

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук
(повна назва)

Кафедра _____ Штучного інтелекту
(повна назва)

АТЕСТАЦІЙНА РОБОТА
Пояснювальна записка

Рівень вищої освіти _____ другий (магістерський) _____
(рівень вищої освіти)

Оцінка показників асортативності безмасштабних мереж

(тема)

Виконав:
студент 2 курсу, групи СШМ-18-1
Обихвост І.М.

(прізвище, ініціали)

Спеціальність 122 – Комп'ютерні науки

(код і повна назва спеціальності)

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного
інтелекту (СШ)

(повна назва освітньої програми)

Керівник доц. Шергін В.Л.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

В.О. Філатов
(прізвище, ініціали)

2019 р.

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____

Кафедра _____ Штучного інтелекту _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 122 – Комп'ютерні науки _____
(код і повна назва)

Тип програми освітньо-професійна _____
(освітньо-професійна або освітньо-наукова)

Освітня програма _____ Системи штучного інтелекту (СШІ) _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«_____» _____ 2019 р.

ЗАВДАННЯ

НА АТЕСТАЦІЙНУ РОБОТУ

студентові _____ Обихвосту Івану Миколайовичу _____

(прізвище, ім'я, по батькові)

1. Тема роботи Оцінка показників асортативності безмасштабних мереж _____

затверджена наказом по університету від 04 листопада 2019 р. № 1623Ст

2. Термін подання студентом роботи до екзаменаційної комісії 18 грудня 2019 р.

3. Вихідні дані до роботи Науково-технічні публікації, дані Інтернет-джерел та відомих наукових проектів щодо розробки та дослідження складних мереж з метою подальшого розрахунку їх коефіцієнта асортативності, згенеровані складні мережі _____

4. Перелік питань, що потрібно опрацювати в роботі Аналіз предметної області та постановка задачі, асортативність у складних мережах і засоби її дослідження, програмна реалізація генерації безмасштабних мереж та розрахунку їх показника асортативності _____

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Рисунок 1 – Мережева структура Інтернету, Рисунок 2 – Мережа сторінок корпоративного веб-сайту, Рисунок 3 – Мережа дружби між членами клубу, Рисунок 4 – Харчова мережа озера Літл Рок, штат Вісконсін, Рисунок 5 – Можливі варіанти уявлення метаболічних мереж, Рисунок 6 – Схема мостів Кенігсбергу, Рисунок 7 – К-компоненти для $k = 1, 2, 3$, Рисунок 8 – Випадкова мережа (ліворуч) і масштабно-інваріантна мережа (праворуч), Рисунок 9 – Процес утворення масштабно-інваріантної мережі, Рисунок 10 – Мережа дружби у середній школі США, Рисунок 11 – Мережа дружби дітей у школі за віком, Рисунок 12 – Графік віку пар друзів у школі, Рисунок 13 – Мережа, що є асортативною за ступенем, Рисунок 14 – Мережа, що є дизасортативною за ступенем, Рисунок 15 – Згенерована мережа

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	доц. Шергін В.Л.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на атестаційну роботу	04.11.19	виконано
2	Аналіз предметної області	04.11.19-12.11.19	виконано
3	Постановка завдання та узгодження з керівником	13.11.19-16.11.19	виконано
4	Дослідження методів розрахунку коефіцієнту асортативності	17.11.19-21.11.19	виконано
5	Розробка алгоритму	22.11.19-28.11.19	виконано
6	Розробка програмного забезпечення	29.11.19-01.12.19	виконано
7	Написання пояснювальної записки	02.12.19-11.12.19	виконано
8	Нормоконтроль	12.12.19	виконано
9	Попередній захист	16.12.19	виконано
10	Захист перед ЕК	18.12.19	

Дата видачі завдання 04 листопада 2019 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Шергін В.Л.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Записка пояснювальна: 89 сторінок, 15 рисунків, 1 таблиця, 25 формул, 15 джерел, 1 додаток.

АСОРТАТИВНІСТЬ, ГОМОФІЛІЯ, ДИЗАСОРТАТИВНІСТЬ,
ІНТЕРНЕТ, КОРЕЛЯЦІЯ, МАСШТАБНО-ІНВАРІАНТНІ МЕРЕЖІ,
СТЕПЕНЕВИЙ ЗАКОН, СКЛАДНІ МЕРЕЖІ

В атестаційній роботі розглядається задача моделювання складних мереж, зокрема, масштабно-інваріантних мереж, а також оцінка їх показників асортативності.

Метою атестаційної роботи є створення алгоритму генерації еластичної моделі масштабно-інваріантної мережі і алгоритму для розрахунку показника асортативності мережі, що використовує формулу коефіцієнта кореляції Пірсона.

Об'єктом дослідження є схильність масштабно-інваріантних мереж до асортативності / дизасортативності за ступенем мережі, зумовлена особливостями їх структури та природи їх формування та розвитку.

Предметом дослідження є масштабно-інваріантна мережа, що розвивається з часом по законам зростання та переважного приєднання.

Методи дослідження базуються на математичному апараті теорії графів, а також елементах теорії оптимізації та регресійного аналізу.

Імітаційне моделювання застосовується для перевірки якості виконаної роботи з використанням коефіцієнту кореляції Пірсона.

РЕФЕРАТ

Записка пояснительная: 89 страниц, 15 рисунков, 1 таблица, 25 формул, 15 источников, 1 приложение.

АССОРТАТИВНОСТЬ, БЕЗМАСШТАБНЫЕ СЕТИ, ГОМОФИЛИЯ, ДИЗАССОРТАТИВНОСТЬ, ИНТЕРНЕТ, КОРРЕЛЯЦИЯ, СТЕПЕННОЙ ЗАКОН, СЛОЖНЫЕ СЕТИ

В аттестационной работе рассматривается задача моделирования сложных сетей, в частности, безмасштабных сетей, а также оценка их показателей ассортативности.

Целью аттестационной работы является создание алгоритма генерации эластичной модели безмасштабной сети и алгоритма для расчета показателя ассортативности сети, использующего формулу коэффициента корреляции Пирсона.

Объектом исследования является склонность безмасштабных сетей к ассортативности / дизассортативности по степени вершины, обусловленная особенностями их структуры и природы их формирования и развития.

Предметом исследования является безмасштабная сеть, развивающаяся со временем по законам роста и предпочтительного присоединения.

Методы исследования базируются на математическом аппарате теории графов, а также элементах теории оптимизации и регрессионного анализа.

Имитационное моделирование применяется для проверки качества выполненной работы с использованием коэффициента корреляции Пирсона.

ABSTRACT

Explanatory note: 89 pages, 15 figures, 1 table, 25 formulas, 15 sources, 1 annex.

ASSORTATIVITY, CORRELATION, COMPLEX NETWORKS, DISASSORTATIVITY, HOMOPHILY, INTERNET, POWER LAW, SCALE-FREE NETWORKS

In the attestation work, the problem of modeling complex networks is being addressed, in particular, scale-free networks, as well as an evaluation of their assortativity coefficients.

The purpose of the attestation is to create an algorithm for generating an elastic model of a scale-free network and an algorithm for calculating the assortativity coefficient of a network using the Pearson correlation coefficient formula.

The object of the study is the tendency of scale-free networks to lean towards assortativity / disassortativity by degree, due to the peculiarities of their structure and the nature of their creation and development.

The subject of the study is a scale-free network that develops over time according to the laws of growth and preferred connection.

Research methods are based on the mathematical apparatus of graph theory, as well as elements of optimization theory and regression analysis.

Simulation is used to verify the quality of work performed using the Pearson correlation coefficient.

ЗМІСТ

Перелік умовних позначень, одиниць, скорочень і термінів	8
Вступ	9
1 Аналіз предметної галузі та формалізована постановка задачі	10
1.1 Вступ до мереж	10
1.2 Теорія графів та складні мережі	25
1.3 Асортативність	39
1.4 Асортативність у складних мережах	41
1.5 Постановка завдання дослідження	47
2 Методи дослідження	48
2.1 Еластична модель масштабно-інваріантної мережі	49
2.2 Асортативне змішування по перераховувальним характеристикам ..	52
2.3 Асортативне змішування по скалярним характеристикам	56
2.4 Асортативне змішування за ступенем	61
3 Практична реалізація та імітаційне моделювання	68
3.1 Вибір мови програмування та програмного забезпечення	68
3.2 Реалізація алгоритму генерації мережі	71
3.3 Реалізація розрахунку коефіцієнта асортативності мережі	78
3.4 Імітаційне моделювання	83
Висновки	86
Перелік посилань	87
Додаток А	89

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

API – Application Programming Interface – спеціальна бібліотека готових функцій і класів для використання у зовнішніх програмних продуктах;

SF-мережі – масштабно-інваріантні (scale-free) мережі;

ER-мережі – мережі Ердеша-Рен'ї;

WWW – World Wide Web – Всесвітня павутина;

URL – Uniform Resource Locator – адреса ресурсу;

Веб-кроулер – програма, призначена для перебору сторінок Інтернету;

Peer-to-peer мережа – комп'ютерна мережа, заснована на рівноправності учасників;

ВСТУП

У 1999 році фізик з університету Нотр Дам (США) А.-Л. Барабаші разом зі своєю аспіранткою Р.Альберт вивчали властивості реальних мереж з дещо незвичної в той час точки зору: вони вирішили дослідити закон розподілу вузлів реальних мереж за кількістю зв'язків. Для багатьох мереж (метаболичні мережі і білкові взаємодії в клітинах, структура авіаційних повідомлень в США, структура Інтернету WWW і т.д.) замість очікуваного імовірнісного розподілу вузлів по числу зв'язків q за законом Пуасона, отриманий розподіл приблизно підпорядковувався властивому всім критичним станам степеневому закону. Таким чином, у багатьох реальних мережах невелике число вузлів містить дуже велике число зв'язків (їх називають хаби, від англійського слова hub-концентратор), а величезна кількість вузлів містить лише кілька зв'язків. Такі мережі отримали назву масштабно-інваріантних мереж (SF-мережі).

Один з підходів до дослідження, що лежать в основі генеративних процесів, полягає у вивченні кореляції або асортативного змішування атрибутів вузла (або «метаданих») по ребрах. Цей аналіз дозволяє нам зробити узагальнення щодо того, з більшою ймовірністю ми спостерігаємо зв'язку між вузлами з однаковими характеристиками (асортативність) або між вузлами з різними (дизасортативність). Соціальні мережі часто містять позитивні кореляції значень атрибутів через з'єднання. Ці кореляції виникають в результаті взаємодоповнюючих процесів відбору (або «гомофілія») і впливу (або «зараження») [1].

У даній роботі буде досліджено явище асортативності / дизасортативності у масштабно-інваріантних мережах шляхом моделювання цих мереж та розрахунку їх показника асортативності.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ФОРМАЛІЗОВАНА ПОСТАНОВКА ЗАДАЧІ

1.1 Вступ до мереж

Є багато систем, що представляють інтерес для вчених, які складаються з окремих частин або компонентів і пов'язані один з одним в деякому роді. Приклади включають в себе Інтернет, набір комп'ютерів, пов'язаних даними зв'язку і людські суспільства, які представляють собою сукупності людей, пов'язаних знайомством або соціальною взаємодією.

Багато аспектів цих систем заслуговують вивчення. Деякі люди вивчають природу окремих компонент – наприклад, як комп'ютер працює, або як людина відчуває або діє – в той час як інші вивчають природу з'єднань або взаємодій – протоколи зв'язку, які використовуються в Інтернеті або динаміку людських відносин. Але є третій аспект цих взаємодіючих систем, яким іноді нехтують, але який майже завжди має вирішальне значення для поведінки системи – патерн з'єднань між компонентами.

Схема з'єднань в даній системі може бути представлена як мережа, компонентами якої є вершини мережі, а сполуками – ребра. Після роздумів не повинно бути сюрпризом те (хоча в деяких областях це відносно недавнє усвідомлення), що структура таких мереж, конкретний характер взаємодії може дуже вплинути на поведінку системи. Наприклад, з'єднання між комп'ютерами в Інтернеті впливають на маршрути передачі даних і ефективність, з якою мережа транспортує ці дані. Зв'язки в соціальних мережах впливають на те, як люди вчаться, формують думки і збирають новини, а також впливає на інші, менш очевидні явища, такі як поширення хвороб. Якщо ми не знаємо щось про структуру цих мереж, ми не можемо сподіватися повністю зрозуміти, як працюють відповідні системи.

Мережа – це спрощене уявлення, яке зводить систему до абстрактної структури, що охоплює тільки основи шаблонів зв'язку і деякі дрібниці.

Вершини і ребра в мережі можуть бути позначені додатковою інформацією, наприклад іменами або сильними сторонами, щоб захопити більше деталей системи, але навіть в цьому випадку багато інформації зазвичай втрачається в процесі приведення всієї системи до мережевого виду. Це, звичайно, має свої недоліки, але є і переваги.

За багато років вчені в самих різних областях розробили великий набір інструментів: математичні, обчислювальні і статистичні – для аналізу, моделювання і розуміння мереж. Багато з цих інструментів починаються з простого уявлення мережі, набору вершин і ребер, і після відповідних розрахунків розкажуть вам про мережі щось, що може виявитися корисним: скажімо, яка з сусідніх вершин є кращою або довжину шляху від однієї вершини до іншої. Інші інструменти приймають форму мережевих моделей, які можуть робити математичні прогнози про процеси, що відбуваються в мережах, таких як рух трафіку через Інтернет або поширення хвороби у суспільстві. Оскільки вони працюють з мережами в їх абстрактній формі, ці інструменти теоретично можуть застосовуватися практично до будь-яких систем, представлених у вигляді мережі. Таким чином, якщо є система, яка вас цікавить, і вона може бути представлена у вигляді мережі, то є сотні вже розроблених різних інструментів, які можуть допомогти вам з аналізом вашої системи. Звичайно, не всі з них дадуть корисні результати – які вимірювання або розрахунки корисні для конкретної системи, залежить від того, що це за система і на які саме питання ви намагаєтеся знайти відповідь. Проте, якщо у вас є правильні питання про мережеву систему, у багатьох випадках вже буде доступний інструмент для цього завдання [2].

Таким чином, мережі є загальним, але потужним засобом представлення моделей зв'язків або взаємодій між частинами системи. Існує безліч прикладів конкретних мереж в різних сферах, а також методів їх аналізу, взятих з математики, фізики, комп'ютерних та інформаційних наук, суспільних наук, біології та ін. Для того, щоб дати всебічне введення в науку

мереж, необхідно об'єднати широкий спектр ідей і досвіду з багатьох дисциплін.

Одним з найбільш відомих і найбільш широко досліджуваних прикладів мережі є Інтернет, комп'ютерна мережа передачі даних, в якій вершинами є комп'ютери, а ребрами є фізичні дані з'єднання між ними, такі як оптоволоконні кабелі або телефонні лінії. Рисунок 1.1 показує картину структури інтернету, знімок мережі, як це було в 2003 році, реконструйовану виходячи зі спостережень за які проходять через мережу великою кількістю інтернет-пакетів даних між різними джерелами і пунктами призначення. Цікаво, що хоч Інтернет і є штучною і ретельно спроектованою мережею, ми не знаємо точно, яка її структура, оскільки вона була побудована багатьма різними групами людей з обмеженим знанням дій один одного і без централізованого контролю. Кращі поточні дані про його структуру отримані з експериментальних досліджень, а не з будь-якого центрального сховища знань або координуючого органу.

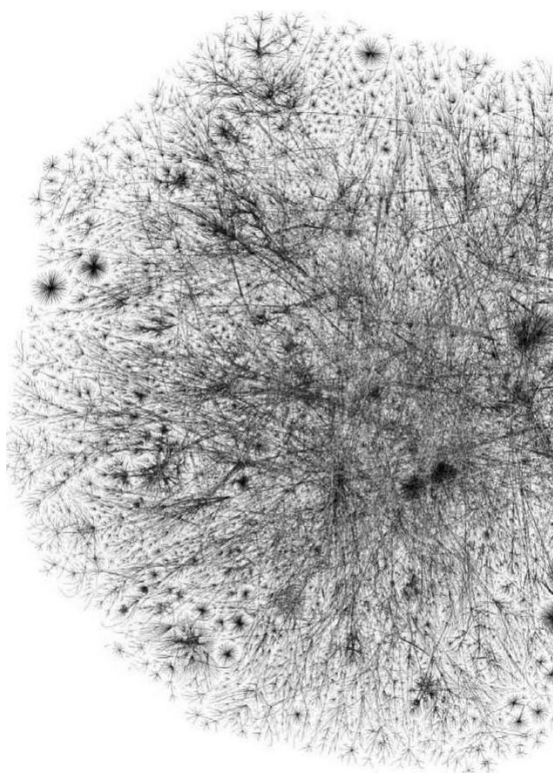


Рисунок 1.1 – Мережева структура Інтернету

Є ряд прекрасних практичних причин, за якими ми могли б вивчити мережеву структуру Інтернету. Функція Інтернету полягає в передачі даних між комп'ютерами (і іншими пристроями) в різних частинах світу, що відбувається шляхом поділу даних на частини або пакети і відправкою їх з вершини в вершину по мережі, поки вони не досягнуть свого передбачуваного пункту призначення. Звичайно, структура мережі буде впливати на те, наскільки ефективно це буде досягнуто, і якщо ми знаємо структуру мережі, ми можемо вирішити багато питань практичної актуальності. Як вибрати маршрут, по якому будуть передаватися дані? Чи завжди найкоротший маршрут – найшвидший? Якщо ні, то що і як ми можемо знайти? Як ми можемо уникнути вузьких місць в транспортному потоці, які можуть уповільнити хід подій? Що відбувається, коли відмовляють вершина або ребро (що вони роблять з деякою регулярністю)? Як ми можемо розробити схеми для обходу таких невдач? Якщо у нас є можливість додати новий вузол в мережу, куди він повинен бути доданий?

Знання структури Інтернету також грає центральну роль в розробці нових стандартів зв'язку. Нові стандарти і протоколи постійно розробляються для спілкування через інтернет, а старі переглядаються. Параметри цих протоколів налаштовані на оптимальну продуктивність з урахуванням структури Інтернету. У перші дні мережі, досить примітивні моделі структури мережі були використані в процесі налаштувань, але в міру того як кращі структурні дані стають доступними, стає можливим краще зрозуміти і поліпшити продуктивність.

Більш абстрактний приклад мережі – Всесвітня павутина. Досить часто слова «Павутина» і «Інтернет» використовуються як взаємозамінні, але технічно вони абсолютно різні. Інтернет являє собою фізичну мережу комп'ютерів, з'єднаних реальними кабелями (або іноді радіозв'язком). Павутина, з іншого боку, являє собою мережу інформації, що зберігається в Інтернеті. Вершини Всесвітньої павутини – це веб-сторінки, а ребра – це «гіперпосилання», виділені фрагменти тексту або кнопки на веб-сторінках,

на які ми натискаємо для переходу від однієї сторінки до іншої. Гіперпосилання – це чисто програмна конструкція; Ви можете зробити посилання зі свого веб-сторінки на сторінку, яка знаходиться на комп'ютері в іншій частині світу, так само легко, як ви можете зв'язатися з другом по коридору. Там немає фізичної структури, як оптичне волокно, яку необхідно побудувати, коли ви робите нове посилання. Посилання – це просто адреса, яка повідомляє комп'ютеру, де шукати.

