

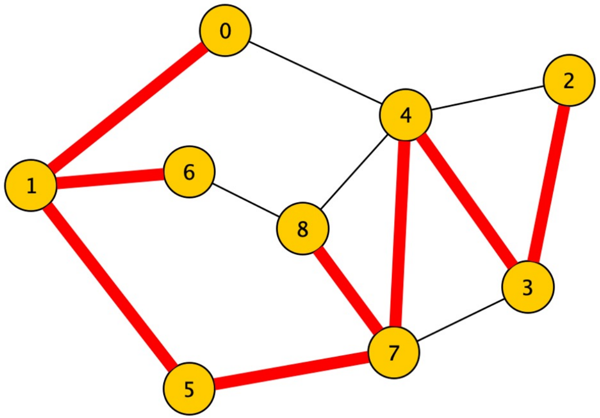
Santos, Carlos
Zudio, Anderson
Santiago, Leandro
Cunha, Luís

t-ADMISSIBILITY
DEFINITION

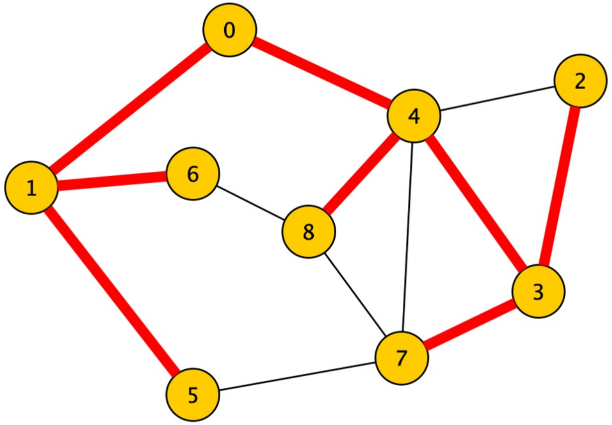
- The **t-admissibility** problem aims to decide whether a graph G has a spanning tree T in which **the distance** between any two adjacent vertices of G is **at most** t.(**stretch factor**)

- **The smallest t** for which the graph is **t-admissible**, we call **stretch index.**

