Evolutionary Computation: CA02

2512308

1 Average and Maximum Path Length of a Graph

(a) Calculate the average path length of the following graphs (with N vertices): line, ring, star, fully connected graph. Show all your reasoning. (You may restrict to odd or even N where convenient.)

Marks: 2

1.1 Line Graphs and Path Graphs

The structure of the line graph L(G) depends on the underlying graph G. Without knowing G, the shortest average path length of a line graph with N nodes is unknown.

In a path graph, the distance between nodes i and j depends on the number of edges separating the nodes. Let i=1 (an exterior node): the distance to node j is j-1. The sum of all shortest paths is the sum of an arithmetic sequence of paths from $d_{12} = 1$ to $d_{1N} = N - 1$:

$$\Sigma_{j=1}^{N-1}(j-1) \tag{1}$$

$$= \frac{N-1}{2} \cdot (1 + N - 1) \tag{2}$$

$$= \frac{N(N-1)}{2}. \tag{3}$$

Now consider all other nodes. As one moves from i=1 to i=2, the shortest path lengths change. For example, $d_{21} = 1$ *and* $d_{23} = 1$, then lengths increase by 1 until $d_{1N} = N - 2$. Generally, the sum of $j > i$ paths decreases by 1 as i increases. The sum of $j < i$ paths increases by 1 as i increases. Consider these arithmetic progressions:

$$j > i : \frac{N(N-1)}{2} + \frac{(N-2)(N-1)}{2} + \cdots + 0 \tag{4}$$

$$j < i : 0 + 1 + \cdots + \frac{N(N-1)}{2} \tag{5}$$

Summing the first progression (4), one obtains:

$$\Sigma_{i=1}^{N-1} \frac{(N-i)(1+N-i)}{2} \tag{6}$$

$$= \frac{N(N-1)(N+1)}{6} \tag{7}$$

Double this value for the sum of both progressions:

$$= \frac{N(N-1)(N+1)}{3} \tag{8}$$

The number of pairwise paths, excluding self-loops is:

$$= N(N-1)(directed) \tag{9}$$

$$= \frac{N(N-1)}{2}(undirected) \tag{10}$$

To find the average path length (definition 1), divide the summed shortest paths (8) by the number of directed shortest paths:

$$= \frac{1}{N(N-1)} \cdot \frac{N(N-1)(N+1)}{3} \tag{11}$$

$$= \frac{N+1}{3} \tag{12}$$

1.2 Ring/Cycle Graph

Consider a cycle graph C with an even number of nodes N. Consider travelling clockwise from x_1 to x_j. The shortest path x_1x_j increases as the distance between x_1 and x_j increases until at $j = (\frac{N}{2} + 1)$:

$$x_1x_j = \frac{N}{2} \tag{13}$$

Thus x_1x_j follows an arithmetic progression:

$$\Sigma_{j=1}^{\frac{N}{2}+1}(j-1) \tag{14}$$

$$S_{\frac{N}{2}+1} = \frac{(\frac{N}{2}+1)}{2}(0 + \frac{N}{2}) \tag{15}$$

$$= \frac{N(N+2)}{8} \tag{16}$$

Thereafter, the shortest path length reduces for $j > (\frac{N}{2} + 1)$ because one travels quicker anticlockwise from x_1 to x_j. The shortest anticlockwise paths x_1x_j are a decreasing arithmetic progression as j increases:

$$\Sigma_{j=\frac{N}{2}+2}^{N}(N+1-j) \tag{17}$$

$$= \frac{\frac{N}{2}-1}{2}(\frac{N}{2} - 1 + 1) \tag{18}$$

$$= \frac{N(N-2)}{8} \tag{19}$$

Thus:

$$\Sigma_{j=1}^{N}(x_1x_j) = \frac{N(N+2)}{8} + \frac{N(N-2)}{8} \tag{20}$$

$$= \frac{2N^2}{8} \tag{21}$$

Since there are N starting nodes ($x_1x_j, x_2x_j, \ldots, x_Nx_j$), the total sum of shortest paths is:

