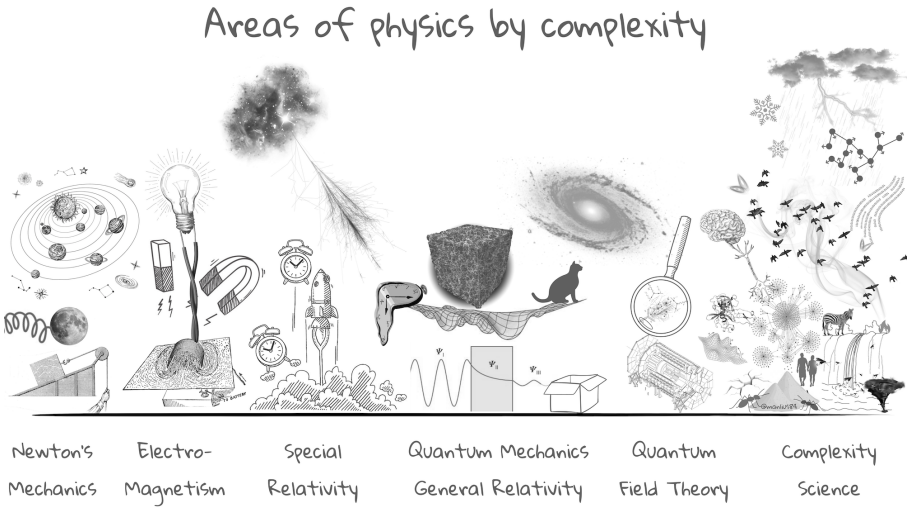
Final Report

Physics of Complex Networks: Structure and Dynamics

Università degli Studi di Padova




# Projects # 01, 16 & 31

**David Weingut**

**Last update**: June 24, 2024

# Contents


**1 Task 01** **1**
- 1.1 Theoretical Basis . . . 1
- 1.2 Simulation . . . 1
- 1.3 Results for Random Graphs . . . 2

**2 Task 16** **4**
- 2.1 Overview . . . 4
- 2.2 Random Graphs . . . 5
- 2.3 Autonomous System Maps . . . 5
- 2.4 Elementwise Application . . . 6

**3 Task 41** **7**
- 3.1 Data and Task Overview . . . 7
- 3.2 Data Processing . . . 7
- 3.3 Analysis of the Resulting Graphs . . . 9

**A The Appendix** **11**

**B Task 01** **12**

**C Task 16** **14**

**D Task 41** **15**

**E Bibliography** **17**

# 1 | #01: Ising Model

**Task leader(s):** *David Weingut*

## 1.1 | Theoretical Basis

The Ising model is a simplified version of the spin-spin interaction on a lattice. The spins are assumed to be $\pm 1$ and interactions are only allowed between direct neighbours.

This is achieved by the following Hamiltonian, equation 1.1, where $J_{ij}$ is the coupling strength between vertices $i$ and $j$ and $s_i$ is the spin value of Vertex $i$. $M$ is the external magnetic field. The sum touches each interacting pair once.

$$\mathcal{H} = -\sum_{i<j} J_{ij} s_i s_j - \sum_i M s_i \tag{1.1}$$

In most cases the external magnetic field is set to zero, which eliminates the latter part and leaves only the interaction term. The sum over spin pairs can also be expressed as a sum over the edges.

For a two dimensional grid the behaviour has been solved analytically in 1944 by Onsager[11]. For an arbitrary structured complex network there hasn't been any due to the the enormous number of configurations.

## 1.2 | Simulation

For the simulating approach simulated annealing was chosen to get the time evolution for the Ising model. A single flip Metropolis sampler was chosen due to its good performance and ease of implementation.

To find the critical temperature $T_c$ I found the magnetization $M$ to be most consistent. This is due to it being a monotonous function of temperature which makes it very easy to find the temperature where it crosses a certain threshold. The magnetical susceptibility $\chi$ was regarded as promising in the beginning because it is supposed to peak at $T = T_c$ but its volatile nature with very large uncertainties near the peak made finding the critical temperature a very unreliable task with strong fluctuations. The heat capacity $C_v$ was very similar, supposed to peak at $T_c$ but rather unreliable in helping find it. Additionally for the network families I tested, $\chi$ and $C_v$ peaked at different temperatures, with no way to discern reliably which of the peaks was the more reliable for my research, the representative behaviour is shown in the appendix, figure B.1.

The theoretical predictions of the model's behaviour are manifold. I tried to match the results of my simulations to the research paper of Leone et al.[8]. They provide a theoretical result for the critical temperature as a function of the first and second moments of the degree distribution. A problem I encountered with this equation is that I couldn't match the behaviour shown in the paper, neither via a replication of the formula nor via my simulation results. The most obvious is the missing divergence of the critical temperature for degree exponents below 3. In figure B.2 I show the dependence of magnetization and energy vs temperature.