t-ADMISSIBILITY
EXAMPLE

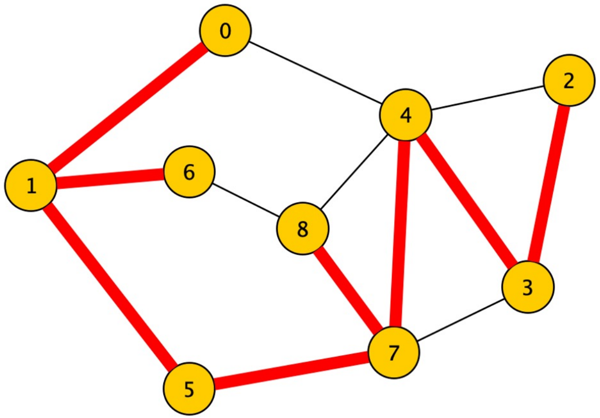


d(0,4) = 4
d(6,8) = 4
d(4,8) = 2
d(3,7) = 2
d(2,4) = 2

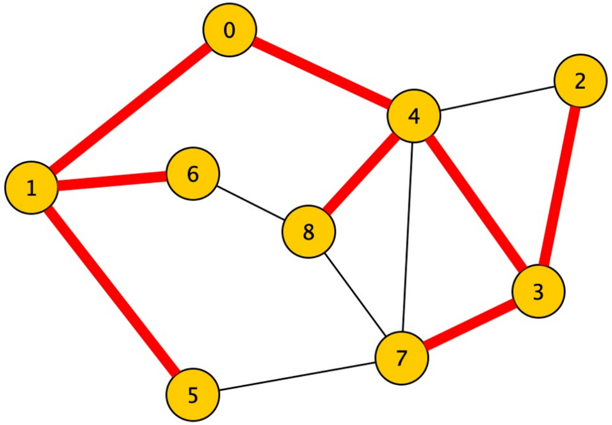


d(5,7) = 5
d(6,8) = 4
d(7,8) = 3
d(4,7) = 2
d(2,4) = 2

t-ADMISSIBILITY
EXAMPLE



d(0,4) = 4
d(6,8) = 4



d(5,7) = 5

Stretch index = 4

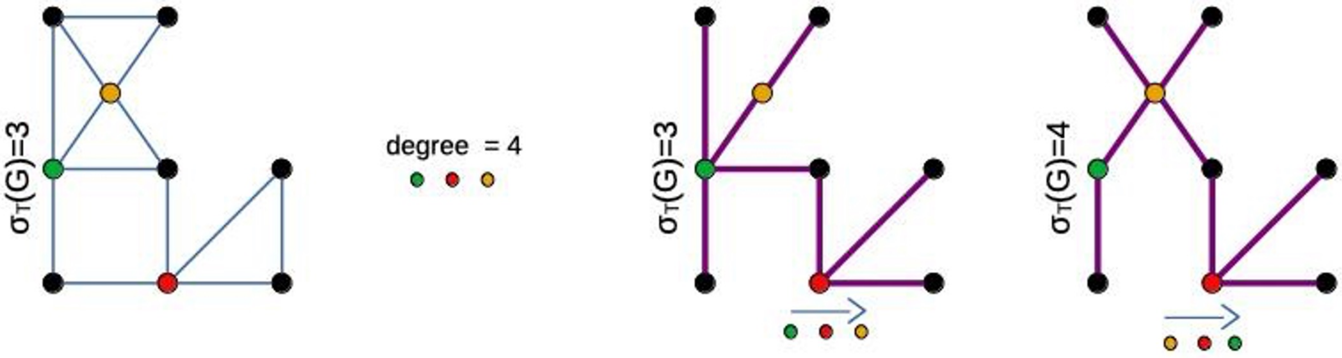
COMPLEXITY CLASSES

	Complexity
$t = 2$	**Polynomial**
$t = 3$	**Open**
$t \geq 4$	**NP-Complete**

1. **Cai and Corneil** https://doi.org/10.1137/S0895480192237403

HEURISTICS

Strategies for generating tree spanners: Algorithms, heuristics and optimal graph classes[2]



2. Couto et al https://doi.org/10.1016/j.ipl.2022.106265

MEDIDAS DE CENTRALIDADE

Degree centrality

$$D_c(v)=\sum_{u=1}^{n} A_{uv}, u \neq v$$

Leverage centrality

$$L_c(v)=\frac{1}{d(v)} \cdot \sum_{v_j \in N(v)} \frac{d(v)-d(v_j)}{d(v)+d(v_j)}$$

Closeness centrality

$$C_c(v)=\frac{1}{\sum_{u \in V(G) \setminus v} d(u,v)}$$

CONTRIBUTION

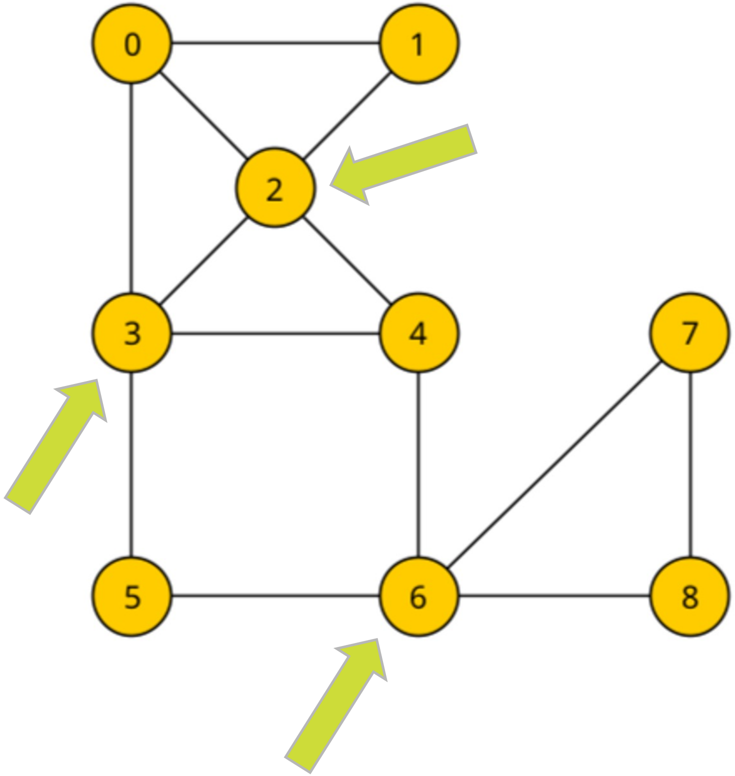
- Utilization of new centrality measures for the tie-breaking problem
- Development of 4 heuristics (2 adapted and 2 new)
- Analysis of the quality of the heuristics.

