**برسی ویژگی های توپولوژیکی شبکه خیابان های شهر رفسنجان**

**حسین رفیعی زاده1، زینب الهدی حشمتی رفسنجانی 2**

1- دانشجوی کارشناسی ارشد، علوم و فناوری شبکه، دانشگاه تهران، تهران

2- استادیار، مهندسی الکترونیک، دانشگاه تهران، تهران

**چکيده**

در این مقاله، شبکه خبابان های رفسنجان را به‌عنوان یک شبکه پیچیده مورد بررسی قرار می‌دهیم که گره ‌ها را تقاطع های شبکه و یال ها مسیر مستقیم بین دو تقاطع را تشکیل دهند. با کمک بررسی ویژگی های توپولوژیکی شبکه، مانند مرکزیت بینیت، مسیرهای آسیب پذیر را پیدا می کنیم. سپس با یافتن پراکندگی درجه، قطر و متوسط طول مسیر شبکه شهری وجود پدیده جهان کوچک را در شبکه بررسی می کنیم و در نهایت به این نتیجه می رسیم که توسعه شبکه شهری رفسنجان از لحاظ اقتصادی بهینه نبوده است.

**واژگان كليدي :**تحلیل شبکه، شبکه پیچیده، شبکه شهری، پدیده جهان کوچک، شبکه بدون مقیاس

**Investigating of topological characteristics of the Rafsanjan city streets network**

**Hossein Rafiee Zade,** **Zainabolhoda Heshmati Rafsanjani**

**Abstract**

In this paper, we study the network analysis of Rafsanjan streets as a complex network so that intersections form network nodes and a direct route between two intersection of network edges. With the help of investigating the topological characteristics of the network, such as betweenness centrality, we find vulnerable stations and routes. Then, by finding the degree distribution, diameter, and average path length of the street network, we examine the existence of the small world phenomenon in the network After realizing that there is no small world phenomenon in Rafsanjan's street network, and finally, we concluded that the development of the urban network of Rafsanjan has not been economically optimal.

Keywords: network analysis, complex network, small world phenomenon, scale free network

**1- مقدمه**

تعداد زیادی از سیستم های اجتماعی، زیستی و انسان را می توان در قالب شبکه ها نشان داد. به عنوان مثال، جامعه توسط تعاملات اجتماعی افراد نزدیک به هم ساخته شده است[1]، در حالی که عملکرد سلول ها توسط شبکه پیچیده ای از متابولیت ها و فعل و انفعالات شیمیایی تضمین می شوند. به همین ترتیب، سیستم های زیرساختی مهم مثل حمل و نقل، اینترنت[2] یا یک سیستم مترو[3] میتوانند به صورت یک شبکه مدل سازی شوند. با افزایش شهرنشینی و ارتباط مرتبط با مطالعات شهری کاوش در ابعاد زیست محیطی، اقتصادی، جمعیتی و اجتماعی شهرها، شبکه‌های خیابانی در پنجاه سال گذشته به موضوع اصلی مورد توجه علمی جهانی تبدیل شده‌ است. از آنجایی که شبکه‌های خیابانی طیف وسیعی از فرآیندهای شهری را پشتیبانی می‌کنند، توجه دانشمندان بسیاری از رشته‌ها، از جمله حمل ‌و نقل و برنامه‌ ریزان شهری، معماران، جغرافی‌دانان، روان‌شناسان محیطی و اخیراً فیزیک دانان را به خود جلب می‌کنند. مطالعات شبکه خیابانی شامل تحقیقات در مورد ساختار شبکه، همبندی، مرکزیت، مدار، پیمایش، سلسله مراتب، گونه شناسی و تکامل است. مطالعات کمی و محاسباتی مختلف شبکه‌های خیابانی عمدتاً از روش‌های مبتنی بر نظریه گراف و علم شبکه استفاده می‌کنند.[4]

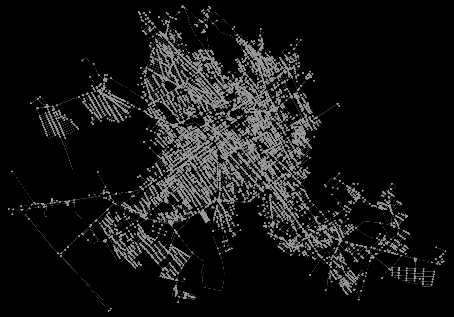
گراف ها تنها یک انتزاع ریاضی هستند، یک نمایش رسمی از یک مدل حذف شده از واقعیت فیزیکی محیط های خیابان از طریق فرآیند انتزاع و مدل سازی می باشند. این شامل ایجاد یک بازنمایی ساده از شبکه خیابان با جدا کردن عناصر اصلی مطالعه و شناسایی روابط آنها است. مهمتر از همه، تعیین می کند که چه چیزی به عنوان گره و یال در یک نمودار نمایش داده می شود و چه پارامترهای اضافی از شبکه خیابانی گراف باید ثبت شود. ما این مرحله را مدل سازی شبکه می نامیم که معمولاً در مدل بزرگتری از یک پدیده خاص تعبیه شده است[4]. توسعه مناسب شبکه راه های حمل و نقل نه تنها هزینه حمل و نقل را چه از نظر مالی و چه از نظر زمانی کاهش می دهد، بلکه به یکپارچگی مناطق مختلف داخل کشور و درک بهتر کشورهای همسایه در سطح بین المللی کمک می کند[5]. حمل و نقل به دو دلیل اصلی به موضوع مهمی برای جغرافیدانان تبدیل شده است. اولاً، حمل و نقل یک فعالیت انسانی مهم با یک جزء فضایی قوی است. ثانیاً، عامل مهمی است که بر تنوع فضایی بسیاری از فعالیت‌های اجتماعی و اقتصادی دیگر تأثیر می‌گذارد[6]. در این پژوهش شبکه خیابان های شهر رفسنجان را به عنوان یک شبکه پیچیده مورد برسی قرار میدهیم و چهار راه ها، میدان ها و تقاطع ها را نود و مسیر های بین آن ها را یال های شبکه در نظر میگیریم و با برسی ویژگی های توپولوژیکی شبکه، مانند مرکزیت بینیت، مسیرهای آسیب پذیر را پیدا می کنیم. سپس با یافتن توزیع درجه، قطر و متوسط طول مسیر شبکه خیابان ها، وجود پدیده جهان کوچک را در شبکه بررسی خواهیم کرد.

**2- نظریه بنیادی گراف**

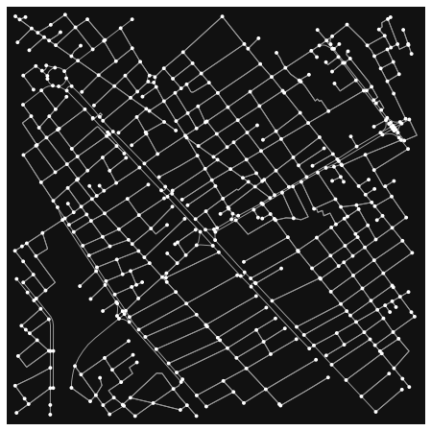
در این بخش، برخی از اصطلاحات پایه نظریه گراف را معرفی می‌کنیم و سعی می‌کنیم اصطلاحات موجود در شبکه خیابان شهر رفسنجان را معرفی کنیم. با تعریف بنیادی شروع می کنیم، گراف چیست؟ یک گراف G = (V, E) از مجموعه ای از گره های V (یا رئوس، نقاط) و مجموعه ای از یال های E (پیوندها، خطوط) تشکیل شده است که نحوه تعامل گره های شبکه با یکدیگر را نشان می دهد. یال ها می توانند جهت دار یا بدون جهت باشند.[8]گره ها اندازه شبکه را نشان می دهند. در این شبکه، گره ها که بین مسیر ها می باشند در سایت OpenStreetMap [[1]](#footnote-2)شان داده شده است که با کمک کتابخانه OSMnx[[2]](#footnote-3) گراف شهر را تولید می کنیم، به عنوان مثال، گذرگاه ها و جاده ها یا بخش های فرعی جاده ها بین دو گره قرار می گیرند. هر گره یا یال می‌تواند ویژگی‌های متفاوتی را در خود جای دهد، به عنوان مثال، گره‌ها را می‌توان به انواع مختلفی از قبیل تقاطع یا بن‌بست اختصاص داد و یال‌ها ممکن است ویژگی عددی خاصی مانند محدودیت سرعت داشته باشند. یال ها گراف می توانند وزن داشته باشند. به گرافی که یال های آن وزن دار باشد، گراف وزن دار [[3]](#footnote-4)می گویند. اولین اندازه گیری برای یک گره در گراف درجه است که جمع تعداد یال هایی است که با گره های دیگر ارتباط دارند که با k نشان داده می شود.[8]

**3- نمایش شبکه خیابان های شهری**

روش های مختلفی زیادی برای نمایش شبکه خیابان های شهر وجود دارد که دو نمایش را در شکل های 1,2 مشاهده می کنید.