Незважаючи на абстрактність, Всесвітня павутина з її мільярдами сторінок і посилань довела свою надзвичайну корисність, не кажучи вже про прибутковість для багатьох людей, і структура мережі посилань представляє істотний інтерес. Оскільки люди схильні додавати гіперпосилання між сторінками зі зв'язаним контентом, структура посилань в Інтернеті розкриває щось про структуру контенту. Більш того, люди схильні частіше посилатися на сторінки, які вони вважають корисними, ніж на ті, які їм не потрібні, так що кількість посилань, що вказують на сторінку, можна використовувати як показник її корисності. Схожа версія цієї ідеї лежить в основі популярного пошукового движка Google, а також деяких інших [3].

Павутина також ілюструє іншу концепцію теорії мереж, спрямовану мережу. Гіперпосилання в Інтернеті йдуть в одному певному напрямку, з однієї веб-сторінки на іншу. Побачивши відповідне посилання на сторінці А, ви можете натиснути його і перейти на сторінку В. Але не потрібно, щоб В містила посилання знову до А (вона може містити таке посилання, але немає закону, який говорить, що так повинно бути і велику частину часу не буде). Ребра Всесвітньої павутини є спрямованими, йдучи від сторінки, що посилається до сторінки, на яку вона посилається.

Як описано вище, Всесвітня павутина – це мережа, в якій вершинами є веб-сторінки, що складаються з тексту, зображень або іншої інформації, а ребра – це гіперпосилання, які дозволяють нам переміщуватися зі сторінки на сторінку. Оскільки гіперпосилання працюють тільки в одному напрямку, мережа є спрямованою. Можна зобразити мережу зі стрілкою на кожному

ребрі, що вказує, яким чином вона працює. Деякі пари веб-сторінок пов'язані гіперпосиланнями в обох напрямках, які можуть бути представлені у вигляді двох спрямованих ребер, по одному в кожному напрямку між відповідними вершинами. Рисунок 1.2 показує зображення невеликої частини веб-мережі, що представляє з'єднання між мережею сторінки на одному сайті.

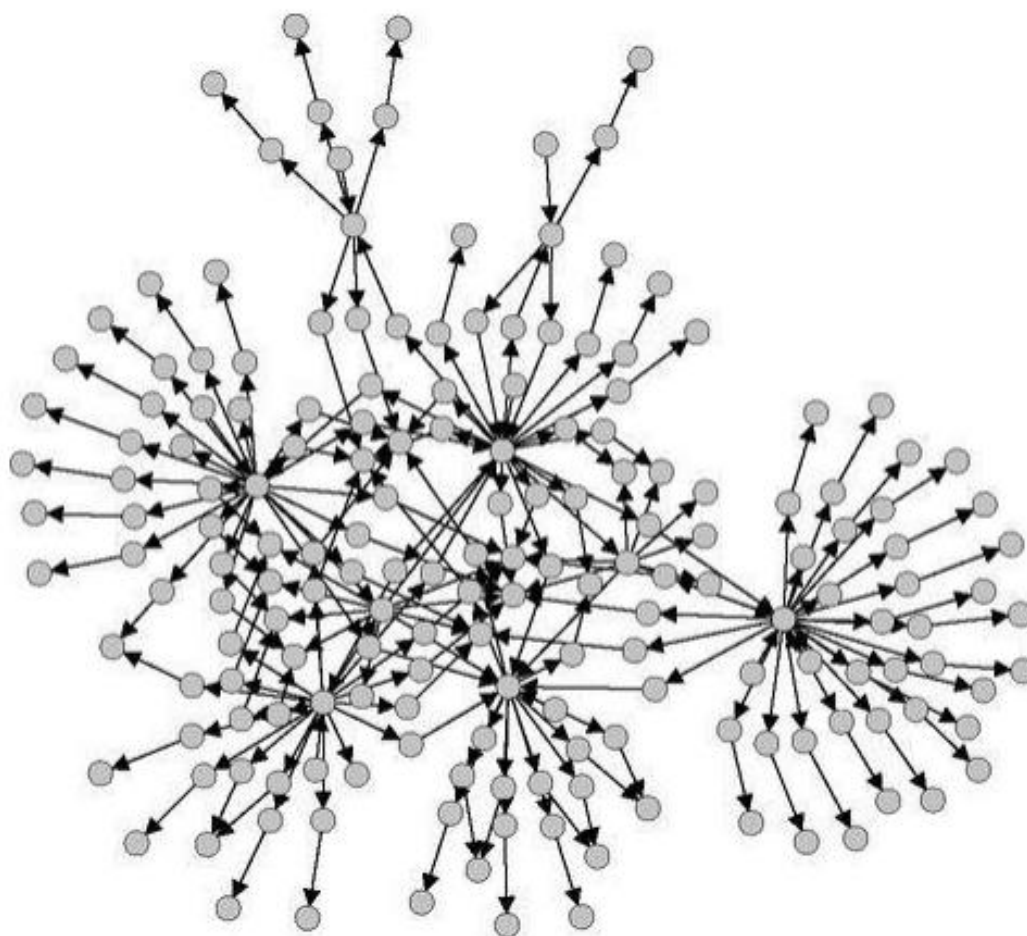


Рисунок 1.2 – Мережа сторінок корпоративного веб-сайту

Структура Інтернету може бути виміряна за допомогою кроулера, комп'ютерної програми, яка автоматично переглядає веб-сторінки в пошуках сторінок. У своїй простій формі кроулер виконує так званий пошук в ширину в мережі, як показано схематично на рисунку 1.2. Починається з будь-якої початкової веб-сторінки, завантажує текст цієї сторінки через

Інтернет і знаходить всі посилання в тексті. Функціонально посилання складається з ідентифікуючого «тега» – короткого фрагмента тексту, що позначає посилання як посилання – і уніфікованого покажчика ресурсу або URL-адреси, стандартизованої адреси комп'ютера, яка говорить, як і де можна знайти пов'язану веб-сторінку. Скануючи теги, а потім копіюючи сусідні URL-адреси, веб-кроулер може швидко витягти URL-адреси для всіх посилань на веб-сторінці, зберігаючи їх у пам'яті або на диску. Коли це зроблено з поточною сторінкою, він видаляє одну з URL-адрес зі свого сховища, використовує її для пошуку нової сторінки в Інтернеті і завантажує текст цієї сторінки, і так процес повторюється. Якщо в якийсь момент кроулер зустрічає URL, що співпадає з уже наявними в його сховищі, то цей URL ігнорується і більше не додається в сховище, щоб уникнути дублювання записів. У сховище додаються тільки ті URL-адреси, які відрізняються від усіх попередніх.

Повторюючи процес завантаження і вилучення URL протягом досить тривалого періоду часу, можна знайти значну частину сторінок в усьому Інтернеті. На практиці, однак, жоден кроулер не здатний знайти всі сторінки в мережі. Є ряд причин для цього. По-перше, деякі веб-сайти забороняють кроулерам сканувати свої сторінки. Веб-сайти можуть розмістити файл під назвою robots.txt в їх кореневий каталог, який вказує, які файли, якщо такі є, кроулери можуть переглядати, і може додатково вказувати, що деякі кроулери можуть переглядати файли, а інші ні. Відповідність з обмеженнями, зазначені в robots.txt є добровільним, але на практиці багато кроулерів виконують його.

По-друге, багато сторінок в Інтернеті генеруються динамічно: вони створюються на льоту за допомогою спеціального програмного забезпечення, що використовує, наприклад, дані з бази даних. Багато корпоративних сайтів, а також веб-сторінки, створені пошуковими системами або службами каталогів, потрапляють в цю категорію. Кількість можливих веб-сторінок, які можуть відображатися в результаті пошуку з

використанням пошукової системи Google, наприклад, настільки велика, що може фактично вважатися нескінченною; було б неможливо для кроулера просканувати всі ці сторінки. Тому кроулер повинен зробити вибір щодо того, що вважається веб-сторінкою, а що ні. Один з варіантів – обмежити себе статичними веб-сторінками, які не генеруються на льоту. Але не завжди просто сказати, які сторінки є статичними, і крім того, багато корисної інформації знаходиться на динамічних сторінках. На практиці рішення, що приймаються кроулером про те, які сторінки включати, як правило, досить довільні, і нелегко вгадати, які сторінки будуть включені в обхід, а які ні. Але можна з упевненістю сказати, що багато сторінок не будуть включені, і в цьому сенсі сканування завжди є неповним.

Однак, можливо, найважливіша причина, по якій веб-обходи не досягають всіх сторінок Павутини – те, що структура мережі не дозволяє цього. Оскільки Павутина є спрямованою мережею, не всі сторінки доступні із заданої початковою точки. Зокрема, зрозуміло, що сторінки які не мають вхідних гіперпосилань – сторінки, на які ніхто не посилається – ніколи не можуть бути знайдені кроулером, який сліпо слідує за посиланнями. Продовжуючи ідею, це також той випадок, коли сторінка ніколи не буде знайдена, якщо на неї посилаються тільки сторінки, на які немає вхідних посилань. І так далі. Насправді, Павутина має спеціальну «компонентну» структуру, і більшість кроулерів знайдуть тільки одну частину цієї структури, «Гігантський зовнішній компонент». У випадку з Всесвітньою павутиною гігантський зовнішній компонент становить лише близько половини всіх веб-сторінок, а інша половина Інтернету недоступна [4].

Відійшовши від технологічної сфери, іншим типом мережі наукового інтересу є соціальна мережа. Соціальна мережа – це, як правило, мережа людей, хоча іноді це може бути мережа груп людей, таких як компанії. Люди або групи утворюють вершини мережі, а ребра є якимись зв'язками між ними, такими як дружба між конкретними особами або ділові відносини між компаніями. Область соціології має, мабуть, найдавнішу і найбільш

розвинену традицію емпіричного вивчення мереж у міру їх виникнення в реальному світі, і багато математичних та статистичних інструментів, які використовуються при вивченні мереж, запозичені прямо або побічно у соціологів.

На рисунку 1.3 показаний відомий приклад соціальної мережі з соціологічної літератури, мережа карате-клубу Захарі Уейна. Ця мережа відображає патерни дружби між членами клубу карате в північноамериканському університеті. Мережа була побудована шляхом прямих спостережень за взаємодіями між членами клубу. Типово для таких досліджень, мережа мала, маючи в цьому випадку тільки 34 вершини. Мережеві уявлення інтернету або Всесвітньої павутини, навпаки, можуть мати тисячі або мільйони вершин. Немає причин, за якими соціальні мережі не можуть бути такими ж великими. Все населення світу, наприклад, можна розглядати як дуже велику соціальну мережу. Але на практиці дані соціальних мереж обмежені відносно невеликими групами через зусилля, пов'язані з їх складанням. Наприклад, дана мережа (рисунок 1.3) була результатом дворічних спостережень одного експериментатора. Останнім часом, завдяки величезним зусиллям з боку великих груп дослідників, було побудовано кілька більш об'ємних соціальних мереж. І онлайн-сервіси соціальних мереж, такі як Facebook, можуть надати мережеві дані в раніше недоступних масштабах. Дослідження структур і властивостей цих великих мереж тільки починають розгортатися [5].

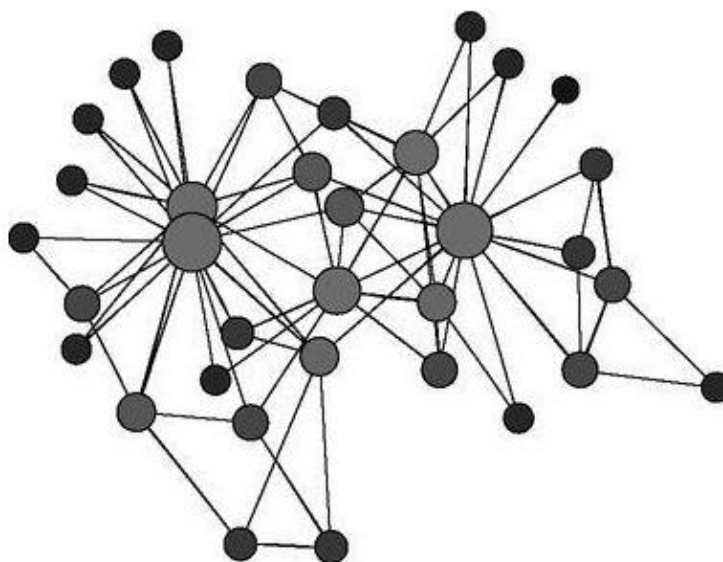


Рисунок 1.3 – Мережа дружби між членами клубу

Третя область, в якій мережі стали важливими в останні роки, – це біологія. Мережі спостерігаються в ряді ситуацій в біології. Деякі конкретні фізичні мережі, такі як нейронні мережі – мережі зв'язків між нейронами в мозку – в той час як інші є більш абстрактними. На рисунку 1.4 показана картина «харчової мережі», екологічної мережі, в якій вершини – це види в екосистемі, а ребра представляють відносини хижак-жертва між ними. Тобто пари видів з'єднуються ребрами в цій мережі, якщо один вид їсть інший. Вивчення харчових мереж формує істотну галузь екології та допомагає нам зрозуміти і кількісно визначити багато екологічних явища, особливо щодо потоків енергії і вуглецю в екосистемах. Харчові мережі також надають нам ще один приклад спрямованої мережі, такий як Всесвітня павутина, що обговорювалася раніше. Ребра в харчовій мережі асиметричні і зазвичай вказують від жертви на хижака, вказуючи напрямок потоку енергії коли жертва з'їдена. (Цей вибір напрямку є всього лише умовним, і, безумовно, можна зробити навпаки. Важливим моментом є асиметрія взаємодії хижак-жертва.).

Рисунок 1.4 узагальнює відомі хижі взаємодії між видами в прісноводному озері в північній частині Сполучених Штатів. Вершини представляють вид, а ребра проходять між парами видів хижак-жертва. Вертикальне положення вершин представляє, грубо кажучи, трофічний рівень відповідного виду [6].

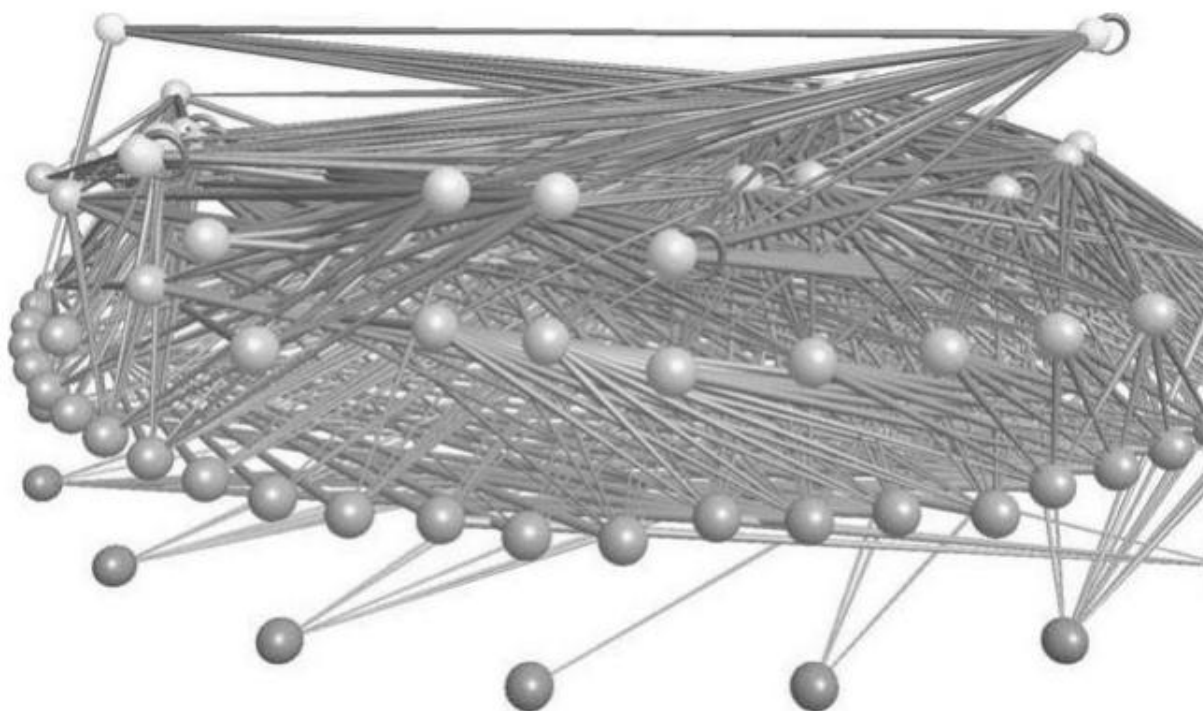


Рисунок 1.4 – Харчова мережа озера Літл Рок, штат Вісконсін

Інший клас біологічних мереж – це біохімічні мережі, такі як метаболічні мережі, мережі міжбілкової взаємодії і генетичні регуляторні мережі. Метаболічна мережа, наприклад, є відображенням хімічних реакцій, які живлять клітини і організми. Читач, можливо, бачив учбові плакати метаболічних реакцій, які прикрашають офіси деяких біохіміків, неймовірно докладні карти з сотнями крихітних написів, пов'язаних лабіринтом стрілок. Надписи – вершини цієї мережі – є метаболітами, субстратами і продуктами метаболізму, а стрілки – спрямовані ребра – це реакції, що перетворюють один метаболіт в інший (рисунок 1.5). Опис

реакцій як мережі є одним з перших кроків до осмислення дивовижного масиву біохімічних даних, отриманих в результаті недавніх і поточних експериментів в молекулярній генетиці.

Деталі метаболічних мереж різняться у різних видів організмів, але принаймні серед тварин, великі частини є загальними для всіх або більшості видів. Багато важливих шляхів, циклів, або інших частин метаболічних мереж практично не змінюється по всьому царству тварин. З цієї причини часто називають просто «метаболізм» без зазначення конкретного виду, що представляє інтерес; з невеликими варіаціями спостереження, зроблені на одному виді, часто відносяться і до інших.

Найбільш правильне відображення метаболічної мережі – це дводольна мережа. У дводольній мережі є два відмінних типи вершин, з ребрами, що проходять тільки між вершинами різних видів. У разі філіальних мереж, наприклад, два типи вершин представляють людей і групи, до яких вони належать. В цьому випадку метаболічної мережі вони представляють метаболіти і метаболічні реакції, з ребрами, що з'єднують кожен метаболіт з реакціями, в яких він бере участь. Насправді, метаболічна мережа дійсно є спрямованою двочастковою мережею, оскільки деякі метаболіти вступають в реакцію (субстрати), а деякі виходять з неї (продукти). Розміщуючи стрілки на ребрах, можна відрізнити вхідні метаболіти від вихідних.

Це дводольне уявлення метаболічної мережі не включає будь-якого способу представлення ферментів, які, хоча і не метаболіти самі по собі, все ще є важливою частиною метаболізму. Хоча це не часто робиться, до мережі все ж можна включити ферменти, якщо ввести третій клас вершин, що представляють їх, з ребрами, що з'єднують їх з реакціями, які вони каталізують. Оскільки ферменти не споживаються в реакціях, ці ребра не є спрямованими – не проникають ні всередину, ні назовні з реакцій, в яких вони беруть участь.