INSTRUCTIONS FOR HEURISTICS 1

- Sort the vertices by

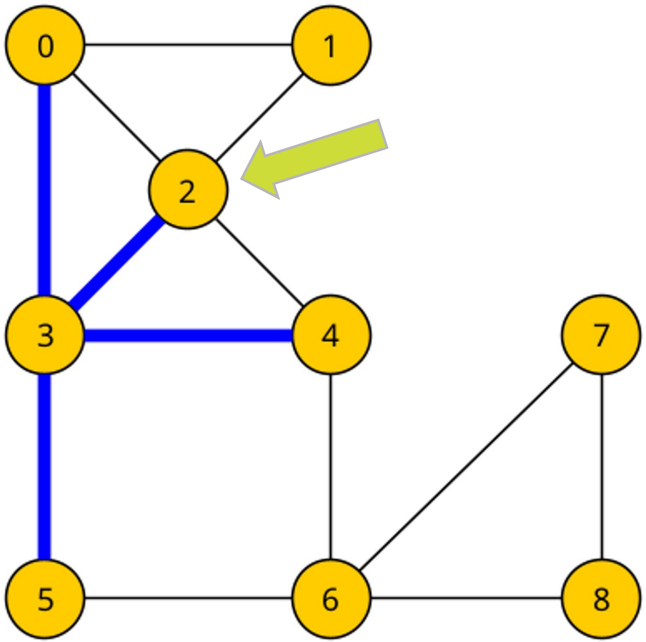
  + degree
  + closeness
  - leverage

v	d(v)	Clos.	Lev.
3	4	0,0434	0,1547
2	4	0,0434	0,1547
6	4	0,0434	0,2857
0	3	0,0370	-0,0285
4	3	0,0454	-0,1428
1	2	0,0344	-0,0266
5	2	0,0416	-0,3333
8	2	0,0344	-0,1666
7	2	0,0344	-0,1666