شکل 1- نمای کلی از شبکه خیابان های شهر رفسنجان



شکل 2- شبکه خیابان در فاصله 0.75 کیلومتری از یک نقطه با طول و عرض جغرافیایی مشخص

ما برای نمایش شبکه از کتابخانه کاربردیه OSMnx استفاده کردیم؛ که به ما این امکان را می دهد داده‌های مکانی را از وب سایت OpenStreetMap دانلود کنیم و شبکه‌ خیابانی و هر هندسه دلخواه جغرافیایی دیگر را مدل‌سازی، پروژه‌سازی، تجسم و تجزیه و تحلیل کنیم و می‌توانیم شبکه‌های شهری مثل پیاده‌روی، رانندگی یا دوچرخه‌سوار را دریافت و مدل‌سازی کنید و سپس به راحتی آن‌ها مورد تجزیه و تحلیل قرار دهیم[11].

**4- داده های پژوهش شده**

شبکه ای که کتابخانه پایتونی OSMnx از شهر رفسنجان به ما می دهد به این ترتیب است که تعداد یال هایش برابر با 20072 و تعداد گره های شبکه برابر با 7963 می باشد و میانگین درجاتش برابر با 5.04 شد. بنابراین اندازه شبکه 7963 است.

خروجی شبکه ای که OSMnx که به ما می دهد به صورت گرافی جهت دار می باشد یه عنوان مثال اگر بین دو نود a , b مسیری وجود داشته باشد یک یال برای رفتن از a به b و یک یال دیگر هم برای b به a در نظر می گیرید ولی ما قصد داریم که شبکه ما یک گراف ساده یعنی بدون جهت و فاقد حلقه باشد که برای انجام محاسبات هم کار ما راحت تر باشد زیرا بسیاری از کتابخانه های تحلیل شبکه مثل Networkx بیشتر اندازه گیری ها و محاسبات شان را بر روی گرافهای ساده انجام می دهند.

پس از تبدیل شبکه به یک گراف ساده تعداد نود ها که ثابت ماند و تعداد یال ها به 11217 کاهش یافت و میانگین درجات هم برابر با 2.81 شد.

**5- ویژگی های توپولوژیکی شبکه خیابان ها**

یک شبکه پیچیده[[4]](#footnote-5) با N گره را می توان به صورت گراف نشان داد، که در آن برابر است با مجموعه گره ها و هم مجموعه پیوندها را نشان می دهند. یک گراف را می توان با یک ماتریس مجاورت A توصیف کرد و نشان داد، که یک ماتریس است که ورودی آن برابر با 1 است اگر پیوندی بین گره های i,j وجود داشته باشد. و در غیر این صورت برابر با صفر می باشد[7].

(4)

در ادامه برخی از اقدامات توپولوژیکی شبکه خیابان شهر رفسنجان معرفی شده است.

**A-5-درجه[[5]](#footnote-6) و یال[[6]](#footnote-7)**

درجه گره i ام در شبکه را با نشان می دهیم. در یک شبکه بدون جهت، تعداد کل یال ها را می توان به صورت مجموع درجات گره بیان کرد [8].

(1)

**B-5- میانگین درجه[[7]](#footnote-8)**

یکی از ویژگی های مهم در یک شبکه غیر جهت دار، متوسط درجه آن شبکه است. برای محاسبه درجه نباید بین یال های جهت دار و غیر جهت دار تمایز قائل شویم در واقع، ما باید بین درجه ورودی و درجه خروجی یک گره تمایز قائل شویم و درجه هر نود را به سادگی اندازه گیری کنیم، که چند یال از یک گره خارج می شوند و چند یال وارد می شوند و در مورد یک گراف غیر جهت دار، می توانیم درجه متوسط ​​را با کمک فرمول زیر محاسبه کنیم[8].

(2)

مشابه این کار را می توان به طور جداگانه برای درجات داخلی و خارجی انجام داد:

(3)

(5)

در شبکه شهری مورد مطالعه، درجه هر گره نشان دهنده تعداد خیابان های متصل به هر کدام از تقاطع هاست و میانگین درجات نشان دهنده میانگین کل درجه هر کدام از تقاطع هاست.

**C-5- قطر شبکه[[8]](#footnote-9)**

قطر یک شبکه را با نشان داده می شود، حداکثر طول کوتاه ترین مسیر در شبکه است. به عبارت دیگر، این بزرگترین فاصله ثبت شده بین هر جفت گره است. برای شبکه های بزرگتر، قطر را می توان با استفاده از الگوریتم جستجوی سطح اول[[9]](#footnote-10) [9] تعیین کرد. در شبکه شهری ما طبق این تعریف، قطر شبکه که کوتاه ترین حداکثر فاصله بین دو نقطه است برابر با 131 یال شد.

**D-5- میانگین طول مسیر[[10]](#footnote-11)**

میانگین طول مسیر که با نشان داده می شود، میانگین کوتاه ترین مسیرها بین تمام جفت گره ها در یک شبکه است. الگوریتم جستجوی سطح اول همچنین می تواند برای تعیین متوسط طول مسیر برای یک شبکه بزرگ استفاده شود[8]. کوتاه ترین مسیر d(i , j)، همانطور که از نامش هم پیداست، یعنی در شبکه مسیری وجود دارد که با کمترین یال می توانیم بین گره های i,j حرکت کنیم. در مورد یک شبکه بدون جهت، کوتاه‌ترین مسیر بین i و j صرف نظر از اینکه از کدام گره شروع می‌کنیم همیشه یکسان است، اما در شبکه‌های جهت دار این درست نیست و بسته به اینکه از کدام گره شروع کنیم کوتاه‌ترین مسیر بین گره‌ها می تواند متفاوت باشد[9].

(6)

**E-5- ضریب خوشه بندی[[11]](#footnote-12)**

معیاری است که درجه گره‌ها را در یک گراف که تمایل به ایجاد یک خوشه دارند را اندازه می‌گیرد. برای گره i با درجه ki، ضریب خوشه بندی به صورت زیر تعریف می شود

(7)

به طور خلاصه، Ci چگالی پیوند های محلی شبکه را اندازه می‌گیرد: هرچه همسایه ها گره i به هم پیوسته‌تر باشد، ضریب خوشه‌بندی محلی آن گره بیشتر است. میزان خوشه‌بندی یک شبکه کامل با میانگین ضریب خوشه‌بندی محاسبه می‌شود که به صورت زیر تعریف می شود[9].

(8)

در مطالعه ما ضریب خوشه بندی یک گره (تقاطع) میزان همبندی همسایه های آن تقاطع یا میدان را به یکدیگر را نشان می دهد و ما علاقه مند هستیم که ضریب خوشه بندی در شبکه ما مقدار بالایی را داشته باشد زیرا که به پدیده جهان کوچک نزدیک می شویم.