$$= N \cdot \frac{2N^2}{8} \tag{22}$$

The number of pairwise paths, excluding self-loops, is:

$$= N(N-1)(directed) \tag{23}$$

$$= \frac{N(N-1)}{2}(undirected) \tag{24}$$

The average path length is thus:

$$= \frac{1}{N(N-1)} \cdot \frac{2N^3}{8} \tag{25}$$

$$= \frac{N^2}{4(N-1)} \tag{26}$$

1.3 Star Graph

Consider a star graph $S_{1,N-1}$ with one central node and N-1 leaf nodes. Notice the centre-leaf path length is always 1. The leaf-leaf path length is always 2 (passing through the centre). For central node $x_1 \in S_{1,N-1}$, the sum of all paths $x_1 x_{leaf}$ is:

$$\Sigma_{l=1}^{N-1}(x_1 x_l) = 2(N-1) \tag{27}$$

Now consider the lth leaf node. The path connecting l to the centre is counted in (33). Edge l connects to N-2 other nodes (excluding the centre and itself), each with path length 2. So, the sum of path lengths $x_l x_{l'}$ is:

$$\Sigma_{l'=1}^{N-2}(x_l x_{l'}) = 2(N-2) \tag{28}$$

This statement extends for all $x_l \neq x_l'$. So, the sum of directed paths between leaf nodes is:

$$\Sigma_{l=1}^{N-1}\Sigma_{l'=1}^{N-2}(x_l x_{l'}) = (N-1) \cdot 2(N-2) \tag{29}$$

$$= (N-1)2(N-2) \tag{30}$$

Adding the centre-leaf paths, one obtains:

$$(N-1)2(N-2) + 2(N-1) \tag{31}$$

$$= 2(N-1)(N-2+1) \tag{32}$$

$$= 2(N-1)^2 \tag{33}$$

The number of pairwise paths, excluding self-loops, is:

$$= N(N-1)(directed) \tag{34}$$

$$= \frac{N(N-1)}{2}(undirected) \tag{35}$$

So, the average shortest path length is:

$$\frac{1}{N(N-1)} \cdot 2(N-1)^2 = \frac{2(N-1)}{N} \tag{36}$$

1.4 Fully-connected Graph

In a fully connected graph (also known as a complete graph), each vertex is connected to every other vertex except itself. Given N vertices the shortest path between any node pair is always 1. Thus, the sum of shortest paths lengths is equal to the number of node pairs:

$$\Sigma_{i=1}^{N}\Sigma_{j=1}^{N}(x_i x_j) = N \cdot (N-1) \tag{37}$$

The number of pairwise paths, excluding self-loops, is:

$$= N(N-1)(directed) \tag{38}$$

$$= \frac{N(N-1)}{2}(undirected) \tag{39}$$

So, the average path length is:

$$\frac{1}{N(N-1)} \cdot N(N-1) = 1 \tag{40}$$

So, the average path length of a fully connected graph is always 1, regardless of the number of vertices.

(b) Identify which of these scale with N as $\mathcal{O}(1)$, $\mathcal{O}(N)$, $\mathcal{O}(N^2)$. Use the mathematical definition of the large $\mathcal{O}$ notation.

Marks: 1

1.5 Complexity

Big O notation (appendix 1) - Landau's symbol - describes the asymptotic behaviour of functions. Suppose an upwardly unbounded interval exists for the size of a graph N s.t. throughout the interval, a linearly scaled function $c \cdot f(N)$ is larger than the average path length, $c \cdot f(N) \geq APL$, then APL is at most increasing on the order $O(f(N))$.