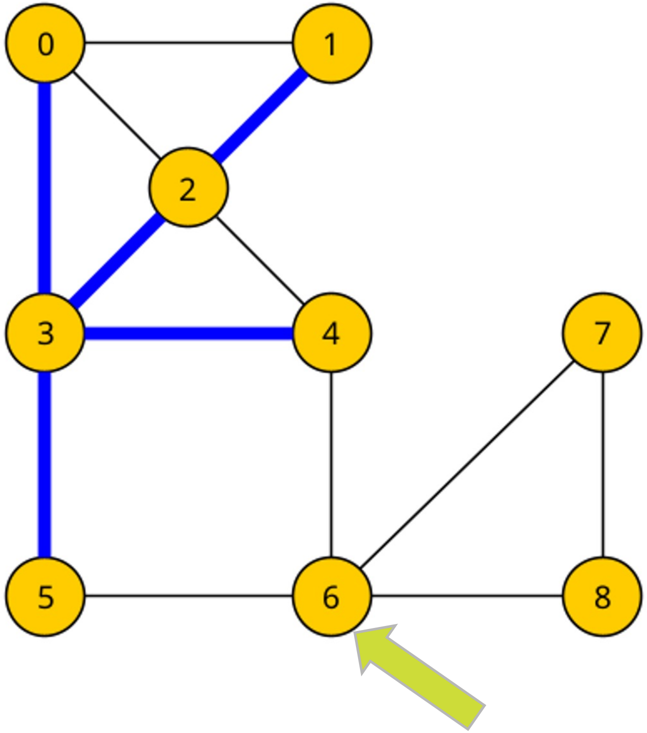
INSTRUCTIONS FOR HEURISTICS 1

3
2
6
0
4
1
5
8
7



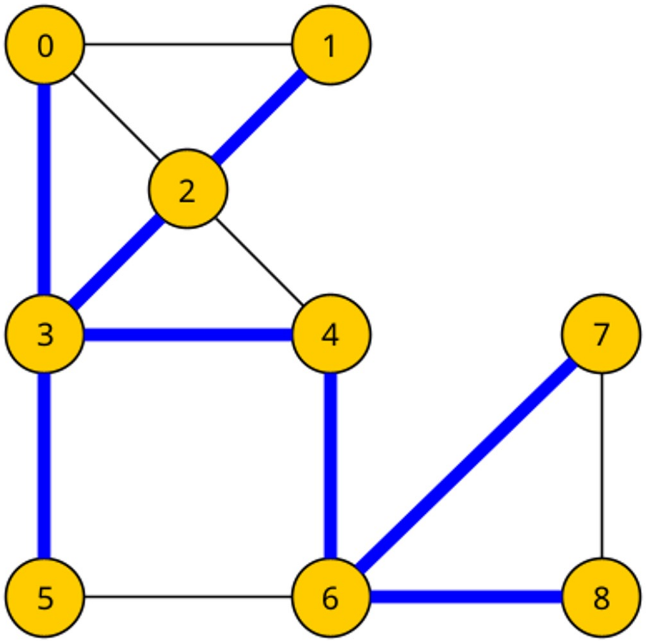
INSTRUCTIONS FOR HEURISTICS 1

3
2
6
0
4
1
5
8
7



INSTRUCTIONS FOR HEURISTICS 1

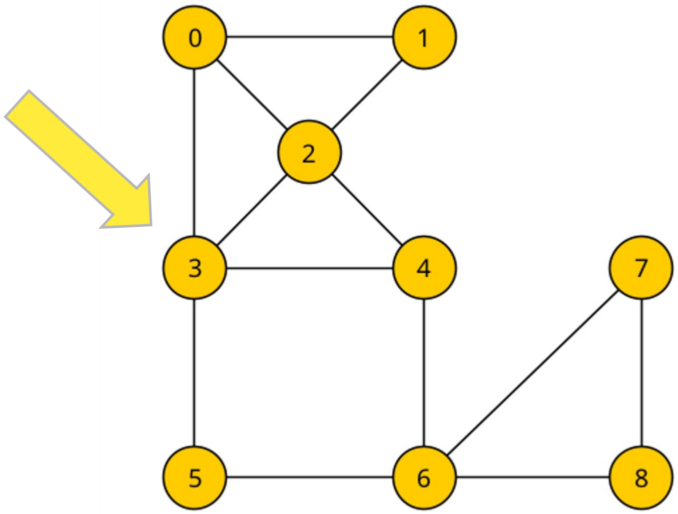
3
2
6
0
4
1
5
8
7



INSTRUCTIONS FOR HEURISTICS 2

- Sort the vertices by
  + degree
  + closeness
  - leverage

v	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2



INSTRUCTIONS FOR HEURISTICS 2

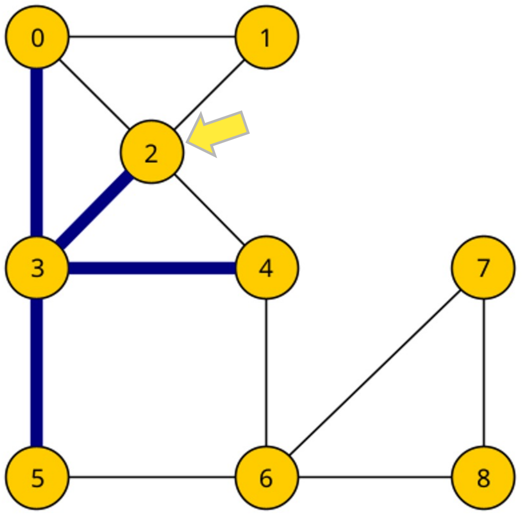
v	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2

Vertex degree in the graph

Number of neighbors of the vertex that are in the tree

v	$f(v) = d_G(v) - A_{tree}(v)$
0	1
2	1
4	1
5	1

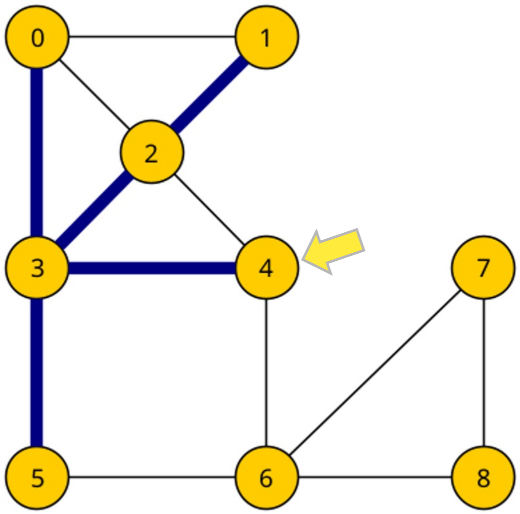
The vertex that has the ability to contribute the largest number of new neighbors to the tree



INSTRUCTIONS FOR HEURISTICS 2

v	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2

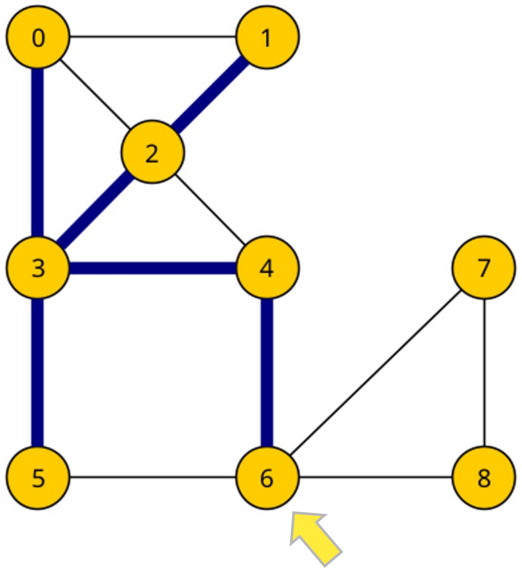
v	$f(v) = d_G(v) - A_{tree}(v)$
0	0
1	0
4	1
5	1



