

POLITECHNIKA WROCŁAWSKA
WYDZIAŁ INFORMATYKI I ZARZĄDZANIA

KIERUNEK: INFORMATYKA

PROJEKT
Teoria i inżynieria ruchu teleinformatycznego

Analiza sieci systemów autonomicznych przy
użyciu algorytmów grafowych

AUTORZY:

Bartosz Cieśla, Bartosz Janusz, Bartosz Kardas

WROCŁAW, 2017

Spis treści

1. Wstęp	5
1.1. Cel projektu	5
1.2. Systemy autonomiczne	5
1.3. Dane sieci	5
1.4. Wstępne przygotowanie danych	5
2. Algorytmy	7
2.1. Centralność w grafach	7
2.2. Betweenness Centrality	7
2.2.1. Algorytm wyznaczania	7
2.3. Closeness Centrality	8
2.4. Eigenvector Centrality - Pagerank	9
3. Opis rozwiązania	11
3.1. Metodologia	11
3.2. Ogólne charakterystyki grafów	11
3.2.1. Odrzucone rzuty - outliers	11
3.2.2. Podstawowe miary	12
3.3. Miary dla wierzchołków	14
3.4. Betweenness	17
3.5. Closeness	20
3.6. Pagerank	24
4. Wizualizacja sieci	29
4.1. Graph-tool draw	29

Spis rysunków

1.1. Wykres liczby wierzchołków dla poszczególnych instancji z zaznaczonymi obserwacjami odstającymi	6
2.1. Działanie Betweenness Centrality na przykładowym grafie	8
2.2. Działanie Closeness Centrality na przykładowym grafie	9
2.3. Działanie Eigenvector Centrality na przykładowym grafie	10
3.1. Wykres wszystkich rzutów danych z sieci	11
3.2. Ilość krawędzi w sieci	12
3.3. Ilość wierzchołków w sieci	12
3.4. Gęstość grafu	13
3.5. Stopnie wierzchołków	14
3.6. Maksymalne stopnie wierzchołków	14
3.7. Minimalne stopnie wierzchołków	15
3.8. Średni stopień wierzchołka w grafie	15
3.9. Mediana stopni wierzchołka	16
3.10. Percentyle dla stopni wierzchołka	16
3.11. Działanie algorytmu betweenness na przykładowym grafie	17
3.12. Wyniki algorytmu betweenness	18
3.13. Maksymalne wartości betweenness	18
3.14. Minimalne wartości betweenness	19
3.15. Średnie wartości betweenness	19
3.16. Mediana wartości betweenness	20
3.17. Działanie algorytmu closeness na przykładowym grafie	21
3.18. Wyniki algorytmu closeness	22
3.19. Maksymalne wartości closeness	22
3.20. Minimalne wartości closeness	23
3.21. Średnie wartości closeness	23
3.22. Mediana wartości closeness	24
3.23. Działanie algorytmu pagerank na przykładowym grafie	25
3.24. Wyniki algorytmu pagerank	26
3.25. Maksymalne wartości pagerank	27
3.26. Minimalne wartości pagerank	27
3.27. Średnie wartości pagerank	28
3.28. Mediana wartości pagerank	28
4.1. Użyta mapa koloru	30
4.2. Graf z początku zebranego zestawu danych dla algorytmu betweenness	31
4.3. Graf z końca zebranego zestawu danych dla algorytmu betweenness	32
4.4. Graf z początku zebranego zestawu danych dla algorytmu closeness	33
4.5. Graf z końca zebranego zestawu danych dla algorytmu closeness	34

4.6. Graf z początku zebranego zestawu danych dla algorytmu pagerank	35
4.7. Graf z końca zebranego zestawu danych dla algorytmu pagerank	36

Spis tabel

Rozdział 1

Wstęp

1.1. Cel projektu

Celem projektu jest ukazanie oraz dokonanie pomiarów zmienności rzeczywistych sieci systemów autonomicznych. Sieci te są reprezentowane za pomocą grafów nieskierowanych. Do ich analizy zastosowano algorytmy typowo wykorzystywane w przypadku badania sieci społecznościowych, tak zwane miary centralności grafu. Oprócz nich szczególną uwagę zwrócono na stopień poszczególnych węzłów.

1.2. Systemy autonomiczne

Oryginalnie projekt miał opierać się na analizie grafów, gdzie każdy wierzchołek odpowiadał pojedynczemu urządzeniu - routerowi. Ze względu jednak na niską dostępność takich zestawów danych oraz ich gigantyczny rozmiar (niemożliwy do przetworzenia na dostępnym sprzęcie), zdecydowano się na badanie większych jednostek. System autonomiczny to sieć lub grupa sieci opartych na protokole IP pod wspólną administracyjną kontrolą, w której utrzymywany jest spójny schemat trasowania. Struktury te są podstawową jednostką budulcową Internetu na poziomie domen. Większość dostępnych topologii sieci internetowej opiera się na ich strukturze, dlatego uznano, że i w opisywanym projekcie znajdą zastosowanie.

1.3. Dane sieci

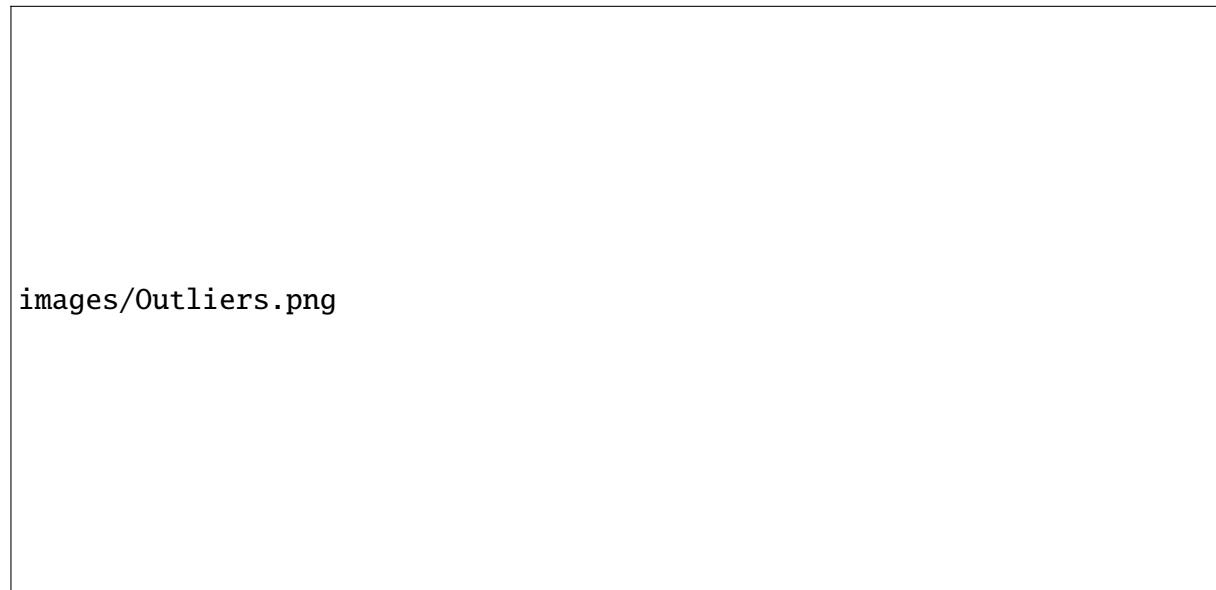
Znalezienie odpowiednich danych wejściowych wymagało trochę czasu. Z racji niedeterministycznej struktury badanych sieci, niemożliwe było wygenerowanie danych testowych. Konieczne okazało się wykorzystanie danych pochodzących z badań istniejącej sieci internetu. Zdecydowano się wykorzystać zestaw pochodzący z projektu University of Oregon Route Views. Zawiera on 733 instancje grafu, tworzonego poprzez codzienne badania struktury sieci na poziomie systemów autonomicznych. Najstarsza instancja pochodzi z 8 listopada 1997 roku, najmłodsza z 2 stycznia 2000 roku. Dane były zbierane za pomocą techniki multi-hop z wykorzystaniem protokołu BGP.

1.4. Wstępne przygotowanie danych

Dane zawarte w zestawie mają postać listy krawędzi. Mimo iż są to grafy nieskierowane, wszystkie krawędzie występują w postaci zdublowanej oraz część węzłów posiada pętle. Warto tu nadmienić, że taki zapis grafu w pewnym kontekście na pewno miał określone znaczenie. Jednakże

z punktu widzenia zamierzonych badań oba zjawiska były niepożądane, w związku z czym napisano własną funkcję wczytującą graf do programu. Jej zadaniem było odfiltrowanie krawędzi wielokrotnych oraz pętli.

Drugim problemem na który natknęto się po wczytaniu grafów, były duże błędy pomiarowe. Po stworzeniu wykresu liczby wierzchołków dla wszystkich instancji zauważono, że istotna ich część znacząco odstaje od widocznego trendu (w kierunku zera). Wykres liczby krawędzi ukazał spadki w tych samych miejscach, co utwierdziło autorów o błędach w pomiarach. Podejrzewają oni, że było to spowodowane przerwaniem procesu zbierania danych o strukturze sieci. Zaskutkowało to mniejszymi instancjami grafów dla niektórych dni. W celu ujednolicenia danych zastosowano funkcję usuwającą obserwacje odstające zawartą w oprogramowaniu Matlab *isoutlier()*. Efekt jej działania można zobaczyć poniżej.