## 1.3 | Results for Random Graphs

In the following the behaviour of the critical temperature, determined as above, is explored for some families of random graphs. As the behaviour should not depend on the size of the graph in a strong way, for ER and WS graphs the size is chosen to be 500 nodes, while for the SBM the size is chosen to

### Erdös-Renyi

For the Erdös-Renyi graphs the behaviour vs the mean degree is studied. One can see

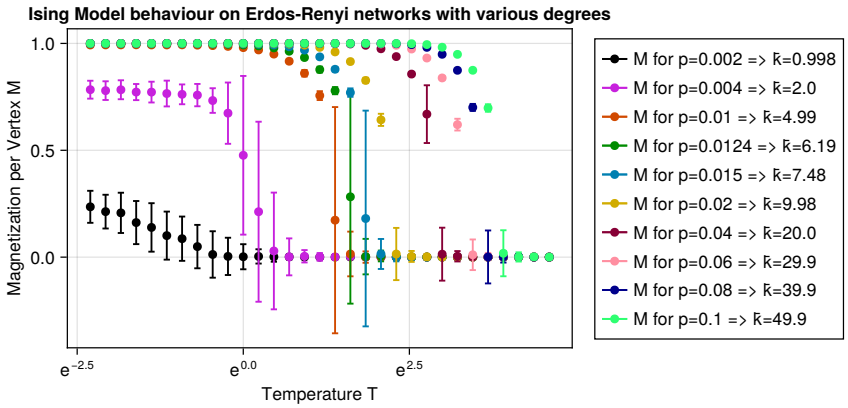



Figure 1.1: Behaviour of the magnetization of an Ising model on-top an Erdös-Renyi graph for a collection of temperatures and mean degrees

that the magnetization for a fixed temperature strongly depends on the average degree. This can be explained intuitively as for the lower degrees there are lot of independent components in the networks who don't interact with each other but by this lower the total average magnetization. The high degree networks in contrast have a structure which makes it very hard for individual spins who are surrounded by parallel peers to flip due to the energy change based flipping probabilities.

### Watts-Strogatz

For the Watts-Strogatz small world graphs the dependence on the rewiring probability is looked into. For low ones the onset temperature rises with the probability but stagnates for around $p = 0.4$ whereafter sees no more changes.

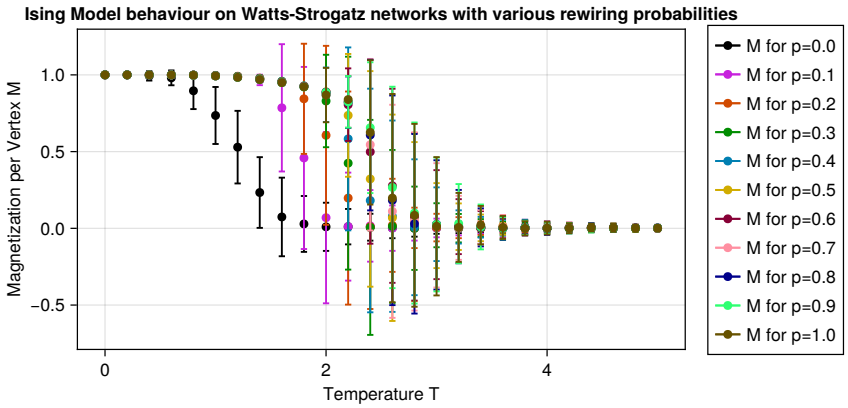



Figure 1.2: Behaviour of the magnetization of an Ising model on-top a Watts-Strogatz graph for a collection of temperatures and rewiring probabilities

### Stochastic Block Model

For the stochastic block model a total of 8 communities with intra-community links having a fixed probability of 0.05 will be used, while the modified parameter is the ratio of inter to intra is modified between 0 and 1. Here one can again see a strong correlation

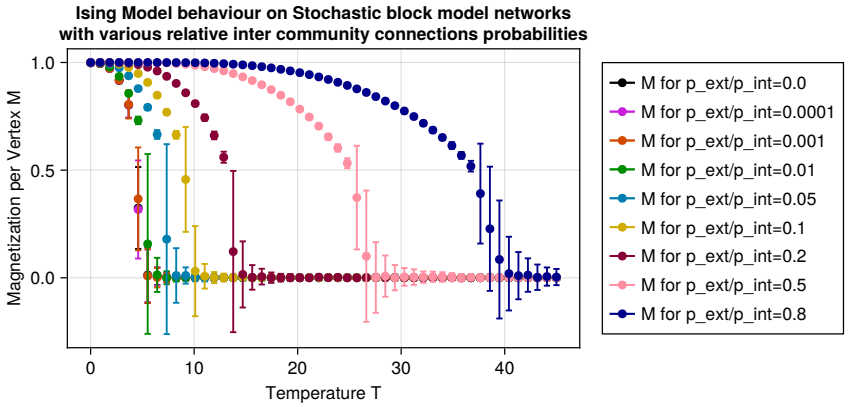



Figure 1.3: Behaviour of the magnetization of an Ising model on-top a stochastic block model graph for a collection of temperatures and relative inter-community edge probabilities

between the mixing parameter and the temperature at which the magnetization begins to decrease. A similar argument to the Erdös-Renyi seems plausible, for very low mixing between the communities, they evolve independently and thus different clusters can both be homogeneously magnetized but opposite to other clusters, which in turn lowers total magnetization of the graph.