INSTRUCTIONS FOR HEURISTICS 2

v	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2

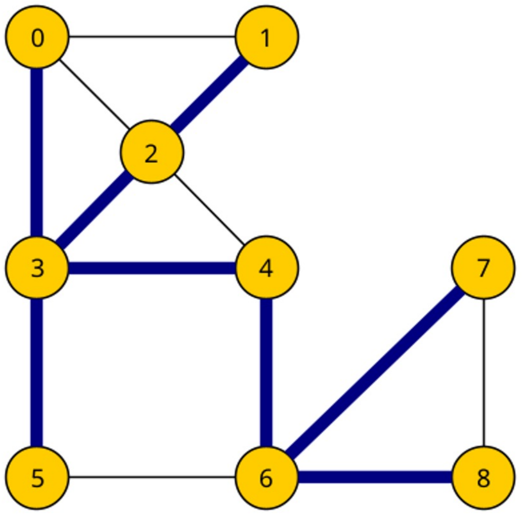
v	$f(v) = d_G(v) - A_{tree}(v)$
0	0
5	0
1	0
6	2



INSTRUCTIONS FOR HEURISTICS 2

v	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2

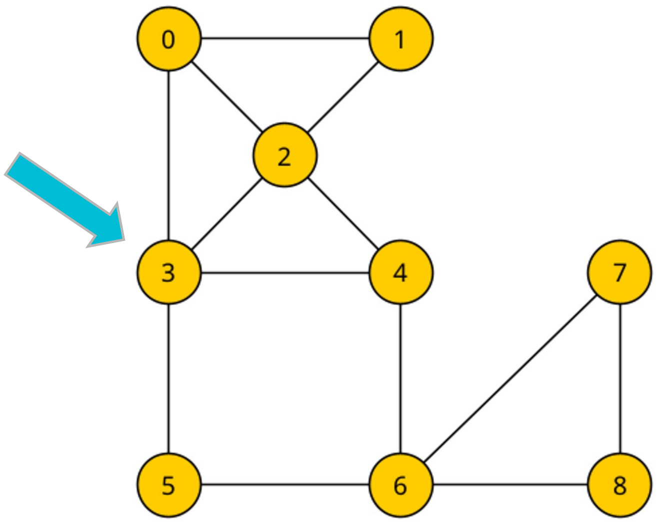
v	$f(v) = d_G(v) - A_{tree}(v)$
0	0
1	0
5	0
7	0
8	0