Rys. 1.1: Wykres liczby wierzchołków dla poszczególnych instancji z zaznaczonymi obserwacjami odstającymi

Rozdział 2

Algorytmy

2.1. Centralność w grafach

W teorii grafów wskaźniki centralności informują o najbardziej znaczących wierzchołkach grafu. Ich przykładowymi zastosowaniami mogą być: znalezienie lidera, przywódcy spośród danej grupy osób, ustalenie kluczowego elementu infrastruktury sieciowej lub miejskiej bądź znalezienie osobnika o największym potencjale do roznoszenia choroby. Istnieje wiele odmiennych wskaźników centralności. Zrealizowany projekt implementuje trzy z nich: Closeness Centrality, Betweenness Centrality oraz Pagerank (jedna z odmian Eigenvector Centrality)

2.2. Betweenness Centrality

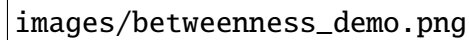
Określa kluczowość wierzchołka w zakresie komunikacji - przechodność, pośredniczenie. Czyli w jakim stopniu dany wierzchołek jest spoiwem dla danej sieci. Jest to miara o bardzo wielkiej wartości, gdyż dzięki niej można znaleźć punkty krytycznej sieci bądź grafu.

2.2.1. Algorytm wyznaczania

1. Wyznaczyć ilość najkrótszych ścieżek między wierzchołkami u i v (d_{uv})
2. Wyznaczyć ilość najkrótszych ścieżek między wierzchołkami u i v , które przechodzą przez wierzchołek w ($d_{uv}(w)$)
3. Suma stosunków oznacza stopień centralności wierzchołka w

$$c_b(w) = \sum_{u \neq v \neq w} \frac{d_{uv}(w)}{d_{uv}}$$

Przykład



images/betweenness_demo.png

Rys. 2.1: Działanie Betweenness Centrality na przykładowym grafie

2.3. Closeness Centrality

Jest to stopień bliskości. Określa jak blisko (daleko) wierzchołek ma do pozostałych w grafie. Wysoki stopień bliskości świadczy o dobrej własności propagacji informacji w grafie - element ten szybko rozprowadzi daną wiadomość (wirusa itp) po całej sieci.

Algorytm wyznaczania

1. Wyznaczyć odległości pomiędzy wierzchołkiem u a pozostałymi wierzchołkami w grafie v (d_{uv})
2. W zależności od rodzaju grafu zsumować otrzymane odległości:
 1. Dla grafów rzadkich

$$c_c(u) = \frac{1}{\sum d_{uv}}$$

2. Dla grafów silnie połączonych

$$c_c(u) = \sum_{u \neq v} \frac{1}{d_{uv}}$$

Przykład



Rys. 2.2: Działanie Closeness Centrality na przykładowym grafie

2.4. Eigenvector Centrality - Pagerank

Określa wpływ, oddziaływanie wierzchołka na pozostałe w grafie. Wykorzystuje nie tylko ilość połączeń danego wierzchołka z innymi, a przede wszystkim ich jakość. Wartości przypisane do każdego z wierzchołków bazują na koncepcji w której wysoko ocenione wierzchołki bardziej wpływają na ostateczną ocenę połączonego wierzchołka, niż te, których ocena jest niska. Jedną z odmian Eigenvector Centrality jest algorytm PageRank. Poniżej przedstawiono uproszczony algorytm jego działania.

Algorytm wyznaczania

1. Wyznaczyć ilość wierzchołków w grafie (N)
2. Wyznaczyć stopień każdego z wierzchołków ($l(u)$)
3. Zainicjować wartości początkowe dla każdego wierzchołka wartością początkową ($c_e(u) = 1$)
4. Określić współczynnik tłumienia, zwykle wynosi on około 0.85 ($d = 0.85$)
5. Obliczyć nową wartość PageRank każdego wierzchołka

$$c_e(u) = \frac{1-d}{N} + d \sum_{v \in B_u} \frac{c_e(v)}{l(v)}$$

B_u oznacza zbiór wszystkich wierzchołków, które odnoszą się do wierzchołka u

Przykład



Rys. 2.3: Działanie Eigenvector Centrality na przykładowym grafie

Rozdział 3

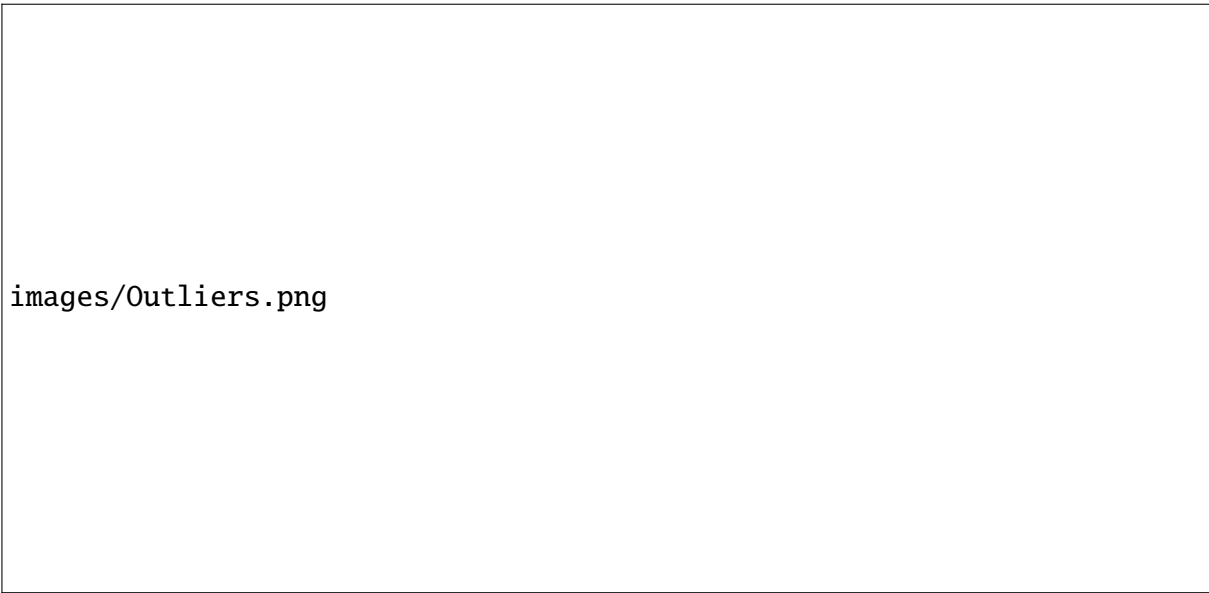
Opis rozwiązania

3.1. Metodologia

Badanie zmienności sieci na przestrzeni czasu podzielono na kilka etapów. W pierwszym z nich usunięto grafy wyraźnie odstające od reszty pod względem ilości wierzchołków. W kolejnych etapach, tak spreparowane dane poddano kolejnym eksperymentom, które pozwalały na uzyskanie różnorodnych miar. W tym celu użyto biblioteki graph-tool dostępnej w języku Python. Jest to wydajne narzędzie do analizy sieci. Dzięki niemu możliwe stało się wygenerowanie wszystkich opisanych niżej miar strukturalnych. W badaniu uwzględniono miary o szerokim zakresie złożoności, począwszy od ilości wierzchołków, krawędzi, poprzez gęstości grafu, skończywszy na obliczaniu centralności. W związku z dużą złożonością algorytmów obliczających miary centralności, wyniki cząstkowe dla każdego z grafu, przechowano w plikach tymczasowych, w celu dalszej analizy. Załączone wykresy utworzono dzięki pythonowej bibliotece matplotlib.

3.2. Ogólne charakterystyki grafów

3.2.1. Odrzucone zrzuty - outliers

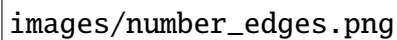


images/Outliers.png

Rys. 3.1: Wykres wszystkich zrzutów danych z sieci

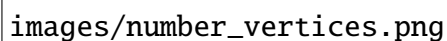
Przy badaniu działania algorytmów grafowych korzystano z serii danych - zrzutów sieci na przestrzeni kilku lat. Zrzuty nie były wykonywane codziennie, dodatkowo niektóre z nich znacząco różnią się wielkością od pozostałych. Źródło danych nie podaje przyczyny, ale takie grafy są nielogiczne biorąc pod uwagę ciągłość sieci. Zastosowano zatem wbudowany w MATLAB-a algorytm do usunięcia tychże grafów, które zostały na wykresie oznaczone czerwonymi krzyżykami.