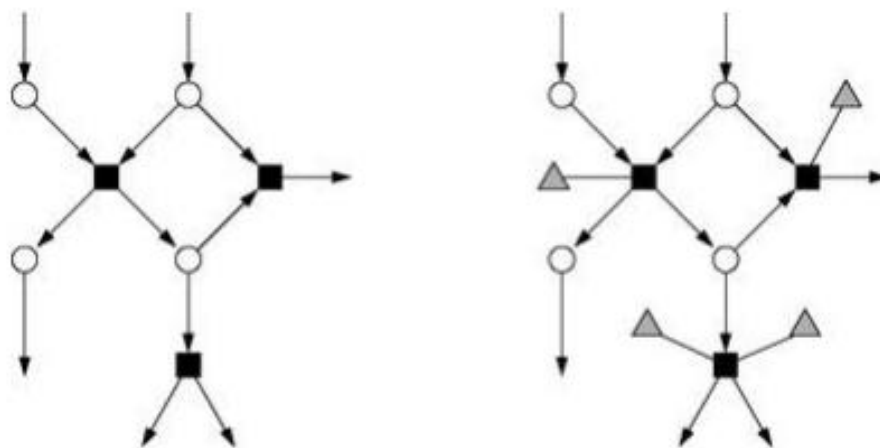


Рисунок 1.5 – Можливі варіанти уявлення метаболічних мереж

Хоч і є правильними і потенційно корисними, жодне з цих уявлень не є часто використовуваним для метаболічних мереж. Найбільш поширені уявлення про метаболічні мережі проєктують мережу тільки на один набір вершин, або метаболіти або реакції, причому останні є більш популярним вибором. В одному підході вершини в мережі представляють метаболіти, і між будь-якими двома метаболітами, які беруть участь в реакції, або як субстрат, або в якості продукту, утворюється ненаправлене ребро. Очевидно, що ця проєкція втрачає велику частину інформації, що міститься в повній двочастковій мережі, але, як уже сказано, вона, тим не менш, широко використовується. Інший підхід, можливо, найпоширеніший, полягає в тому, щоб представляти спрямовану мережу з одним типом вершини, які представляють метаболіти і спрямоване ребро від одного метаболіту до іншого створюється в тому випадку, якщо є реакція, в якій перший метаболіт з'являється в якості субстрату, а другий – як продукт. Це уявлення містить більше інформації від повної мережі, але все ще досить незадовільно, так як реакція з багатьма субстратами або багатьма продуктами відображається як сукупність великої кількості ребер, і немає простого способу сказати, що ці ребра представляють собою аспекти однієї реакції. Популярність цього подання обумовлена тим, що для багатьох реакцій

тільки один продукт і один субстрат відомі або вважаються важливими, і, отже, реакція може бути представлена тільки одним спрямованим ребром без плутанини. Число компаній виробляють великі діаграми, що показують найбільш важливі частини метаболічної мережі в такому виді [7].

Це всього лише кілька прикладів типів мереж, вивчення яких знаходиться в центрі уваги. Є багато інших, з якими ми зустрічаємося в повсякденному житті. Серед них одні з найбільш відомих – телефонні мережі, автомобільні, залізничні та повітряні мережі, електромережі, мережі цитування, рекомендаційні мережі, peer-to-peer мережі, мережі електронної пошти, колабораційні мережі, мережі поширення хвороб, річкові мережі та мережі слів.

За останні кілька десятиліть складні мережі, такі як Інтернет змінили те, як ми живемо. Вони також змінили поняття демократії, освіти, охорони здоров'я, розваги та комерції. У той час як складні мережі надають різні можливості, вони також створюють проблеми, такі як великий обсяг надлишкової інформації і різні проблеми безпеки. Для вирішення таких завдань, необхідні нові математичні інструменти і рамки для моделювання, аналізу, розуміння, і прогнозування структур і функцій різних складних мереж в надії зробити їх більш безпечними, доступними, передбачуваними і надійними.

Багато питань у сфері мережевого проектування пов'язані з розробкою, використанням, контролем і захистом комплексу мережі і пропонують велику кількість можливостей для більш ретельних досліджень. Наприклад, повна керованість і спостерігаємість великомасштабних спрямованих і зважених складних мереж нелінійних динамічних систем великих розмірностей відкривають багато питань для дослідників в різних полях схем і систем. Хоча вони є центральними питаннями в багатьох цікавих і важливих реальних додатках, які включають в себе складні мережі, мало що відомо про те, як вирішити ці питання через

відсутність загальної теорії і ефективних методів кількісного дослідження різних складних мереж.

Як видно, різні системи можуть бути представлені у вигляді мереж. Якщо зібрати дані про структуру однієї з цих мереж, що можна зробити з цими даними? Що вони можуть сказати про форму і функціонуванні системи, яку представляє мережу? Які властивості мережі системи можна виміряти або змодельовати і як ці властивості пов'язані з практичними питаннями, які нас хвилюють?

Першим кроком в аналізі структури мережі часто є її відображення. Рисунки 1.1-1.5 є типовими прикладами. Кожен з них був згенерований спеціалізованою комп'ютерною програмою, розробленою для візуалізації мереж, і існує безліч таких програм, як комерційних, так і безкоштовних. Візуалізація може бути надзвичайно корисним інструментом для аналізу мережевих даних, що дозволяє відразу побачити важливі структурні особливості мережі, які в іншому випадку було б важко виділити з необроблених даних. Людське око надзвичайно обдароване у виборі моделей, а візуалізація дозволяє нам використовувати цей дар для роботи над завданнями мереж. З іншого боку, пряма візуалізація мереж насправді корисна тільки для мереж до декількох сотень або тисяч вершин, а також для мереж, які є дещо розрідженими, що означає, що кількість ребер досить невелика. Якщо є занадто багато вершин або ребер в мережі, тоді зображення мережі будуть занадто складними для сприйняття оком і їх корисність стає обмеженою. У відповідь на цю потребу, теорія мереж розробила великий набір інструментів і метрики, які можуть допомогти нам зрозуміти, що говорять дані нашої мережі, навіть в тих випадках, коли корисна візуалізація неможлива.

1.2 Теорія графів та складні мережі

Дослідники, які вивчають властивості живої матерії, соціальних систем, великих техногенних об'єктів, все частіше приходять до висновку, що знань про властивості окремих елементів (гени, нейрони, банки, дорожні перехрестя) недостатньо для скільки-небудь повного розуміння функціонування відповідних складних систем в цілому (живої клітини, мозку, економіки, транспорту). Справді, у хворих на рак немає окремого «гена раку», а є мутації в декількох десятках взаємодіючих генів. Свідомість не може бути зведене до поведінки окремого нейрона, а вимагає врахування взаємодії мільярдів синапсів. Економічна криза не буде зрозумілий без вивчення складної павутини взаємних заборгованостей, властивих світовій фінансовій системі. Вивчення причин утворення дорожніх пробок і знаходження рішень для їх запобігання вимагає вивчення структури всієї дорожньої мережі міста. У цих науках, поряд з парадигмою редукціонізму, яка ще зовсім не вичерпала свій потенціал, повинна мати місце нова парадигма, що відображає складність відповідних систем.

З кінця 1990-х років для вивчення складних систем розвивається новий ефективний інструмент дослідження – теорія складних мереж. Вузли в таких мережах є елементи цих складних систем, а зв'язки посилянь – взаємодії між елементами. Такі мережі утворюють своєрідний каркас відповідних складних систем, а дослідження властивостей цих мереж дає змістовну інформацію про властивості складних систем в цілому, що дозволяє подолати деякі недоліки, властиві редукціонізму.

Мережевий підхід вивчення оточуючого нас світу бере свій початок з 1735 року, коли Леонард Ейлер, що жив тоді в Петербурзі, вирішив знамениту задачу про сім мостів в Кенігсберзі (рисунок 1.6), і заклав тим самим основи нової математичної дисципліни – теорії графів.

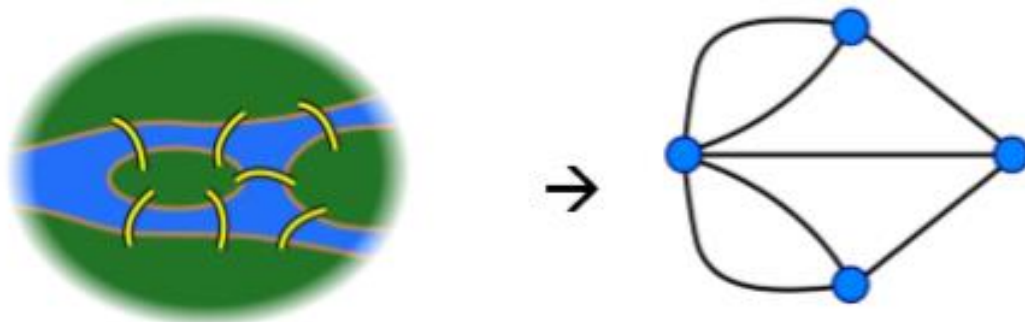


Рисунок 1.6 – Схема мостів Кенігсбергу

Завдання виглядало таким чином: як пройти по всіх семи мостах, не проходячи по жодному з них двічі. Довести або спростувати можливість існування такого маршруту ніхто не міг. Леонард Ейлер знайшов правило, користуючись яким, легко визначити, чи можна пройти по всіх мостах, не проходячи двічі по жодному з них:

- число непарних вершин має бути парним;
- якщо всі вершини парні, то можна не відриваючи олівця від паперу накреслити граф. При цьому можна починати з будь-якої вершини і завершити в тій самій вершині;
- граф з більш ніж двома непарними вершинами неможливо накреслити одним розчерком.

Граф Кенігсбергських мостів правило не задовольняв, тобто, мости не можна обійти, не проходячи двічі по якомусь з них.

Основою теорії складних мереж є теорія графів. При цьому, теорія графів може отримувати змістовні твердження, як правило, для графів (мереж) невеликого розміру або спеціальної структури. У той час як теорія складних мереж має справу з великою кількістю, як правило, випадковим чином з'єднаних вузлів. У зв'язку з цим, з одного боку, багато поширених питання теорії графів не представляють інтересу для теорії складних мереж. З іншого боку, багато дуже важливих понять в теорії

складних мереж для графів з невеликим числом вузлів і зв'язків або не представляють інтересу, або їх просто важко змістовно сформулювати.

У теорії складних мереж віділяють три основних напрямки: дослідження статистичних властивостей, які характеризують поведінку мереж; створення моделей мереж; передбачення поведінки мереж при зміні структурних властивостей. У прикладних дослідженнях зазвичай застосовують такі типові для мережевого аналізу характеристики, як розмір мережі, мережева щільність, ступінь центральності і т.п. Під час аналізу складних мереж, як і в теорії графів, досліджуються параметри окремих вузлів; параметри мережі в цілому; мережеві підструктури.

Поняття «складних мереж» охоплює мережі, що мають властивості:

- великий розмір;
- елементи випадковості при формуванні;
- зростання (зміна) з часом.
- деякі вузли можуть формувати компактні групи – ансамблі.

Для подальшої ясності має сенс виділити деякі основні характеристики складних мереж. Їх можна розділити на характеристики окремих вузлів і характеристики мережі в цілому.

Для окремих вузлів виділяють наступні параметри:

- вхідна ступінь вузла – кількість ребер графа, що входять до вузла;
- вихідна ступінь вузла – кількість ребер графа, що виходять з вузла;
- відстань від одного вузла до інших;
- ексцентричність – найбільша з мінімальних відстаней від даного вузла до інших;
- посередництво, що показує, скільки найкоротших шляхів проходить через даний вузол;
- центральність – загальна кількість зв'язків даного вузла по відношенню до інших.

Мережі в цілому характеризуються такими параметрами, як кількість вузлів, кількість зв'язків, відстань між вузлами, середня відстань від одного вузла до інших, мережева щільність, кількість симетричних, транзитивних і циклічних тріад, діаметр мережі – найбільша відстань між вузлами в мережі і інші.

При розгляді мережевої моделі Всесвітньої павутини була представлена концепція компонент. Компонентою в ненаправленій мережі називається така максимальна підмножина вершин, що кожна з них досяжна по деякому шляху від кожної з решти. Корисним узагальненням цього поняття є k -компонента. K -компонента (іноді також називається k -зв'язкова компонента) – максимальна підмножина вершин, така, що кожна досяжна з кожної з інших принаймні k вершинно-незалежними шляхами (рисунок 1.7). Два шляхи називають незалежними від вершин, якщо вони не мають жодної спільної вершини, крім початкової і кінцевої. Для загальних окремих випадків $k = 2$ і $k = 3$, k -компоненти також називаються бікомпонентами і трікомпонентами відповідно.

1-компонента за цим визначенням є просто звичайною компонентою – існує як мінімум один шлях між кожною парою вершин – і k -компоненти для $k \geq 2$ вкладені одна в одну. А 2-компонента або бікомпонента, наприклад, обов'язково є підмножиною 1-компонента, оскільки будь-яка пара вершин, які пов'язані принаймні двома шляхами, також пов'язані принаймні одним шляхом. Точно таким же чином трікомпонента обов'язково є підмножиною бікомпоненти і так далі.

Число вершинно-незалежних шляхів між двома вершинами дорівнює розміру зрізу, встановленому між одними і тими ж двома вершинами, тобто числу вершин, яке повинно було бути видалено, щоб роз'єднати. Таким чином, другий спосіб визначення k -компоненти полягає в тому, щоб сказати, що це максимальна підмножина вершин, таке, що ніяка пара вершин не може від'єднатися одна від одної, видаливши менш k вершин.

К-компоненту також можна визначити, використовуючи вершинно-незалежні шляхи, щоб вершини перебували в одній і тій самій k-компоненті, якщо вони з'єднані k або більш вершинно-незалежними шляхами, або еквівалентно, якщо вони не можуть бути виключені видаленням менш ніж k ребер. Цей варіант може бути корисний при певних обставинах, але на практиці він використовується рідко.

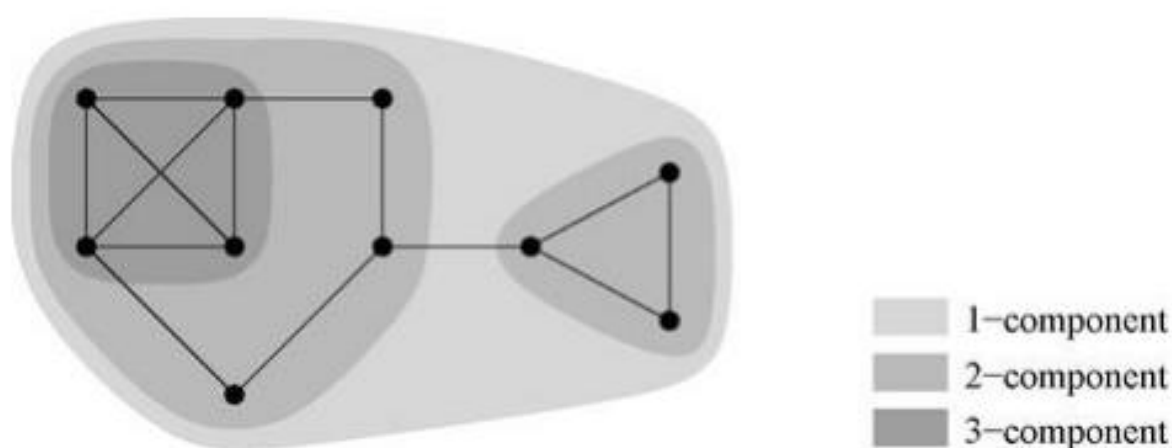


Рисунок 1.7 – К-компоненти для $k = 1, 2, 3$

Ідея k-компонент є природною в мережевому аналізі, оскільки пов'язана з ідеєю надійності мережі. В мережі передачі даних, наприклад, Інтернет, число вершинно-незалежних шляхів між двома вершинами також є числом незалежних маршрутів, які дані можуть взяти між тими ж двома вершинами, а розмір відрізка, встановленого між ними, є числом вершин в мережі – як правило, маршрутизаторів – які могли б вийти з ладу або іншим чином розірвати з'єднання даних між двома кінцевими точками. Таким чином, пара вершин, з'єднаних двома незалежними шляхами не можуть бути відключені один від одного через відмову якогось одного маршрутизатора, пара вершин, з'єднаних трьома шляхами, не може бути відключена відмовою будь-яких двох маршрутизаторів, і так далі. Можна сподіватися, наприклад, що велика частина магістралі мережі – системи глобальних

каналів зв'язку, які передають дані на великі відстані – це k -компоненти з високим k , так що точкам на магістралі буде важко втратити зв'язок один з одним.

Варто звернути увагу, що при $k \geq 3$ k -компоненти в мережі можуть бути несуміжними. Звичайні компоненти (1-компоненти) і бікомпоненти, навпаки, завжди є суміжними. В межах літератури з соціальних мереж, де несуміжні компоненти часто вважаються небажаними, k -компоненти іноді визначаються трохи по-іншому: k -компонент визначається як максимальний підмножина вершин, таке, що кожна пара в безлічі пов'язана не менше ніж з k незалежними від вершини шляхами, які самі містяться цілком у підмножині. Це визначення виключає несуміжні k -компоненти, але з математичної точки зор'яний з ними складніше працювати, ніж з компонентами з стандартного визначення. З цієї причини і тому що є також багато випадків, в яких підрахунок несуміжних k -компонент є актуальним, стандартне визначення залишається найбільш широко використовуваним в інших областях, крім соціології [8].

Для великих мереж з випадковою структурою однієї з найважливіших характеристик є функція розподілу $P(x)$ за ступенями вузлів. Розподіл вузлів по числу зв'язків $P(x)$ є ймовірність того, що випадково обраний вузол у випадковій мережі має ступінь x . Найбільша частина реальних складних мереж схожі (близькі) до наступних трьох:

- випадкова мережа або мережа Ердеша-Рен'ї (ER-мережа, рисунок 1.8). У випадку мережі ER функція розподілу є функцією Пуасона;
- мережа з експотенціальним розподілом;
- мережа зі ступеневим розподілом (SF-мережа, рисунок 1.8).

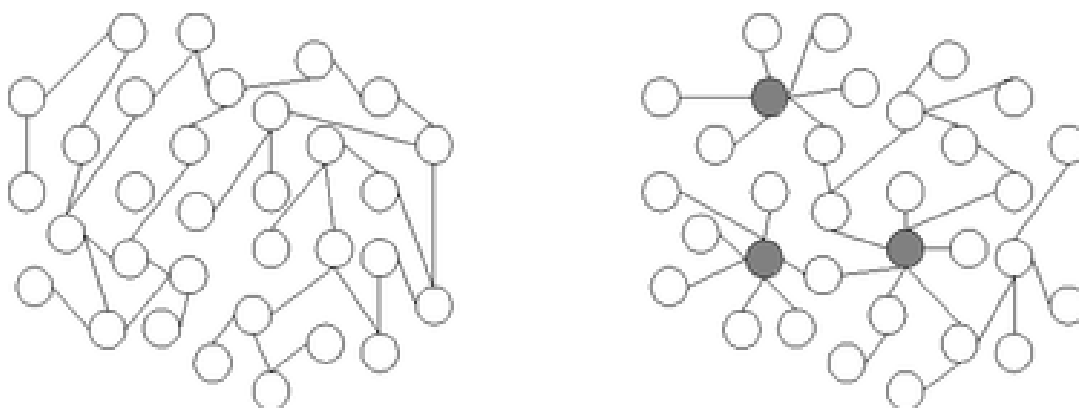


Рисунок 1.8 – Випадкова мережа (ліворуч) і масштабно-інваріантна мережа (праворуч)

У цій роботі увагу буде зосереджено на мережах останнього типу. Соціолог з Гарвардського університету Стенлі Мілграм в 1967 році емпірично показав, що будь-яких двох людей на Землі можна з'єднати ланцюжком з шести знайомих. Мілграм був зацікавлений в кількісному визначенні типової відстані між акторами в соціальних мережах. Це твердження отримало широку популярність і стало називатися «явищем тісного світу» (small world phenomenon), а після виходу Бродвейської п'єси, а потім і художнього фільму з відповідним сюжетом, ще й «шість ступенів поділу» («six degrees of separation»). Математичні аргументи припускають, що ця відстань повинна бути досить маленькою для більшості пар вершин у більшості мереж, і це вже було добре відомо за часів Мілграма. Мілграм хотів перевірити цю гіпотезу в реальних мережах і щоб зробити це, він придумав наступний експеримент.