INSTRUCTIONS FOR HEURISTICS 3

- Sort the vertices by
  + degree
  + closeness
  - leverage

V'	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2

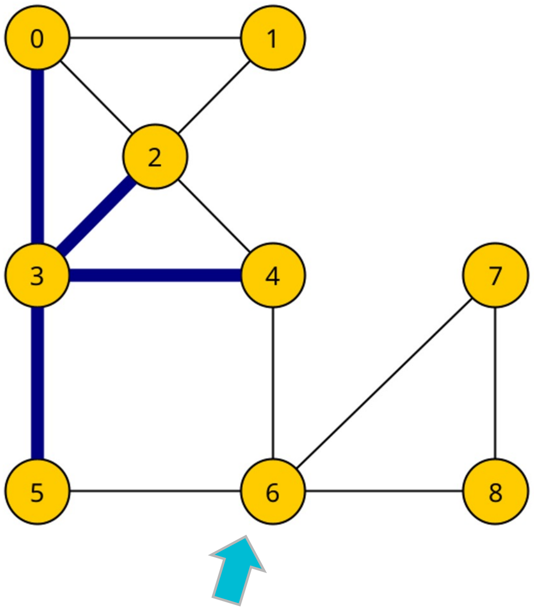


INSTRUCTIONS FOR HEURISTICS 3

V	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2

v	f(v) = $d_G(v) - A_{tree}(v)$
1	0
0	1
2	1
4	1
5	1
6	2
7	2
8	2

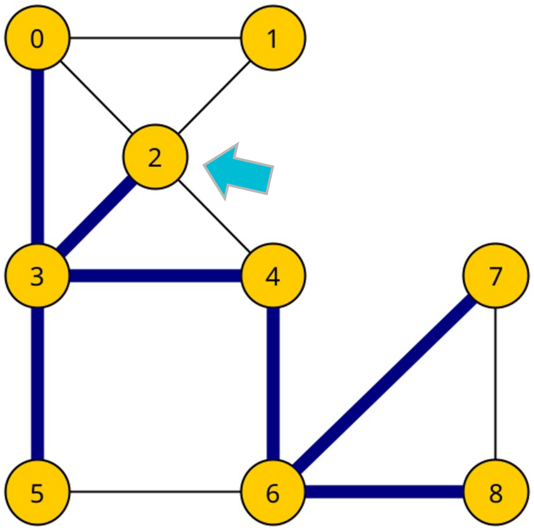
Compute for all vértices of G



INSTRUCTIONS FOR HEURISTICS 3

v	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2

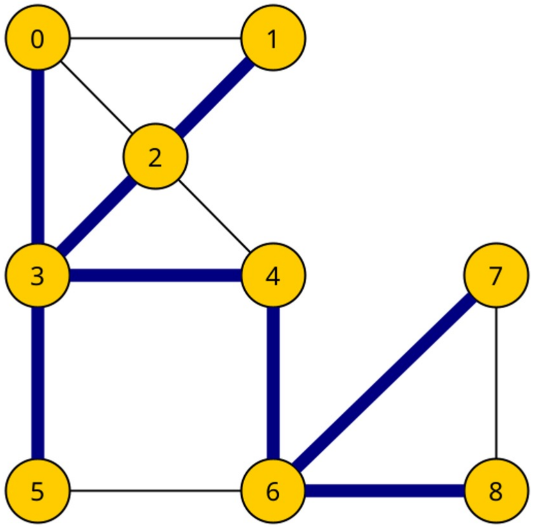
v	$f(v) = d_G(v) - A_{tree}(v)$
1	0
4	0
5	0
7	0
8	0
0	1
2	1



INSTRUCTIONS FOR HEURISTICS 3

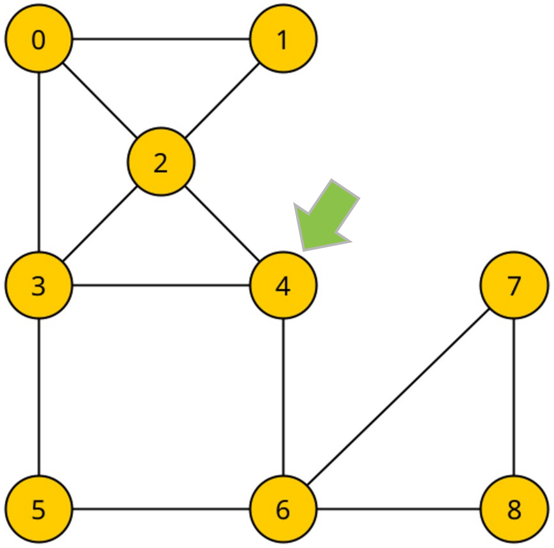
v	d(v)
3	4
2	4
6	4
0	3
4	3
1	2
5	2
8	2
7	2

v	$f(v) = d_G(v) - A_{tree}(v)$
1	0
4	0
5	0
7	0
8	0
0	0
2	0

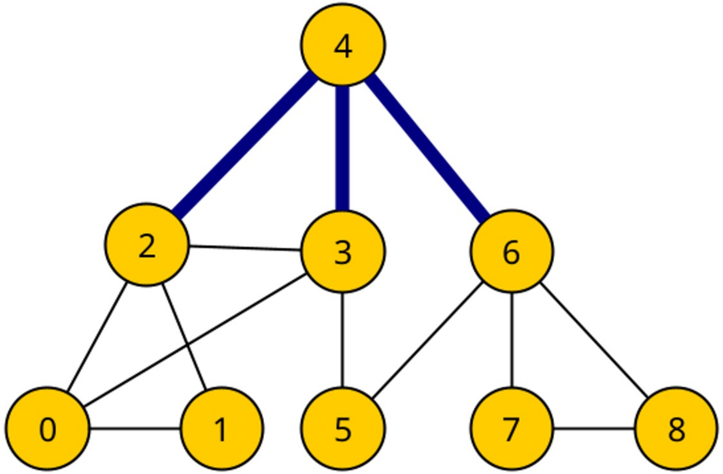


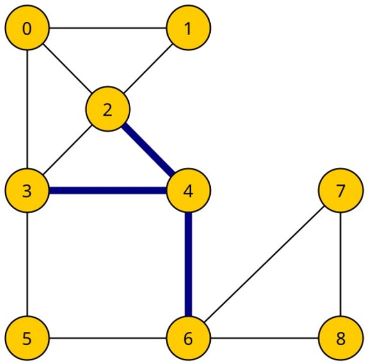
INSTRUCTIONS FOR HEURISTICS 4

v	Closeness	Leverage
0	0,0370	-0,0285
1	0,0344	-0,0266
2	0,0434	0,1547
3	0,0434	0,1547
4	**0,0454**	**-0,1428**
5	0,0416	-0,3333
6	0,0434	0,2857
7	0,0344	-0,1666
8	0,0344	-0,1666