3.2.2. Podstawowe miary



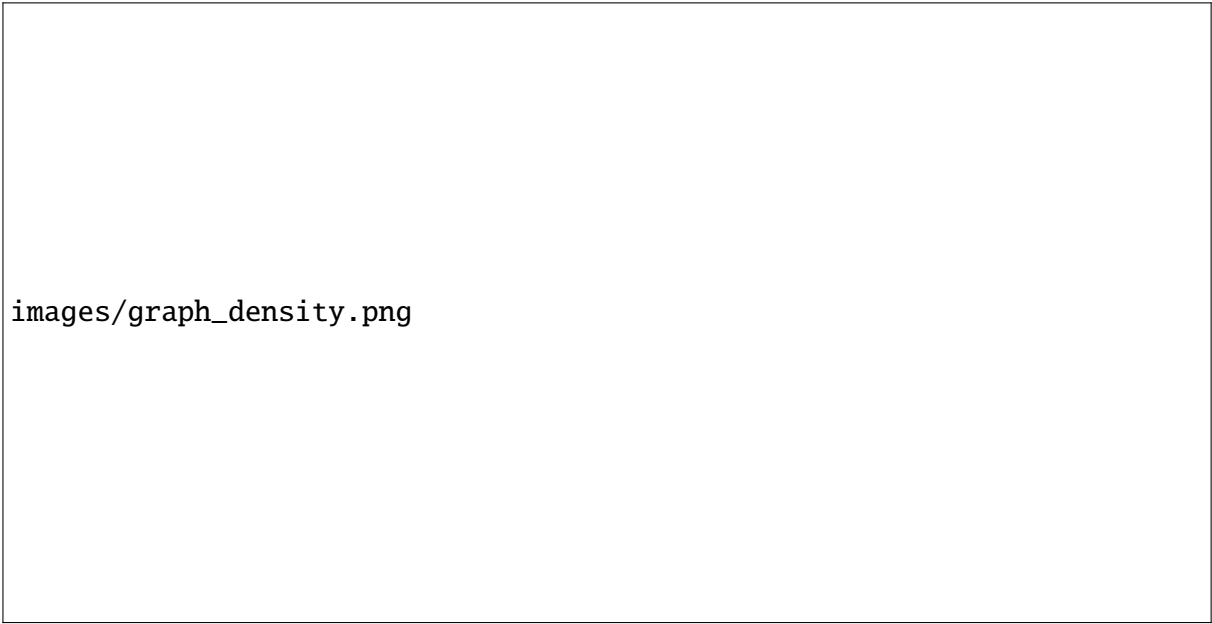
images/number_edges.png

Rys. 3.2: Ilość krawędzi w sieci



images/number_vertices.png

Rys. 3.3: Ilość wierzchołków w sieci



images/graph_density.png

Rys. 3.4: Gęstość grafu

Malejąca gęstość grafu wynika wprost z tego, że stosunek ilości wierzchołków do krawędzi jest stały w miarę upływu czasu, mimo tego że obie wartości rosną. Gęstość grafu liczymy ze wzoru:

$$d = \frac{2m}{n(n-1)} \quad (3.1)$$

d - gęstość grafu, m - ilość krawędzi, n - ilość wierzchołków

Da się łatwo zauważyć, że jeśli stosunek $\frac{m}{n} = \text{const}$, a w tym przypadku $\frac{m}{n} \approx 2 \Rightarrow m \approx 2n$ to wzór 3.1 można zapisać następująco:

$$d \approx \frac{4n}{n(n-1)} \equiv d \approx \frac{4}{n-1} \quad (3.2)$$

Jak widać po przekształceniu uzyskujemy hiperbolę, którą dla dużych n na wąskim przedziale $[3000, 6500]$ można przybliżyć malejącą funkcją liniową.

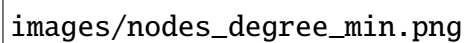
3.3. Miary dla wierzchołków

images/nodes_degrees.png

Rys. 3.5: Stopnie wierzchołków

images/nodes_degree_max.png

Rys. 3.6: Maksymalne stopnie wierzchołków



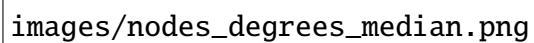
images/nodes_degree_min.png

Rys. 3.7: Minimalne stopnie wierzchołków



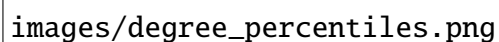
images/nodes_degrees_mean.png

Rys. 3.8: Średni stopień wierzchołka w grafie



images/nodes_degrees_median.png

Rys. 3.9: Mediana stopni wierzchołka

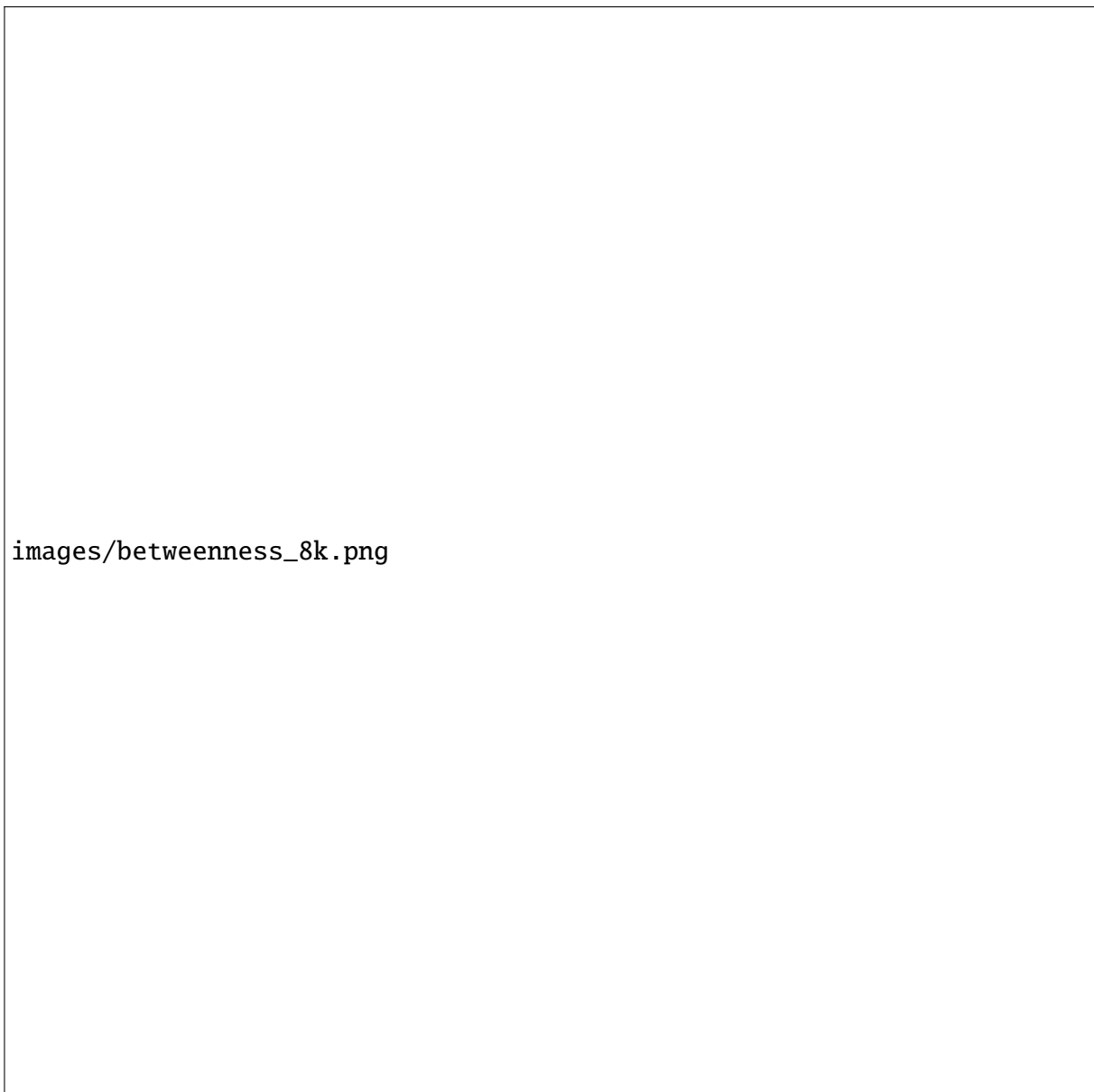


images/degree_percentiles.png

Rys. 3.10: Percentyle dla stopni wierzchołka

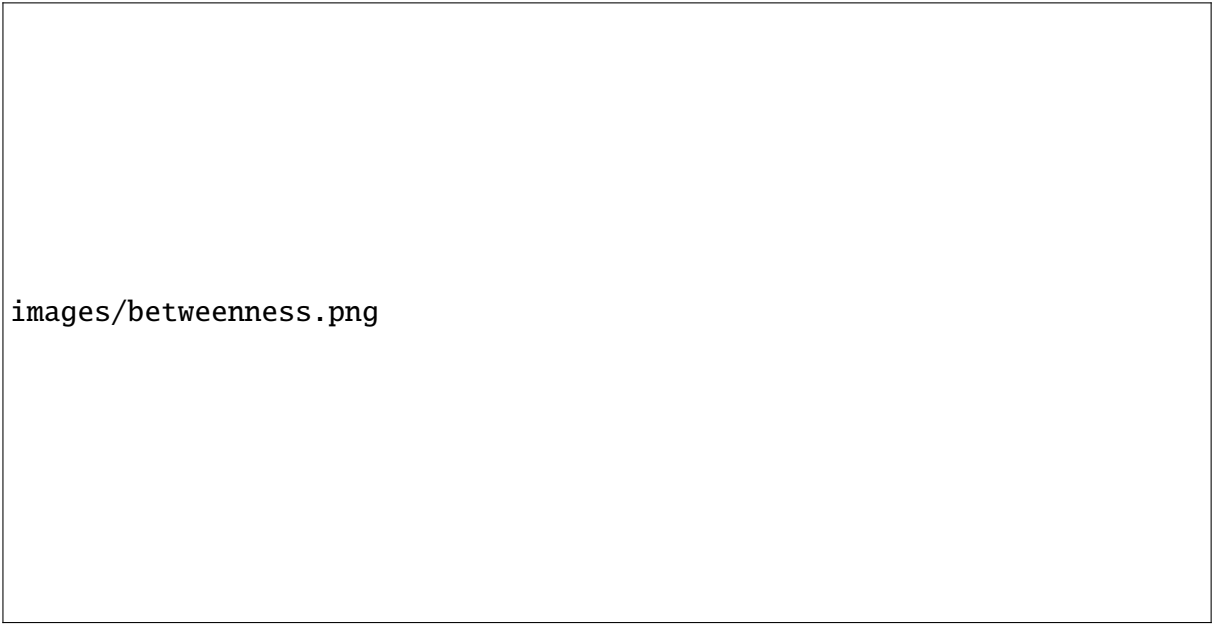
Z powyższych wykresów wynika, że graf jest duży i ma relatywnie niską gęstość połączeń. Przez cały okres duża ilość wierzchołków (95%) ma niski stopień, co przekłada się na niską średnią oraz medianę i może wpływać na wyniki algorytmów.

