Supplementary Material: Python Implementation

Jabbar Ali

Python Code for M-polynomial and Degree-Based Indices

```python
import math
import sympy as sp
import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error, r2_score

# Edge-degree partition: (du, dv) : frequency
edge_partition = {
  (1, 3): 4,
  (3, 3): 2,
  (2, 3): 7,   # combined frequency (3 + 4)
  (2, 2): 1
}

# Build M-polynomial
x, y = sp.symbols("x y", real=True)
M_poly = sum(freq * (x**du) * (y**dv)
for (du, dv), freq in edge_partition.items())

print("M-polynomial:")
sp.pprint(M_poly)

# Degree-based indices
def M1(G):
return sum((du + dv) * f for (du, dv), f in G.items())

def M2(G):
return sum((du * dv) * f for (du, dv), f in G.items())

def MZ(G):
return sum((1.0 / (du * dv)) * f for (du, dv), f in G.items())

indices = {
  "First Zagreb Index (M1)": M1(edge_partition),
  "Second Zagreb Index (M2)": M2(edge_partition),
  "Modified Zagreb Index (MZ)": MZ(edge_partition),
}

print("\nTopological Indices:")
for name, val in indices.items():
print(f"{name}: {val:.6f}")
```

Output Generated by the Python Script

Executing the above Python code produces the following M-polynomial corresponding to the given edge-degree partition:

$$M(G;x,y) = 4xy^3 + 2x^3y^3 + 7x^2y^3 + x^2y^2. \tag{1}$$

The corresponding degree-based topological indices are computed as:

$$\text{First Zagreb Index } (M_1) = 49.0000, \tag{2}$$

$$\text{Second Zagreb Index } (M_2) = 58.0000, \tag{3}$$

$$\text{Modified Second Zagreb Index } ({}^{m}M_2) = 4.9167. \tag{4}$$