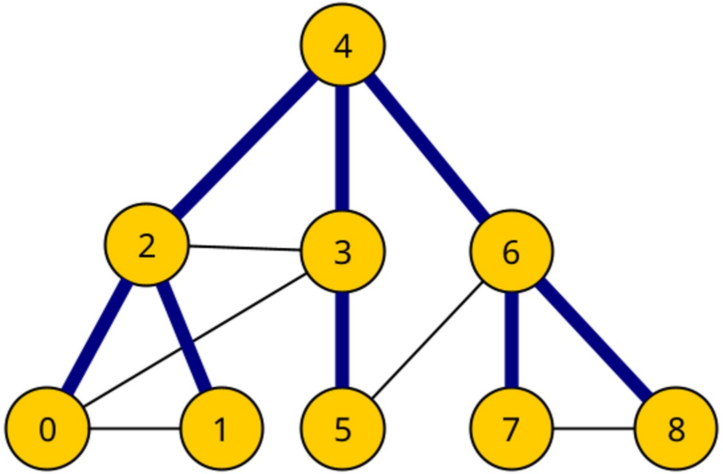


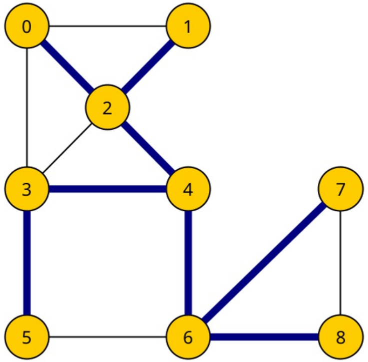
INSTRUCTIONS FOR
HEURISTICS 4





INSTRUCTIONS FOR HEURISTICS 4





COMPUTATIONAL EXPERIMENTS

- Generate 11 random graphs with 10 to 20 vertices and a maximum of 34 edges.

- Generate 400 graphs distributed between 100 and 1000 vertices from the Bipartite, Erdos, Watts, and Barabási classes.

QUALITY OF HEURISTICS

Type / Vertices	H1v1	H1v2	H2v1	H2v2	H3v1	H3v2	H4v1	H4v2r1	H4v2r3
10	**0.0**	**0.0**	**1.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
11	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
12	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
13	**0.0**	**0.0**	**0.0**	**1.0**	**1.0**	**1.0**	**0.0**	**0.0**	**0.0**
14	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
15	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
16	**0.0**	**1.0**	**0.0**	**1.0**	**0.0**	**0.0**	**0.0**	**0.0**	**0.0**
17	**0.0**	**0.0**	**0.0**	**0.0**	**3.0**	**3.0**	**0.0**	**1.0**	**0.0**
18	**2.0**	**1.0**	**1.0**	**1.0**	**2.0**	**1.0**	**0.0**	**0.0**	**0.0**
19	**0.0**	**0.0**	**1.0**	**0.0**	**1.0**	**1.0**	**0.0**	**0.0**	**0.0**
20	**1.0**	**1.0**	**2.0**	**1.0**	**2.0**	**2.0**	**0.0**	**0.0**	**0.0**
	DC	DC CC LC	DC	DC CC LC	DC	DC CC LC	DC CC LC	CC LC	CC LC

DC Degree centrality

CC Closeness centrality

LC Leverage Centrality

QUALITY FOR HEURISTIC 1

Class \ n / $Av(m)$	100.0 1.9k	200.0 7.6k	300.0 17.4k	400.0 30.5k	500.0 46.9k	600.0 68.8k	700.0 93.6k	800.0 122.3k	900.0 154.8k	1000.0 191.1k
Barabasi	0.3 1.9	0.0 2.0	0.0 2.0	0.0 2.0	0.0 2.0	0.0 2.0	0.0 2.0	0.0 2.0	0.0 2.0	0.0 2.0
Erdos	0.39 2.8	0.6 3.2	0.78 3.3	0.74 3.8	0.77 4.0	1.2 4.4	0.87 3.8	0.91 3.6	0.64 4.3	0.78 4.3
Watts	0.8 3.6	1.07 4.8	1.04 5.1	0.6 4.8	1.07 5.2	1.00 5.3	1.07 5.2	1.20 5.5	1.1 5.3	0.63 5.0
Bipartite	1.32 4.8	1.49 5.6	1.2 5.6	0.97 6.8	1.61 7.0	1.32 6.8	2.03 7.2	2.0 8.0	1.28 6.6	1.83 7.2

DC CC LC

- The values at the bottom of the cell represent the average between the heuristic's stretch factor and the lower limit value as specified in the literature.
- The values at the top of the cell represent the standard deviation.