3.4. Betweenness



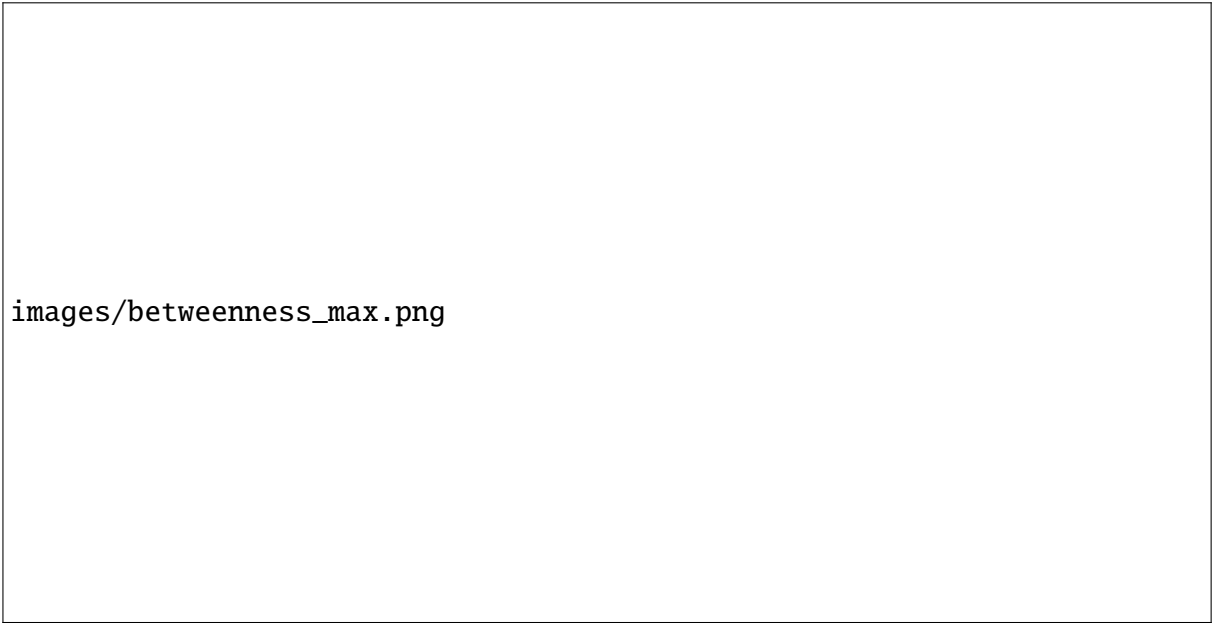
Rys. 3.11: Działanie algorytmu betweenness na przykładowym grafie

Powyżej przedstawione jest działanie algorytmu betweenness na wybranym grafie. Kolor oraz wielkość wierzchołka zależy od wyniku algorytmu. Duże wierzchołki są „mostami” które spinają sieć - algorytm zwrócił dla nich największą wartość. Przekłada się to bezpośrednio na to, że dany wierzchołek - fragment sieci może być w dużym stopniu obciążony ruchem.



images/betweenness.png

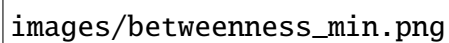
Rys. 3.12: Wyniki algorytmu betweenness



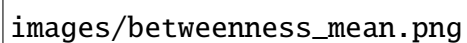
images/betweenness_max.png

Rys. 3.13: Maksymalne wartości betweenness

Niestałe zmiany maksymalnej wartości betweenness świadczą o tym, że struktura komunikacji w sieci zmieniała się dość znacznie. Im mniejsza jest różnica pomiędzy wartościami maksymalną i minimalną tym sieć jest bardziej równomierna w dostępie. Znaczy to że podczas dużego obciążenia jest mniejsze ryzyko, że większość ruchu będzie przechodziła przez jeden bądź kilka węzłów. Zmniejsza to prawdopodobieństwo awarii lub braku odpowiedzi.

The image area is mostly blank, with the text 'images/betweenness_min.png' located in the lower-left corner. This likely represents a visualization of minimal betweenness centrality values for a graph.

Rys. 3.14: Minimalne wartości betweenness

The image area is mostly blank, with the text 'images/betweenness_mean.png' located in the lower-left corner. This likely represents a visualization of average betweenness centrality values for a graph.

Rys. 3.15: Średnie wartości betweenness

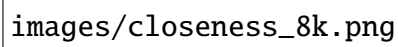
Średnia wartość algorytmu spada w czasie. Jest to spowodowane głównie tym, że ilość wierzchołków w grafie rośnie a ich średni stopień maleje.



images/betweenness_median.png

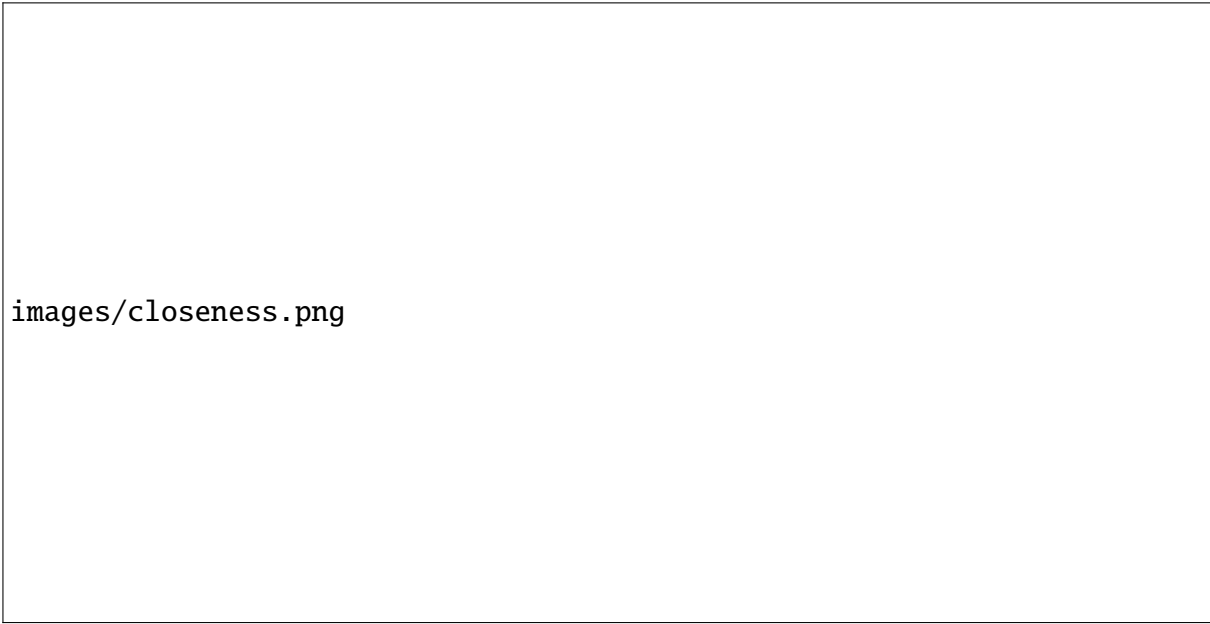
Rys. 3.16: Mediana wartości betweenness

3.5. Closeness



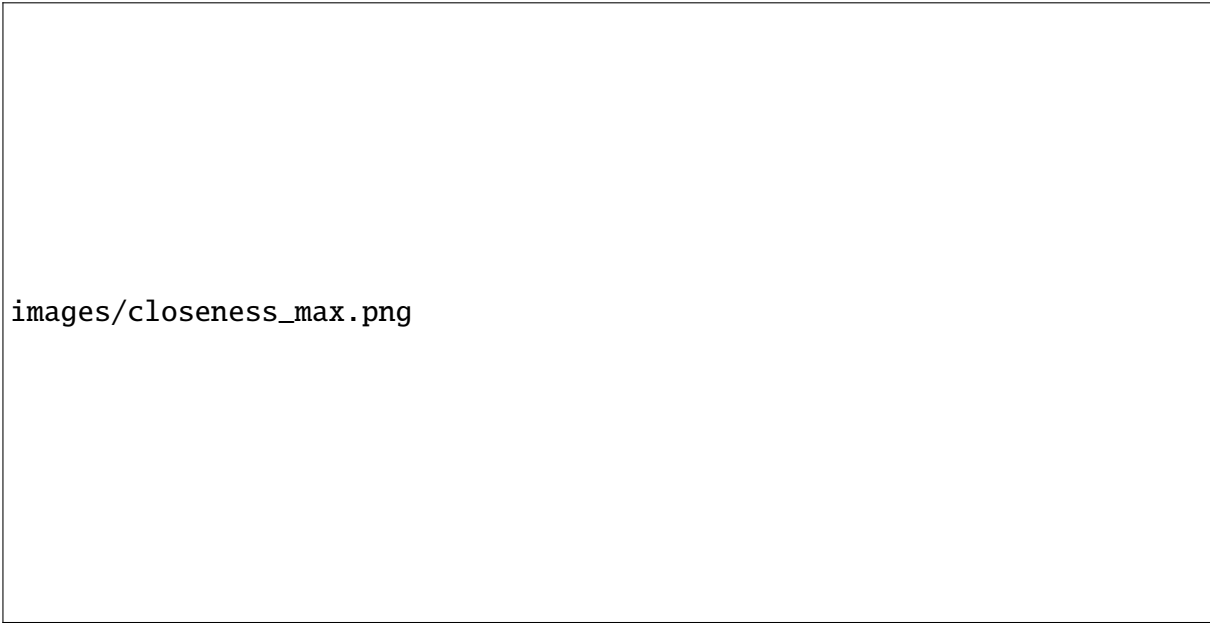
Rys. 3.17: Działanie algorytmu closeness na przykładowym grafie

Powyżej przedstawione jest działanie algorytmu closeness na wybranym grafie. Kolor oraz wielkość wierzchołka zależy od wyniku algorytmu. Duże i jasne wierzchołki są wierzchołkami, z których najłatwiej dostać się do innych - wymagają średnio najmniej przeskoków by dostać się do innego wierzchołka. Przekłada się to bezpośrednio na to, że dany wierzchołek - fragment sieci może być często wybierany jako pośrednik.