The average path length of a path graph is:

$$\frac{N+1}{3}$$

This function scales faster than O(1). To see this, consider:

$$0 \leq \frac{N+1}{3} \leq c \cdot 1 \tag{41}$$

$$0 \leq \frac{N+1}{3} \leq c \tag{42}$$

For all $c, N \geq 0$: N=3c-2 violates this inequality, so the APL_{path} increases above the order O(1). Consider instead O(N).

$$0 \leq \frac{N+1}{3} \leq cN \tag{43}$$

$\exists c, n_0 : (47)$ holds $\forall N > N_0$. To see this, consider c=1. First, notice:

$$0 \leq \frac{N+1}{3} \leq N \tag{44}$$

$$0 \leq \frac{N+1}{3} \tag{45}$$

$$-1 \leq N \tag{46}$$

$$\frac{N+1}{3} \leq N \tag{47}$$

$$2 \leq 2N \tag{48}$$

$$1 \leq N \tag{49}$$

So for all $N \geq 1$, the second inequality holds. Thus, $N_0 = 1$ when $c = 1$. So the O(N) condition (appendix 1) obtains. APL_{path} is approximately bounded by the complexity of order N.

Applying the same arguments to the APL of ring, star, and fully connected graphs, one finds:

1. APL_{ring}'s complexity is bounded by the order $O(N)$
2. APL_{star}'s complexity is bounded by the order $O(1)$
3. $APL_{complete}$'s complexity is bounded by the order $O(1)$, when $c \geq 1$

(c) State the diameter (maximal path length) of the above graphs (a hand-written sketch of the largest path is acceptable).

Marks: 1

1.6 Diameter

The diameter of a graph is the longest, shortest path between two nodes.

1. The diameter of a line graph is $N - 1$: all edges enclosed by the exterior nodes.
2. The diameter of a ring graph with an even number of nodes N is $\frac{N}{2}$: the edges traversed between two nodes on opposite sides of the ring.
3. The diameter of the star graph is 2: the distance between any two leaf nodes.
4. The diameter of a fully connected graph is 1. As all shortest path lengths are 1, the largest shortest path is 1.

2 Adjacancy Matrix Calculations

Use the adjacency matrix (or the link list) to represent graphs G with N nodes:

Marks: 3

---

1. Path Graph:

$$A_{\text{path}} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 1 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}_{N \times N}$$

2. Ring Graph:

$$A_{\text{ring}} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 1 \\ 1 & 0 & 1 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix}_{N \times N}$$

3. Star Graph:

$$A_{\text{star}} = \begin{bmatrix} 0 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix}_{N \times N}$$

4. Fully Connected Graph:

$$A_{\text{complete}} = \begin{bmatrix} 0 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 1 & \cdots & 1 \\ 1 & 1 & 0 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 0 \end{bmatrix}_{N \times N}$$

Implement (and test, e.g. on examples from above) the following operations
(a) Calculate the average path length =: $f_1(G)$
(b) Calculate the diameter (maximal path length) =: $f_2(G)$
(c) Calculate total number of links =: $f_3(G)$
(d) Check that G is connected

You may use libraries, but would need to demonstrate their correct working, e.g., on one connected and one not connected graph, for question (c2).

(See Code file)

3 Network Mutation Operators

Implement the following operations:

(a) Add a link at a randomly chosen position (between two randomly chosen links). You will need to check the link was absent before, and eventually handle the case of a fully connected graph.
(b) Remove a randomly chosen link, provided that the graph stays connected.
(c) Random rewiring (a combination of the above)

Marks: 3

(See Code file)

4 Evolutionary Algorithm:

Implement an evolutionary algorithm that uses (with probabilities $p1, p2, p3 = 1 - p1 - p2$*) the above mutation operators, using some fitness function(s) to be specified.*

Marks: 2

Test the algorithm on maximizing $f_2(G) - f_1(G)$*, i.e., finding a graph (of fixed size N) with largest difference between maximal and average path length. Choose a graph size* $7 \leq N \leq 20$ *and display the graph (and* $f1, f_2, f_2 - f_1$*) for the three best graphs.*

Marks: 2

(See Code File)

Comment on your results: were they as expected, and why?

Marks: 1

The fitness function is the distance between the diameter and the average shortest path length. The initial population in my evolution contains 50 identical ring graphs with 10 vertices. The mutation probabilities are [p1 = 0.4, p2 = 0.4, p3 = 0.2]. The number of generations is 1000.