**F-5- مرکزیت بینیت[[12]](#footnote-13)**

مرکزیت بینیت گره را می توان برای گره ها یا یال ها شبکه اندازه گیری کرد، مرکزیت بینیت گره میزان تاثیرگذار یک گره در کوتاه ترین مسیرهای بین همه جفت گره ها را نشان می دهد. این معیار به طور گسترده ای در شبکه های حمل و نقل برای بازتاب گذر استفاده می شود. جاده های با مرکزیت بالا در کوتاه ترین مسیرهای شبکه قرار دارند و باید به عنوان جاده های مهم شبکه در نظر گرفته شوند که ممکن است حجم ترافیک را افزایش دهند. در رابطه مرکزیت هر چه بالاتر باشد، نقش آن گره i در انتقال مهمتر است [9]. بینیت گره i به عنوان کسری از کوتاه ترین مسیرهایی که از گره i می گذرد تعریف می شود که رابطه آن به صورت زیر می باشد

(9)

تعداد کوتاهترین مسیرها از s به t است که از i می گذرد و تعداد کل کوتاه ترین مسیرها از s به t است. تقاطع و مسیرهایی که مرکزیت بینیت بالاتری دارند، می‌توانند در معرض آسیب‌پذیری بیشتری قرار گیرند، یعنی آسیب به این تقاطع ها یا مسیرهای خیابانی می‌تواند مشکلات جدی‌تری را برای شبکه ایجاد کند [9].

**G-5- همبندی و چگالی[[13]](#footnote-14)**

همبندی در یک شبکه نشان می دهد که آیا پیوندهای مستقیم بین جفت گره ها وجود دارد یا خیر، در حالی که چگالی نشان دهنده نسبت یالهای موجود به کل یال ها است [10]. برای این شبکه چگالی 0.00035 به دست آمد که منطقی است زیرا در یک شبکه خیابانی همه گره ها نمی تواند به هم مسیر داشته باشند.

(10)

* در یک شبکه کامل، بین هر جفت گره یال وجود دارد و تمام پیوند های بین گره ها برقرار است، پس چگالی آن برابر با 1 است.
* در یک شبکه که یالی وجود نداره چگالی آن برابر با صفر می باشد.
* در یک شبکه همبند قوی، حداقل یک مسیر بین هر جفت گره وجود دارد، بنابراین هیچ گره ای از شبکه جدا نمی باشد.
* در یک شبکه ناهمبند، ممکن است گره ها یا جفت گره ها از هم جدا باشند. اگر هر گره ای جدا یا از شبکه قطع شود، چگالی آن برابر با صفر می شود.

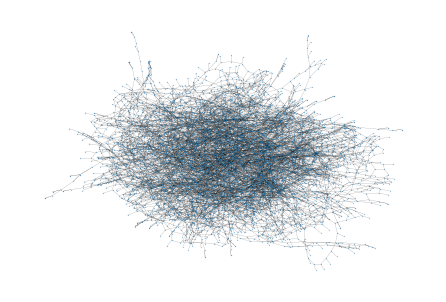
**H-5-پدیده جهان کوچک[[14]](#footnote-15)**

در علم شبکه، پدیده جهان کوچک به این معناست که فاصله بین دو گره تصادفی در مقایسه با اندازه شبکه کم است و فقط به صورت لگاریتمی با اندازه شبکه افزایش می یابد. از این رو به طور معمول ویژگی جهان کوچک با رابطه زیر تعریف می شود[8]. اکثر شبکه های مورد مطالعه دارای خاصیت جهان کوچک هستند

(11)

**6- نتایج به دست آمده**

با استفاده از مجموعه داده ای که در بخش 4 تهیه کردیم و با کمک نرم افزار گفی[[15]](#footnote-16) ، قطر شبکه، میانگین فاصله[[16]](#footnote-17)، میانگین درجات[[17]](#footnote-18)، چگالی شبکه شهری را به محاسبه و آن را ترسیم کردیم که در شکل 3 مشاهده می کنید.

****

شکل3- ترسیم شبکه خیابان های شهر رفسنجان که با کمک کتابخانه Networks

**A-6- نوع شبکه**

با کمک زبان برنامه نویسی پایتون و کتابخانه Networks [[18]](#footnote-19)توزیع درجات[[19]](#footnote-20) شبکه شهر رفسنجان و توزیع مرکزیت ها را محاسبه کردیم، همچنین با استفاده از پکیج Powerlaw [[20]](#footnote-21)توزیع دقیق درجات و همچنین مناسب ترین توزیع درجه و توان درجه [[21]](#footnote-22)را بدست آوردیم.

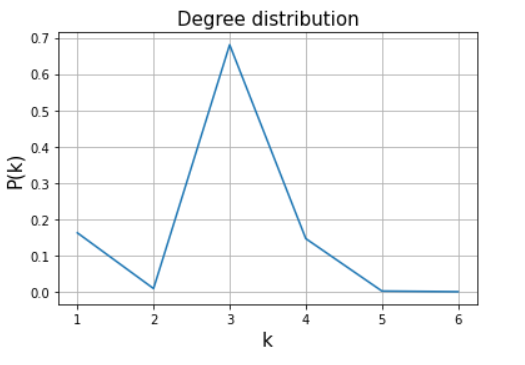
نتایج به دست آمده در جدول 1 نشان می دهد که بیشتر تقاطع ها از درجه 3 می باشند و سپس از درجه های4 , 1 هستند و از بقیه درجه ها هم تقاطع های کمتری را داریم.

تقاطع های با درجه بالا را در اصلاح علمی شبکه هاب می نامند. وجود هاب ها در شبکه خیابان های یک شهر باعث ایجاد میانبر ها و کاهش میانگین طول مسیر و در نتیجه سبب به وجود آمدن پدیده جهان کوچک می شوند همانطور که بعدا ثابت خواهیم کرد، شبکه خیابان های شهر رفسنجان به دلیل تعداد بسیار کم هاب ها، دارای پدیده جهان کوچک نمی باشد، و هاب ها به دلیل تعداد بسیار کم شان تاثیر قابل توجهی را بر روی ساختار شبکه نمی گذارند.

|  |  |
| --- | --- |
| **Percent** | **Degree** |
| 16.25% | 1 |
| 0.88% | 2 |
| 68.03% | 3 |
| 14.64% | 4 |
| 0.19% | 5 |
| 0.01% | 6 |

جدول 1- درصد توزیع درجه گره های شبکه شهر رفسنجان

شکل 3 نمودار توزیع درجات شبکه شهر رفسنجان را نشان داده است. که محور افقی در شکل 4 نشان دهنده درجه گره ها و محور عمودی نشان دهنده توزیع درجه ها می باشدحال سوالی که پیش می آید ساختار شبکه ما از چه نوعی است؟ و فرایند تولید شبکه به چه صورت می باشد؟ ما فعلا به این نتیجه رسیدیم که توزیع درجات در شبکه رفسنجان نمایی می باشند و با کمک پکیج power law دو توزیع نمایی [[22]](#footnote-23)و power law را فیت کردیم و دیدم که توزیع نمایی بهتر فیت می شود. و دوباره آمدیم توزیع log normal و توزیع نمایی را مقایسه کردیم و دیدیم که توزیع log normal بر روی دنباله درجات فیت تر می شود و از آنجایی که دو توزیع log normal و power law از یک خانواده می باشند پس می توان گفت که شبکه ما مقیاس آزاد [[23]](#footnote-24)می باشد و تعدادی از گره ها وجود دارند که نسب به سایر گره ها توجه بیشتری را به خود جلب می کنند و موفق می شوند یال های بیشتری را به خود جذب کنند و در نتیجه درجه بالاتری را دارند که هاب نامیده می شوند.



شکل4- توزیع درجه گره های شبکه شهر رفسنجان

در شبکه بدون مقیاس، شیب خط نمودار توزیع را توان درجه می نامند و با γ نمایش می دهند و با کمک γ می توان پدیده جهان کوچک را در شبکه برسی کرد و برای شبکه خیابان های شهر رفسنجان مقدار 1.63 به دست آمد و همین طور قطر شبکه رفسنجان برابر با 131 می باشد که یعنی برای عبور از یک نقطه به نقطه دیگر شهر ما باید حداکثر از 131 یال عبور کنیم. میانگین طول مسیر شبکه خیابان های رفسنجان هم برابر با 46.54 شد. با توجه به این نتایج و طبق فرمول های 12 و 11 ما پدیده جهان کوچک در این شبکه نداریم.