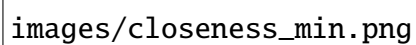
images/closeness.png

Rys. 3.18: Wyniki algorytmu closeness

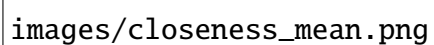


images/closeness_max.png

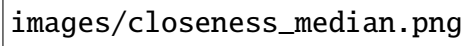
Rys. 3.19: Maksymalne wartości closeness

A rectangular box containing the text "images/closeness_min.png", which serves as a placeholder for a visualization of minimal closeness values.

Rys. 3.20: Minimalne wartości closeness

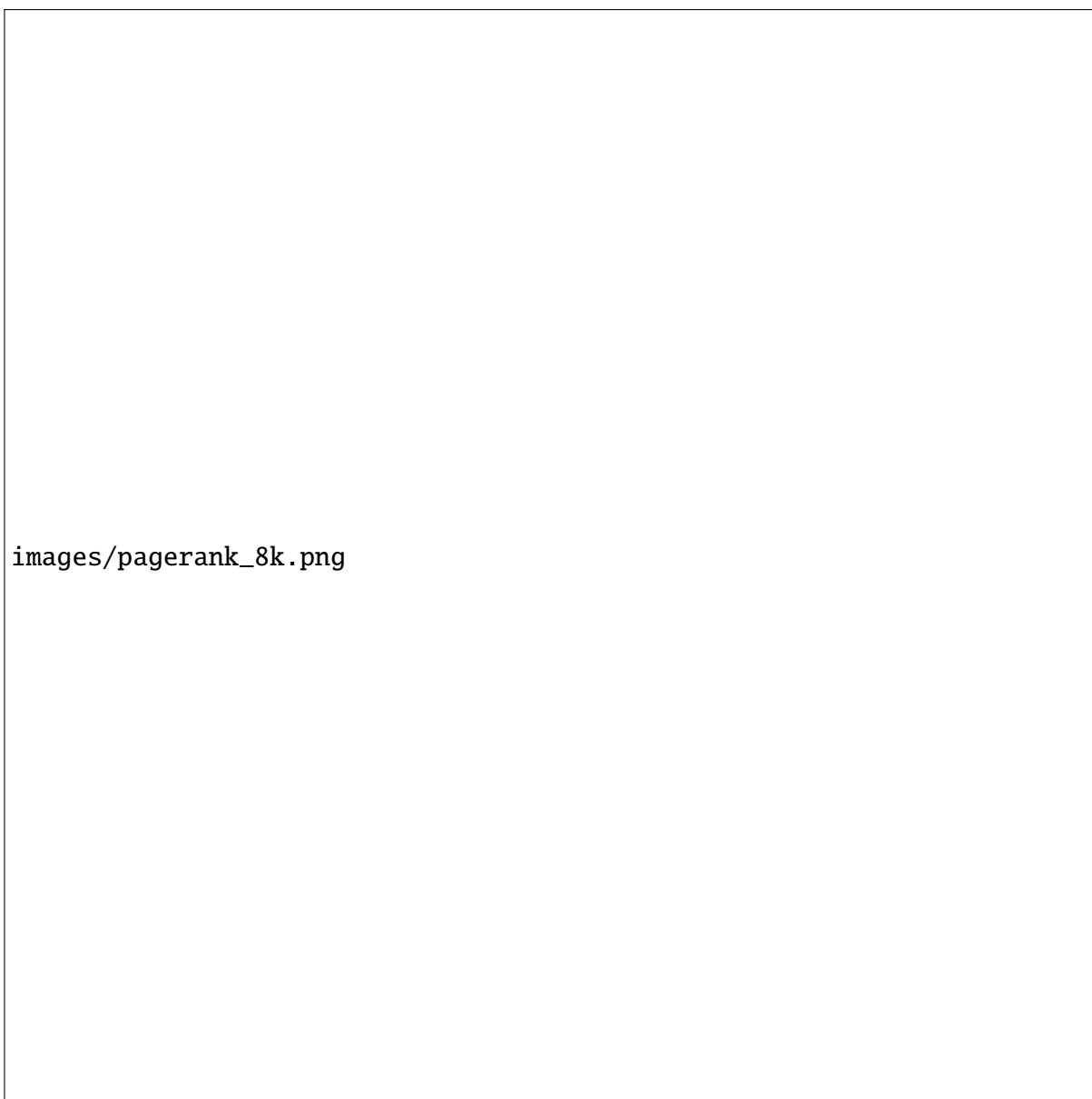
A rectangular box containing the text "images/closeness_mean.png", which serves as a placeholder for a visualization of average closeness values.

Rys. 3.21: Średnie wartości closeness

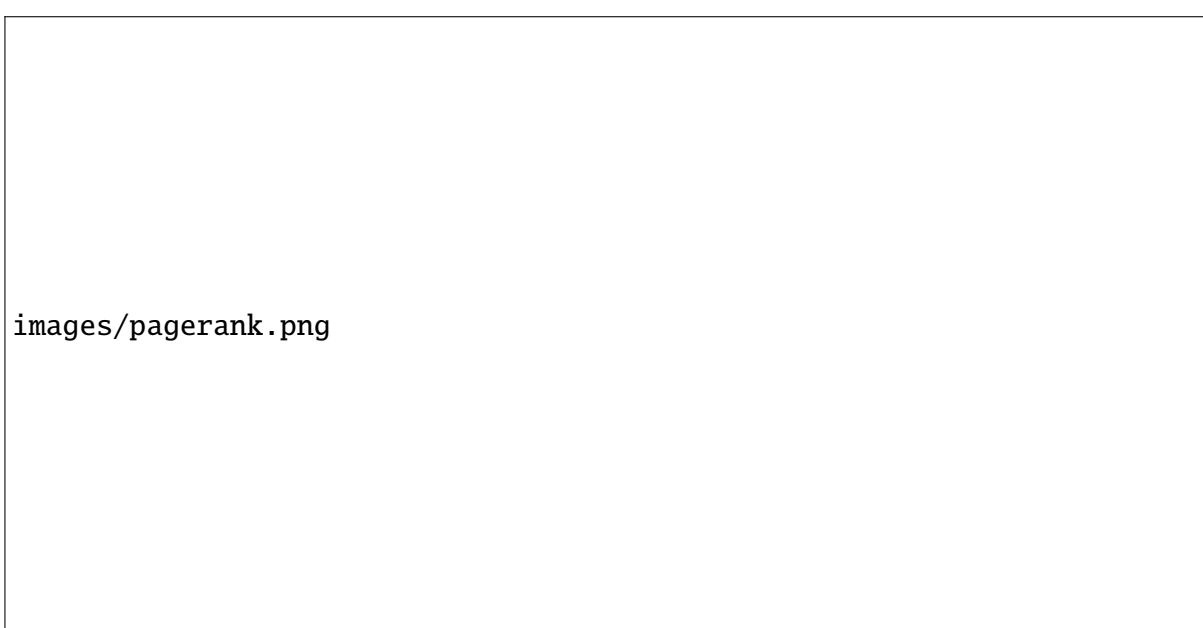
The image area is mostly blank, with the text 'images/closeness_median.png' located on the left side. This appears to be a placeholder for a visualization of closeness median values.