The first 1000 generations produces three identical solutions: three complete graphs. This result is intuitive. The complete graph has a diameter and average path length of 1, so the difference $f2 - f1$ is 0, which is low. The tendency towards three complete graphs probably arises from the relatively large probability of adding a node ($p1 = 0.4$), and indeed several non-terminal generations displayed fully connected graphs.

Despite this result, fluctuations occurred, whereby intermediary generations included varying numbers of fully connected graphs. Once fully connected, a graph can only be perturbed by removing an edge. Given all candidates are perturbed each generation, the persistence of fully connected graphs through generations is low.

The author suspects complete graphs are one of several similarly good solutions. Specifically, star graphs with a single, long axon produce low fitness scores. The authors starting belief was that a star graph would produce a low average path length and the large axon would produce a comparatively larger diameter. This, in the author's mind, would produce a negative $f2 - f1$ result. Indeed, when varying the probability vector $[p1, p2, p3]$ s.t. the 'f6_remove_link' function is most probably, one observes star-like graphs with singular long axons. These graphs appeared with {pop=50,gen=1000, p1=0.1, p2=0.8, p3=0.1}.

5 From Single to Multi-Objective Optimization

From single to multi-objective optimization: Transportation networks. We assume distances (travel times, costs) of each link to be equal and of value 1. Further, we assume revenue from tickets is independent of distance (or no revenue, if public transport is free). Further:

- *Assume transportation costs are per link, then the total cost is proportional to f1.*
- *Assume network maintenance costs are proportional to the total number of links f3*
- *Assume complaint+refund cost sare proportional to the maximal path length f2.*

(a) Use a linearly weighted fitness $a_1 f_1 + a_2 f_2 + a_3 f_3 =: f_w$ and mimimize fw for three qualitatively different choices of the weights. Display your results in a meaningful way and comment on your results.

Marks: 3

Run 1 (Parameters and Results):

1. Probabilities $[p1 = 0.33, p2 = 0.33, p3 = 0.34]$
2. Population size = 50
3. Generations = 1000
4. Weights $\{a1 = 10, a2 = 5, a3 = 1\}$

These parameters produce a set of graphs where the average number of links per node is low (appendix 7.2). One node in each graph (node 0) is connected to all other nodes, so the graphs exhibit a 'hub' and 'periphery' network type. On close inspection, each graph also contains smaller secondary hubs, like node 7 in solution 1.

These graphs minimise a joint fitness function where the average path length is twice as highly weighted as the diameter, and 10 times as highly weighted as the edge count. So, the tendency for these networks to evolve hubs and secondary spokes is intuitive. Each hub allows shorter journeys between peripheral nodes (like connecting flights in airtravel networks). We also observe secondary hubs which may reflect a the low weight on the edge count as secondary hubs are an expensive duplication of larger primary hubs.

Run 2 (Parameters and Results):

1. Probabilities $[p1 = 0.33, p2 = 0.33, p3 = 0.34]$
2. Population size = 50
3. Generations = 1000
4. Weights $\{a1 = 5, a2 = 10, a3 = 1\}$

The three best solutions are identical star graphs with node 3 at their centre. This means all nodes have only one connecting edge aside node 3 which has 9. The longest chain is therefore 2.

This result adheres to the author's expectations because star graphs have a diameter

of 2 which is the second lowest feasible score above 1. Yet, given the relatively low weight assigned to the edge count, one might expect a greater number of edges in the graph, which reduce the average path length. The author suspects the three results emerge as a result of the random graph perturbations which, once made, were a local optimum that the algorithm did not deviate from.

Run 3 (Parameters and Results):

1. Probabilities $[p1 = 0.33, p2 = 0.33, p3 = 0.34]$
2. Population size = 50
3. Generations = 1000
4. Weights $\{a1 = 1, a2 = 5, a3 = 10\}$

The final three solutions are more similar than the solutions in run 1 but more heterogeneous than the identical solutions in run 2. In each graph, 8/10 nodes connect to only one other node. Again, each graph contains a main hub (now with 8 connections). These graphs all include one node connecting a peripheral node to a hub.