(12)

توان درجه ما تقریبا نزدیک به 2 شد و در این حالت هم فرم شبکه غیر عادی[[24]](#footnote-25) است و با توجه به رابطه 12 برای γ = 2 درجه بزرگ‌ترین هاب به شکل خطی با اندازه شبکه رشد می‌کند، یعنی در این حالت، پیکربندی هاب و گره های اقماری ایجاد می شود، که در آن همه‌ گره‌ها به یکدیگر نزدیک هستند زیرا همه آن‌ها به یک هاب مرکزی یکسان متصل هستند. در این ناحیه میانگین طول مسیر به   
وابسته نیست[10].

**B-6- مقاومت شبکه[[25]](#footnote-26)**

حال می توان این سوالات را مطرح کرد که آیا برای شبکه خیابانی بهتر است که توزیع درجات دارای خاصیت مقیاس آزاد باشند و یا تصادفی؟ و این موضوع چگونه بر استحکام شبکه تأثیر می‌گذارد به این معنا که اگر جاده‌های خاصی بسته یا مسدود شوند چه اتفاقی می‌افتد، چگونه بر جریان ترافیک شهر تأثیر می‌گذارد؟ برای از بین بردن جنبش شهر و حرکت ماشین ها و ایجاد ترافیک در شهر رفسنجان چه چیزی لازم است؟ چه جاده هایی محسوس تر هستند به این معنا که اگر این جاده ها بسته شوند تأثیر آن بر کل جریان ترافیک شبکه شدیدتراست. استحکام شبکه تلاش می‌کند تا اندازه‌گیری‌هایی را تعریف کند که نشان دهند که یک شبکه چقدر در برابر حملات، خرابی‌ها یا چیزی شبیه ترافیک مقاومت از خود نشان می دهند.

**A** -**B-6- همبندی نود ها[[26]](#footnote-27)**

همبندی نود ها تعداد گره هایی را که باید از گراف G حذف کنیم تا گراف ناهمبند شود را شرح می دهد، همبندی به این معناست که هر گره در گراف G بتواند از طریق یال ها به هر گره دیگر دلخواه در شبکه برسد و در غیر این صورت گراف G ناهمبند می باشد. یکی از ویژگی های مهم هر گراف باید داشته باشد این است که نباید به راحتی ناهمبند شود[9]. این نوعی تعریف کمی غیر معمول و مبهم است، به خصوص برای یک شبکه جاده ای، زیرا که ممکن است در جاده ها یا خیابان ها بن بست وجود داشته باشد، حذف گره همبند بن بست بلافاصله باعث ناهمبندی اتصلات گراف G شود در صورتی که از هر گره به گره دلخواه در شبکه مسیری وجود دارد .

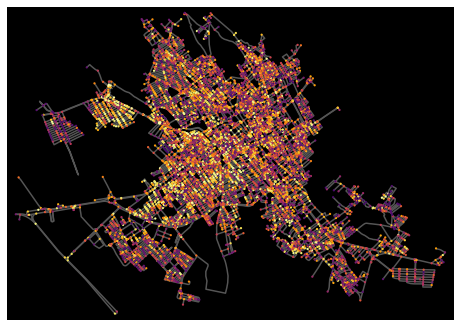
با کمک کتابخانه networkx و تابع هایش، همبندی گراف را به دست آوردیم و خروجی یک را به ما داد و به این معناست که شبکه شهر رفسنجان پس از حذف فقط یک گره ناهمبند می شود، آیا حذف یک نود برای ما مهم است؟ خیر، زیرا اندازه زیرگراف حذف شده فقط یک گره است و بقیه شبکه همچنان متصل است. با این حال، اگر اندازه زیر گراف قطع شده نسبتاً بزرگ باشد، این نشان دهنده وجود مشکل در ساختار شبکه است.

**B** -**B-6- مرکزیت ها[[27]](#footnote-28)**

می‌توانیم با محاسبه مرکزیت یک گره یا یال در یک شبکه، اندازه‌گیری و تجسم کنیم که تا چه اندازه آن یال یا گره مورد نظر «مهم» است. مرکزیت های زیادی در علوم شبکه وجود دارد، از جمله نزدیکی، بینیت، درجه، بردار ویژه و رتبه صفحه.

**A**- **B**-**B-6- مرکزیت بینیت**

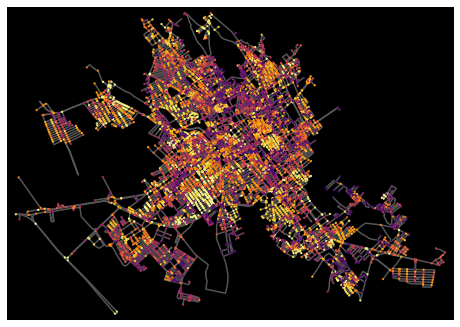
مرکزیت بینیت راهی برای تشخیص میزان تأثیر یک گره بر جریان اطلاعات در یک گراف است. اغلب برای یافتن گره هایی استفاده می شود که به عنوان یک پل بین دو قسمت از گراف عمل می کنند. خیابان های با مرکزیت بینیت بالا در کوتاه ترین مسیرهای شبکه قرار دارند و باید به عنوان مسیر های مهم شبکه در نظر گرفته شوند. در شکل 5 مرکزیت بینیت هر کدام از تقاطع ها و خیابان های شهر رفسنجان را با رنگ مشاهده می کنید که کم ترین مرکزیت را با رنگ بنفش تیره و بیشترین مرکزیت با رنگ زرد روشن نمایش داده شده است.



شکل5 – نمایش مرکزیت بینیت در شبکه خیابان های رفسنجان

**B** - **B**-**B- 6- مرکزیت نزدیکی[[28]](#footnote-29)**

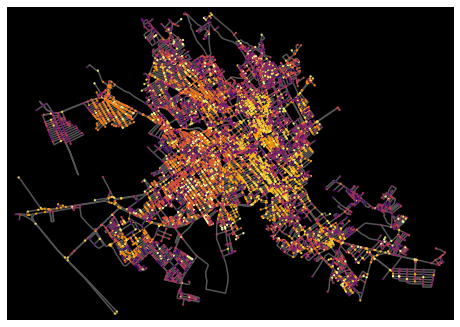
مرکزیت نزدیکی نشان می دهد که یک گره چقدر به سایر گره های شبکه نزدیک است و میانگین کوتاهترین طول مسیر از هر گره به گره های دیگر را در شبکه محاسبه می کند. برای هر گره u، الگوریتم مرکزیت نزدیکی مجموع فواصل آن را تا تمام گره‌های دیگر را بر اساس محاسبه کوتاه‌ترین مسیرها بین همه جفت گره‌ها محاسبه می‌کند. سپس مجموع حاصل معکوس می شود تا امتیاز مرکزیت نزدیکی برای آن گره تعیین شود. در شکل 6 مرکزیت نزدیکی هر کدام از تقاطع ها و خیابان های شهر رفسنجان را با رنگ مشاهده می کنید که کم ترین مرکزیت را با رنگ بنفش تیره و بیشترین مرکزیت با رنگ زرد روشن نمایش داده شده است.



شکل6– نمایش مرکزیت نزدیکی در شبکه خیابان های رفسنجان

**C**- **B**-**B-6- مرکزیت درجه[[29]](#footnote-30)**

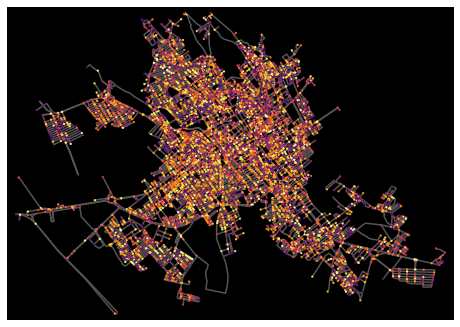
مرکزیت درجه ساده ترین معیار مرکزیت برای محاسبه است. درجه یک گره صرفاً شمارش تعداد خیابان های متصل (یعنی یال ها) به آن گره است. مرکزیت درجه برای یک گره به سادگی درجه آن است. یک گره با 10 اتصال خیابان دارای مرکزیت درجه 10 خواهد بود. گره با 1 یال دارای مرکزیت درجه 1 خواهد بود. در شکل 7 مرکزیت درجه هر کدام از تقاطع ها و خیابان های شهر رفسنجان را با رنگ مشاهده می کنید که کم ترین مرکزیت را با رنگ بنفش تیره و بیشترین مرکزیت با رنگ زرد روشن نمایش داده شده است.