# 2 | #16: Traffic Congestion

**Task leader(s):** *David Weingut*

## 2.1 | Overview

This project is concerned with a simplified model of traffic on top of a network. The simulated dynamics will follow the behaviour outlined in the research papers[6][5]. The system consists of a network and packets moving in this network. The packets are simple entities, they just contain their own destination as data. Each node of the network can hold a potentially infinite number of packets. A packet gets effectively destroyed upon arrival at its destination. Each time step a single packet is sent from each node that has a packet to send to one of its neighbours. The neighbour is for a deterministic approach chosen according to equation 2.1, where $b$ is the best next node for the packet, chosen by being the neighbour of current vertex $l$ minimizing the effective distance function $d_{\text{eff}}$.

$$b = \underset{i \in \{1,\dots,k_l\}}{\arg\min}\, d_{\text{eff}}\,(i, \text{destination}) \tag{2.1}$$

The dependence of the behaviour of the system on the choice of distance function is the main interest in this project. A naive approach would be to just follow the shortest path to the destination, so $d_{\text{eff}}(i,j) = hd(i,j)$, with $d$ being the geodesic distance. A possible improvement suggested in the aforementioned papers is to "incorporate local traffic information"[5, Abstract]. This happens via an configurable interpolation of shortest path information and queue length for the candidate, the distance function takes the form $d_{\text{eff}}(i,j) = hd(i,j)+(1-h)n_i$ where $n_i$ is the number of packets currently at node $i$.

The papers differ in the approach they take to the temporal distribution of the addition of packets. I am using the approach of [5], that a continuous influx of $p$ packets per time step (pps) is realized and the amount of packets currently in the system is observed. They introduce an order parameter $\rho$ describing the relative change of active packets.

$$\rho = \lim_{t\to\infty} \frac{A(t+\tau) - A(t)}{\tau p} \tag{2.2}$$

It takes values in the range 0, a stationary state meaning free flow, and 1, no packages reach their destination, total congestion.

In the following I will investigate the behaviour of the dynamics for different network topologies.

## 2.2 | Random Graphs

A first idea was to test different routing strategies on different types of random graphs like Erdös-Renyi or Watts-Strogatz based ones. This idea was then discarded again when I read that the behaviour was strongly dependent on the exact clustering of the graph[5, p. 2] and after I discovered in some preliminary simulations that the spread of the active packets is quite large even for the same exact network. In connection with wanting to fairly represent the ensemble I thought it too computationally complex to get a faithful estimate of the performance across both graphs in an ensemble and different simulation runs for one topology.

## 2.3 | Autonomous System Maps

For this reason I decided that I would take a real network, inspired by the original papers and motivations for the algorithm chosen to be an autonomous system map. The one I chose is from 29 December 1998, collected for [10] and downloaded from [9]. It has 493 nodes and 1234 edges with an average clustering coefficient of 0.1756 and a diameter of 8 hops.

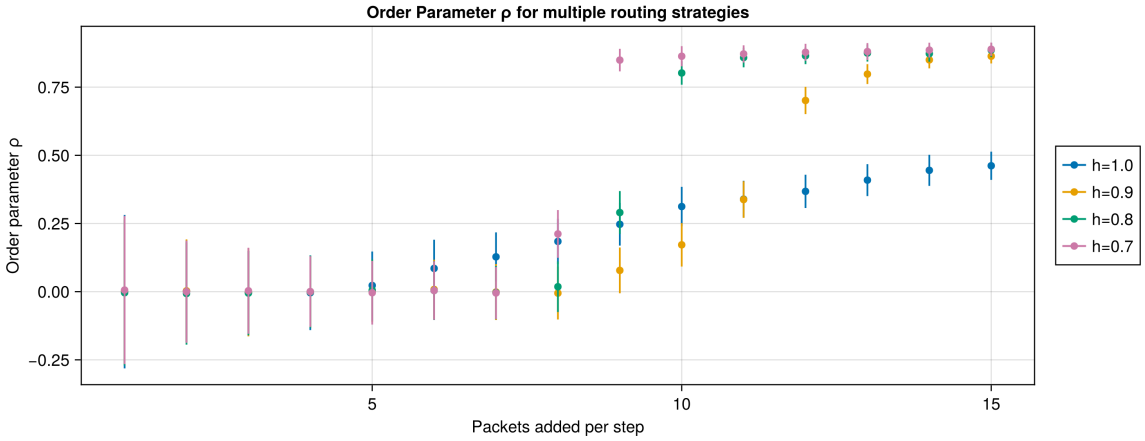



Figure 2.1: Comparison of neighbour choice algorithms. More details in the text.

The behaviour depending on the weight of the path length vs the queue length can be seen in figure 2.1. It can be clearly seen that for high package creation rates the choice based only on the shortest path is the one which handles it most effectively, even though there still is a growth of the active packet count by almost half the input rate, the network seems to be over-saturated. For the queue length aware algorithms there is an area between 5 pps and 8 pps to 10 pps, depending on the exact value of $h$, where they reign superior. This increased performance however rapidly increases if a certain threshold is crossed. While $h = 1$ needs 4 more packets to reach $\rho = 0.25$ after initially departing from 0 while for $h = 0.7$ it only takes 2 packets to go from $\rho \approx 0$ at 7 pps to reach $\rho > 0.75$ for 9 pps. For higher values of $h$ the transition is more gradual and comes later. The nature of the phase transition agrees with [5] while they didn't find a correlation of $h$ with the onset of it.