Мілграм відправив набір посилок (всього 96) одержувачам, випадково обраним з телефонного довідника в американському місті Омаха, штат Небраска. Пакети містили офіційний буклет, або «паспорт», прикрашений зображенням рідного дому Мілграма, Гарвардського університету. Були включені письмові інструкції з проханням до одержувачів спробувати доставити паспорт вказаній людині, одному з друзів Мілграма, який жив в

Бостоні, штат Массачусетс, за тисячі миль звідси. Єдиною інформацією, наданою про ціль, було його ім'я (і, отже, побічно факт того, що він був чоловіком), його адреса і його діяльність в якості біржового маклера. Але власникам паспорта не дозволяється просто відправляти свій паспорт за вказаною адресою. Замість цього їх попросили передати паспорт комусь, кого вони знали по імені і, більш конкретно, людині в цій категорії, у якої, як вони відчували, буде більше шансів доставити паспорт до цілі. Таким чином, вони можуть вирішити відправити його комусь, кого вони знали, хто жив в Массачусетсі, або, можливо, комусь, хто працював у фінансовій індустрії. Вибір був за ними. Кому б вони не відправили паспорт, людину, що його отримала, просили повторити процес, відправивши його одному зі своїх знайомих, щоб після послідовності таких кроків, паспорт, якщо пощастить, потрапив до рук його передбачуваного одержувача. Оскільки кожен крок процесу відповідав переходу рук паспорта між парою знайомих по імені, весь пройдений шлях відповідав шляху по ребрах соціальної мережі, утвореної безліччю таких знайомств. Таким чином, довжина шляху надавала верхню межу геодезичного відстані в цій мережі між крайніми особистостями в ланцюзі.

З 96 відісланих паспортів 18 знайшли дорогу до біржового агента в Бостоні. Хоча це на перший погляд може здатися низькою цифрою, насправді вона є досить високою — недавні спроби повторити роботу Мілграма привели до того, що кількість відповідей на порядки нижче. Мілграм попросив учасників записувати в паспорт кожен крок пройденого шляху, щоб він знав, серед іншого, яка довжина кожного шляху, і він виявив, що середня довжина завершених шляхів від Омахи до мети була всього 5,9 кроку. Цей результат є джерелом ідеї «шести ступенів поділу» — що між будь-якими двома людьми в світі існує всього шість кроків.

Є, звичайно, багато причин, чому цей результат є тільки приблизним. Мілграм використовував тільки єдину ціль в Бостоні, і немає ніякої гарантії, що ціль була якимось чином типовою для населення в цілому. І всі початкові

одержувачі в дослідженні були в одному і тому ж місті в тій же країні (жоден з завершених ланцюжків, які досягли мети, не вийшов за межі країни). Також немає ніякої гарантії, що ланцюги вибрали найкоротший шлях до мети. Ймовірно, що ні, тож довжини знайдених шляхів забезпечують, як вже сказано, тільки верхню межу фактичної геодезичної відстані між вершинами. І більшість ланцюжків, зрозуміло, ніколи не були завершені. Паспорти були скинуті або загублені і не потрапили до мети. Розумно припустити, що шанси загубитися були більше для паспортів, які займали більш довгі шляхи, і, отже, що шляхи, які були завершені, були упередженим зразком, тобто, мали типові довжини коротше, ніж в середньому.

За всіх цих та кількох інших причин експерименти Мілграма повинні прийматися з дрібкою солі. Проте, широко приймається основний результат, що полягає в тому, що пари вершин в соціальних мережах мають тенденцію в середньому бути з'єднаними короткими шляхами, і, крім того, було показано, що він поширюється на також багато інших видів мереж. Досить велика кількість експериментів підтвердили даний ефект мережі настільки, щоб, які б побоювання не існували про конкретну техніку Мілграма, загальний результат серйозно не ставився під сумнів.

Експерименти Мілграма також в якості бонусу виявили деякі інші цікаві особливості мережі знайомств. Наприклад, Мілграм виявив, що більшість паспортів, які дійшли до кінця, зробили це через тільки трьох друзів цілі. Тобто велика частка зв'язків цілі з зовнішнім світом, здавалося, були пов'язані тільки з кількома його знайомими, явище, яке іноді називають «ефектом воронки». Мілграм назвав таких добре пов'язаних один з одним знайомих «соціометричними суперзірками», і їх існування іноді зазначалося також у інших мережах, таких як мережі співпраці, хоча не в деяких інших.

Ще одне цікаве наслідок експерименту Мілграма було виділено Клейнбергом. Той факт, що помірне число паспортів дійсно знайшли шлях до передбачуваного цільовим особі показує не тільки те, що короткі шляхи існують в мережі знайомств, але також і те, що люди вміють знаходити ці шляхи. У ретроспективі це досить дивний результат. Як показав Клейнберг, це можливо і характерно для мережі мати короткі шляхи між вершинами, але їх важко знайти, якщо тільки у людини немає повної інформації про структуру всієї мережі, якої у учасників досліджень Мілграма не було. Клейнберг запропонував можливе пояснення того, як учасники знайшли шляхи, ґрутуючись на припущеннях про структуру мережі.

Відносно недавно експеримент в маленькому світі був повторений Додсом з використанням електронної пошти в якості медіума. У цій версії експерименту учасники пересилали електронні листи знайомим в спробі донести їх до певної цільової людини, про яку їм було сказано кілька основних фактів. Експеримент покращився у порівнянні з мілграмовським з точки зору чистого обсягу, а також наявності набагато більш численних і різноманітних цільових осіб і відправних точок для повідомлень: було запущено 24 000 ланцюжків, більшість (хоча і не всі) з унікальними стартовими особами, і з 18 різними цілями в 13 різних країнах. З іншого боку, експеримент зазнав набагато нижчих показників участі, ніж Мілграм, можливо, тому що громадськість до теперішнього часу налаштована досить негативно до небажаної пошти. Із 24 000 ланцюгів тільки 384, або 1,5%, досягли поставленої мети, в порівнянні з 19% у випадку Мілграма. Проте, основні результати були схожі на результати Мілграма. Завершення ланцюга мали середню довжину трохи більше чотирьох кроків. Завдяки присутності кращих даних і значно більш ретельного статистичного аналізу Додса, можливо було також компенсувати ухили через незакінчені ланцюги і підрахувати, що справжня середня довжина шляху для експерименту була десь між п'ятьма і сімома кроками – дуже схоже на результат Мілграма. Однак в цьому випадку не спостерігається еквівалента «соціометричних

суперзірок» експерименту Мілграма, піднімаючи питання про те, чи була їх поява в справі Мілграма просто випадковістю конкретної цільової особистості, яку він вибрав, а не загальною властивістю соціальних мереж.

Цікавий варіант експерименту в маленькому світі був запропонований Кіллвортом і Бернардом, які цікавилися тим, як люди «орієнтуються» в соціальних мережах, і зокрема, як учасники експериментів в маленькому світі вирішують, кому пересилати повідомлення в спробі досягти зазначеної мети. Вони провели те, що вони називали «зворотним маленьким світом», експерименти, в яких вони просили учасників уявити, що вони беруть участь в експерименті в маленькому світі. (Фіктивне) повідомлення повинно було бути передано цільовій людині і учасників запитали, що вони хочуть знати про цілі, щоб прийняти рішення про те, кому переслати повідомлення. Фактична передача повідомлення ніколи не мала місце – експериментатори просто записували, які питання учасники задавали про мету. Вони виявили, що три характеристики були знайдені в переважній більшості частіше, ніж будь-які інші, а саме найменування цілі, їх географічне положення і рід занять – ті ж три елементи інформації, що Мілграм надав в своєму первісному експерименті. Деякі інші характеристики спливали з помірною частотою, особливо коли експеримент проводився в незахідних культурах або серед меншин: в деяких культурах, наприклад, походження або релігія вважалися важливими ідентифікуючими характеристиками мети [9].

У 1998 році прикладний математик з Корнельського університету Стівен Строгатц і його аспірант Данкан Воттс опублікували в журналі Nature статтю, в якій показали, що такі реальні мережеві структури як мережа нейронних зв'язків хробака нематода, мережа акторів Голлівуду і мережу з приблизно 5000 електростанцій США також мають властивість «тісного світу». Одночасно вони запропонували математичну модель мереж «тісного світу», властивістю якої є суперпозиція властивостей регулярних, періодичних решіток і пуасонівських випадкових мереж.

У 1999 році фізик з університету Нотр Дам (США) А.-Л. Барабаші разом зі своєю аспіранткою Р.Альберт вивчали властивості реальних мереж з дещо іншої точки зору. Якщо Строгатц і Воттс в своєму дослідженні мереж виходили з феномена «тісного світу», то Барабаші і Альберт вирішили дослідити закон розподілу вузлів реальних мереж за кількістю зв'язків.

Для багатьох мереж (метаболичні мережі і білкові взаємодії в клітинах, структура авіаційних повідомлень в США, структура Інтернету World Wide Web і т.д.) замість очікуваного імовірнісного розподілу вузлів по числу зв'язків за законом Пуассона, отриманий Барабаші і Альберт розподіл приблизно підпорядковується властивому всім критичним станам степеневому закону (формула 1.1):

$$P(x) = x^{-\gamma}, \quad (1.1)$$

де γ – постійна величина, для більшості реальних мереж $2 < \gamma \leq 3$;

x – передбачувана ступінь вузла.

Таким чином, у багатьох реальних мережах невелике число вузлів містить дуже велике число зв'язків (їх називають хабами, від англійського слова hub – концентратор), а величезна кількість вузлів містить лише кілька зв'язків. Такі мережі отримали назву масштабно-інваріантних мереж (scale-free networks). Ця назва не була придумана спеціально для цього типу мереж, а була взята з теорії критичних явищ, де флуктуації в критичних станах також підкоряються степеневому закону, а саму теорію масштабно-інваріантних мереж стали розглядати як один із сценаріїв виходу складних систем в критичний стан.

Барабаші і Альберт запропонували просту і елегантну модель виникнення та еволюції масштабно-інваріантних мереж. Вони показали, що для виникнення масштабно-інваріантних мереж необхідні дві умови:

- зростання – починаючи з невеликої кількості m_0 вузлів, на кожному кроці у часі додається один новий вузол з $m \leq m_0$ зв'язками, що поєднують цей новий вузол з m різноманітними вже існуючими вузлами;

- переважне приєднання (Preferential attachment). Принцип переважного приєднання полягає в тому, що чим більше зв'язків має вузол, тим більш імовірно для нього створення нових зв'язків. Вузли з найбільшим ступенем мають більше можливостей забирати собі зв'язки, що додаються в мережу.

Інтуїтивно, принцип переважного приєднання може бути зрозумілий, якщо ми думаємо в термінах соціальних мереж, які об'єднують людей. Тут зв'язок від А до В означає, що людина А «знає» або «знайома» з людиною В. Сильно пов'язані вузли представлені відомими людьми з великим числом зв'язків. Коли новачок потрапляє в співтовариство, для нього / неї більш імовірно зв'язатися з одним з відомих людей, ніж з відносно невідомим. Подібним чином у всесвітній мережі сторінки зв'язуються з хабами, наприклад, з добре відомими сайтами, як Гугл або Вікіпедія, ніж зі сторінками, які мало кому відомі. Якщо вибирати для зв'язку нову сторінку випадковим чином, то ймовірність вибору певної сторінки буде пропорційна її ступені. Це пояснює принцип переважного приєднання.

Сценарій зростання і еволюції масштабно-інваріантних мереж, запропонований Барабаші і Альберт, можна віднести до добре відомого в соціології явища, що одержало назву «ефект Матфея», маючи на увазі біблійний вислів «імущому дається ...». Є й інші афористичні позначення цього явища «багатий стає багатшим», «гроші до грошей», «успіх породжує успіх» і т.д. Цей принцип був узятий Прайсом і Саймоном для пояснення походження степеневого закону в соціальних явищах. Фізики розглядають такого роду явища як властивості критичних станів.

Обчислення показують, що принцип переважного приєднання дійсно призводить до масштабно-інваріантної мережі з показником ступеня $\gamma = 3$. Це досить рідко зустрічається в реальних мережах закон розподілу, але

цінність цієї моделі Барабаш-Альберт полягає в тому, що вона показує принципову можливість генерації масштабно-інваріантної мережі на базі простих припущень. На рисунку 1.9 представлений процес утворення масштабно-інваріантної мережі.

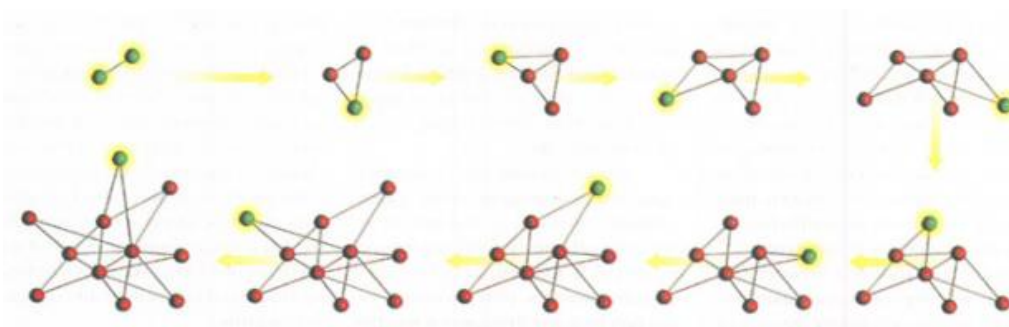


Рисунок 1.9 – Процес утворення масштабно-інваріантної мережі

Дослідження показують, що при довільному видаленні вузлів з пуассонівської випадкової мережі Ердеша-Рен'ї, існує певне критичне значення, що вимірюється відношенням числа видалених вузлів до загальної кількості вузлів в мережі, вище якого мережа розпадається на окремі фрагменти. Для масштабно-інваріантних мереж, коли показник ступеня $\gamma \leq 3$, такого критичного числа не існує. Отже, масштабно-інваріантні мережі дуже стійкі до випадкових пошкоджень або зовнішніх випадкових впливів. Ця стійкість пояснюється неомогенною топологічною структурою цих мереж. Однак, у таких мереж існує своєрідна «Ахіллесова п'ята» – цілеспрямоване пошкодження одного або кількох вузлів з великим числом зв'язків (degree centrality) веде до дезінтеграції мережі. Наприклад, хакер може істотно зашкодити WWW, якщо виведе з ладу один або кілька сайтів з великим числом зв'язків. З проблемою стійкості мережі при випадкових пошкодженнях тісно пов'язана задача дослідження поширення інфекції: як поширення вірусів в мережі WWW, мобільних телефонів або електронної пошти, так і поширення вірусів біологічної природи в соціальних мережах. Низька величина швидкості поширення захворювання

відповідає високій частці видалених вузлів або зв'язків гігантської пов'язаної компоненти мережі в випадково пошкодженій мережі. Відсутність епідемічного порогу означає неможливість розбити гігантську пов'язану компоненту на окремі фрагменти випадковим видаленням вузлів або зв'язків в нескінченних мережах. Якщо мережа скінченна, а всі реальні мережі скінченні, то гігантська пов'язана компонента може бути дезінтегрована випадковим видаленням досить великої, але скінченної частки вузлів, і, отже, епідемічний поріг існує.

1.3 Асортативність

Асортативність (англ. Assortative mating, assortment, nonrandom mating) – не випадковий підбір пар, в яких індивіди вибирають схожих (позитивна асортативність) або несхожих (негативна асортативність) за фенотипом партнерів. В результаті позитивної асортативності утворюються пари, схожі з яких-небудь рис (тобто зі схожим фенотипом), а при негативній асортативності або дизасортативності утворюються пари, які характеризуються додатковістю рис. Однак, як показують дослідження, практично завжди асортативність відбувається по позитивному шляху, тобто вибирають один одного індивіди, які найбільш схожі один на одного.

Асортативність вивчають такі науки, як генетика, психогенетика і психологія. З точки зору генетики, асортативність впливає на індивідуальні відмінності особин в популяції. Якщо властивість, по якій відбувається відбір, генетично обумовлена, то при асортативності відбувається:

- зникнення гетерозигот (що особливо важливо при патологіях, пов'язаних з рецесивним успадкуванням);
- зміна частоти генів у популяції (якщо якийсь варіант отримує репродуктивні переваги);
- зміна розподілу риси в популяції.

Вперше про доцільність дослідження асортативності заявлено в роботах Карла Пірсона (англійський математик, засновник математичної статистики, філософ, біолог, професор євгеніки), який в 1896 році запропонував використовувати коефіцієнт кореляції для математичної оцінки рівня подібності подружжя, а пізніше (в 1902 році) провів експериментальне дослідження асортативності за такими антропологічними ознаками, як зростання, довжина передпліччя і довжина руки з розправленими пальцями. У 1918 році Рональд Фішер (фахівець в області математичної статистики, прихильник євгеніки, теоретик і практик еволюційної генетики, автор концепції прогресивного відбору та інбридингу) на прикладі результатів, отриманих Пірсоном, математично обґрунтував необхідність і методологічні принципи оцінки асортативності при аналізі подібності родичів. До теперішнього часу в науці склалися численні напрямки вивчення не випадковості підбору подружніх пар, в тому числі: асортативність особистісної сфери подружжя, компліментарність (взаємодоповнюваність) подружжя, тобто наявність дизасортативності, вплив на ступінь асортативності шлюбів рівня інтелекту, освіти і соціального статусу подружжя, зміна подібності подружжя зі збільшенням стажу шлюбу, задоволеність шлюбом, взаєморозуміння подружжя, оцінка у подружжя перцептивно-сприйманих характеристик, таких як вік, зріст, вага, привабливість і володіння певними властивостями особистості, асортативність за національною приналежністю і місцем народження подружжя, за віросповіданням, вплив типу асортативності на кількість дітей в сім'ї.

Асортативне спарювання можна розділити на два типи: генетичне і соціо-культурне. Перший тип почали вивчати досить давно. Як вже говорили вище, в 1903 році Пірсон і його колеги повідомили про сильні кореляції по висоті, обсягу і довжині лівого передпліччя між чоловіками і дружинами в 1000 парах. На цьому асортативне спарювання щодо зовнішнього вигляду не закінчується, чоловіки вважають за краще жіночі

обличчя, які нагадують свої власні, в досвіді, коли їм показують зображення трьох жінок, причому одне зображення модифіковано так, щоб нагадувати їх власну фотографію. Однак той же результат не виявлено у жінок, які вибирають чоловічі особи.

1.4 Асортативність у складних мережах

Недавні дослідження структури мереж були зосереджені на невеликій кількості властивостей, які, як видається, є загальними для багатьох мереж, і можна очікувати, що вони вплинуть на функціонування мережевих систем. Серед них, мабуть, найкраще вивчені «ефект малого світу», транзитивність мережі або «кластеризація», а також розподіл ступенів. Однак багато інших властивостей були розглянуті і можуть бути однаково важливі, принаймні, в деяких системах. Приклади включають стійкість до видалення мережевих вузлів, можливість навігації або можливості пошуку в мережах, структуру співтовариства і спектральні властивості. Тут ми вивчаємо ще одну досить важливу особливість мережі – кореляцію між властивостями суміжних вузлів мережі, відому в літературі по екології і епідеміології як «асортативне змішування».