شکل7– نمایش مرکزیت درجه در شبکه خیابان های رفسنجان

**C**- **B**-**B-6- مرکزیت پیج رنک[[30]](#footnote-31)**

الگوریتم پیج رنک بر اساس تعداد روابط ورودی و اهمیت گره های منبع مربوطه، اهمیت هر گره را در گراف اندازه گیری می کند. فرض اساسی به طور کلی این است که یک گره فقط به اندازه گره هایی که به آن پیوند دارند اهمیت دارد. در شکل 8 مرکزیت پیج رنک هر کدام از تقاطع ها و خیابان های شهر رفسنجان را با رنگ مشاهده می کنید که کم ترین مرکزیت را با رنگ بنفش تیره و بیشترین مرکزیت با رنگ زرد روشن نمایش داده شده است.



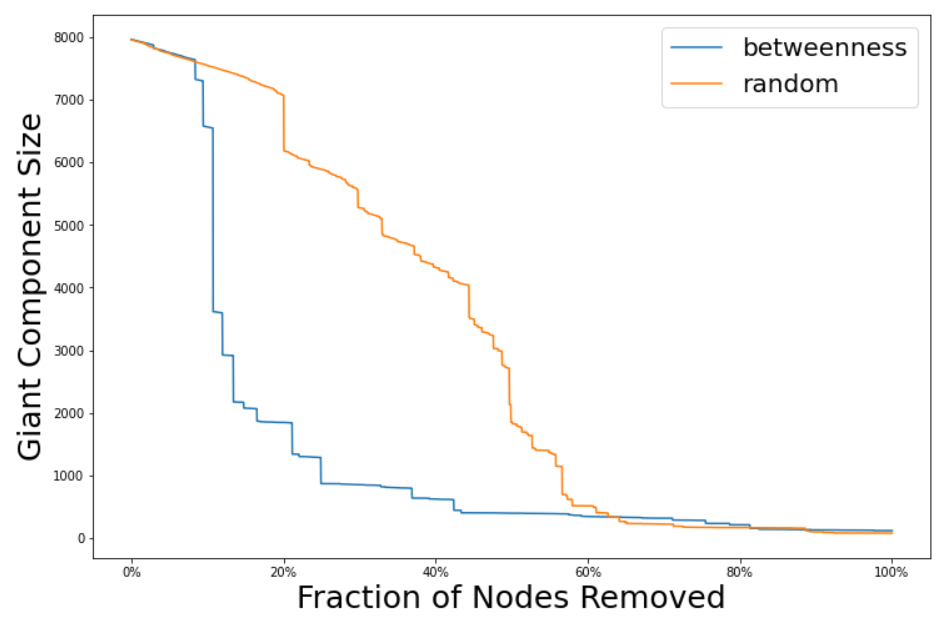
شکل8– نمایش مرکزیت پیج رنک در شبکه خیابان های رفسنجان

**C** -**6- حمله به شبکه[[31]](#footnote-32)**

برخی از اندازه‌گیری‌های اولیه را برای برسی مقاوت شبکه انجام دادیم، بنابراین زمان آن رسیده است که ببینیم واقعاً شبکه ما چقدر قوی است؟ برای این کار با دو رویکرد به گره های شبکه حمله خواهیم کرد:

1. گره ها را با توجه به مرکزیت بینیت محاسبه شده، از گره های با امتیاز بالا به سمت گره های کم امتیاز شروع به حذف کردن می کنیم.
2. نود ها را به صورت تصادفی از شبکه حذف کنیم.

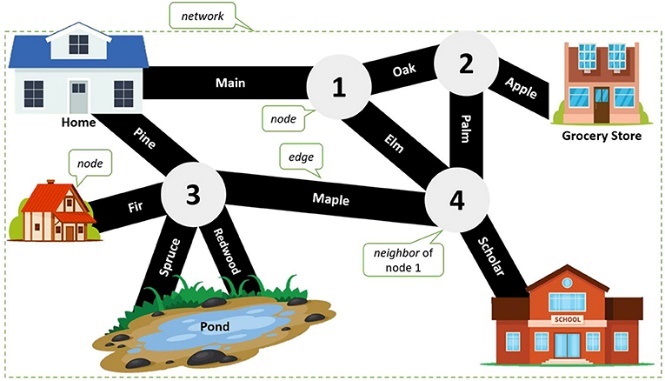
حذف گره‌ها باعث می‌شود که بزرگترین مولفه یا بزرگترین مؤلفه متصل در گراف، کوچک شود و برخی از گره‌ها ممکن است نقش مهمی در این فرآیند داشته باشند که باعث انقباض شدید بزرگترین مولفه می‌شوند[12]. همانطور که در شکل 9 مشاهده می کنید حذف گره هایی که نقش مهمی را در شبکه ایفا می کنند، منجر به کوچک شدن سریع بزرگترین مولفه نسبت به حذف تصادفی گره ها می شود، اما فقط تا یک درصد معین و در ابتدا فرقی نمی‌کند که گره‌ها به ‌طور تصادفی انتخاب شوند یا بر اساس اهمیتشان، که این نشان‌دهنده استحکام شبکه ما است. با این حال، در نقطه ای از آنجا حدود 10 درصد از گره ها حذف می شوند و حذف گره های مهم خاص منجر به کاهش بسیار سریعتر در اندازه بزرگترین مولفه می شود. پس گره های با مرکزیت بالا تاثیر مهمی را در اندازه مولفه همبند شبکه می گذارند. نکته جالب این جاست که تنها پس از حذف حدود 50 درصد از گره ها، اندازه بزرگترین مولفه به سرعت به سایز تقریباً صفر گره می رسد.



شکل 9 – مقایسه حمله به گره های با اهمیت و حمله به صورت تصادفی به نود ها در شبکه و تاثیر آن بر روی بزرگترین مولفه

**D** -**6- مسیر یابی[[32]](#footnote-33)**

ما هر روز تصمیم می گیریم که از چه مسیرهایی برای سفر بین مکان های مختلف شهر استفاده کنیم. در خانه، ممکن است از اتاق خواب به آشپزخانه سفر کنیم. در خارج از خانه، ممکن است از خانه به دانشگاه سفر کنیم. فرض کنید ما یک مجموعه ای از اشیاء (به نام "گره") و اتصالات (به نام "یال") بین آنها داریم. همانطور که قبلا هم گفتیم نقاط توسط خیابان ها و گذرگاه ها به یکدیگر متصل می شوند. هر کدام از این نقاط یک گره نامیده می شوند. شئ ها در یک شبکه به هم متصل هستند. به عنوان مثال، در شکل 1، هر مکان و هر تقاطع خیابان یک گره است، و خیابان ها و گذرگاه ها یال نامیده می شوند که دو گره را به یکدیگر متصل می کنند. عنوان مثال، هنگام سفر از خانه به دانشگاه، هر خیابان یک یال است. "همسایگان" یک گره، مجموعه گره هایی هستند که توسط یک یال به آنها متصل است. یک دنباله از گره مبدا به یک گره مقصد و مجموعه ای از یال ها بین یک گره مبدا و گره مقصد می باشند [9،13].



شکل 10– در این نقشه شهری کوچک، که نمونه ای از یک شبکه است، مکان ها (یعنی گره ها) در تقاطع بین خیابان ها (یعنی یال ها) هستند.