DC Degree centrality CC Closeness centrality LC Leverage Centrality

QUALITY FOR HEURISTIC 2

Class \ n / $Av(m)$	100.0 1.9k	200.0 7.6k	300.0 17.4k	400.0 30.5k	500.0 46.9k	600.0 68.8k	700.0 93.6k	800.0 122.3k	900.0 154.8k	1000.0 191.1k
Barabasi	0.45 / 1.3	0.3 / 1.9	0.0 / 2.0	0.0 / 2.0	0.0 / 2.0	0.0 / 2.0	0.0 / 2.0	0.0 / 2.0	0.0 / 2.0	0.0 / 2.0
Erdos	0.48 / 2.4	0.67 / 2.5	0.3 / 2.9	0.53 / 2.9	0.83 / 2.9	0.8 / 3.4	0.66 / 3.4	0.39 / 3.2	0.6 / 3.2	0.6 / 3.2
Watts	0.63 / 3.0	0.6 / 3.2	0.66 / 3.6	0.63 / 3.0	0.48 / 3.6	0.78 / 3.7	0.64 / 3.7	0.8 / 3.4	0.66 / 3.4	0.64 / 3.7
Bipartite	0.79 / 4.4	1.49 / 4.4	0.97 / 4.8	1.2 / 4.4	0.91 / 4.6	0.97 / 4.8	1.0 / 5.0	1.32 / 5.2	0.91 / 4.6	1.28 / 5.4

DC CC LC

- The values at the bottom of the cell represent the average between the heuristic's stretch factor and the lower limit value as specified in the literature.
- The values at the top of the cell represent the standard deviation.

DC Degree centrality CC Closeness centrality LC Leverage Centrality

QUALITY FOR HEURISTIC 3

n / $Av(m)$ / Class	100.0 1.9k	200.0 7.6k	300.0 17.2k	400.0 30.5k	500.0 46.9k	600.0 68.8k	700.0 93.6k	800.0 122.3k	900.0 154.8k	1000.0 191.8k
Barabasi	0.78 2.7	0.5 3.5	0.80 3.6	0.45 3.7	0.0 4.0	0.6 3.8	0.0 4.0	0.0 4.0	0.3 4.1	0.0 4.0
Erdos	1.34 4.3	1.3 5.1	0.66 4.4	1.0 5.0	0.0 0.0	0.94 5.1	1.28 5.4	1.13 4.9	1.79 5.7	1.49 5.6
Watts	1.26 5.3	1.28 6.4	1.49 5.4	1.40 6.2	1.54 6.0	1.55 7.3	1.85 6.5	1.74 6.4	1.5 6.5	2.15 6.4
Bipartite	1.88 5.33	1.78 6.0	2.33 6.88	1.0 7.0	1.56 7.4	1.95 8.4	1.28 8.6	2.2 7.4	1.2 8.4	1.66 8.2

DC CC LC

- The values at the bottom of the cell represent the average between the heuristic's stretch factor and the lower limit value as specified in the literature.
- The values at the top of the cell represent the standard deviation.

DC Degree centrality CC Closeness centrality LC Leverage Centrality

QUALITY FOR HEURISTIC 4

Class \ n / Av(m)	100.0 1.9k	200.0 7.6k	300.0 17.4k	400.0 30.5k	500.0 46.9k	600.0 68.8k	700.0 93.6k	800.0 122.3k	900.0 154.8k	1000.0 191.1k
Barabasi	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0
Erdos	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0
Watts	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0
Bipartite	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0	**0.0** 2.0

DC CC LC

- The values at the bottom of the cell represent the average between the heuristic's stretch factor and the lower limit value as specified in the literature.
- The values at the top of the cell represent the standard deviation.

DC Degree centrality CC Closeness centrality LC Leverage Centrality

CONCLUSION

- The centrality measures improved the selection of vertices.
- The heuristic 4 presented trees with better solutions for the stretch factor
- We need to evaluate the heuristics with new classes of graphs.

THANKS!

GITHUB

Questions?

@cthadeusantos & cthadeusantos@gmail.com

& carlosthadeu@id.uff.br