Один з підходів до дослідження, що лежать в основі генеративних процесів, полягає у вивченні кореляції або асортативного змішування атрибутів вузла (або «метаданих») по ребрах. Цей аналіз дозволяє нам зробити узагальнення щодо того, з більшою ймовірністю ми спостерігаємо зв'язку між вузлами з однаковими характеристиками (асортативність) або між вузлами з різними (дизасортативність). Соціальні мережі часто містять позитивні кореляції значень атрибутів через з'єднання. Ці кореляції виникають в результаті взаємодоповнюючих процесів відбору (або «гомофілія») і впливу (або «зараження»). Наприклад, асортативність часто спостерігається у відношенні віку, раси і соціального статусу, а також таких поведінкових моделей, як куріння і вживання алкоголю. Прикладами

дизасортативних мереж є гетеросексуальні мережі знайомств (стать), екологічні харчові мережі (метаболічна категорія), а також технологічні та біологічні мережі (ступінь вузла).

Найпростішим відображенням мережі є набір точок, зазвичай званих вершинами або вузлами, з'єднаних попарно в лінії, зазвичай звані ребрами або зв'язками. Більш складні мережеві моделі можуть вводити інші властивості вершин або ребер. Наприклад, ребра можуть бути спрямовані – вони вказують в одному конкретному напрямку – або можуть мати вагу, довжину або силу. Вершини можуть також мати ваги або інші числові величини, пов'язані з ними, або можуть бути взяті з деякого дискретного набору типів вершин. При вивченні соціальних мереж, патернів зв'язків між людьми в суспільстві, давно відомо, що ребра що з'єднують вершини незалежно від їх властивості або типу. Наприклад, зразки дружби між людьми сильно залежать від мови, раси і віку розглянутих людей, серед іншого.

Розглянемо рисунок 1.10, де показана мережа дружби дітей в американській школі, побудована на основі опитувальника. З рисунка випливає поділ мережі на дві групи. Виявляється, цей поділ в основному за ознакою раси. Різні відтінки вершин на зображенні відповідають учням різних рас, як зазначено в легенді, і показують, що школа розділена між групою, що складається в основному з чорношкірих дітей, і групою, що складається з переважно дітей білого кольору.

Це не новина для соціологів, які давно спостерігають і обговорюють такі поділи. Люди знаходять друзів, знайомства, бізнес-відносини, і багато інших типів зв'язків на основі всіх видів характеристик, в тому числі віку, національності, мови, доходу, рівня освіти і багато чого іншого. Майже будь-який соціальний параметр може бути використаний людьми для вибору своїх друзів. У людей спостерігається сильна тенденція до спілкування з іншими людьми, яких вони в деякому роді вважають схожими

на себе. Ця тенденція називається гомофілією або асортативним змішуванням.

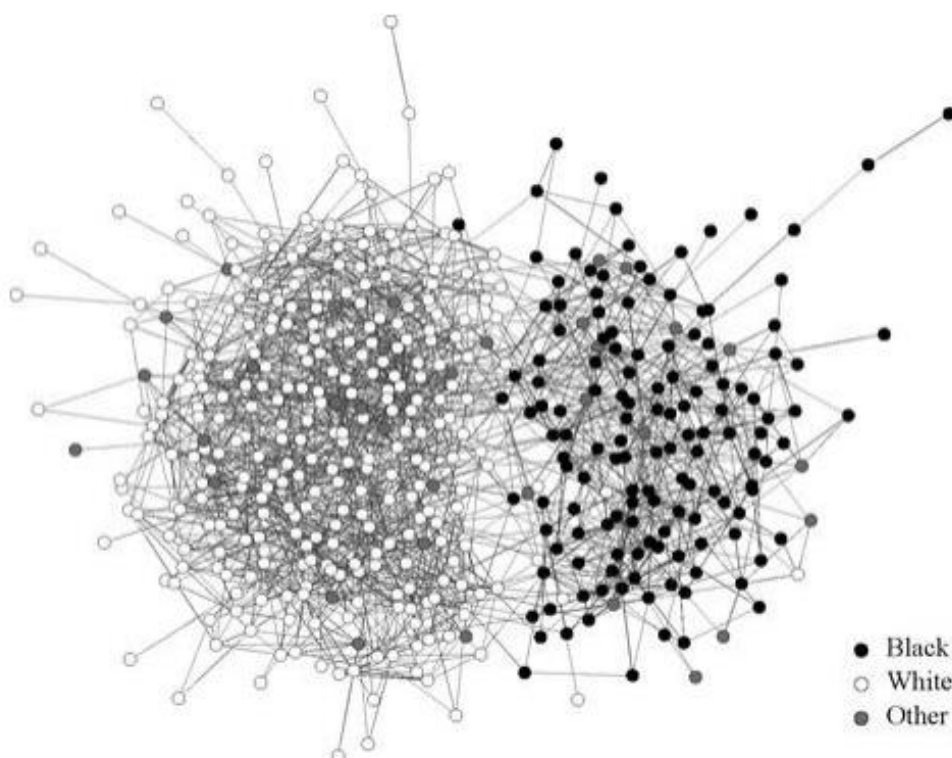


Рисунок 1.10 – Мережа дружби у середній школі США

Якщо люди вважають за краще спілкуватися з іншими людьми, які схожі на них, ми говоримо, що мережа демонструє асортативне мікшування або асортативне змішування. Якщо вони вважають за краще спілкуватися з тими, хто відрізняється, це показує дизасортативне змішування. Дружба, як правило, виявляється допоміжною по більшості характеристик. Ймовірно, найпоширеніший і знайомий приклад дизасортативного змішування – це змішування по статі в мережах сексуальних контактів. Більшість сексуальних партнерств відбувається між людьми протилежної статі, тому вони представляють собою зв'язок між людьми, які відрізняються по статі. Звичайно, одностатеві партнерства також зустрічаються, але вони представляють значно меншу частку зв'язків в мережі.

Асортативне змішування може надати глибокий вплив на структурні властивості мережі. Наприклад, асортативне змішування мережі по дискретній характеристиці матиме тенденцію розбивати мережу на окремі спільноти. Якщо люди вважають за краще дружити з іншими людьми, які говорять, наприклад, на своїй рідній мові, то можна очікувати, що країни з більш ніж однією мовою розділяться на співтовариства з мови. Асортативне змішування за віком може привести до розшарування суспільства за віком. І хоча основна увага в даному прикладі приділяється соціальним мережам, розумно припустити, що подібні ефекти змішування спостерігаються і в несоціальних мережах.

Характеризація рівня асортативності грає важливу роль в розумінні процесів формування зв'язків у складних системах. Однак глобальне сортування може не відображати відмінності в шаблонах мікшування в мережі в цілому. Розподіл мікшування в реальних мережах часто спотворено, перерозподілено і, можливо, є мультимодальним. Фактично, для деяких конфігурацій унімодальний розподіл може бути навіть неможливим.

У міру того, як мережеві дані ростуть, у гетерогенних підгруп з'являється більше можливостей для співіснування в загальній популяції. Наявність цих субпопуляцій доповнює продовжувані обговорення взаємодії між метаданими вузла і структурою мережі і передбачає, що, хоча ми можемо спостерігати взаємозв'язок між конкретними властивостями вузла і формуванням з'єднання в частині мережі, це не означає, що ці відносини існують по всій мережі в цілому.

У нашому суспільстві, в значній мірі орієнтованому на дані, такі області, як реклама, кредити, освіту, зайнятість і багато інших, мають величезну вигоду. Важливо, щоб ми пам'ятали про фундаментальні припущення, які ми робимо при аналізі цих даних. Одним з найбільш поширених припущень, зроблених при аналізі даних, є те, що спостережувані дані незалежно і однаково розподілені. У мережевій науці

припущення про незалежність явно порушується структурою мережі і наявністю (диз) асортативного змішування.

Ця неоднорідність не тільки кидає виклик традиційним допущенням в статистиці і машинному навчанні, але і відкриває нові потенційні можливості. Недавні результати вивчення задач прогнозування в мережах з гетерогенними схемами змішування демонструють, що при правильно сконструйованому алгоритмі навчання все ще можливо робити точні прогнози, а в деяких випадках навіть використовувати гетерогенність для подальшого підвищення продуктивності.

Стандартний підхід до кількісної оцінки рівня асортативності в мережі полягає в обчисленні коефіцієнта асортативності. Така зведена статистика корисна для захоплення середнього шаблону змішування по всій мережі. Однак таке узагальнення дійсно має сенс тільки в тому випадку, якщо воно відображає сукупність вузлів в мережі. Тобто, якщо асортимент більшості людей зосереджений навколо середнього значення, оскільки ми втрачаємо інформацію про різноманітність моделей змішування, присутніх в мережі. Коли мережі неоднорідні і містять різні схеми змішування, глобальна міра може не дати точного опису. Крім того, вона не надає засоби для кількісної оцінки різноманітності або виявлення аномалій або викидів моделей взаємодії.

Далі буде розглянуто, як можна кількісно оцінити асортативне змішування. Асортативне змішування дискретних характеристик, таких як раса, стать або національність, принципово відрізняються від змішування з скалярною характеристикою, такий як вік або доходи, тому два випадки необхідно розглядати окремо.

Припустимо, є мережа, в якій вершини класифікуються відповідно до деякої характеристики, що представляє із себе кінцевий набір можливих значень. Значення є перераховувальними – вони не потрапляють ні в який певний порядок. Наприклад, вершини можуть представляти людей і класифікуватися відповідно до національності, раси або статі. Або вони

можуть бути веб-сторінками, класифікованими по тому, якою мовою вони написані, або біологічні види, класифіковані по природному середовищі, або будь-яка з багатьох інших можливостей.

Мережа є асортативною, якщо значна частина її ребер проходить між вершинами одного і того ж типу, і простий спосіб кількісного визначення асортативності – виміряти цю частку. Однак це не дуже хороша міра, тому що, наприклад, це 1, якщо всі вершини належать до одного і того ж типу. Це тривіальний різновид асортативності: всі друзі людини, наприклад, теж люди, але це не дуже цікаве твердження. Замість цього хотілося б, щоб ця величина була великою в нетривіальних випадках, але малою в тривіальних.

Доброю мірою виявляється наступне. Знаходиться частка ребер, які проходять між вершинами одного типу, а потім з цього вираховується частка таких ребер яку можна було б очікувати, якби ребра були розташовані випадково, без урахування типу вершини. Для тривіального випадку, коли всі вершини одного типу, наприклад, 100% ребер проходить між вершинами одного типу, але це також очікувана цифра, так як інакше бути не може. Тоді різниця двох чисел дорівнює нулю, що говорить про те, що в цьому випадку нетривіальна асортативність не має місце бути. Тільки коли частка ребер між вершинами одного типу значно більше, ніж ми очікувана на підставі випадковості, величина дасть позитивний результат.

1.5 Постановка завдання дослідження

Мета даної атестаційної роботи – оцінювання показників асортативності масштабно-інваріантних мереж. Вона буде досягнута шляхом моделювання мереж і розрахунку їх показників асортативності. Необхідно провести аналіз того, що буде потрібно для успішного виконання даного завдання.

В першу чергу – розробка алгоритму генерації масштабно-інваріантної мережі. Для цього знадобиться вивчити природу даних мереж,

характерні для них архітектуру, тенденції зростання і розвитку. Можна також задіяти візуалізацію мереж / графів для зручності відображення результуючої мережі.

Після цього, коли буде готовий метод генерації / моделювання мережі, необхідні засоби для підрахунку її показника асортативності. Потрібно проаналізувати існуючі методи розрахунку показника асортативності мережі, порівняти їх переваги і недоліки, після чого вибрати підходящий і реалізувати.

І нарешті, необхідно вибрати програмні засоби, що дозволяють реалізувати розроблені алгоритми і методи – ті, що будуть зручні для розрахунків і мають бібліотеки / пакети для роботи з графами з реалізаціями операцій на зразок пошуку шляху і т.д.

2 МЕТОДИ ДОСЛІДЖЕННЯ

Асортативне змішування дискретних характеристик, таких як раса, стать або національність, принципово відрізняються від змішування з скалярною характеристикою, такий як вік або доходи, тому два випадки необхідно розглядати окремо.

Припустимо, є мережа, в якій вершини класифікуються відповідно до деякої характеристики, що представляє із себе кінцевий набір можливих значень. Значення є перераховувальними – вони не потрапляють ні в який певний порядок. Наприклад, вершини можуть представляти людей і класифікуватися відповідно до національності, раси або статі. Або вони можуть бути веб-сторінками, класифікованими по тому, якою мовою вони написані, або біологічні види, класифіковані по природному середовищі, або будь-яка з багатьох інших можливостей.

Мережа є асортативною, якщо значна частина її ребер проходить між вершинами одного і того ж типу, і простий спосіб кількісного визначення асортативності – виміряти цю частку. Однак це не дуже хороша міра, тому що, наприклад, це 1, якщо всі вершини належать до одного і того ж типу. Це тривіальний різновид асортативності: всі друзі людини, наприклад, теж люди, але це не дуже цікаве твердження. Замість цього хотілося б, щоб ця величина була великою в нетривіальних випадках, але малою в тривіальних.

Доброю мірою виявляється наступне. Знаходиться частка ребер, які проходять між вершинами одного типу, а потім з цього вираховується частка таких ребер яку можна було б очікувати, якби ребра були розташовані випадково, без урахування типу вершини. Для тривіального випадку, коли всі вершини одного типу, наприклад, 100% ребер проходить між вершинами одного типу, але це також очікувана цифра, так як інакше бути не може. Тоді різниця двох чисел дорівнює нулю, що говорить про те, що в цьому випадку нетривіальна асортативність не має місце бути. Тільки коли частка ребер між вершинами одного типу значно більше, ніж ми

очікувана на підставі випадковості, величина дасть позитивний результат [10].

2.1 Еластична модель масштабно-інваріантної мережі

Теорія масштабно-інваріантних мереж, як уже було згадано раніше, базується на двох концепціях: зростання і переважне приєднання. Концепція зростання має на увазі, що, починаючи з певної кількості вузлів, в кожен момент часу до графу / мережі додається новий вузол з деякою кількістю ребер з уже наявними вузлами. Переважне приєднання передбачає більшу ймовірність утворення ребра з новим вузлом для вузлів з найбільшим ступенем (хабів), ніж для вузлів з відносно меншим ступенем. Хоч подібні моделі і мають очевидні переваги, у них також є ряд обмежень і недоліків. У класичних моделях масштабно-інваріантних мереж показник розподілу ступенів вершин перевищує 2, в той час як в реальному світі час від часу зустрічаються мережі з показником, рівним 2, або менше. Крім того, в деяких мережах середня ступінь змінюється в міру спостережень, але ця залежність або не піддається розгляду, або вважається зовнішнім фактором середовища. Крім того, в класичних моделях показник розподілу ступенів вершин залежить безпосередньо від середньої їх ступеня. Щоб уникнути вищезгаданих недоліків і обмежень в роботі буде використана модель мережі, що додає до наявних двох концепцій третю: відносна швидкість збільшення кількості ребер і вершин, що змінюється з часом, тобто концепція еластичності.

Однією з найбільш відомих моделей неорієнтованих масштабно-інваріантних мереж є модель Барабаші-Альберт, запропонована фізиком з університету Нотр Дам (США) А.-Л.Барабаші разом з його аспіранткою Р.Альберт, що вивчали у той час властивості реальних мереж з точки зору розподілу вузлів по числу зв'язків. Дана модель заснована на наступних правилах:

- граф неорієнтований і має обмежену початкову кількість вершин і ребер;
- на кожному кроці до графу додається нова вершина, поєднана з m існуючими вершинами;
- кінці доданих ребер розподіляються між існуючими вершинами за принципом переважного приєднання: ймовірність того, що ребро інцидентне деякій вершині, пропорційно ступеню цієї вершини – тобто чим більше ребер інцидентно вершині, тим імовірніше вона створить ребро з новою вершиною.

Припустимо, що деякий граф $G(V, E)$, сполучений із множини вершин V та множини ребер E спочатку має дві вершини, поєднані ребром. Кожен момент часу до нього буде додаватися одна нова вершина і поєднуватися з однією вже існуючою вершиною, таким чином: $V(t) = t+1$, $E(t) = t$, $\Delta V(t) = 1$, $\Delta E(t) = 1$. Якщо порівняти відносні темпи зростання кількості вершин $\delta V(t) = \Delta V(t)/V(t)$ і ребер $\delta E(t) = \Delta E(t)/E(t)$, то можна отримати формулу 2.1:

$$\delta E(t) = 1 * \delta V(t-1), \quad (2.1)$$

де t – момент часу;

δE – відносний темп зростання кількості ребер;

δV – відносний темп зростання кількості вершин.

При розгляді випадку повного графу (нова вершина поєднується з усіма існуючими вершинами) з тими самими початковими умовами вищезгадане відношення прийме вид $\delta E(t) = 2 * \delta V(t-1)$. Двійка у цьому відношенні та одиниця на її місці у попередньому є так званим параметром еластичності, λ . Зазвичай в теорії мереж цей термін застосовується як позначення величини стійкості мережі щодо видалення вузлів. У більш широкому математичному розумінні він позначає відношення відносних швидкостей приросту функції і аргументу, в цьому випадку – відношення

відносного приросту кількості ребер до відносного приросту кількості вузлів (формула 2.2):

$$\lambda = \lim_{t \rightarrow \infty} \frac{\delta E(t)}{\delta V(t)}, \quad (2.2)$$

де t – момент часу;

δE – відносний темп зростання кількості ребер;

δV – відносний темп зростання кількості вершин;

λ – коефіцієнт еластичності.

Одним з важливих і корисних застосувань поняття еластичності є те, що вона встановлює відповідність між швидкістю приросту кількості ребер і поточним середнім значенням цієї кількості $e(t)$. Таким чином, при фіксованому значенні λ кількість нових ребер буде визначатися таким чином (формула 2.3):

$$\Delta E(t) = \lambda * e(t), \quad (2.3)$$

де t – момент часу;

ΔE – величина зміни кількості ребер;

$e(t)$ – середнє по доданим ребрам;

λ – коефіцієнт еластичності.

Величина $\lambda > 1$ означає, що кожна нова вершина додає більше ребер, ніж в середньому на поточний момент часу. У всіх існуючих моделях масштабно-інваріантних мереж ці співвідношення рівні, тобто $\lambda = 1$.

В роботі використовується модифікація розглянутої базової моделі масштабно-інваріантної мережі з використанням концепції різної відносної швидкості збільшення кількості дуг і вершин. Головна відмінність модифікованої моделі від базової полягає в тому, що середня кількість дуг

$m(t)$, які можна додавати в мережу, буде змінною і залежною від поточної кількості ребер і коефіцієнта еластичності (формула 2.4):

$$m(t) = \Delta E(t) \approx \lambda * e(t), \quad (2.4)$$

де t – момент часу;

ΔE – величина зміни кількості ребер;

$e(t)$ – середнє по доданим ребрам;

λ – коефіцієнт еластичності;

$m(t)$ – кількість ребер, що буде додано на поточному кроці.

2.2 Асортативне змішування по перераховувальним характеристикам

У математичних термінах позначимо через c_i клас або тип вершини i , яка є цілим числом $1...n_c$, де n_c – загальна кількість класів. Тоді загальна кількість ребер, що проходять між вершинами одного типу:

$$\sum_{\text{edges}(i, j)} \delta(c_i, c_j) = \frac{1}{2} \sum_{ij} A_{ij} \delta(c_i, c_j), \quad (2.5)$$

де $\delta(c_i, c_j)$ є дельтою Кронекера, а фактор $1/2$ бере в розрахунок той факт, що кожна пара вершин i, j рахується двічі у другій сумі.