در سفر به نقاطی که نزدیک به هم هستند، تنها چند تقاطع خیابان (گره‌) وجود دارد و مسیرهای مختلف را امتحان می کنیم تا کوتاه‌ترین مسیر [[33]](#footnote-34)را پیدا کنیم. اما اگر نقاط دورتر از هم باشند، پیدا کردن کوتاه‌ترین مسیر بسیار دشوارتر است. اما چگونه ابزارهای ناوبری مانند نقشه گوگل بهترین راه برای رسیدن به مقصد را تعیین می کنند؟ یکی از راه‌ حل ها، مسئله کوتاه‌ترین مسیر است که یک مسئله ریاضی، برای یافتن مسیر بین دو تا نقطه است به نحوی که مجموع هزینه‌ها یال های در مسیر به حداقل برسد.

با استفاده از الگوریتم دایجسترا[[34]](#footnote-35)، می‌توانیم کوتاه‌ترین مسیر را از گره مبدا به هر گره دیگری در شبکه بیابیم. اگر خانه خود را به عنوان گره مبدا و مقصد خود را به عنوان گره دیگری در یک شبکه در نظر بگیرید، می توانید مسیر کوتاه و بهینه ای را از خانه خود به هر مکانی که می خواهید بروید تعیین کنید. یافتن کوتاه ترین مسیرها برای حل مشکلات شبکه های مختلف مهم و مفید است. به عنوان مثال، کوتاه ترین مسیرها می تواند کارایی برنامه ریزی شهری را بهبود بخشند. مهندسان عمران می توانند یک شهر را به عنوان یک شبکه نمایش دهند و بهترین مکان ها را برای ساخت جاده ها برای کاهش تراکم ترافیک و بهترین مکان ها برای قرار دادن لوله های آبیاری برای توزیع آب بین جمعیت تعیین کنند [13]. یافتن کوتاه‌ترین مسیرها همچنین انتقال داده‌ها را از یک کامپیوتر به کامپیوتر دیگر با سرعت‌های بالا امکان‌پذیر می‌سازد و به حجم عظیمی از اطلاعات اجازه می‌دهد در چند ثانیه انتقال یابند[9,13].

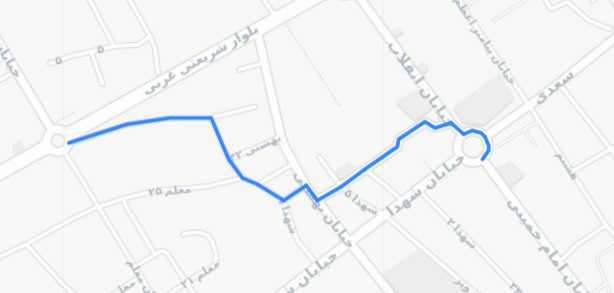
همچنین نمونه های زیادی از کوتاه ترین مسیرها در ارتباطات و شبکه های اجتماعی وجود دارد. به عنوان مثال، فرض کنید هر فرد در یک شبکه اجتماعی یک گره است و هر یال نشان دهنده یک دوستی است. شما می توانید از طریق ارتباطات دوستان خود بفهمید که چگونه با فردی خارج از گروه های دوستی خود ارتباط برقرار کنید. کوتاه ترین مسیرهای ارتباط (مانند دوستی) بین دو فرد تصادفی در ایالات متحده کوتاه تر از آن چیزی است که تصور می شود. در چنین مسیری به طور میانگین کمتر از شش پله بین فرد مبدأ و فرد مقصد وجود داشت[14]، کوتاه‌ترین مسیرها بین افراد، «پدیده جهان کوچک» را نشان می‌دهد، و این طول‌ مسیر کوتاه نیز الهام ‌بخش عبارت «شش درجه جدایی[[35]](#footnote-36)» است[9].

مثال دوم مربوط به شیوع بیماری هاست است. در طول همه‌گیری کویید نوزده[[36]](#footnote-37)، یافتن مسیرهای کوتاه برای محدود کردن افراد در معرض ویروس که باعث بیماری می‌شود مفید بوده است [16، 15].

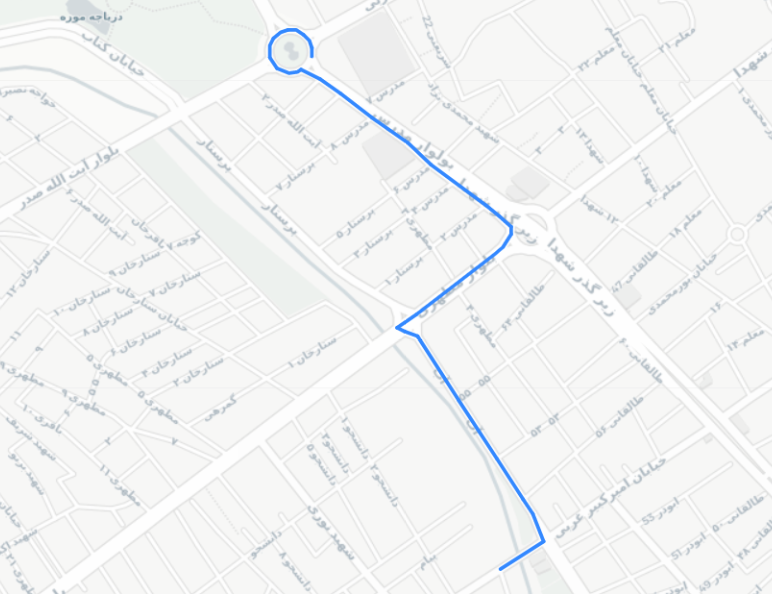
پس به طور کلی کوتاه ترین مسیرها هنگام سفر از یک نقطه به نقطه دیگر مهم هستند. آنها در بسیاری از شبکه ها کاربردهای متعددی دارند و می توانند به حل انواع مشکلات دنیای واقعی کمک کنند. از برنامه ریزی برای تعطیلات خانوادگی گرفته تا بررسی نحوه اتصال جهان ما، مطالعه کوتاه ترین مسیرها در شبکه ها بسیار مهم است و مبنایی برای تحقیقات پیچیده تر است.

در این بخش آخر خواهیم دید که با دادن نقطات مبدا و مقصد به تابع، کوتاه ترین مسیر را بین دو نقطه را محاسبه می کند،

ما مختصات دو میدان ابراهیم و میرافضلی را به ورودی تابع دادیم و همانطور که در شکل 6 مشاهده می کنید کوتاه ترین یا نزدیک ترین یا سریع ترین مسیر بین این دو نقطه را به ما نمایش می دهد.



شکل 11 – نمایش کوتاه ترین مسیر بین دو میدان ابراهیم و میر افضلی واقع در شهر رفسنجان



شکل 12 – نمایش کوتاه ترین مسیر بین میدان بسیج و پارک لاله واقع در شهر رفسنجان

**7- نتیجه گیری**

این مقاله شبکه خیابان های شهر رفسنجان را یک شبکه پیچیده می داند و با استفاده از ویژگی های شبکه آن را از دو جنبه اقتصادی و آسیب پذیری بررسی می کند. از نظر اقتصادی نیز نتایج نشان داد که به دلیل نبود پدیده جهان کوچک، توسعه خیابان های شهر رفسنجان به سمت اقتصادی و بهینه حرکت نکرده است. همچنین با یافتن تقاطع ها و مسیرهای آسیب‌پذیر شبکه و توزیع آن‌ها به کمک ویژگی‌های شبکه مانند مرکزیت بینیت، متوجه شدیم که آسیب‌پذیرترین تقاطع های هاب هستند که در صورت آسیب‌دیدگی می‌توانند تا حد زیادی به شبکه خیابان های رفسنجان آسیب وارد کنند. همچنین با توجه به نتایج به‌دست‌آمده، شبکه خیابان های شهر رفسنجان را می‌توان به‌عنوان یک شبکه سلسله مراتبی[[37]](#footnote-38) در نظر گرفت زیرا دارای درجه های نامتعادل و دارای نودهای هاب است. برای اقتصادی‌تر شدن شبکه خیابان های شهر رفسنجان، باید هاب ها را افزایش دهیم که باعث ایجاد مسیر های بیشتری در شبکه می شوند که این امر باعث می شود که پدیده جهان کوچک در شبکه ظاهر شود و علاوه بر کاهش متوسط طول مسیر و قطر شبکه، آسیب پذیری شبکه را کاهش می دهد. برای پرداختن عمیق به جنبه های اقتصادی، ما باید فرآیندهای موجود در این شبکه را بررسی کنیم و مرکزیت شبکه را بر اساس این فرآیندها و ساختار توپولوژیکی شبکه بازتعریف کنیم.