## 2.4 | Elementwise Application

As an additional idea I wanted to test if there was any difference in behaviour if the effective distance according to equation 2.1 would not be applied during the final step but during the search for the shortest path. For this I applied the respective distance function to each element of the weight matrix. This made the graph into a weighted graphs where the edge weights are constantly changing according to the number of packets residing in the target node of the edge. This will be called the element wise approach in the following. For computational reasons I used the smallest network in the dataset I used in section 2.3. This network is from 29 August 1999 has 103 nodes and 248 edges with an average clustering coefficient of 0.2703 and a diameter of 6 hops.

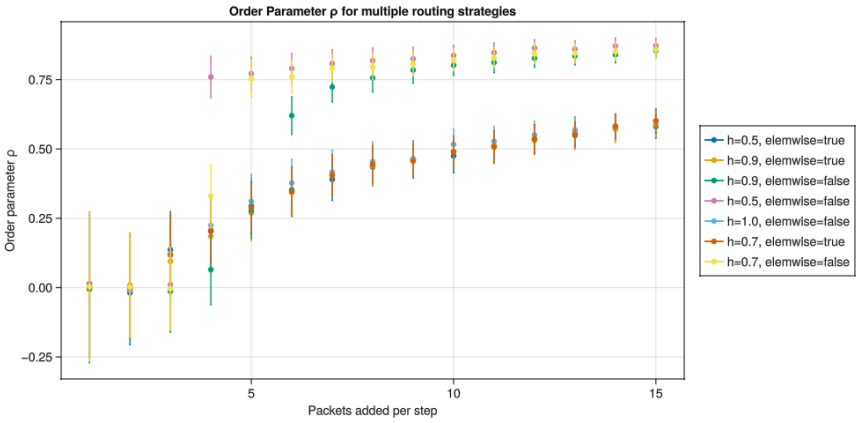



Figure 2.2: Comparison of different path length modifiers and neighbour choice algorithms. More details in the text.

The results can be seen in figure 2.2. For this simulation run I chose to use the aforementioned adjusting of the weights wherever the label in the legend says `elemwise=true`. Here the change of single weights in turn used just the length of the shortest path found by the weighted search algorithm, the neighbour selection was greedy and not taking into account the traffic information at the next node, only on its further path. The performance of these modified protocols were within the margin of error identical to the protocol that completely disregards traffic information. In figure C.1, shown in the appendix, the original behaviour as described in equation 2.1 was kept additionally to adding traffic information along the path. Here the element wise approach's performance was pretty much indiscernible from the one without my addition. In conjunction these figures seem to imply that the choice of the next node to travel to is the most important one while the traffic information later in the path has a negligible impact. This may be connected to the unpredictable nature of the traffic in this scenario as while you look at the second next node, there could be up to $k - 1$ more packets waiting for delivery. I did not look at the detailed behaviour of the model for this algorithm, the crowding behaviour in connection to the betweenness centrality as done in [5] may be an interesting avenue for further research. In total I would say the additional computational complexity of the approach described in this section is not worth it for unclear gains.

# 3 | Public Transport in Large Cities Worldwide

**Task leader(s):** *David Weingut*

## 3.1 | Data and Task Overview

This task aims to convert the unstructured data available on Citylines.co[1]. The data is community sourced networks of public transport in cities around the world. For this purpose there exists a website with the ability to both view and, after a registration, edit the available data. To facilitate the data editing it is split into multiple interoperating sets. The basic building blocks are sections and stations. There sections are collections of individual geographical coordinates which together form a multiline usually representing a part of a line, for example (part of) a bus route or (part of) a railway. A station is a single coordinate. Both these primitives contain metadata. Both sections and stations can be assigned to lines, which are logical units of a transport network, for example a bus line, identified at the city level by a number or name. The assignment of sections and stations to lines must not be one-to-one as lines can share parts of their routes or stations. For this reason there exist two separate datasets, one mapping IDs to basic data as where they are (city and geographical coordinates) and years regarding their build start and opening, the other provides the mapping between IDs and lines and additionally information regarding the web interface like when it was created or last updated in the web interface. The data regarding the internal workings of the website is ignored in the following as there is no relevance for the underlying networks. Lines can then be joined together to form a system, so the collection of all bus lines in a city can be found under the umbrella of the bus system. Each system is assigned a mode which represents the usual carriage used to perform passenger transportation, in most cases either a bus or a variation of rail transport, trams and subways are common subtypes. A city can be composed of multiple systems.

The task is to untangle this raw dataset and try to create a collection of easily readable network files including metadata.

## 3.2 | Data Processing

First of all the CSV files from the website were downloaded, due to the random nature of the file names on the server they were downloaded once and afterwards not updated, as trying to get the current filename and try to redownload the file was regarded as out-of-scope for this project.

