1. **Graph Types**

In order to test the effectiveness of our techniques, we need different types of graphs based on degree distribution as a test bed. For this purpose, we handle various graph models addressing different kinds of problems.

* 1. **Erd ̋os-R ́enyi based Graphs**

First generation model is Erd ̋os-R ́enyi (ER) where probability of selecting an edge between two vertices is *p* which is independant from other edges [1].

The following graphs are generated according to ER model:

* TSP-LIB: Instances actually represent city coordinates, not the graph. We turn them into graphs via ER procedure using *p* and euclidean distance between points [6].
* ER graphs: Pure ER implementation.

In all of the graph generations above, the following parameters are applied *p*=0.3.

**1.2. Scale-Free Graphs**

ER does not cover real network assumptions. To this end, some scale-free network models are handled in the literature. We generate synthetic graphs for each of them via using their software packages.

* + 1. **Albert-Barabasi**

In contrast to ER model, real networks stem from growing number of nodes and new nodes are inclined to link to the nodes having higher degree which is known as “rich-get-richer” rule. Albert-Barabasi create their model inspired by this *growth* and *preferential attachment* phenomenon [2,4].

**1.2.2. Benchmark graphs**

Real networks are composed of community structures representing the internal node organization. However, the proposed algorithm benchmark models in the literature assume that nodes have same degree and communities have equal size. Yet, communities have different sizes according to power law and node degrees must have a skewed degree distribution. Benchmark model fill this gap [5].

**1.2.3. Forest fire model**

It is developed to challenge these two assumptions handled in Albert-Barabasi growth model: Constant average degree and slowly growing diameter. Unlike them, networks become denser with more average degree and diameters shrink as the network grows. Forest Fire Model satisfies these conditions by “burning” through existing edges when a new node attaches to an existing node [8].

* 1. OR-LIB

They represent “uncapacitated p-median problem” instances and are directly adopted from the literature with given edge weights [3,7].

1. **Test Instances**

Edge weights (w\_ij) values between nodes are set uniformly between 30 and 50 excluding TSP-LIB and OR-LIB based graphs. Gateway deployment cost f\_i is selected uniformly between 50000 and 60000 whereas node demand d\_i is picked according to Uniform (100, 200). Afterwards, all-pairs shortest path algorithm is performed on w\_ij to find t\_ij, transportation cost values. At last, given that connection cost is c\_ij, we set c\_ij = 100 w\_ij. We solve all instances using C++ calling Cplex 12.7 solver on a PC with x64 Intel(R) Core(TM) i5-3210M 2.50 GHz CPU and 8 GB RAM. We give 2 hr 20 mins time limit for each instance.

**References**

1. Geeksforgeeks, <https://www.geeksforgeeks.org/erdos-renyl-model-generating-random-graphs/> , accessed date: 27.3.24
2. Network science, <http://networksciencebook.com/chapter/5#growth>, accessed date: 27.3.24
3. ORLIB, <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/files/> , accessed date: 27.3.24
4. Albert-L ́aszl ́o Barab ́asi and R ́eka Albert. Emergence of scaling in random networks. science, 286(5439): 509–512, 1999
5. Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. Physical review E, 78(4):046110, 2008
6. Gerhard Reinelt. Tsplib95. Interdisziplin ̈ares Zentrum f ̈ur Wissenschaftliches Rechnen (IWR), Heidelberg, 338:1–16, 1995
7. John E Beasley. A note on solving large p-median problems. European Journal of Operational Research, 21 (2):270–273, 1985.
8. Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pages 177–187, 2005