Given the high weights on the edge count and diameter, the author expected a star-like graph. A star graph contains the minimal number of edges needed to connect all nodes (N). Consequently, these results aligned with the author's expectations. An unexpected feature of these graphs is the consistent display of a node connecting the hub and a single peripheral node. This result could arise due to the low average path-length weight(a_1). Given this structure, the path from all peripheral nodes to the two-part axon node is higher than in a normal star-graph.

(b) Implement a multiobjective evolutionary algorithm (of your choice), and aim at minimizing each of the three costs. Display the 10 best solutions, projected on the (f1, f2), (f1, f3), (f2, f3) planes, and a table of their f1, f2, f3 values.

Marks: 5

Network costs are an increasing function of (f_1) the average path length, (f_2) the diameter, and (f_3) the edge count of a graph. To minimise these costs, one can use multi-objective algorithms in the network literature like the NSGA-II (Non-dominated Sorting Genetic Algorithm II) (Deb et al., 2002). The accompanying code file demonstrates the algorithm's implementation, where the costs are equally weighted $a1 = a2 = a3 = 1$.

The results in appendix 7.2 display the Pareto front solutions in the planes: (f_1, f_2), (f_1, f_3), and (f_2, f_3). Further, the f_1, f_2, and f_3 values for each solution appear in the Pareto front. The NSGA-II algorithm computes the Pareto-optimal solutions for minimizing the three cost functions: f1 (diameter), f2 (average path length), and f3 (total number of links).

6 Challenge question

Choose one of the following:

(a) The resulting networks may be unrealistic due to oversimplified assumptions. Try to define and optimize a modified fitness function and discuss your results.

(b) Calculate (and try to sketch / plot as a function of p) the expected values of $f1, f2, f3, fw$ for a (N, p) random graph (you may use and properly cite results from the literature). Comment on the shape or $fw(p)$ and what we learn from this about our problem.

Marks: 2

To calculate the expected values of f_1 (diameter), f_2 (average path length), f_3 (total number of links), and f_w (weighted fitness) for an (N, p) random graph, one can use the Erdős–Rényi model. An Erdős–Rényi random graph, denoted G(N, p) has N nodes, and any node pair is connected with probability p (Gilbert, 1959),(Erdos, Renyi, et al., 1960), (Bollobas and Bollobas, 1998).

To calculate the objective functions as functions of p, consider that for an Erdős–Rényi random graph, the approximate expected values are (Bollobas, Riordan, et al., 2001):

1. Average Path Length $f_1 \approx \frac{log(N)}{log(<k>)}$, where $<k>$ is the average degree, and $<k> = (N-1) \cdot p$
2. Diameter $f_2 \approx \frac{log(N)}{log(N-1)} \cdot p$
3. Edge Count $f_3 \approx (N \cdot \frac{(N-1)}{2}) \cdot p$ (As the expected number of edges is $N_2^C \cdot p$)

As in question 5b, let the weights a1, a2, and a3 all equal 1 s.t.

$$f_w = f1 + f2 + f3.$$

Notice immediately that if the graph is not connected, APL and diameter are equal to infinity. The graph is said to be in the connected regime when the average degree of each node is high $(n-1) \cdot p >> ln(n)$.

If connected, the average path length reduces as p increases because the log function is monotonically increasing. APL reaches 1 when $p = 1$. The diameter also decreases as the graph becomes connected and the number of connecting edges increases. Like APL, the expected diameter decays exponentially as p increases. The expected edge count increases linearly in p until $\frac{N(N-1)}{p}$ when $p = 1$. The weighted fitness (f_w) is initially extremely high therefore but starts decreasing as p approaches the connectivity threshold $(n-1) \cdot p >> ln(n)$. Between this threshold and 1, f_w will increase as the tradeoff between falling APL and diameter and increasing edge count occurs.

This analysis can help understand the impact of p on the properties of a random graph. For the transportation network problem, one learns that highly connected networks are preferable to poorly connected networks and fully connected networks. Leveraging these insights, one can make informed decisions about network designs by considering the tradeoffs among diameter, average path length, and the total number of links.