Then all the files were read into dataframes. The approach for all the cities is

identical, so the following is repeated for each city. For a city then you see if any of the relevant subsets of the data is empty. This can happen due to the data being crowd sourced, so for example some cities only route segments were mapped out, while not a single station was added. First a common, at this point empty, graph is initialized for the city. After this check an iteration through all the systems of the city is performed. If a system has no lines it is just skipped. The next focus are the individual lines of the system. Here first the segments and stations relevant to the line are extracted from the data. Next each section is translated to a vector of 2D points. Under the simplifying assumption that a line contains no fork, the sections are joined. This assumption is reasonable as a fork is in most cases split into two cases, for example it is often avoided to have one line with multiple possible endpoints to reduce possible confusion for the customer of the transport services.[1] That assumption makes the workings of my algorithm possible. The algorithm keeps one segment of so far sorted sections, this is initialized to the first section in the dataset for reasons of simplicity, and a vector of all the other sections. Then in each step one finds the section that seems to most plausibly extend the already continuous segment. To obtain this the distances of heads and tails of all remaining sections to either end of the sorted segment are compared and it is determined whether a reversal of the section is necessary to get the smoothest possible continuation. An example of an internal state can be seen in the appendix in figure D.1. This is repeated until there are no more unmatched sections. This somewhat convoluted approach is necessary because there can be a difference between end of one section and the start of the next and because the order of the sections in the dataset is not necessarily logical for that line but only due to the order of insertion in the database. The next step is to use the reached continuous route to find out which stations are connected to which other stations or in which order the route passes the stations. For this purpose some trigonometry is used to calculate the distance between a line and a point. If the distance between the line and a station is below a threshold it is added to a vector containing all the stations encountered so far. A station is not added again if it was also the previously added one. The threshold was chosen as 20 m by me, based on the intuition to allow some leeway in mapping but avoid false positives. This needs to be done because the points in the sections are not regularly spaced and waypoints not necessarily coincide with locations of stations. A previous approach based only on distances to section points had to be discarded because in the case of very straight routes the distance between waypoints can exceed the distance between stations. As the next step the stations that were found to be on the route are deduplicated based on the exact name. This happens because frequently the stations for the different directions are both entered in the dataset which seems to be not wanted for a network analysis, so here the approach of simplification was chosen. Next all the stations are added to the graph and edges between those adjacent in the found route are created. After all the systems were iterated the resulting graph is written to the file system as two files, one for the nodes and their metadata and one for the edges and metadata.

## 3.3 | Analysis of the Resulting Graphs

[1] Sometimes this leads to disconnected regions, as can be seen for Algiers. But as this seemed to be a rare occurrence based on a quick visual inspection of the outputs, this was deemed to be a worthy trade-off for mostly reliable section joins.

For the analysis of the resulting graphs I decided not to look at any of the graphs in detail but more to try and see if there are any underlying patterns that seem to emerge from the pool of data. For this the files written in the previous step are read back into graphs. Then they are analysed both as they are and after a simple name based connection mechanism and adding edges for stations less than 100 m apart. The latter is to study the general structure after taking into consideration that even though stations might not be in the exact same place, if they share a name they belong to the same logical unit of a station. This also leads to the effect of better connecting different lines and systems and thereby transport modes. The analysis consisted of calculating the size (vertex count), the mean degree, the assortativity, the diameter in terms of the geodetic distance between stations connected by a route, the diameter in terms of edges needed, the relative size of the largest connected component(LCC) and the power law exponent as obtained by a fit to the degree distribution as implemented by the igraph software[3]. For the not further connected networks, for the diameters and power law fits only the largest connected component was used.

The full collection of pairwise plots can be seen in figure D.2 in the appendix while here only follows a short analysis of some findings deemed interesting. The size distribution is strongly heterogeneous with its value stretching from 3 to 3272 with a mean of 139 and a median of 57. After the connecting measures the median gets reduced by 3, the maximum changes to 2608 and the mean to 122.84. This shows that the amount of logical stations gets strongly overestimated by the simple data extraction mechanism. The mean degree is 1.9 and 2.0 respectively. This makes sense in conjunction with the assumption that most stations are part of just one line and are therefore only connected to the next and previous stations on the route. The assortativity is centered on 0 pretty exactly, slightly below for the unconnected and slightly above after the connection run, in both cases around one tenth of a standard deviation or less away from 0. Unsurprisingly the measures for the both diameters strongly correlate with each other and are both significantly lowered by the adding of connections. Further evidence consolidating the need for the connection adding is the distribution of the size of the LCC, this quantity has a mean of 0.65 before and 0.91 afterwards. In general I would expect a traffic network to be mostly connected. This could of course also be of either a faulty dataset or a faulty processing of the data and the shortcomings of this are smoothed over by the connection algorithm. This could be an avenue for further analysis and improvement of the pipeline. For the power law exponent there are about 5 % to 10 % of the network where the fit fails and the p-value for the fit is below 0.05. The average is 1.5 for the untreated network and gets increased by half a standard deviation to 1.6 due to the added connections. These exponents nonetheless seem to be quite low compared to the typical exponents of 2 to 3 as were presented as usual in the lecture. In general there were no clear correlations between metrics.