Rys. 3.22: Mediana wartości closeness

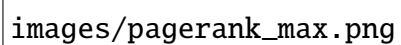
3.6. Pagerank



Rys. 3.23: Działanie algorytmu pagerank na przykładowym grafie

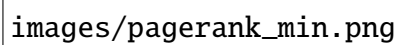


Rys. 3.24: Wyniki algorytmu pagerank



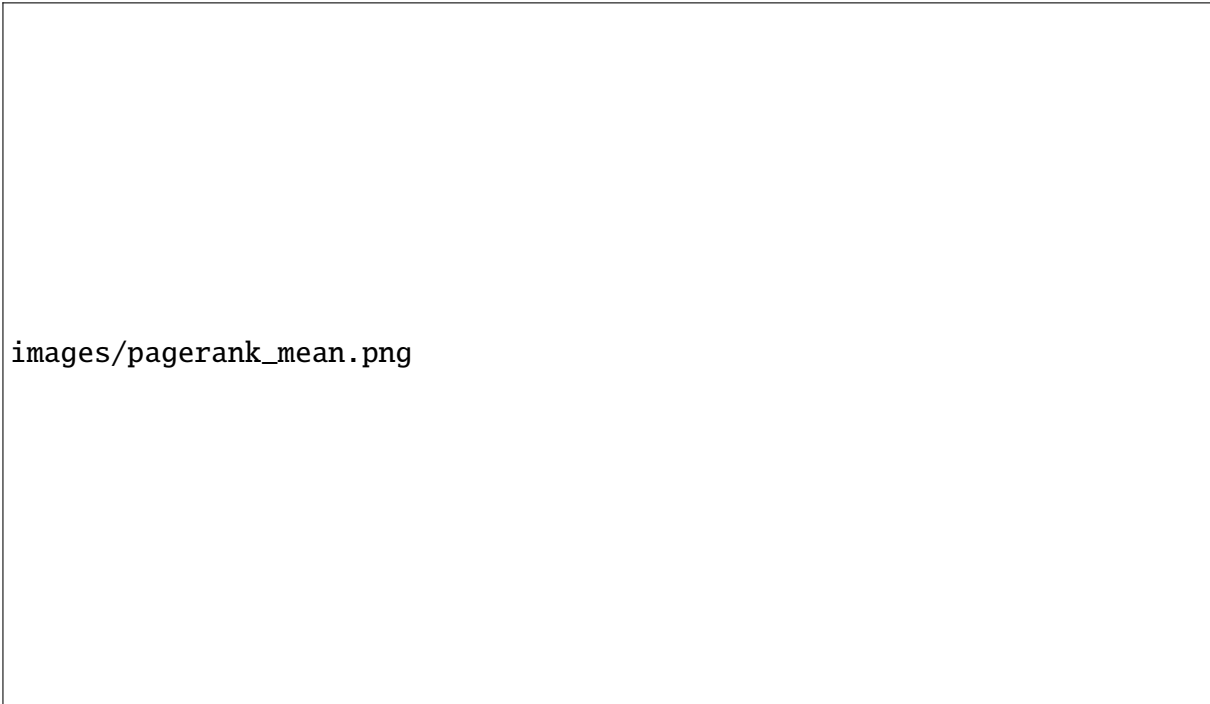
images/pagerank_max.png

Rys. 3.25: Maksymalne wartości pagerank



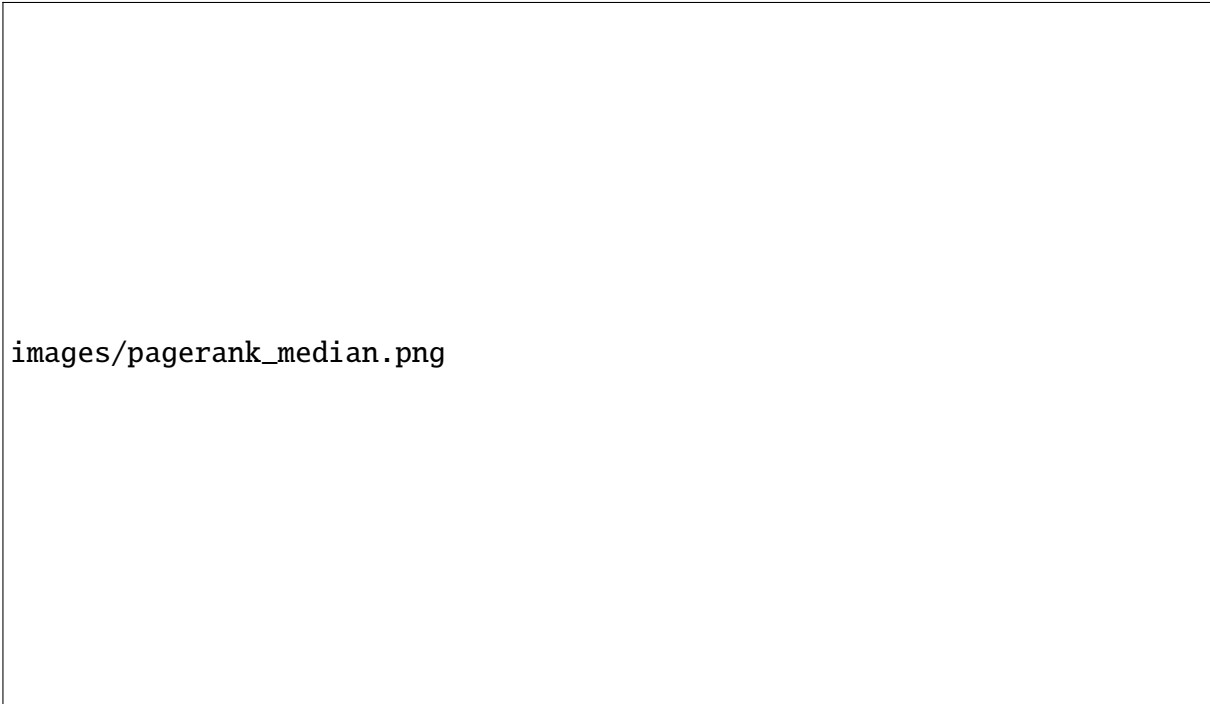
images/pagerank_min.png

Rys. 3.26: Minimalne wartości pagerank



images/pagerank_mean.png

Rys. 3.27: Średnie wartości pagerank



images/pagerank_median.png

Rys. 3.28: Mediana wartości pagerank

Pagerank jest miarą tego jak dużo innych wierzchołków odwołuje się do badanego. Nie jest to równoznaczne ze stopniem wierzchołka ale ma on duży wpływ na wynik. W badanym okresie gęstość sieci malała, co daje się zauważyć również w spadku średniego wyniku algorytmu. Mimo tego maksymalna wartość pagerank, mimo drobnych fluktuacji, zachowuje podobny poziom. Wskazuje to na to, że nowe wierzchołki były w dużej mierze dodawane na obrzeżach a nie w środku sieci.

Rozdział 4


Wizualizacja sieci

Na potrzeby projektu intensywnie poszukiwano metod skutecznego wizualizowania badanych sieci. Przetestowaliśmy większość dostępnych narzędzi, w tym bardzo popularne oprogramowanie Gephi oraz bibliotekę NetworkX. Niestety, ze względu na znaczną wielkość grafów, żadne z łatwo dostępnych narzędzi nie umożliwiała stworzenia ich interaktywnej animacji. W związku z tym zdecydowano się na dwie, uzupełniające się metody.

4.1. Graph-tool draw

Pierwszą z nich jest funkcja rysująca z biblioteki Graph-tool. Posiada ona szerokie możliwości odnośnie formy wyjściowej rysowanych grafów. Okno interaktywne nie zapewniało odpowiedniej płynności animacji, co w praktyce uniemożliwiało korzystanie z niego. Zadowalająca z kolei okazała się możliwość generacji statycznych obrazów PNG w dużej rozdzielczości. Maksymalna rozdzielczość nie była ograniczona przez bibliotekę, z sukcesem generowano obrazy 20 000 x 20 000 pixeli. Jednakże ilość pamięci operacyjnej na dostępnym sprzęcie testowym narzucała ograniczenie na rozdzielczość i w efekcie zdecydowano się na 10 000 x 10 000. Zapewnia to odpowiednią ostrość nawet przy dużym powiększeniu.

Funkcja udostępniała kolorowanie grafu i zmianę rozmiarów jego wierzchołków zgodnie z wyliczoną miarą centralności. Kolorowanie odbywało się według skali *gnuplot*, im wartość miary była większa, tym kolor był cieplejszy (bliżej żółtego).