References

Bollobas, Bela and Bollobas, Bela (1998). *Random graphs*. Springer.

Bollobas, Bela, Riordan, Oliver, et al. (2001). "The degree sequence of a scale-free random graph process". In: *Random Structures & Algorithms* 18.3, pp. 279–290.

Deb, Kalyanmoy et al. (2002). "A fast and elitist multiobjective genetic algorithm: NSGA-II". In: *IEEE transactions on evolutionary computation* 6.2, pp. 182–197.

Diestel, Reinhard (2018). *Graph theory*. Springer.

Erdos, Paul, Renyi, Alfred, et al. (1960). "On the evolution of random graphs". In: *Publ. Math. Inst. Hung. Acad. Sci* 5.1, pp. 17–60.

Gilbert, Edgar N (1959). "Random graphs". In: *The Annals of Mathematical Statistics* 30.4, pp. 1141–1144.

McKelvey, Robert and Nguyen, Hien (1983). "Urban Operations Research (Richard C. Larson and Amedeo R. Odoni)". In: *SIAM Review* 25.1, pp. 129–131.

Wilson, Robin J (1979). *Introduction to graph theory*. Pearson Education India.

Wong, Stephen (2020). *"big-O" notation*. URL: `https://www.clear.rice.edu/comp310/course/design/big_o.html`.

7 Appendices

7.1 Definitions

Definition 1 *Average (shortest) Path Length $\langle d \rangle$ - The arithmetic mean of all shortest paths between node pairs in a graph. The shortest path between nodes i and j is the path with the fewest number of links. The shortest path is often called the distance between nodes i and j, denoted by d_{ij}, or simply d. We can have multiple shortest paths of the same length d between a pair of nodes. The shortest path never contains loops or intersects itself. (McKelvey and Nguyen, 1983).*

Definition 2 *Line Graph - The line graph L(G) of a simple graph G is the graph whose vertices are in one-one correspondence with the edges of G, two vertices of L(G) being adjacent iff the corresponding edges of G are adjacent (Wilson, 1979).*

Definition 3 *Path Graph - A path graph is a set of interior and exterior nodes arranged in sequence s.t. interior nodes are adjacent to two immediately neighbouring nodes and two exterior nodes are adjacent to one immediately neighbouring node. (Wilson, 1979).*

Definition 4 *Ring or Cycle Graph - Consider a path graph P, and an edge connecting node i and j, $x_i x_j$. If $P = x_0 \dots x_{k-1}$ is a path and $k \geq 3$, then the graph $C := P + x_{k-1}x_0$ is a ring or cycle graph. (Diestel, 2018).*

Definition 5 *Star graph - A star graph $S_{1,N}$ is a bipartite graph with a single central node connected to N leaf nodes. $S_{1,N}$ contains no other edges.*

Definition 6 *Large O notation (Wong, 2020) - A function g(n) is O(f(n)):*

$$g(n) = O(f(n))$$

if

$$\exists c, n_0 > 0 : \forall n \geq n_0 :$$
$$0 \leq g(n) \leq cf(n)$$

Alternatively, O(f(n)) is the set of all functions h(n):