*Structure as*[2]*:*

- *A short (max 1 page) explanation of the task, including references. Include mathematical concepts.*

[2]Remove this part from the report

- *Max 2 pages for the whole task (including figures)*
- *It is possible to use appendices for supplementary material, at the end of the report. Max 5 pages per task*

*A total of 3 pages + 5 supplementary pages per task*

# A | The Appendix

This appendix lists mainly figures that highlight interesting or surprising behaviours but only serve to support the want for a more detailed picture and are not relevant for the main points.

The code for the simulations in this report were written using Julia[2] and would not have been possible without its extensive standard library and package ecosystem, including but not limited to `Graphs.jl`[7] for the network side and `Makie.jl`[4] for visualization.

# B | #01 Ising Model

As seen in figure B.1 the curves of susceptibility and heat capacity do not peak for the same temperature and are relatively noisy with large variations.

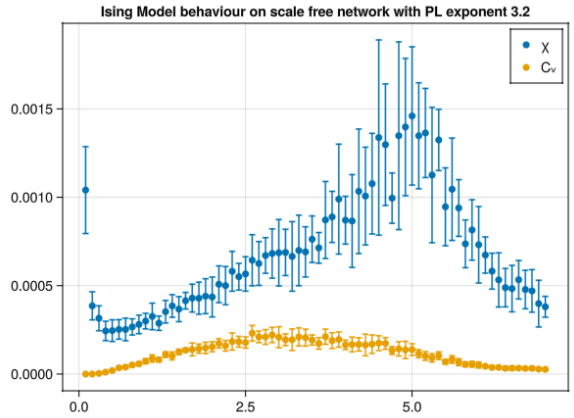



Figure B.1: Behaviour of the susceptibility and heat capacity for a scale free network with a degree exponent of 3.2

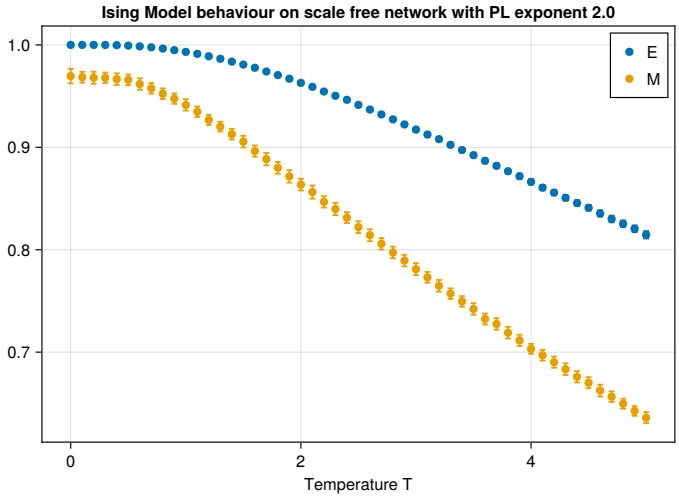



Figure B.2: Behaviour of the energy and magnetization for a scale free network with a degree exponent of 2

In figure B.2 I show that M and E do not stay near 1, which would be indicative of the always ferromagnetic behaviour, as it is described in [8] to happen. It is remarkable though that the transition seems to begin for rather low temperatures and then continue very slowly.

In figure B.3 a variety of target thresholds is compared to determine the critical temperature from. It is visible that for a threshold too large the temperature is much

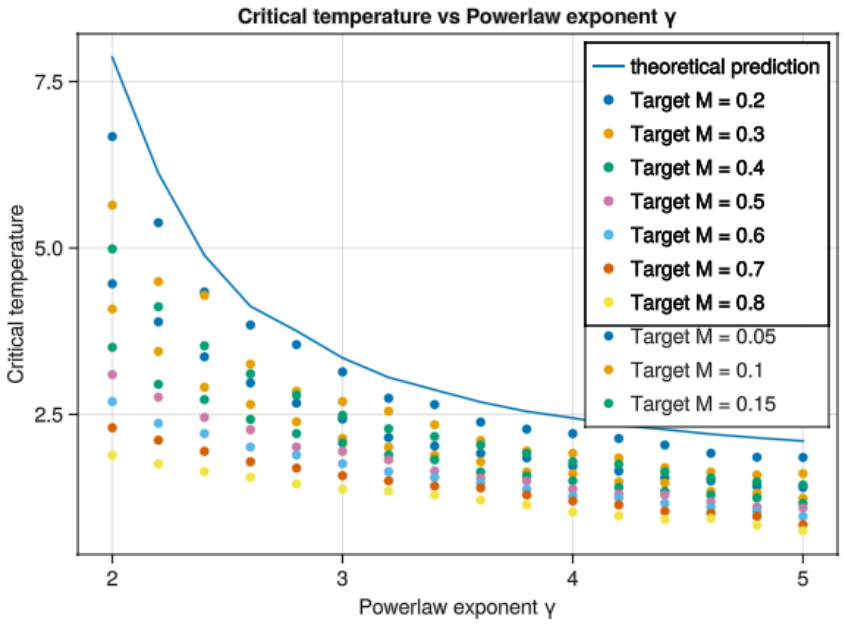



Figure B.3: Behaviour of the critical temperature as determined by various thresholds

too low while a very small one leads to very noisy estimate. The optimal one seems to be at 0.03 to 0.05.