Для розрахунку очікуваного числа ребер між вершинами для випадку, якщо ребра розташовані випадковим чином, необхідно виконати трохи більше роботи. Розглянемо конкретне ребро, прикріплене до вершини i , що має ступінь k_i . По визначенню є $2m$ кінців ребер у всій мережі, де m , як завжди, загальна кількість ребер, та ймовірність того, що другий кінець нашого конкретного ребра є одним з кінців k_j , прикріплених до вершини j , дорівнюють $k_j/2m$, якщо з'єднання виконані випадково. Враховуючи всі ребра k_i , прикріплені до вершини i , загальна очікувана кількість ребер між

вершинами i та j тоді дорівнює $k_i k_j / 2m$, а очікувана кількість ребер між усіма парами вершин одного типу:

$$\frac{1}{2} \sum_{ij} \frac{k_i k_j}{2m} \delta(c_i, c_j), \quad (2.6)$$

де фактор $\frac{1}{2}$, як і раніше, не дозволяє повторно рахувати пари вершин. Приймаючи різницю формул 2.5 та 2.6, можна отримати вираз для різниці між фактичною та очікуваною кількістю ребер, що поєднують вершини однакового типу у мережі:

$$\frac{1}{2} \sum_{ij} A_{ij} \delta(c_i, c_j) - \frac{1}{2} \sum_{ij} \frac{k_i k_j}{2m} \delta(c_i, c_j) = \frac{1}{2} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) \delta(c_i, c_j). \quad (2.7)$$

Традиційно розраховують не кількість таких ребер, а частку, яка визначається діленням цього виразу на кількість ребер m :

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) \delta(c_i, c_j), \quad (2.8)$$

Ця величина Q називається модульністю і є мірою ступені, в якій вершини пов'язані зі схожими собі в мережі. Це строго менше 1, набуває додатних значень, якщо є більше ребер між вершинами того ж типу, ніж ми очікували, і негативних, якщо менше.

Наприклад, для розглянутої раніше мережі, де типами є три етнічні класифікації: «чорний», «білий», і «інше», значення модульності $Q = 0,305$, що вказує (позитивне) асортативне змішування за расою в цій конкретній мережі. Негативні значення модульності вказують на дизасортативне змішування. Можна побачити негативну модульність, наприклад, в мережі сексуальних партнерств, де більшість партнерств відбувається між особами протилежної статі.

Величина

$$B_{ij} = A_{ij} - \frac{k_i k_j}{2m} \quad (2.9)$$

з'являється в ряді ситуацій при вивченні мереж. У деяких контекстах корисно розглядати B_{ij} як елемент матриці B , яка сама називається матрицею модульності.

Модульність завжди менше 1, але в цілому не досягає значення 1 навіть для ідеально змішаної мережі, в якій кожна вершина пов'язана тільки з іншими того ж типу. Залежно від розмірів груп і ступенів вершин, максимальне значення Q може бути значно менше, ніж 1. Це в деякому сенсі незадовільно: як відрізнити, коли присутнє сильне асортативне змішування, а коли ні? Щоб виправити проблему, можна нормалізувати Q шляхом ділення на його значення для ідеально змішаної мережі. З ідеальним змішуванням всі ребра знаходяться між вершинами одного типу i , отже, $\delta(c_i, c_j) = 1$ всякий раз, коли $A_{ij} = 1$. Це означає, що перший член суми у формулі 2.8 сумується до $2m$ і модульність для ідеально змішаних мереж дорівнює

$$Q = \frac{1}{2m} (2m - \sum_{ij} \frac{k_i k_j}{2m} \delta(c_i, c_j)), \quad (2.10)$$

тоді нормалізоване значення модулярності

$$\frac{Q}{Q_{\max}} = \frac{\sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) \delta(c_i, c_j)}{2m - \sum_{ij} \frac{k_i k_j}{2m} \delta(c_i, c_j)}. \quad (2.11)$$

Ця величина, яку іноді називають коефіцієнтом асортативності, тепер приймає максимальне значення 1 на ідеально змішаній мережі. Хоча це може бути корисною мірою в деяких обставинах, однак, формула 2.11 рідко

використовується. Найчастіше модульність використовується в своїй ненормалізованих формі, формула 2.8. Альтернативна форма для модульності, яка іноді корисна в практичних ситуаціях, може бути представлена наступним чином:

$$e_{rs} = \frac{1}{2m} \sum_{ij} A_{ij} \delta(c_i, r) \delta(c_j, s), \quad (2.12)$$

яка представляє собою частку ребер, які з'єднують вершини типу r з вершинами типу s , і

$$a_r = \frac{1}{2m} \sum_i k_i \delta(c_i, r), \quad (2.13)$$

яка являє собою частину решт ребер, прикріплених до вершин типу r . Потім, зазначивши, що

$$\delta(c_i, c_j) = \sum_r k_i \delta(c_i, r) \delta(c_j, r), \quad (2.14)$$

з формули 2.8 можна отримати:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) \sum_r \delta(c_i, r) \delta(c_j, r) =$$

$$\sum_r \left[\frac{1}{2m} \sum_{ij} A_{ij} \delta(c_i, r) \delta(c_j, r) - \frac{1}{2m} \sum_i k_i \delta(c_i, r) \frac{1}{2m} \sum_j k_j \delta(c_j, r) \right] = \quad (2.15)$$

$$\sum_r (e_{rr} - a_r^2)$$

Ця форма може бути корисна, наприклад, коли у нас є мережеві дані у вигляді списку ребер і типів вершин на їх кінцях, але немає явних даних про ступені вершин. В такому випадку e_{rs} та a_r буде відносно легко розрахувати, у той час як формула 2.8 буде досить незручною для коритсування.

2.3 Асортативне змішування по скалярним характеристикам

Гомофілія також може проявлятися в мережі і зі скалярними характеристиками, такими як вік або доходи. Це характеристики, значення яких йдуть в певному порядку, так що можна сказати не тільки коли дві вершини абсолютно однакові відповідно до характеристики, але і коли вони приблизно однакові. Наприклад, хоча двоє людей, безумовно, можуть бути ровесниками – навіть народжені в один і той же день – вони можуть бути приблизно одного віку – народжені з проміжком у пару років один від одного, скажімо, – і люди можуть (і насправді часто так і роблять) вибирати, з ким вони зв'язуються на основі таких приблизних вікових груп. Немає еквівалентного приблизного визначення подібності для раніше обговорюваних перераховувальних характеристик з попереднього розділу: немає сенсу, в якому люди з Франції та Німеччини, скажімо, більш близькі за національністю, ніж люди з Франції та Іспанії.

Якщо мережеві вершини з однаковими значеннями скалярної характеристики мають тенденцію з'єднуватися разом частіше, ніж ті, які мають різні значення, то мережа вважається асортативно змішаною згідно з цією характеристикою. Якщо, наприклад, люди дружать з іншими людьми того ж віку що і вони, то мережа є асортативно змішаною за віком. Іноді також можна почути, що мережа стратифікована за віком, що означає те ж саме – можна думати про вік як про одновимірну шкалу або ось, при цьому індивідууми різного віку утворюють пов'язані «страти» всередині мережі.

Можна розглянути рисунок 2.1, який показує дані про дружбу для того ж набору школярів США, що і раніше, але тепер як функцію віку. Кожна точка на малюнку відповідає одній парі друзів і положення точки вздовж двох осей дає вік друзів. Як видно з малюнка, серед учнів спостерігається істотне асортативне змішування за віком: багато точок лежать в прямокутниках поряд з діагональною лінією, що представляє дружбу між учнями в одному класі. У цьому випадку також спостерігається помітна тенденція для учнів мати більше друзів широкого спектру віку у міру їх дорослішання, так що щільність точок у верхньому правому квадраті менше, ніж в нижньому лівому.

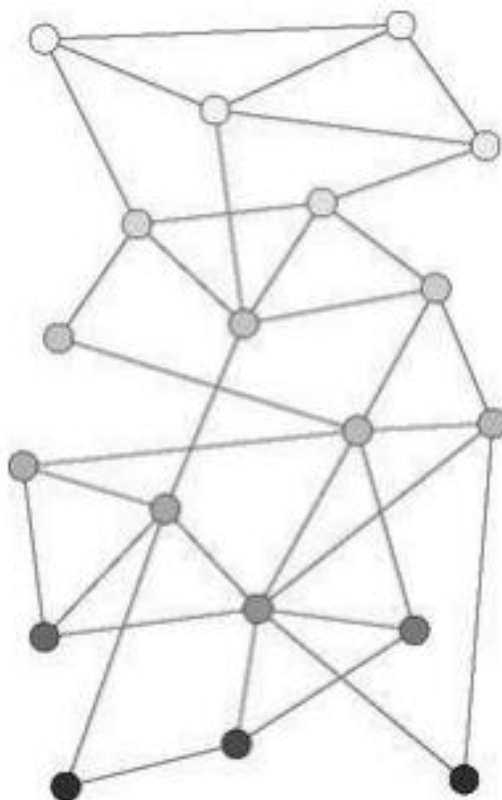


Рисунок 2.1 – Мережа дружби дітей у школі за віком

На графіку на рисунку 2.2 кожна точка відповідає одному з ребер на графі з рисунка 2.1, а його положення уздовж горизонтальної та вертикальної осей дає вік двох чоловік на кожному кінці цього ребра. Вік вимірюється в термінах класів учнів, які йдуть від 9 до 12. Насправді, класи в шкільній системі США не відповідають в точності віку, оскільки учні можуть поступати до школи рано чи пізно, залишатися на другий рік і т.д. (Кожен учень розташований випадковим чином в межах інтервалу, що відображає його клас, щоб розсіяти точки на графіку. Варто звернути увагу, що кожна дружба з'являється двічі, вище і нижче діагоналі.) [11].

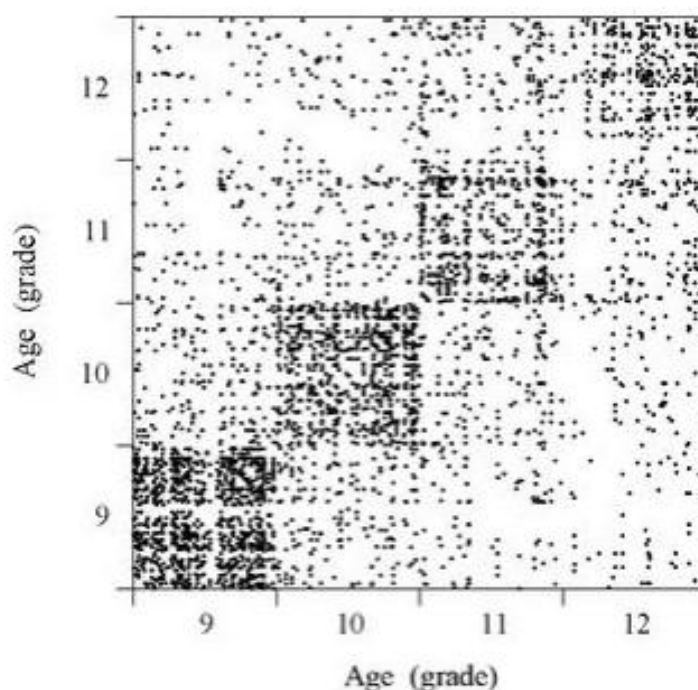


Рисунок 2.2 – Графік віку пар друзів у школі

Можна зробити грубу міру асортативного змішування по скалярним характеристикам, адаптуючи ідеї попереднього розділу. Можна згрупувати вершини згідно характеристиці, що нас цікавить (скажімо, вік), а потім розглядати групи як окремі «типи» вершин. Наприклад, можна згрупувати людей за віком в діапазоні від одного року до десяти років, однак це упускає

більшу частину пункту про скалярні характеристики, оскільки розглядає вершини, зазначені в одній і тій самій групі як вершини однакового типу, в той час як вони можуть бути рівні тільки приблизно, а вершини в різних групах як абсолютно різні, хоча насправді вони можуть бути досить схожими.

Кращим підходом є використання коваріаційної міри наступним чином. Нехай x_i буде значенням для вершини і скалярної величини, що нас цікавить (вік, доходи і т. д.). Розглянемо пари значень (x_i, x_j) для вершин на кінцях кожного ребра (i, j) у мережі, і розрахуємо їх коваріацію по всім ребрам наступним чином. Середнє μ по значенням x_i на кінці ребра визначається так:

$$\mu = \frac{\sum_{ij} A_{ij} x_i}{\sum_{ij} A_{ij}} = \frac{\sum_i k_i x_i}{\sum_i k_i} = \frac{1}{2m} \sum_i k_i x_i. \quad (2.16)$$

Варто звернути увагу, що це не просто середнє значення x_i , усереднене по всім вершинам. Це середнє по ребрам, а оскільки вершина зі степенню k_i лежить на кінцях k_i ребер та фігурує у середньому k_i разів при розрахунку середнього (звідси коефіцієнт k_i в сумі). Тоді коваріація x_i та x_j по ребрам дорівнює

$$\begin{aligned} \text{cov}(x_i, x_j) &= \frac{\sum_{ij} A_{ij} (x_i - \mu)(x_j - \mu)}{\sum_{ij} A_{ij}} = \frac{1}{2m} \sum_{ij} A_{ij} (x_i x_j - \mu x_i - \mu x_j + \mu^2) = \frac{1}{2m} \sum_{ij} A_{ij} (x_i x_j - \\ &\mu^2) = \frac{1}{2m} \sum_{ij} A_{ij} x_i x_j - \frac{1}{(2m)^2} \sum_{ij} k_i k_j x_i x_j = \\ &\frac{1}{2m} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) x_i x_j. \end{aligned} \quad (2.17)$$

Коваріація буде позитивною, якщо значення x_i, x_j на будь-якому кінці ребра обидва є великими або обидва маленькими і негативні, якщо вони мають тенденцію змінюватися в протилежних напрямках. Іншими словами,

коваріантність буде позитивною, коли у нас є асортативне змішування, і негативною для дизасортативного змішування.

Як і у випадку з модульністю, іноді зручно нормалізувати коваріацію, так, щоб вона приймала значення 1 в ідеально змішаній мережі, в якій всі ребра проходять між вершинами з точно рівними значеннями x_i (хоча в більшості випадків таке було б вкрай мало ймовірно на практиці). Після вставки $x_j = x_i$ до формули 2.16 можна отримати ідеальне значення змішування у вигляді:

$$\frac{1}{2m} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) x_i^2 = \frac{1}{2m} \sum_{ij} (k_i \delta_{ij} - \frac{k_i k_j}{2m}) x_i x_j \quad (2.18)$$

і нормалізована міра, іноді звана коефіцієнтом асортативності, являє собою співвідношення:

$$r = \frac{\sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) x_i x_j}{\sum_{ij} (k_i \delta_{ij} - \frac{k_i k_j}{2m}) x_i x_j} \quad (2.19)$$

Хоча це може бути не одразу очевидно, насправді це приклад коефіцієнта кореляції Пірсона, який має коваріацію в чисельнику і дисперсію в знаменнику. Коефіцієнт кореляції варіюється в значенні між максимумом 1 для ідеально асортативної мережі і мінімумом -1 для ідеально дизасортативної. Нульове значення означає, що значення x_i на кінцях ребер некорельовані.

Для даних з рисунка 2.2 коефіцієнт кореляції приймається рівним $r = 0,616$, що вказує на те, що мережа учнівської дружби має значуще асортативне змішування за віком – учні, як правило, схильні дружити з тими, хто їм близький за віком.

Було б також можливо мати асортативне (або дизасортативне) змішування відповідно з векторними характеристиками, з вершинами,

вектори яких мають схожі значення, виміряні деякої відповідною метрикою, будучи більш (або менш) ймовірно з'єднаними ребром. Один із прикладів такого змішування – це формування дружніх відносин між людьми відповідно до їх географічного положення, місце розташування визначається двовимірним вектором, наприклад, координати широти / довготи. Це, безумовно, той випадок, коли люди зазвичай дружать з тими, хто живе географічно близько до них, тому можна очікувати, що змішування цього типу буде асортативним. Однак, в даний час в літературі по мережах не приділяється великої уваги методам асортативного змішування векторів [12].

2.4 Асортативне змішування за ступенем

Окремим випадком асортативного змішування по скалярній величиною, що становить особливий інтерес, є змішування за ступенем. У мережі, яка показує ознаки асортативного змішування за ступенем, вершини високого ступеня будуть переважно пов'язані з іншими вершинами високого ступеня, а низького ступеня – відповідно, з іншими вершинами низького ступеня. У соціальній мережі, наприклад, ми маємо асортативне змішування за ступенем, якщо товариські люди дружать з іншими товариськими людьми, а відлюдники – з іншими відлюдниками. І навпаки, могло б мати місце дизасортативне змішування за ступенем, що означало б, що товариські люди водяться з відлюдниками і навпаки.

Причина, по якій цей конкретний випадок цікавий, полягає в тому, що, на відміну від віку або доходу, ступінь вершини сама по собі є властивістю структури мережі. Маючи одну структурний властивість (ступені вершин), що впливає в іншу (положення ребер) породжує деякі цікаві особливості в мережах. Зокрема, в асортативній мережі, де вузли високого ступеня мають тенденцію триматися разом, можна очікувати сукупність або ядро таких вузлів високого ступеня в мережі, оточене менш щільною периферією

вузлів з більш низьким ступенем. Ця структура ядра / периферії є спільною рисою соціальних мереж, багато з яких виявляються асортативно змішаними за ступенем. Рисунок 2.3 показує невелику асортативно змішану мережу, в якій добре видно структуру ядра / периферії.

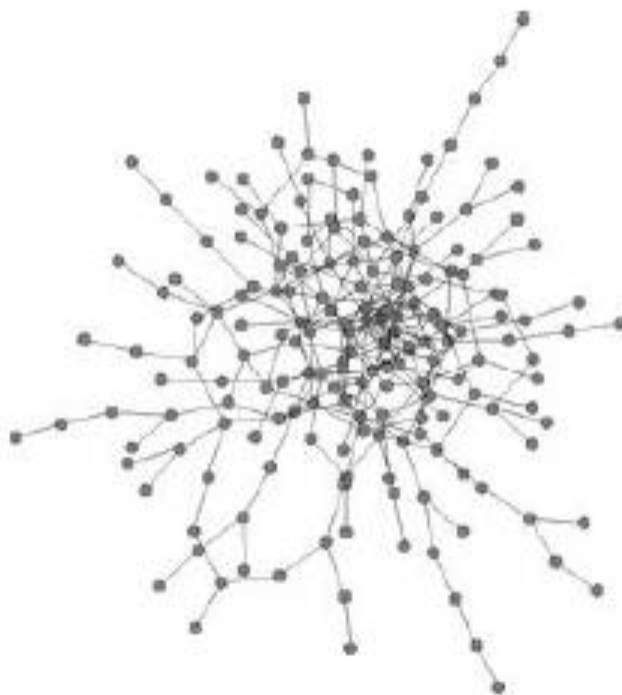


Рисунок 2.3 – Мережа, що є асортативною за ступенем

З іншого боку, якщо мережа є дизасортативно змішаною за ступенем, то вершини з високим ступенем схильні до підключення до вершин з низьким ступенем, створюючи зіркоподібні фігури в мережі, які можна легко бачити на рисунку 2.4, на якому наведено приклад невеликої дизасортативної мережі. Дизасортативні мережі зазвичай не розділяються на ядро / периферію, а є більш однорідними.