با توجه به این که ضریب خوشه بندی 0.02 که عدد پایینی هست در شبکه شهر رفسنجان و توان هم نزدیک به 2 و قطر شبکه برابر با 131 میانگین طول مسیر هم 46 به دست آمد بنابراین می توان نتیجه گرفت که پدیده جهان کوچک در شبکه وجود ندارد.

نبود پدیده جهان کوچک در شبکه خیابان های شهر باعث می شود فاصله نقاط و زمان سفر بسیار زیاد شود و این امر از نظر اقتصادی باعث کاهش کارایی شبکه می شود.

در نهایت می توان گفت که از نظر توپولوژیکی به دلایلی مانند قطر زیاد شبکه و متوسط طول مسیر، ضریب خوشه بندی نزدیک به صفر و توزیع نامناسب گره ها و مسیرهای آسیب پذیر، شبکه خیابان های شهر رفسنجان در سال های اخیر از نظر اقتصادی توسعه نیافته است و بهینه نمی باشد.

**منابع و مراجع**

[1] S. Wasserman, K. Faust, Social Networks Analysis, Cambridge University Press, Cambridge, 1994.

[2] R Pastor-Satorras, A. Vespignani, Evolution and Structure of the Internet: A Statistical Physics Approach, Cambridge University Press, Cambridge, 2004.

[3] V. Latora, M. Marchiori, Physica A 314 (2002) 109.

[4] Street network studies: from networks to models and their representations S Marshall, J Gil, K Kropf, M Tomko… - Networks and Spatial 2018 - Springer.

[5] The Role of The Transport Road Network in The Economic Development of Saudi ArabiaX Han, H Zhu, P Yu, Z M. Aldagheiri, 2009

[6] Hoyle, B. S., Transport and Development in Tropical Africa: Case Studies in the Developing World, London: John Murray, 1988.

[7] Investigating of topological characteristics of the Iranian railway network: A network science approach, Sina Firuzbakht & Mohammad Khansari,

[8] A. Barabasi, Network Science, 2nd ed. Northeastern University, Boston, 2016.

[9] M. E. J. Newman, Networks, 2nd ed. Oxford: Oxford University Press, 2018.

[10] Y. Jiang, B. Yao, L. Wang, T. Feng, and L. Kong, “Evolution trends of the network structure of Spring Airlines in China: A temporal and spatial analysis,” *J. Air Transp. Manage.*, vol. 60, pp. 18–30, 2017.

[11] <https://osmnx.readthedocs.io/en/stable>

[12] Graph measures and network robustness W Ellens, RE Kooij - arXiv preprint arXiv:1311.5064.

[13] Cramer, C., Porter, M. A., Sayama, H., Sheetz, L., and Uzzo, S. (eds). 2015. Network Literacy: Essential Concepts and Core Ideas.

[14] Milgram, S. 1967. The small-world problem.

[15] rooks, H. Z., Kanjanasaratool, U., Kureh, Y. H., and Porter, M. A. 2021. Disease detectives: Using mathematics to forecast the spread of infectious diseases. Front. Young Minds 9:577741. doi: 10.3389/frym.2020.577741

[16] Ying, F., and O’Clery, N. 2021. Modelling COVID-19 transmission in supermarkets using an agent-based model. PLoS ONE 16: e0249821. doi: 10.1371/journal.pone.0249821