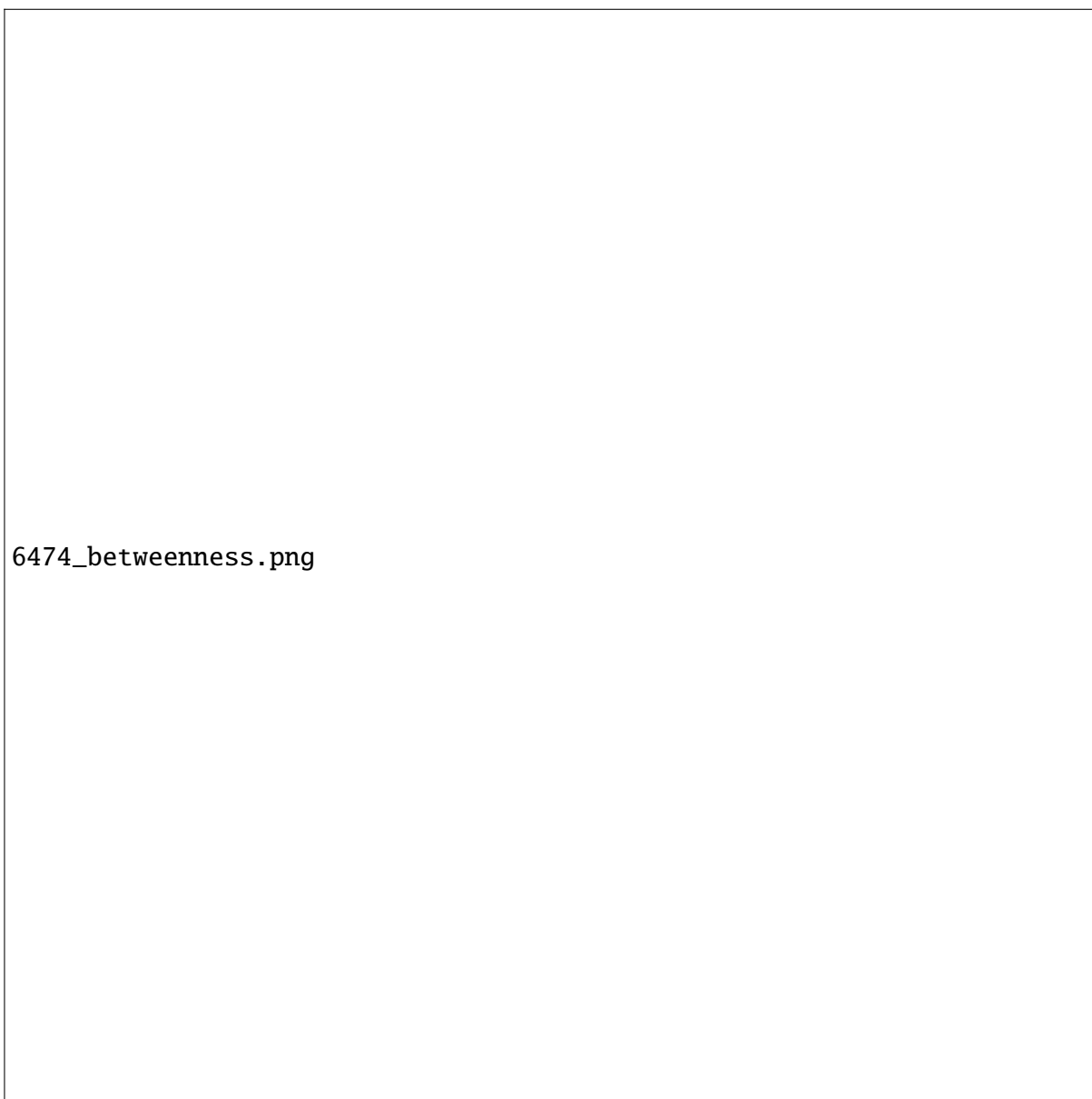
colormaps_reference_05.png

Rys. 4.1: Użyta mapa koloru

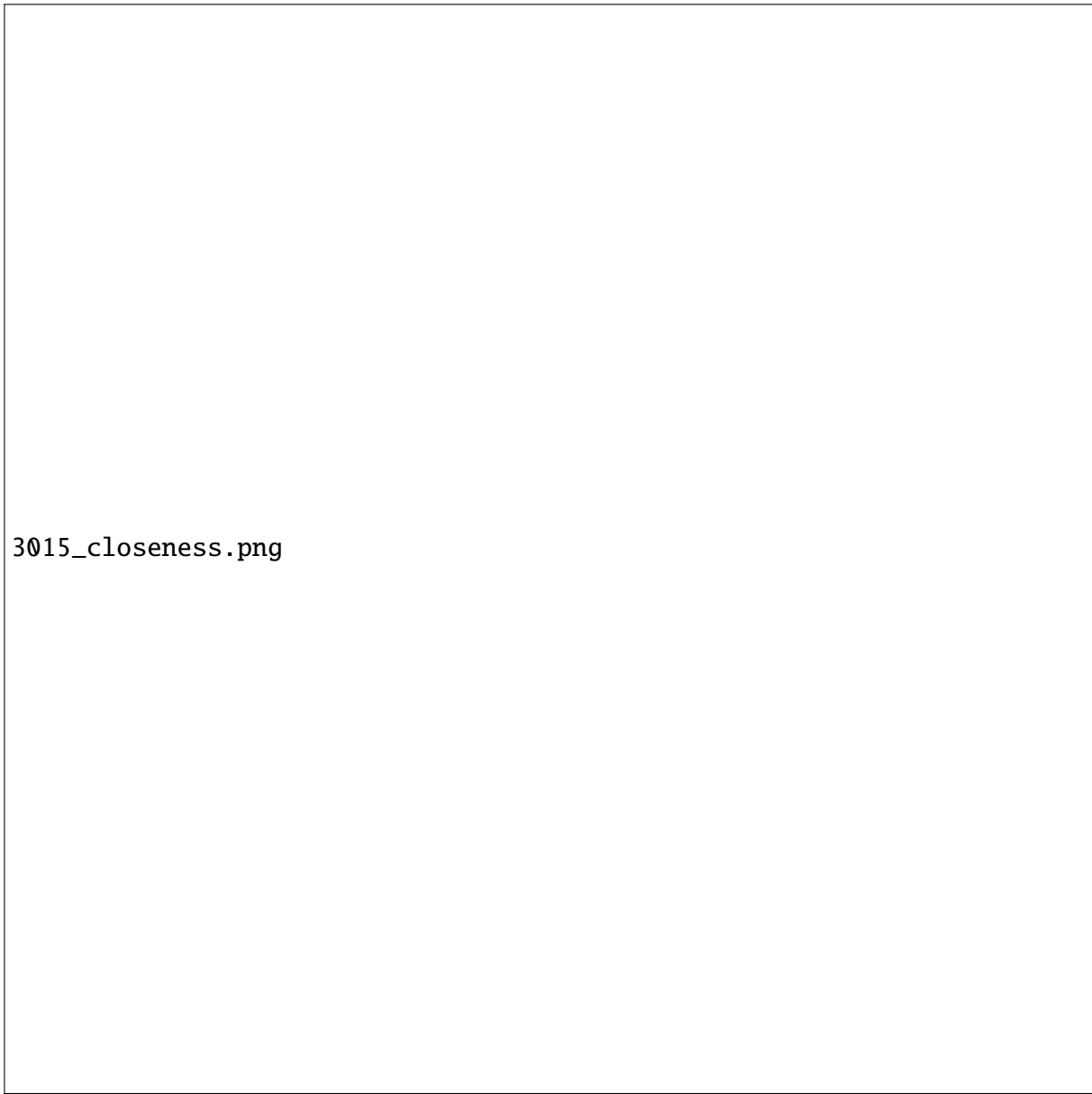
Poniżej znajdują się wizualizacje dla wszystkich 3 algorytmów dla najstarszego grafu (1997 rok) oraz najmłodszego (2000 rok).