Рисунок 2.4 – Мережа, що є дизасортативною за ступенем

Асортативне змішування за ступенем може бути виміряне так само, як змішування відповідно до будь-якої іншої скалярної величини. Ми визначаємо коваріацію типу, описаного формулою 2.17, але з x_i теперь дорівнюючим ступеню k_i :

$$\text{cov}(k_i, k_j) = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) k_i k_j, \quad (2.20)$$

або, якщо потрібно, можна нормалізувати за максимальним значенням коваріації, щоб отримати коефіцієнт кореляції або коефіцієнт асортативності:

$$r = \frac{\sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) k_i k_j}{\sum_{ij} (k_i \delta_{ij} - \frac{k_i k_j}{2m}) k_i k_j}. \quad (2.21)$$

Варто зазначити, що для розрахунку формул 2.20 і 2.21 потрібна тільки структура мережі і ніякої іншої інформації (на відміну від розрахунків для інших форм асортативного змішування). Як тільки стає відомою матриця суміжності (i , отже, ступені) всіх вершин, можна розрахувати коефіцієнти без жодних труднощів. Можливо, з цієї причини змішування за ступенем є одним з найбільш часто вивчаємих типів асортативного змішування [13].

Таким чином, асортативне змішування або гомофілія – це тенденція вершин з'єднуватися з іншими, схожими на них якимось чином. В цьому і попередньому розділі було наведено кілька прикладів з соціальних мереж, таких як дружба в старшій школі, в яких школярі схильні більше спілкуватися з людьми того ж етнічного походження або віку, що і вони самі. Особливий інтерес представляє асортативне змішування за ступенем, схильність вершин з'єднувати інші зі ступенями, які схожі на їхні власні. Може також мати місце дизасортативне змішування за ступенем, у разі якого вершини з'єднуються з іншими з дуже різними ступенями. Як було показано, асортативне / дизасортативне змішування може вказати істотний вплив на структуру мережі. Його можна оцінити шляхом розрахунку коефіцієнта асортативності (формула 2.21). Якщо ми збираємося розрахувати значення цього коефіцієнта, однак, ми не повинні робити це безпосередньо з цього рівняння, тому що подвійна сума по вершинах i і j має забагато доданків (n^2 , якщо бути точніше) і в випадку мереж досить великих розмірностей може вилитися у величезні витрати по часу роботи і пам'яті. Замість цього можна використовувати

$$r = \frac{S_1 S_e - S_2^2}{S_1 S_3 - S_2^2}, \quad (2.22)$$

де

$$S_e = \sum_{ij} A_{ij} k_i k_j = 2 \sum_{\text{edges}(i,j)} k_i k_j, \quad (2.23)$$

а також

$$S_1 = \sum_i k_i, \quad S_2 = \sum_i k_i^2, \quad S_3 = \sum_i k_i^3. \quad (2.24)$$

Комп'ютерний час, необхідний для розрахунків різних показників мереж, є важливою темою, яка потребує окремого обговорення. Той, хто мав досвід програмування, а конкретніше – написання програми для виконання конкретного розрахунку і його налаштування майже напевно опинявся в ситуації, коли запустив написану ним програму, тільки щоб виявити, що вона все ще продовжує працювати через годину або навіть через день. Виконавши деякі додаткові розрахунки, людина з жахом виявляє, що на те, щоб закінчити розпочате потрібна тисяча років, і, отже, що написана ним програма в основному марна.

Концепція обчислювальної складності (або просто «складність» для стислості), по суті, корисна саме тому, що допомагає не витратити енергію на програми, які не будуть завершені ні за яких обставин за невизначену кількість часу. З огляду на складність алгоритму, перш ніж навіть приступити до написання комп'ютерної програми, потрібно бути впевненим в тому, що пишеш програму, яка в змозі завершити свою роботу, і що не гаєш час попусту.

Обчислювальна складність є мірою часу роботи комп'ютерного алгоритму. Можна розглянути простий приклад: скільки часу потрібно, щоб знайти найбільше число у списку з n чисел? Якщо припустити, що числа не даються в якомусь особливому порядку (наприклад, у порядку зростання величини), то немає швидшого способу знайти найбільше число, ніж просто

пройти по всьому списку, елемент за елементом, зберігаючи найбільшого число, яке ми бачили, поки не дійдемо до кінця.

Це дуже простий приклад комп'ютерного алгоритму. Його можна було б використовувати, наприклад, щоб знайти вершину в мережі, яка має вищий ступінь. Алгоритм складається з ряду кроків, один для кожного елемента в списку. На кожному кроці вивчається наступний елемент списку і питається, чи є він більше, ніж найбільший, який ми бачили до сих пір. Якщо це так, він стає новим найбільшим, в іншому випадку нічого не відбувається, і ми переходимо до наступного кроку.

І тут настає ключовий момент: в гіршому випадку найбільша частина роботи, що нам доведеться зробити для цього алгоритму – на кожному кроці перевірити наступне число, порівняти його з нашим попереднім найбільшим і замінити попереднього найбільшого на новий елемент. Тобто найбільший обсяг роботи, яку доведеться виконати, відбувається в тому випадку, коли кожне число більше, ніж всі попередні.

У цьому випадку обсяг виконуваної роботи однаковий на кожному етапі i , отже, загальний час виконання алгоритму одно $n\tau$, де τ – час, витрачений на кожен крок. Якщо пощастить, фактичний час може бути менше, ніж ця величина, але ніколи не буде більше. Таким чином йдеться про те, що час виконання або тимчасова складність цього алгоритму порядку n або просто $O(n)$ для стислості.

Технічно, обчислювальна складність алгоритму є показником того, як час роботи алгоритму залежить від розмірності його вхідних даних. У нашому прикладі вхідними даними є список чисел, а розмір цих вхідних даних – довжина n списку. Якби цей алгоритм був використаний для того, щоб знайти вузол найвищого ступеня в мережі, тоді розмір вхідного сигналу був би кількістю вершин в мережі. У багатьох мережевих алгоритмах кількість вершин n буде важливим параметром, який буде розглядатися. В інших випадках важливим параметром буде число ребер m в мережі, в той час як в інших знадобляться обидва m і n , щоб повністю вказати розмір

вхідних даних – в алгоритмі можуть бути різні частини, наприклад, які працюють залежно окремо від вершин і ребер, і тоді загальний час роботи залежить від обох. Так, наприклад, алгоритм, відомий як «пошук в ширину», який використовується для пошуку геодезичних шляхів в мережах, має обчислювальну складність $O(m) + O(n)$ для мережі з m ребрами і n вершинами, що означає, що вона працює в часі $am + bn$, де a і b – константи, або швидше [14].

Як можна бачити, сума у формулі 2.23 має m членів, де m - кількість ребер в мережі, а суми в формулі 2.24 мають n членів кожна, тому формулу 2.22 зазвичай набагато швидше оцінити, ніж формулу 2.21.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ

3.1 Вибір мови програмування та програмного забезпечення

Для практичної реалізації були обрані мова програмування Python, що володіє чітким і послідовним синтаксисом, продуманою модульністю і масштабованістю, завдяки чому вихідний код написаних на ньому програм легко читається, і в якості модуля для роботи з графами – бібліотека NetworkX, призначена для створення, маніпуляції і вивчення структури, динаміки і функціонування складних мережевих структур. Вона дає більшість стандартних мережевих метрик, дозволяє роботу з різними типами зв'язків і вузлів (вузли та зв'язки можуть бути чим завгодно), а найголовніше – весь функціонал, простоту і швидкість Python. Це вільне ПЗ, яке розповсюджується під BSD ліцензією.

Основні можливості бібліотеки:

- класи для роботи з простими, орієнтованими та зваженими графами;
- вузлом може бути практично що завгодно: текст, зображення, XML;
- збереження / завантаження графів у / з найбільш розповсюджених форматах файлів зберігання графів;
- вбудовані процедури для створення графів базових типів;
- методи для знаходження підграфів, клік та K-дольних графів (максимальний підграф, у якому кожна вершина має хоча б рівень K);
- здобуття таких характеристик графа, як ступені вершин, висота графа, діаметр, радіус, довжини шляхів, центр, проміжності та ін.;
- візуалізація мереж у вигляді 2D та 3D графіків.

Заявляється, що бібліотека вільно може оперувати дуже великими мережевими структурами, рівня графа з 10 мільйонами вузлів і 100 мільйонами дуг між ними. З причини того, що він базується на низкоуровневої структурі даних мови Python під назвою «словник-

словників», пам'ять витрачається ефективно, графи добре масштабуються, мало залежать від особливостей операційної системи, в якій виконується скрипт і відмінно підходять для популярного на даний момент напрямку аналізу даних із соціальних мереж і графів.

Бібліотека організована у вигляді ієрархії пакетів. Верхній рівень в кожному пакеті надає загальні методи маніпуляції його структурами, ті, що нижче, набувають більшої і більшої спеціалізації.

Підтримуються наступні основні типи графів:

- Graph – реалізація простого неорієнтованого графа. Додаткові вершини між двома вузлами ігноруються, можливі вузли, що зв'язані самі з собою (петлі);
- DiGraph – орієнтований граф, додані функції і обмеження, специфічні для цього типу графів;
- MultiGraph – реалізація мультиграфів, в таких графах можливе існування пар вершин, які з'єднані більш ніж одним ребром (ненаправленим), або більш ніж двома дугами протилежних напрямків;
- MultiDiGraph – власне, орієнтований мультиграф.

Внутрішнє представлення графів реалізовано у вигляді списків суміжності. Проте, щоб уникнути появи неузгодженості, всі операції з графами повинні проводитися не напряму з цим списком, а з використанням API функцій бібліотеки.

Будь-який вузол або дуга мають унікальний ідентифікатор, за яким можна отримати всю інформацію, що пов'язана з ним, також додатково можуть існувати імена, більш зручні для реалізації поточного алгоритму ніж ідентифікатори, що також дозволяють отримати ці дані.

Додатково кожен вузол або дуга можуть мати будь-яку кількість атрибутів, що зберігають різні типи даних. Зважені графи мають службовий атрибут з назвою «weight» і ця назва не може бути використана для зберігання іншої інформації, щоб уникнути руйнування внутрішньої логіки його подання.

На даний момент граф може бути створений з використанням одного з трьох методів:

- генератор графів – вбудовані класи для створення графів загальних топологій, таких як повні графи різноманітних рівнів, збалансовані дерева, циклічні графи, графи Дороговцева-Гольтцева-Мендеса, випадкові біноміальні графи, та багатьох інших типів;
- завантаження даних і формування графа на основі файлу або потоку даних одного з підтримуваних форматів;
- послідовне додавання вузлів і дуг.

Окрім створення графа зазвичай потрібно отримувати інформацію про його вузли, дуги, шляхи та ін. Основними методами для цього є отримання масивів вузлів і дуг (`edges ()` і `nodes ()` відповідно), а також отримання ітератора по вузлах і дуг (`edges_iter ()` і `nodes_iter ()` відповідно).

Додатково існує велика кількість функцій отримання більш специфічної інформації про граф, наприклад `nx.triangles (G, n)` поверне кількість трикутників в графі `G`, в яких вершина `n` є одним з вузлів.

У бібліотеці реалізовано велику кількість типових для роботи над графами алгоритмів. Реалізовані такі алгоритми як знаходження найкоротшого шляху, пошуку в висоту і ширину, кластеризація, знаходження ізоморфізму графів і багато іншого [15].

Основною метою бібліотеки є робота з графами, і їх візуальне відображення відходить на другий план, але все ж реалізовано, оскільки є важливим інструментом аналізу.

Надано зручні методи для відображення графів з використанням Python бібліотеки `Matplotlib` або зовнішнього модуля `Graphviz` для більш складних випадків.

3.2 Реалізація алгоритму генерації мережі

Основна ідея еластичної моделі масштабно-інваріантної мережі полягає в наступному: починаючи з деякого вже існуючого початкового графа, в кожен момент часу з'являється одна нова вершина і утворює деяку кількість нових ребер з уже існуючими вершинами, виходячи з принципу переважного приєднання – чим більше ступінь вершини, тим імовірніше утворення з нею нового ребра. У класичній моделі Барабаші-Альберт ця кількість є постійною, в еластичній же моделі, де змінюється відносна швидкість збільшення кількості ребер і вершин, воно визначається множенням коефіцієнта еластичності на середню кількість доданих вершин на даний момент часу.

Для успішної реалізації даного алгоритму можна скористатися вже наявною в бібліотеці NetworkX реалізацією генерації моделі мережі Барабаші-Альберт, `barabasi_albert_graph`. Еластична модель при генерації слідує всіма тими же кроками, що і класична модель Барабаші-Альберт (додавання вершини в граф, визначення інцидентних вершин за принципом переважного приєднання), за винятком одного – визначення кількості нових ребер. У еластичній моделі, на відміну від класичної, це число не є постійним і визначається заново на кожній ітерації, виходячи з коефіцієнта еластичності, встановленого користувачем / оператором і історії доданих ребер на даний момент (кількість ребер, доданих на кожному кроці, доведеться зберігати і тягнути за собою протягом роботи алгоритму). У прикладі 3.1 наведено вихідний код генератора моделі мережі Барабаші-Альберт в незайманому вигляді:

```

def barabasi_albert_graph(n, m, seed=None):
    if m < 1 or m >= n:
        raise nx.NetworkXError("Barabási-Albert network must have "
                                "m >= 1 and m < n, m = %d, n = %d" % (m, n))
    if seed is not None:
        random.seed(seed)

    # Add m initial nodes (m0 in barabasi-speak)
    G = empty_graph(m)
    # Target nodes for new edges
    targets = list(range(m))
    # List with nodes repeated once for each adjacent edge
    repeated_nodes = []
    # Start adding the other n-m nodes. The first node is m.
    source = m
    while source < n:
        # Add edges to m nodes from the source.
        G.add_edges_from(zip([source] * m, targets))
        # Add one node to the list for each new edge just created.
        repeated_nodes.extend(targets)
        # And the new node "source" has m edges to add to the list.
        repeated_nodes.extend([source] * m)
        # Now choose m unique nodes from the existing nodes
        # Pick uniformly from repeated_nodes (preferential attachment)
        targets = _random_subset(repeated_nodes, m)
        source += 1
    return G

```

Приклад 3.1 – Реалізація генерації моделі мережі Барабаші-Альберт в бібліотеці NetworkX

При виконанні функції `barabasi_albert_graph` в неї передається три параметра: два обов'язкових і один опціональний. Параметр `n` відповідає за кількість вершин, що додаються – кількість проведених ітерацій, тому що за одну ітерацію додається одна вершина. Параметр `m` – кількість ребер, утворених новою вершиною при додаванні в граф. І, нарешті, параметр `seed` присутній тут для генератора випадкових чисел `random`. Передача `random.seed(seed)` в `random` параметра `seed` перед викликом `random.random`

дозволяє отримати з генератора деяку гарантовану послідовність чисел для даного параметра `seed`. При звичайному виклику `random.random()` на виході виходять різні числа. Це ясно і природньо, але може бути не дуже зручним, коли потрібно знайти помилку і тоді зручніше отримувати один і той самий розклад для кожного запуску програми. Для цього і існує функція `random.seed()`, що викликається відразу ж перед початком генерації чисел. Якщо не встановити `seed` вручну, то модуль встановлює його за значенням поточного часу. Тому кожен раз `seed` матиме інше значення, отже, і набори випадкових чисел будуть різні.

Спершу алгоритм перевіряє, чи не є переданий параметр `m` менше одиниці або дорівнює або більше, ніж переданий параметр `n`. Кількість ребер, утворених новою вершиною при додаванні в граф, не може бути негативним або бути рівним нулю – якщо нова вершина не буде утворювати нові ребра, генерована модель піде врозріз з концепцією зростання моделі Барабаші-Альберт. Якщо кількість додаваних ребер дорівнюватиме або більше підсумкової кількості доданих ребер, алгоритм закінчується на першій ітерації, навіть не розпочавшись, що теж йде врозріз з концепціями генерованої моделі. Якщо хоча б одна з умов не задоволена – алгоритм видасть помилку і повідомленням сповістить користувача про те, що він зробив не так.

Після цього перевіряється, чи був переданий в функцію параметр `seed` для генератора випадкових чисел. Якщо так, то викликається функція `random.seed()` з передачею в неї відповідного параметра, якщо немає – алгоритм просто продовжує роботу і в подальшому при виклику `random.random()`, як вже було згадано раніше, в якості `seed` буде використано значення поточного часу.

Для того, щоб нова вершина могла утворити `m` нових ребер, необхідно, щоб в графі було присутнє щонайменше `m` вершин, окрім тієї, що додається. З цієї причини перед запуском циклу необхідно створити граф, що має хоча б `m` вершин, що і відбувається при виконанні функції

`empty_graph(n)`. Даний граф порожній, що означає, що присутні в ньому вершини не пов'язані один з одним ребрами. Насправді, початковому графу бути порожнім зовсім не обов'язково, їм може бути будь-який граф, але в даній реалізації він порожній.

Далі створюється список `targets`, в якому надалі будуть зберігатися вершини, обрані для утворення з ними нових ребер. Оскільки початковий граф має рівно m вершин, вибирати нема з чого – всі наявні вершини додаються в список.

Після створюється список `repeated_nodes`. У ньому ідентифікатор кожної вершини графа буде зберігатися в стільки ж примірниках, скільки зв'язків має дана вершина. Саме з цього списку в подальшому будуть вибиратися m вершин для утворення нових ребер. Це зроблено для того, щоб у вершин з більшим ступенем був більший шанс бути обраними, тому що вони будуть зустрічатися в списку частіше. Таким чином реалізується принцип переважного приєднання.

Потім створюється змінна `source`, яка є ідентифікатором нової вершини. І потім запускається цикл, який буде тривати до тих пір, поки `source` менше наявної кінцевої кількості вершин n – так як лічильник ідентифікатора вершин починається з нуля, наявність вершини з ідентифікатором $n - 1$ має на увазі, що в графі є n вершин.

Виконання тіла циклу починається з того, що в граф додаються ребра нової вершини `source` з кожної з обраних цільових вершин `targets`. При цьому немає необхідності додавати саму вершину – замість того, щоб при спробі утворити ребро з ще не створеної вершиною видати помилку, функція додавання ребра попередньо додасть в граф необхідну вершину.

Після цього в `repeated_nodes` спершу додаються всі вершини зі списку `targets`, а після – m раз додається нова вершина `source`. Потім при виклику `_random_subset(repeated_nodes, m)` зі списку `repeated_nodes` обирається m унікальних ідентифікаторів вершин так, як описано у прикладі 3.2:

```
def _random_subset(seq, m):
    targets = set()
    while len(targets) < m:
        x = random.choice(seq)
        targets.add(x)
    return targets
```

Приклад 3.2 – Функція `_random_subset`

Як видно в прикладі 3.2, в функцію `_random_subset` при виклику передаються параметри `seq` і `m`, де `seq` – деяка послідовність (в даному випадку – список ідентифікаторів вершин), а `m` – кількість унікальних значень з цієї послідовності, які треба вибрати. Під задачу вибору значень з послідовності виділена окрема функція, оскільки `random.sample` при наявності в послідовності повторюваних значень цілком може вибрати однакові, в той час як алгоритму необхідні унікальні значення і список ідентифікаторів вершин напевно містить безліч значень, що повторюються. Результуюча колекція `targets` представлена у вигляді множини `set`, в якій не можуть міститися повторювані значення – спроба додати в нього вже наявний член буде проігнорована. Таким чином, поки довжина множини не буде дорівнює кількості необхідних значень, буде відбуватися наступне: з послідовності `seq` вибирається випадкове значення за допомогою `random.choice` і додається в безліч `targets`. Як уже було згадано вище, спроба додати вже наявне в `targets` значення не увінчається успіхом, тобто довжина `targets` не зміниться, і таким чином цикл триватиме, поки в `targets` не буде `m` унікальних значень. Після виходу з циклу результуюча множина `targets` повертається з функції.