# C | #16: Traffic Congestion

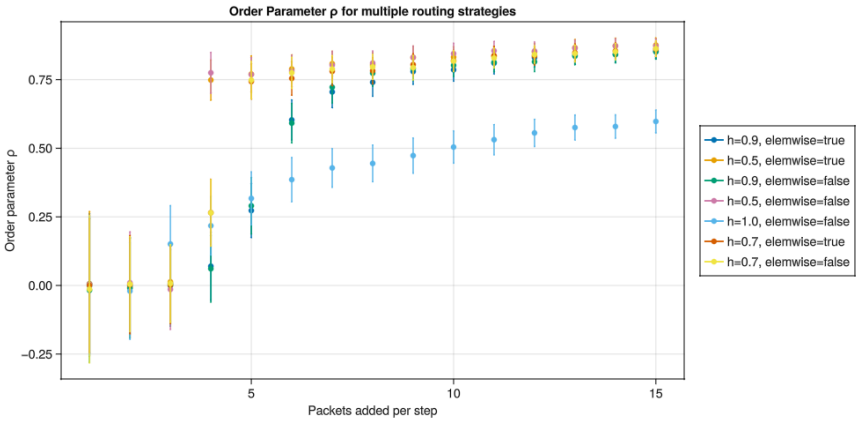



Figure C.1: Comparison of different path length modifiers and neighbour choice algorithms. Application of the effective distance function for each step of the way.

# D | #41 Public Transport in Large Cities Worldwide

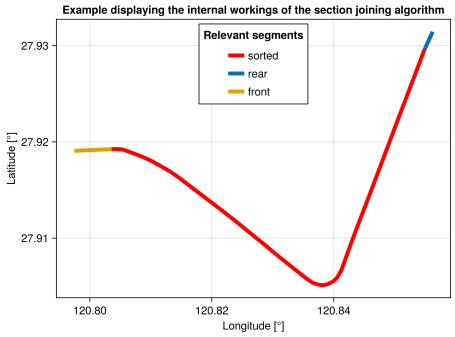



Figure D.1: This figure shows a snapshot of the algorithm described in section 3.2. 'sorted' in the legend refers to the already joined segments, 'rear' and 'front' to the tail and head end of 'sorted' respectively

| **variable** | **mean** | **min** | **median** | **max** | **nmissing** |
|---|---|---|---|---|---|
| size | 139.0 | 3.0 | 57.0 | 3272.0 | 0 |
| mean degree | 1.895 | 1.07 | 1.91 | 2.54 | 0 |
| assortativity | -0.0335 | -1.0 | -0.0371 | 0.495 | 0 |
| diameter [m] | 46997.1 | 1028.13 | 27822.2 | 4.17e5 | 0 |
| diameters [#edges] | 26.6 | 2.0 | 24.0 | 107.0 | 0 |
| relative size LCC | 0.646 | 0.0476 | 0.636 | 1.0 | 0 |
| $\alpha$s | 1.521 | 1.192 | 1.486 | 2.12 | 18 |