پیوست

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143  144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166  167  168  169  170  171  172  173  174  175  176  177  178  179  180  181  182  183  184  185  186  187  188  189  190  191  192  193  194  195  196  197  198  199  200  201  202  203  204  205  206  207  208  209  210  211  212  213  214  215  216  217  218  219  220  221  222  223  224  225  226  227  228  229  230  231  232  233  234  235  236  237  238  239  240 | *"""*  *Author: Hossein Rafiee zade*  *"""*  **import** **osmnx** **as** **ox**  **import** **networkx** **as** **nx**  **import** **matplotlib.pyplot** **as** **plt**  **import** **numpy** **as** **np**  **import** **seaborn** **as** **sns**  **from** **collections** **import** Counter  **import** **random**  **import** **pandas** **as** **pd**  ax = ox.graph\_from\_place('Rafsanjan', network\_type='drive')  ax = ax.to\_undirected()  G=ox.project\_graph(ax)  ox.plot.plot\_graph(G,filepath="image2.png" , save=True)  nx.info(G)  k = sum([v **for** k, v **in** G.degree()]) / len(G)  G = nx.Graph(G)  nx.info(G)  k = sum([v **for** k, v **in** G.degree()]) / len(G)  between = nx.betweenness\_centrality(G)  *# plot it*  df = pd.DataFrame(data=pd.Series(between).sort\_values(), columns=['cc'])  df['colors'] = ox.plot.get\_colors(n=len(df), cmap='inferno', start=0.2)  df = df.reindex(G.nodes())  nc = df['colors'].tolist()  fig, ax = ox.plot\_graph(ax, bgcolor='k', node\_size=5, node\_color=nc, node\_edgecolor='none', node\_zorder=2,  edge\_color='#555555', edge\_linewidth=1.5, edge\_alpha=1 ,filepath="between.png" , save=True)  close = nx.closeness\_centrality(G)  *# plot it*  df = pd.DataFrame(data=pd.Series(close).sort\_values(), columns=['cc'])  df['colors'] = ox.plot.get\_colors(n=len(df), cmap='inferno', start=0.2)  df = df.reindex(G.nodes())  nc = df['colors'].tolist()  fig, ax = ox.plot\_graph(ax, bgcolor='k', node\_size=5, node\_color=nc, node\_edgecolor='none', node\_zorder=2,  edge\_color='#555555', edge\_linewidth=1.5, edge\_alpha=1,filepath="close.png" , save=True)  degree = nx.degree\_centrality(G)  *# plot it*  df = pd.DataFrame(data=pd.Series(degree).sort\_values(), columns=['cc'])  df['colors'] = ox.plot.get\_colors(n=len(df), cmap='inferno', start=0.2)  df = df.reindex(G.nodes())  nc = df['colors'].tolist()  fig, ax = ox.plot\_graph(ax, bgcolor='k', node\_size=5, node\_color=nc, node\_edgecolor='none', node\_zorder=2,  edge\_color='#555555', edge\_linewidth=1.5, edge\_alpha=1,filepath="degree.png" , save=True)  PageRank=nx.pagerank(G)  *# plot it*  df = pd.DataFrame(data=pd.Series(PageRank).sort\_values(), columns=['cc'])  df['colors'] = ox.plot.get\_colors(n=len(df), cmap='inferno', start=0.2)  df = df.reindex(G.nodes())  nc = df['colors'].tolist()  fig, ax = ox.plot\_graph(ax, bgcolor='k', node\_size=5, node\_color=nc, node\_edgecolor='none', node\_zorder=2,  edge\_color='#555555', edge\_linewidth=1.5, edge\_alpha=1,filepath="PageRank.png" , save=True)  ox.plot\_graph(ox.graph\_from\_place('Rafsanjan', network\_type='drive'), bgcolor='k',filepath="image5.png", node\_size = 4,  node\_color='#999999', node\_edgecolor='none', node\_zorder=1,  edge\_color='#555555', edge\_linewidth = 0.3, edge\_alpha=1,save=True)  ox.io.save\_graphml(ox.graph\_from\_place('Rafsanjan', network\_type='drive'), filepath=None, gephi=True, encoding='utf-8')  NumnerOfNodes=G.number\_of\_nodes()  NumnerOfedges=G.number\_of\_edges()  **print**(NumnerOfNodes)  **print**(NumnerOfedges)  degrees = [val **for** (node, val) **in** G.degree()]  Prob\_list=[]  **for** i **in** range(1,7):  Prob\_list.append(degrees.count(i)/NumnerOfNodes)  Deg\_list=[1,2,3,4,5,6]  plt.figure()  plt.plot(Deg\_list,Prob\_list)  plt.xlabel("k", fontsize=15)  plt.ylabel("P(k)", fontsize=15)  plt.title("Degree distribution", fontsize=15)  plt.savefig('fig00.png')  plt.grid(True)  plt.show(True)  plt.loglog(Deg\_list,Prob\_list)  plt.xlabel("k", fontsize=15)  plt.ylabel("P(k)", fontsize=15)  plt.title("Degree distribution", fontsize=15)  plt.savefig('fig2.png')  plt.show(True)  degree\_sequence = sorted([d **for** n, d **in** G.degree()], reverse=True)  **import** **powerlaw**  fit = powerlaw.Fit(degree\_sequence,xmin=1)  fit.alpha  fig2 = fit.plot\_pdf(color='b', linewidth=2)  fit.power\_law.plot\_pdf(color='g', linestyle='--', ax=fig2)  plt.savefig('fig3.png')  R, p = fit.distribution\_compare('exponential','power\_law',normalized\_ratio=True)  R, p = fit.distribution\_compare('exponential','lognormal\_positive',normalized\_ratio=True)  diameter = max([max(j.values()) **for** (i,j) **in** nx.shortest\_path\_length(G)])  degree\_dic = Counter(dict(G.degree()).values())  degree\_hist = pd.DataFrame({"degree": list(degree\_dic.values()),  "Number of Nodes": list(degree\_dic.keys())})  plt.figure(figsize=(20,10))  sns.barplot(y = 'degree', x = 'Number of Nodes',  data = degree\_hist,  color = 'darkblue')  plt.xlabel('Node Degree', fontsize=30)  plt.ylabel('Number of Nodes', fontsize=30)  plt.tick\_params(axis='both', which='major', labelsize=20)  plt.show()  G2 = G.to\_undirected()  nx.draw(G2, pos=nx.spiral\_layout(G2,scale=4, center=None, dim=2, resolution=0.5, equidistant=False), node\_size=0.03, width=0.1, node\_color='lightblue' , edge\_color='k')  plt.savefig('fig5.png')  G2 = G.to\_undirected()  nx.draw(G2, pos=nx.spring\_layout(G2,scale=2), node\_size=0.03, width=0.1)  plt.savefig('fig6.png')  g = G.to\_undirected()  pos=nx.spring\_layout(g)  fig, ax = plt.subplots()  fig.set\_facecolor('black')  nx.draw(g,pos,node\_color='purple',edge\_color='lime',width=0.1,edge\_cmap=plt.cm.Blues,node\_size=1, node\_shape='^')  ax.axis('off')  fig.set\_facecolor('black')  plt.savefig("all\_graph.pdf")  plt.savefig("all\_graph.png")  **print**(nx.info(g))  nx.node\_connectivity(G)  density=nx.density(G)  sorted\_x = sorted(between.items(), key=operator.itemgetter(1), reverse=True)  rand\_x = list(range(0,4426 ))  random.shuffle(rand\_x)  between\_giant = []  between\_rand = []  avg\_degs = []  G\_simple = nx.Graph(G)  G\_simple2 = nx.Graph(G)  **for** x **in** range(3000):    remove = sorted\_x[x]  remove2 = sorted\_x[rand\_x[x]]  G\_simple.remove\_nodes\_from(remove)  G\_simple2.remove\_nodes\_from(remove2)    connected\_component\_subgraphs1 = (G\_simple.subgraph(c) **for** c **in** nx.connected\_components(G\_simple))  connected\_component\_subgraphs2 = (G\_simple2.subgraph(c) **for** c **in** nx.connected\_components(G\_simple2))  giant = len(max(connected\_component\_subgraphs1, key=len))  giant2 = len(max(connected\_component\_subgraphs2, key=len))  between\_giant.append(giant)  between\_rand.append(giant2)  y1 = between\_giant  y2 = between\_giant  y1= y1[ :-1]  y2= y2[1: ]  perc = np.linspace(0,100,len(between\_giant))  fig = plt.figure(1, (12,8))  ax = fig.add\_subplot(1,1,1)  ax.plot(perc, between\_giant)  ax.plot(perc, between\_rand)  fmt = '**%.0f%%**' *# Format you want the ticks, e.g. '40%'*  xticks = mtick.FormatStrFormatter(fmt)  ax.xaxis.set\_major\_formatter(xticks)  ax.set\_xlabel('Fraction of Nodes Removed',fontsize=25)  ax.set\_ylabel('Giant Component Size',fontsize=25)  ax.legend(['betweenness','random'],fontsize=20)  plt.savefig('fig7.png')  plt.show()  **import** **folium**  **import** **osmnx** **as** **ox**  **import** **networkx** **as** **nx**  ox.config(use\_cache=True, log\_console=True)  G = ox.graph\_from\_point((30.40631, 55.9821), dist=3000, network\_type='drive')  G = ox.speed.add\_edge\_speeds(G)  G = ox.speed.add\_edge\_travel\_times(G)  orig = ox.get\_nearest\_node(G, (30.40631, 55.9821))  dest = ox.get\_nearest\_node(G, (30.39866, 55.98592))  route = nx.shortest\_path(G, orig, dest, 'travel\_time')  route\_map = ox.plot\_route\_folium(G, route)  route\_map.save('test2.html') |

1. [https://www.openstreetmap.org/search?query=rafsanjan#map=12/30.3862/55.9868](https://www.openstreetmap.org/search?query=rafsanjan%23map=12/30.3862/55.9868) [↑](#footnote-ref-2)
2. <https://osmnx.readthedocs.io/en/stable> [↑](#footnote-ref-3)
3. weighted graph [↑](#footnote-ref-4)
4. complex network [↑](#footnote-ref-5)
5. Degree [↑](#footnote-ref-6)
6. link [↑](#footnote-ref-7)
7. Average degree [↑](#footnote-ref-8)
8. Network diameter [↑](#footnote-ref-9)
9. BFS [↑](#footnote-ref-10)
10. Average path length(APL) [↑](#footnote-ref-11)
11. Clustering coefficient [↑](#footnote-ref-12)
12. Betweenness centrality [↑](#footnote-ref-13)
13. Connectivity and Density [↑](#footnote-ref-14)
14. Small world phenomenon [↑](#footnote-ref-15)
15. Gephi [↑](#footnote-ref-16)
16. Average distance [↑](#footnote-ref-17)
17. Average degrees [↑](#footnote-ref-18)
18. [networkx.org](file:///C:\Users\Hossein\Desktop\New%20folder%20(3)\networkx.org) [↑](#footnote-ref-19)
19. Degree distribution [↑](#footnote-ref-20)
20. [pypi.org/project/powerlaw](file:///C:\Users\Hossein\Desktop\New%20folder%20(3)\pypi.org\project\powerlaw) [↑](#footnote-ref-21)
21. Degree exponent [↑](#footnote-ref-22)
22. Exponential distribution [↑](#footnote-ref-23)
23. Scale free network [↑](#footnote-ref-24)
24. Anomalous Regime [↑](#footnote-ref-25)
25. Network Robustness [↑](#footnote-ref-26)
26. Node Connectivity [↑](#footnote-ref-27)
27. centralities [↑](#footnote-ref-28)
28. closeness centrality [↑](#footnote-ref-29)
29. degree centrality [↑](#footnote-ref-30)
30. page rank [↑](#footnote-ref-31)
31. Network Attacks [↑](#footnote-ref-32)
32. Routing [↑](#footnote-ref-33)
33. Shortest path length [↑](#footnote-ref-34)
34. Dijkstra's algorithm [↑](#footnote-ref-35)
35. Six degrees of separation [↑](#footnote-ref-36)
36. COVID-19 [↑](#footnote-ref-37)
37. hierarchical network [↑](#footnote-ref-38)