Список `targets` з викликаючої функції `barabasi_albert_graph` перезаписується списком, отриманим в результаті роботи функції `_random_subset` для поточного стану списку `repeated_notes`, після чого значення ідентифікатора нової вершини `source` збільшується на 1. На цьому тіло циклу закінчується.

Після того, як кількість доданих вершин досягне встановленого значення n , відбудеться вихід з циклу і отриманий граф G повернеться з функції в зовнішнє середовище.

Після докладного розбору функції-генератора моделі мережі Барабаш-Альберт стає зрозуміло, що для створення її еластичного варіанту необхідно всього лише виправити пару команд. Краще виправити готове, ніж робити з нуля, коли дозволяє ситуація. Отримана функція виглядає так, як описано в прикладі 3.3:

```
def barabasi_albert_graph(n, m, seed=None):
    if m < 1 or m >= n:
        raise nx.NetworkXError("Barabási-Albert network must have "
                                "m >= 1 and m < n, m = %d, n = %d" % (m, n))
    if seed is not None:
        random.seed(seed)

    G = empty_graph(m)
    targets = list(range(m))
    repeated_nodes = []
    added = []
    source = m
    while source < n:
        G.add_edges_from(zip([source] * m, targets))
        repeated_nodes.extend(targets)
        repeated_nodes.extend([source] * m)
        added.append(m)
        m = int((sum(added)/len(added)) * ec)
        targets = _random_subset(repeated_nodes, m)
        source += 1
    return G
```

Приклад 3.3 – Реалізація генерації еластичної моделі мережі Барабаші-Альберт

Наведений в прикладі 3.3 код дещо відрізняється від початкового варіанту. У списку аргументів функції з'явився новий обов'язковий аргумент – коефіцієнт еластичності `es`. Він буде використаний всередині тіла циклу для визначення кількості додаваних вершин на поточній ітерації. Окрім цього, перед входом в цикл присутній декларація списку `added`. У цьому списку буде зберігатися історія доданих ребер – перший елемент відображає кількість ребер, доданих на першій ітерації, другий – на другій, третій – на третій і так далі. Це, так само, як і коефіцієнт еластичності `es`, необхідно для визначення кількості додаваних ребер на поточній ітерації. Після додавання `m` ребер в граф, `m` додається в список `added`, потім розраховується значення `m` для наступної ітерації. Шляхом ділення суми списку `added` на його довжину виходить середнє по доданим вершинам і множиться на коефіцієнт еластичності `es`. Отримане значення `m` передається для вибору ребер в функцію `_random_subset`. За допомогою скрипта з прикладу 3.4 була згенерована і намальована мережа, що складається зі ста вершин:

```
G = elastic_ba_graph(100, 4, 1.25)
nx.draw_networkx(G)
plt.show()
```

Приклад 3.4 – Скрипт генерації мережі та її візуального відображення

В якості початкового значення для кількості додаваних вершин було передано число 4, коефіцієнт еластичності – 1.25. На першій ітерації буде додано 4 вершини, але поступово цей показник буде змінюватися. Результат відтворення мережі представлений на рисунку 3.1:

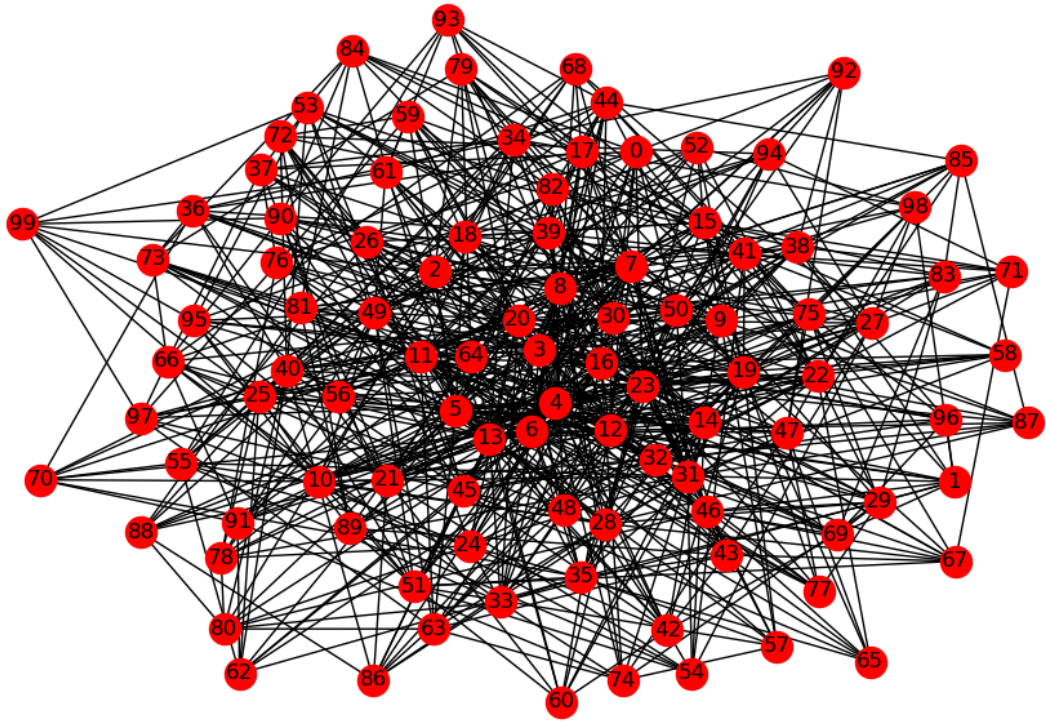


Рисунок 3.1 – Згенерована мережа

Вже було згадано, що отрисовка графів не є сильною стороною бібліотеки NetworkX. Проте, можна помітити, що вершини з великими номерами ближче до 100 (в даному випадку номер відображає порядок додавання) знаходяться «на поверхні» і мають меншу кількість зв'язків, в той час як в центрі в тісному клубку зв'язків зосереджені вершини, які були додані в самому початку, в результаті чого стали «концентраторами».

3.3 Реалізація розрахунку коефіцієнта асортативності мережі

Як все було сказано у попередньому розділі, найбільш популярним і розповсюдженим серед досліджень асортативності мереж є асортативність мереж за ступенем. Ступінь є скалярним параметром вершин мережі, тож це означає, що асортативність за ступенем – окремий випадок асортативності за скалярним параметром. Але чому так? Що відрізняє ступінь вершини на фоні інших скалярних або дискретних характеристик? Справа в тому, що

ступінь вершини, або точніше – розподіл ступенів вершин є природною властивістю мереж, – отримати всю необхідну інформацію про нього можна, маючи на руках лише відомості про структуру мережі, у той час, як, скажімо, вік або доходи, або ще яка-небудь спеціалізована характеристика є специфічною для конкретної мережі і потребує додаткових зусиль для зберігання / передачі інформації та ін. Тим більше, навіть у випадку необхідності налаштувати програму під іншу скалярну характеристику (ті ж самі вік або доходи) не має виникнути завеликих труднощів доти, доки ця характеристика є скалярною.

Можна почати зверху, поки що не прив'язуючись ні до якої конкретної скалярної характеристики – функція `numeric_ac` виконує розрахунок коефіцієнта кореляції Пірсона, який, як вже було сказано, у даному випадку є еквівалентним коефіцієнту асортативності мережі (приклад 3.5):

```
def numeric_ac(M):
    if M.sum() != 1.0:
        M = M / float(M.sum())
    nx, ny = M.shape
    x = numpy.arange(nx)
    y = numpy.arange(ny)
    a = M.sum(axis=0)
    b = M.sum(axis=1)
    vara = (a * x**2).sum() - ((a * x).sum())**2
    varb = (b * x**2).sum() - ((b * x).sum())**2
    xy = numpy.outer(x, y)
    ab = numpy.outer(a, b)
    return (xy * (M - ab)).sum() / numpy.sqrt(vara * varb)
```

Приклад 3.5 – Розрахунок коефіцієнта кореляції Пірсона для масиву чисел

Як можна бачити, функція розрахунку коефіцієнта кореляції Пірсона є максимально абстрагованою від конкретних скалярних характеристик – на вході функції присутній лише масив чисел без будь-якої додаткової

інформації про природу або призначення цих чисел, тобто, при бажанні та / або необхідності можна застосувати функцію для інших потреб – можливо, для розрахунку асортативності мережі за іншою скалярною характеристикою, або навіть використати її за первинним призначенням – для розрахунку коефіцієнту кореляції Пірсона (не для асортативності мережі).

У першу чергу масив, що поступив на вхід функції, приводиться до необхідного для подальших розрахунків вигляду за допомогою методів `share` та `arrange` пакету `numpy`. Після цього виконуються необхідні за формулою Пірсона розрахунки і їх результат повертається до зовнішнього середовища (функції, що викликала цю функцію).

На цьому проектування не закінчується – функція розрахунку коефіцієнта кореляції Пірсона є лише малою частиною роботи. Вище згадувалося, що вона абстрагується від конкретних скалярних характеристик. Те, що відокремлює ступінь вершини мережі від віку або доходів – вхідний масив даних `M`. Результат роботи функції повертається до функції, що її викликала з параметром `M`, отже у ній знаходиться і джерело цього масиву даних (приклад 3.6):

```
def degree_assortativity_coefficient(G, x='out', y='in', weight=None,
                                    nodes=None):
    M = degree_mixing_matrix(G, x=x, y=y, nodes=nodes, weight=weight)
    return numeric_ac(M)
```

Приклад 3.6 – Функція розрахунку коефіцієнту асортативності мережі

Функція `degree_assortativity_coefficient` є невеликою, бо вона – головна, координаційна функція, що керує вхідним / вихідним потоком даних допоміжних функцій. Результатом її роботи є розрахований за допомогою функції `numeric_ac` коефіцієнт кореляції Пірсона для масиву `M`,

а вміст цього масиву визначається результатом роботи функції `degree_mixing_matrix`, що отримує на вхід граф `G`, переданий до функції `degree_assortativity_coefficient`, а також додаткову інформацію про цей граф у випадку її наявності (масив вхідних ребер, масив вихідних ребер, ваги ребер та ін.), а повертає розподіл ступенів вершин графу `G` у вигляді матриці змішування ступенів (приклад 3.7):

```
def degree_mixing_matrix(G, x='out', y='in', weight=None,
                        nodes=None, normalized=True):
    d = degree_mixing_dict(G, x=x, y=y, nodes=nodes, weight=weight)
    s = set(d.keys())
    for k, v in d.items():
        s.update(v.keys())
    m = max(s)
    mapping = {x: x for x in range(m + 1)}
    a = dict_to_numpy_array(d, mapping=mapping)
    if normalized:
        a = a / a.sum()
    return a

def degree_mixing_dict(G, x='out', y='in', weight=None,
                      nodes=None, normalized=False):
    xy_iter = node_degree_xy(G, x=x, y=y, nodes=nodes, weight=weight)
    return mixing_dict(xy_iter, normalized=normalized)

def mixing_dict(xy, normalized=False):
    d = {}
    psum = 0.0
    for x, y in xy:
        if x not in d:
            d[x] = {}
        if y not in d:
            d[y] = {}
        v = d[x].get(y, 0)
        d[x][y] = v + 1
        psum += 1
    if normalized:
        for k, jdict in d.items():
            for j in jdict:
                jdict[j] /= psum
    return d
```

Приклад 3.7 – Функції для отримання матриці змішування ступенів мережі

На найнижчому рівні тут знаходиться функція `degree_mixing_dict`, що спершу здобуває з переданого до неї графу `G` списки вузлів та ребер за допомогою функції `node_degree_xy`, після чого передає результат до

`mixing_dict`, де ці списки аналізуються і на їх основі формуються дані про змішування ступенів вершин у графі у вигляді словника (спеціальна структура вигляду ключ-значення).

Функція `degree_mixing_dict`, в свою чергу, є допоміжною функцією для `degree_mixing_matrix`, яка на основі отриманого словника змішування ступенів вершин у графі формує відповідну матрицю і нормалізує її у разі необхідності.

Отримана матриця у подальшому використовується в якості джерела даних для розрахунку коефіцієнта кореляції Пірсона у головній координуючій функції `degree_assortativity_coefficient`.

Описані тут дії призначені для пошуку коефіцієнта кореляції для матриці змішування ступенів вершин у графі, але, як вже неодноразово згадувалося – на місці ступеня вершини могла бути будь-яка інша скалярна характеристика, будь то вік чи доходи. При бажанні / необхідності налаштувати розрахунки під іншу скалярну характеристику, усе, що потрібно зробити – змінити код функції `degree_mixing_dict`, у якій відбувається здобуття списків вершин / ребер графа і формування словника змішування ступенів вершин, таким чином, щоб формувався словник змішування необхідної скалярної величини.

3.4 Імітаційне моделювання

Тепер, коли всі необхідні функції та налаштування готові, можна перейти безпосередньо до імітаційного моделювання – генерації мереж та оцінювання їх показників асортативності. Для порівняння було вирішено використати в процесі моделювання не тільки еластичну модель масштабно-інваріантних мереж, а й класичну модель Барабаші-Альберт разом з моделями випадкової мережі Ердеша-Рен'ї і малого світу Ваттса-Строгатца. Функції генерації цих мереж потребують різноманітних наборів параметрів: кінцева кількість вузлів (спільно для всіх), кількість додаваних

ребер за ітерацію (константа у випадку класичної мережі Барабаші-Альберт, для еластичної моделі масштабно-інваріантної мережі задається лише початкове значення і змінюється з ходом будування мережі), імовірність утворення ребра (випадкова мережа), кількість найближчих сусідів (малий світ), як показано у прикладі 3.8:

```
G1 = elastic_ba_graph(500, 4, 1.25)
r1 = degree_assortativity_coefficient(G1)
G2 = nx.barabasi_albert_graph(500, 4)
r2 = degree_assortativity_coefficient(G2)
G3 = nx.erdos_renyi_graph(500, 0.3)
r3 = degree_assortativity_coefficient(G3)
G4 = nx.watts_strogatz_graph(500, 2, 0.2)
r4 = degree_assortativity_coefficient(G4)
print(r1)
print(r2)
print(r3)
print(r4)
```

Приклад 3.8 – Моделювання мереж та вимірювання їх показників асортативності

Візуалізацію мереж буде опущено, адже вона має сенс (принаймні для того рівня реалізації, який надано у модулі NetworkX) лише у випадку відносно невеликих графів / мереж, але на відносно невеликих розмірностях не завжди встигають проявитися властивості складних мереж (масштабно-інваріантність, малий світ тощо). Результати моделювання приведено у таблиці 4.1:

Таблиця 4.1 – Результати імітаційного моделювання

Кількість вершин					
Вид мережі	500	1000	1500	2000	2500
Еластична модель	-0.035	-0.04	-0.033	-0.023	-0.027
Барабаші- Альберт	-0.063	-0.08	-0.047	-0.055	-0.037
Ердеша- Рен'ї	-0.004	-0.003	0	-0.001	0
Ваттса- Строгатца	-0.059	-0.124	-0.09	-0.108	-0.125

Більшість розглянутих мереж мають тенденцію схилитися в сторону дизасортативності за ступенями вершин. Структура як мережей малого сіту, так і масштабно-інваріантних мережей передбачає утворення кількох вузлів, що мають ступінь, набагато більшу у порівнянні з більшістю вузлів мережі. У мережах малого світу це зумовлено правилом «шести ступеней розділу», у масштабно-інваріантних мережах – принципом переважного приєднання. Наявність «хабів» означає, що більшість вузлів малого ступеню приєднані до вузлів високого ступеню – тобто, ступені більшості «сусідів» не схожі, а навпаки, відрізняються, що є свідченням дизасортативності мережі за ступенем вершин. Еластична модель дещо менше схилена до дизасортативності, ніж класична модель Барабаші-Альберт – кількість додаваних ребер на кожному кроці змінюється, у той час у класичній моделі залишається постійною. Помітно, що результати випадкових мереж Ердеша-Рен'ї знаходяться дуже близько до нуля – мережа випадкова і не має тенденції віддавати перевагу тому чи іншому типу вершин.

ВИСНОВКИ

В ході виконання даної роботи були проведені дослідження в декількох областях, зокрема, загальні принципи структури і роботи складних мереж, що дозволяють досить достовірно відобразити чимало мереж реального світу – Інтернет, міжбілкові взаємодії, харчові ланцюги та багато інших.

Також були проаналізовані методи дослідження, що полягають у вивченні кореляції або асортативного змішування атрибутів вузла по ребрах. Цей аналіз дозволяє нам зробити узагальнення щодо того, з більшою ймовірністю ми спостерігаємо зв'язку між вузлами з однаковими характеристиками (асортативність) або між вузлами з різними (дизасортативність).

На основі виконаних досліджень було розроблено засіб, що дозволяє моделювати роботу складних мереж (еластичної моделі масштабно-інваріантної мережі, якщо точніше) та оцінювати їх показники асортативності. Після цього було проведене імітаційне моделювання роботи різноманітних типів складних мереж та проаналізований вплив характерних властивостей їх структур на асортативність / дизасортативність мережі за ступенем.

Було виявлено, що мережі на зразок мереж малого світу та масштабно-інваріантних мереж, архітектура яких зосереджена навколо вузлів-концентраторів, проявляють дизасортативні властивості (вершини низького ступеню здебільшого мають тенденцію створювати ребра з вершинами високого ступеню), у той час як випадкові мережі Ердеша-Рен'ї зазвичай не є ні асортативними, ні дизасортативними, бо їх випадкова природа не оперує за принципом переважного приєднання.

ПЕРЕЛІК ПОСИЛАНЬ

1. Newman M. Networks: An Introduction // Oxford University Press, 2010. P. 734-749.
2. Kleinberg J. Networks, Crowds, and Markets: Reasoning about a Highly Connected World // Cambridge University Press, 2010. P. 61-70.
3. Network Science – Barabasi. URL : <http://barabasi.com> (дата звернення 30.10.2019).
4. Complex Networks 2019 URL : <https://www.complexnetworks.org/> (дата звернення 30.10.2019).
5. Complex Networks – Latest research and news | Nature. URL : <https://www.nature.com/subjects/complex-networks> (дата звернення 21.10.2019).
6. Complex networks: Structure and dynamics – ScienceDirect. URL : <https://www.sciencedirect.com/science/article/pii> (дата звернення 21.10.2018).
7. Complex networks and collective behavior in nature – MIT. URL : http://web.mit.edu/8.334/www/grades/projects/projects10/Hernandez-Lopez-Rogelio/structure_1.html (дата звернення 1.11.2019).
8. Journal of Complex Networks | Oxford Academic. URL : <https://academic.oup.com/comnet> (дата звернення 2.11.2019).
9. Complex Networks – Complex Systems Lab. URL : <http://complex.upf.edu/14-ricard-sol%C3%A9/19-complex-networks> (дата звернення: 2.11.2019).
10. Barabasi A.-L. Network Science // Cambridge University Press, 2016. P. 253-270.
11. Barabasi A.-L. Linked: The New Science of Networks // Perseus Books Group, 2002. P.141-150
12. Hansen D. Analyzing Social Media Networks with NodeXL: Insights from a Connected World // University of Maryland, 2009. P. 25-40.

13. Adar, E. GUESS: A Language and Interface for Graph Exploration, Proc // ACM Conference on Human Factors in Computing Systems. 2006.
14. Shneiderman B., Aris A. Network visualization with semantic substrates // IEEE Symposium on Information Visualization and IEEE Trans. Visualization and Computer Graphics. 2006. 12 (5). P. 733-740.
15. Structure of Complex Networks: Theory and Applications – Oxford Scholarship. URL : <https://www.oxfordscholarship.com/> (дата звернення 6.11.2019).