Table D.1: Metric distribution in the graphs as extracted from the data

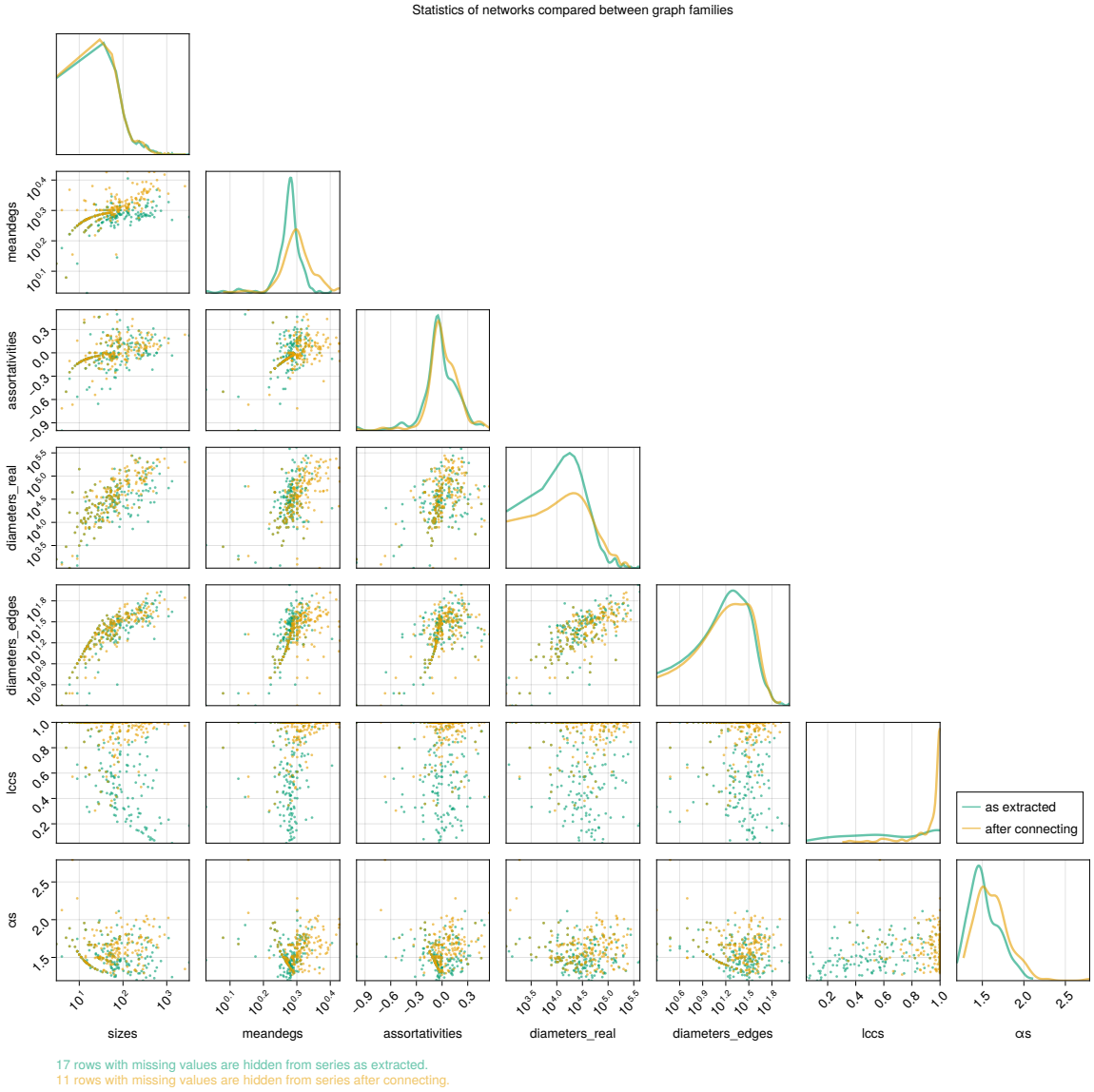



Figure D.2: Pairwise comparisons of the main graph metrics as named in section 3.3

| variable | mean | min | median | max | nmissing |
|---|---|---|---|---|---|
| size | 122.8 | 3.0 | 54.0 | 2608.0 | 0 |
| mean degree | 2.024 | 1.2 | 2.0 | 2.683 | 0 |
| assortativities | 0.00566591 | -1.0 | -0.00813008 | 0.552941 | 0 |
| diameter [m] | 57132.6 | 1028.13 | 37181.1 | 2.75689e5 | 0 |
| diameter [#edges] | 27.1854 | 2.0 | 26.0 | 83.0 | 0 |
| relative size LCC | 0.914744 | 0.304348 | 1.0 | 1.0 | 0 |
| $\alpha$ | 1.61813 | 1.2724 | 1.59582 | 2.79234 | 11 |

Table D.2: Data distribution in the graphs after a node joining pass

# E | Bibliography


[1] citylines.co. https://www.citylines.co/. [Accessed 13-June-2024].

[2] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017. doi: 10.1137/141000671. URL https://epubs.siam.org/doi/10.1137/141000671.

[3] Gábor Csárdi and Tamás Nepusz. The igraph software package for complex network research. *InterJournal, Complex Systems*, page 1695, 2006.

[4] Simon Danisch and Julius Krumbiegel. Makie.jl: Flexible high-performance data visualization for Julia. *Journal of Open Source Software*, 6(65):3349, 2021. doi: 10.21105/joss.03349. URL https://doi.org/10.21105/joss.03349.

[5] P. Echenique, J. Gómez-Gardeñes, and Y. Moreno. Dynamics of jamming transitions in complex networks. *Europhysics Letters*, 71(2):325, jun 2005. doi: 10.1209/epl/i2005-10080-8. URL https://dx.doi.org/10.1209/epl/i2005-10080-8.

[6] Pablo Echenique, Jesús Gómez-Gardeñes, and Yamir Moreno. Improved routing strategies for internet traffic delivery. *Phys. Rev. E*, 70:056105, Nov 2004. doi: 10.1103/PhysRevE.70.056105. URL https://link.aps.org/doi/10.1103/PhysRevE.70.056105.

[7] James Fairbanks, Mathieu Besançon, Schölly Simon, Júlio Hoffiman, Nick Eubank, and Stefan Karpinski. Juliagraphs/graphs.jl: an optimized graphs package for the julia programming language, 2021. URL https://github.com/JuliaGraphs/Graphs.jl/.

[8] M. Leone, A. Vázquez, A. Vespignani, and R. Zecchina. Ferromagnetic ordering in graphs with arbitrary degree distribution. *The European Physical Journal B - Condensed Matter and Complex Systems*, 28(2):191–197, Jul 2002. ISSN 1434-6036. doi: 10.1140/epjb/e2002-00220-0. URL https://doi.org/10.1140/epjb/e2002-00220-0.

[9] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. http://snap.stanford.edu/data, June 2014.

[10] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, page 177187, New York, NY, USA, 2005. Association for

Computing Machinery. ISBN 159593135X. doi: 10.1145/1081870.1081893. URL https://doi.org/10.1145/1081870.1081893.

[11] Lars Onsager. Crystal Statistics. I. A Two-Dimensional Model with an Order-Disorder Transition. *Physical Review*, 65(3-4):117–149, February 1944. doi: 10.1103/PhysRev.65.117.