Rys. 4.2: Graf z początku zebranego zestawu danych dla algorytmu betweenness

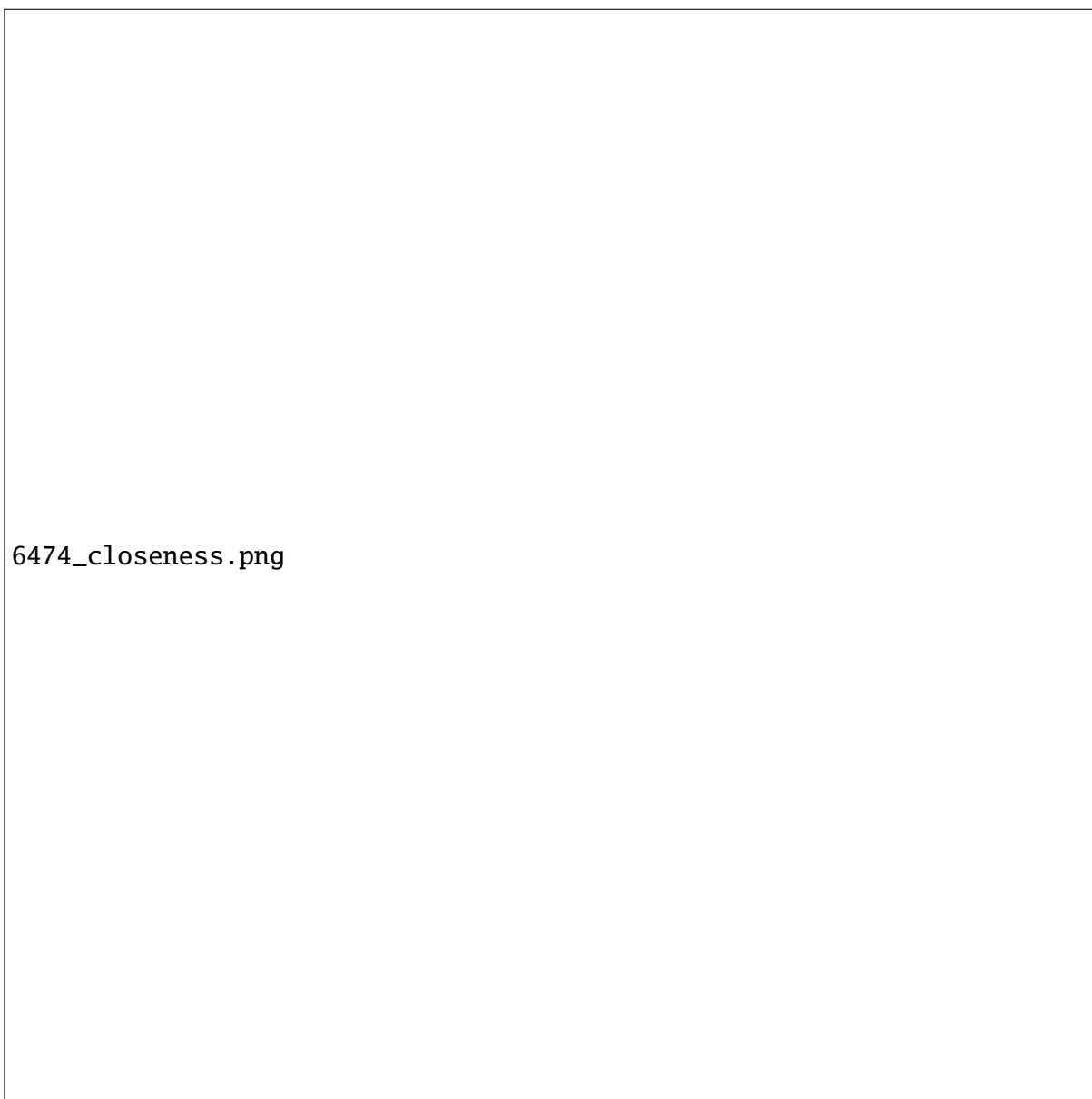


Rys. 4.3: Graf z końca zebranego zestawu danych dla algorytmu betweenness



3015_closeness.png

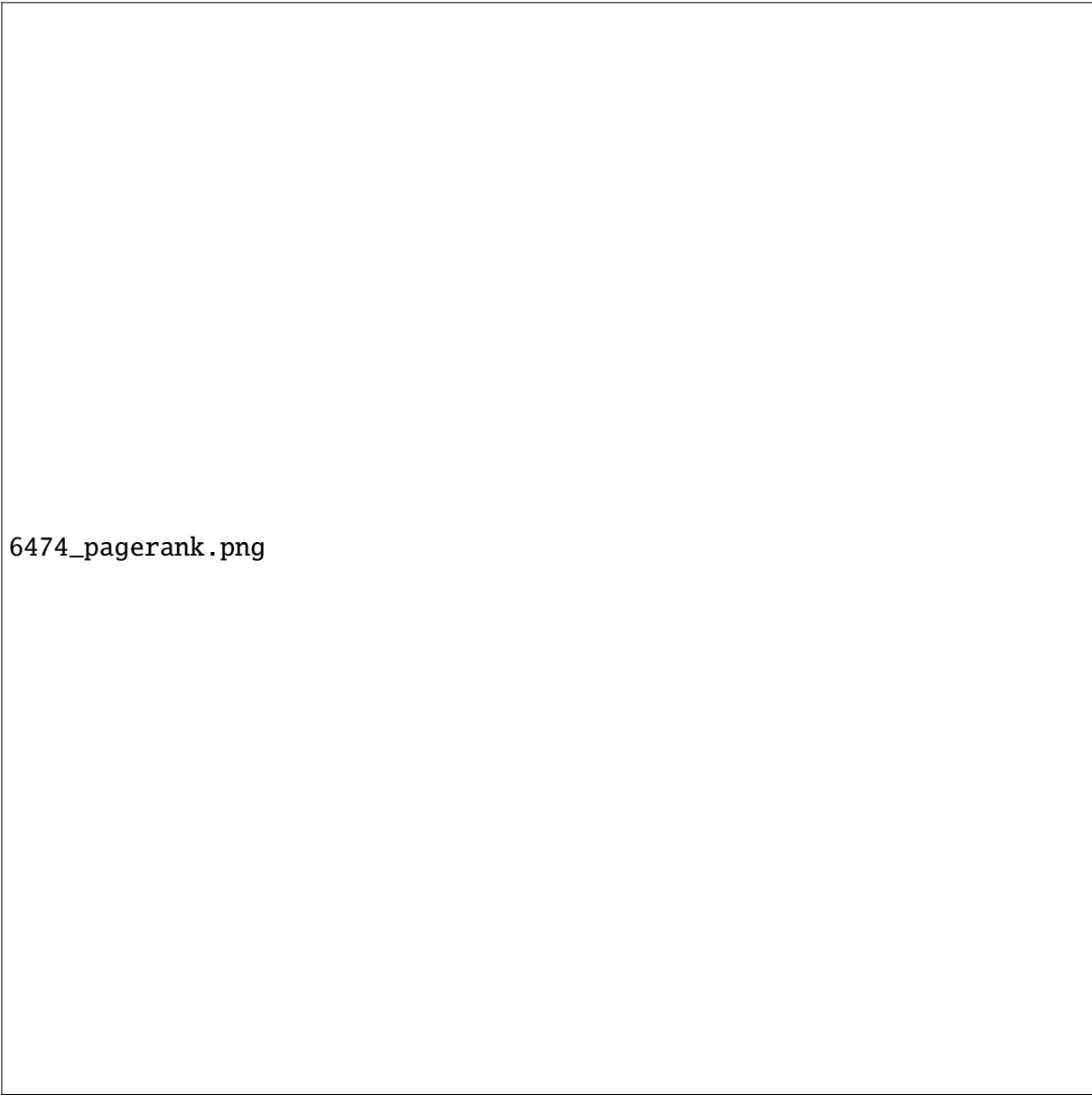
Rys. 4.4: Graf z początku zebranego zestawu danych dla algorytmu closeness



Rys. 4.5: Graf z końca zebranego zestawu danych dla algorytmu closeness



Rys. 4.6: Graf z początku zebranego zestawu danych dla algorytmu pagerank



6474_pagerank.png

Rys. 4.7: Graf z końca zebranego zestawu danych dla algorytmu pagerank

4.2. Walrus

Druga metoda wizualizacji opierała się na oprogramowaniu Walrus, stworzonym w okolicach 2000 roku przez organizację CAIDA, zajmującą się badaniem topologii sieci. Jego rozwój zakończono w 2005 roku, mimo to jest to jedyne narzędzie będące w stanie interaktywnie zwizualizować bardzo duże grafy na stosunkowo słabym sprzęcie i to w przestrzeni trójwymiarowej. Mimo tych niewątpliwych zalet program jest toporny w użytkowaniu i nadaje się głównie do grafów zbliżonych do drzewa rozpinającego. Wymaga on pliku wsadowego w egzotycznym formacie LibSea, konieczne więc było napisanie specjalnego generatora. Do innych ograniczeń zaliczyć można:

- obsługa wyłącznie grafów skierowanych
- obsługa tylko grafów spójnych
- podanie w pliku wsadowym pełnego drzewa rozpinającego
- brak obsługi krawędzi wielokrotnych i cykli w grafie

- brak obsługi pętli w grafie

Powyższe ograniczenia wymusiły dokonania transformacji grafu. Efekty widoczne są poniżej. Podobnie jak w pierwszej metodzie, kolory odpowiadają przeskalowanej wartości miar centralności. Kolorowane są również krawędzie, według zasady: krawędź łącząca dwa węzły przyjmuje kolor wierzchołka o większej przypisanej wartości miary. Zdecydowano się na takie rozwiązanie w celu zwiększenia widoczności wierzchołków. Z tego samego powodu poniższe wizualizacje zawierają tylko drzewo spinające badanych grafów. Ostatnia obejmuje cały graf w celach porównawczych.



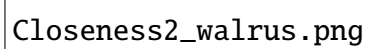
Rys. 4.8: Najstarszy graf, wizualizacja algorytmu betweenness



Rys. 4.9: Najstarszy graf, wizualizacja algorytmu betweenness - ujęcie 2

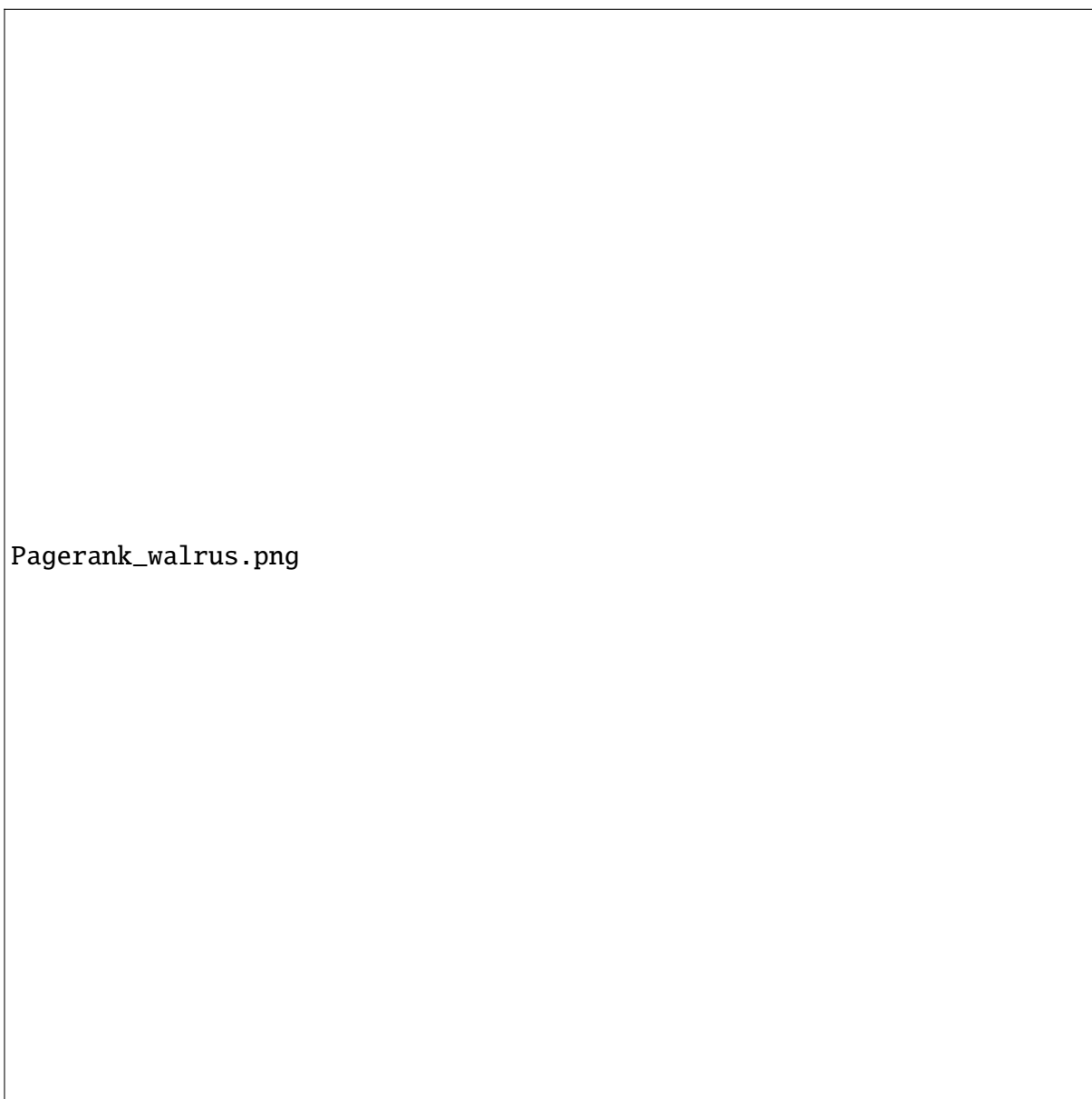


Rys. 4.10: Najstarszy graf, wizualizacja algorytmu closeness



Closeness2_walrus.png

Rys. 4.11: Najstarszy graf, wizualizacja algorytmu closeness - ujęcie 2



Rys. 4.12: Najstarszy graf, wizualizacja algorytmu pagerank



Rys. 4.13: Najstarszy graf, wizualizacja algorytmu pagerank - cały graf

Na powyższej ilustracji doskonale widać ogrom badanego grafu. Tak duża liczba krawędzi powoduje, że z wizualizacji ciężko cokolwiek wyczytać. Stąd rezygnacja z części z nich w pozostałych obrazach.