$$\exists c, n_0 > 0 :$$
$$0 <= h(n) <= cf(n) \forall n >= n0.$$

7.2 Graphs

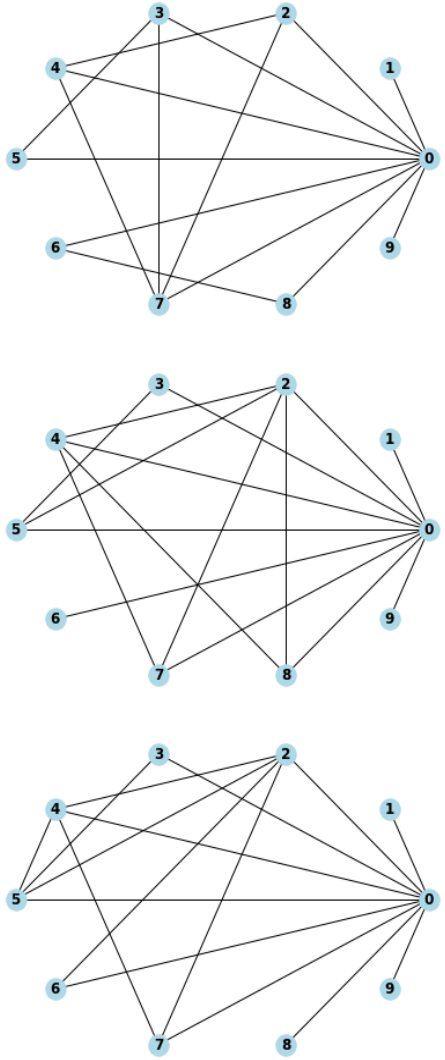

Figure 1: Q5a Run 1 Top 3 Solutions

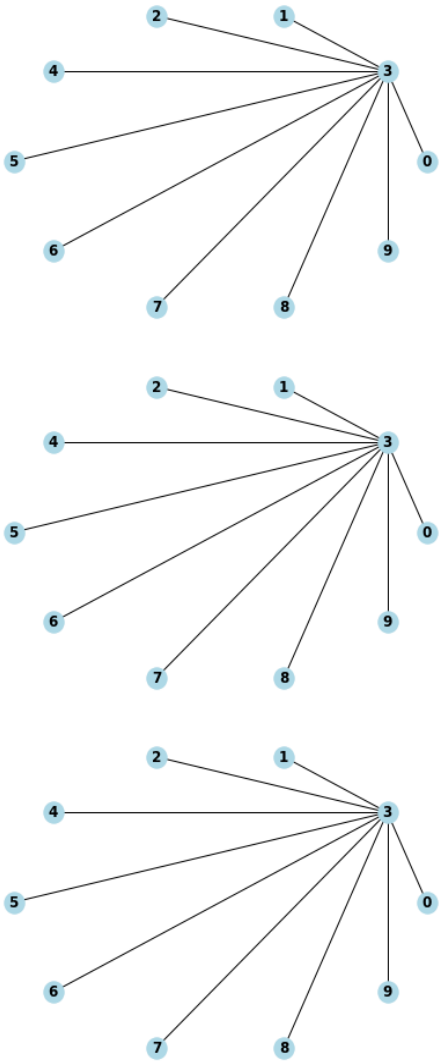

Figure 2: Q5a Run 2 Top 3 Solutions

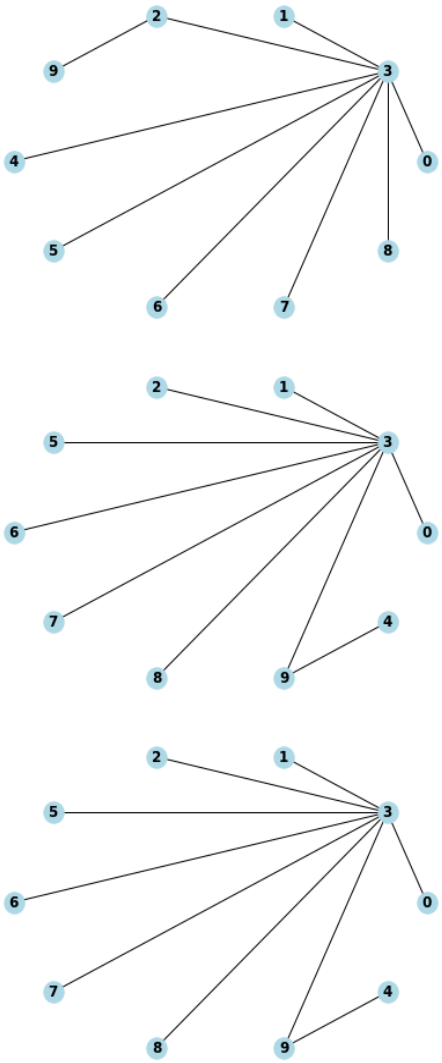

Figure 3: Q5a Run 3 Top 3 Solutions