28.03.2022

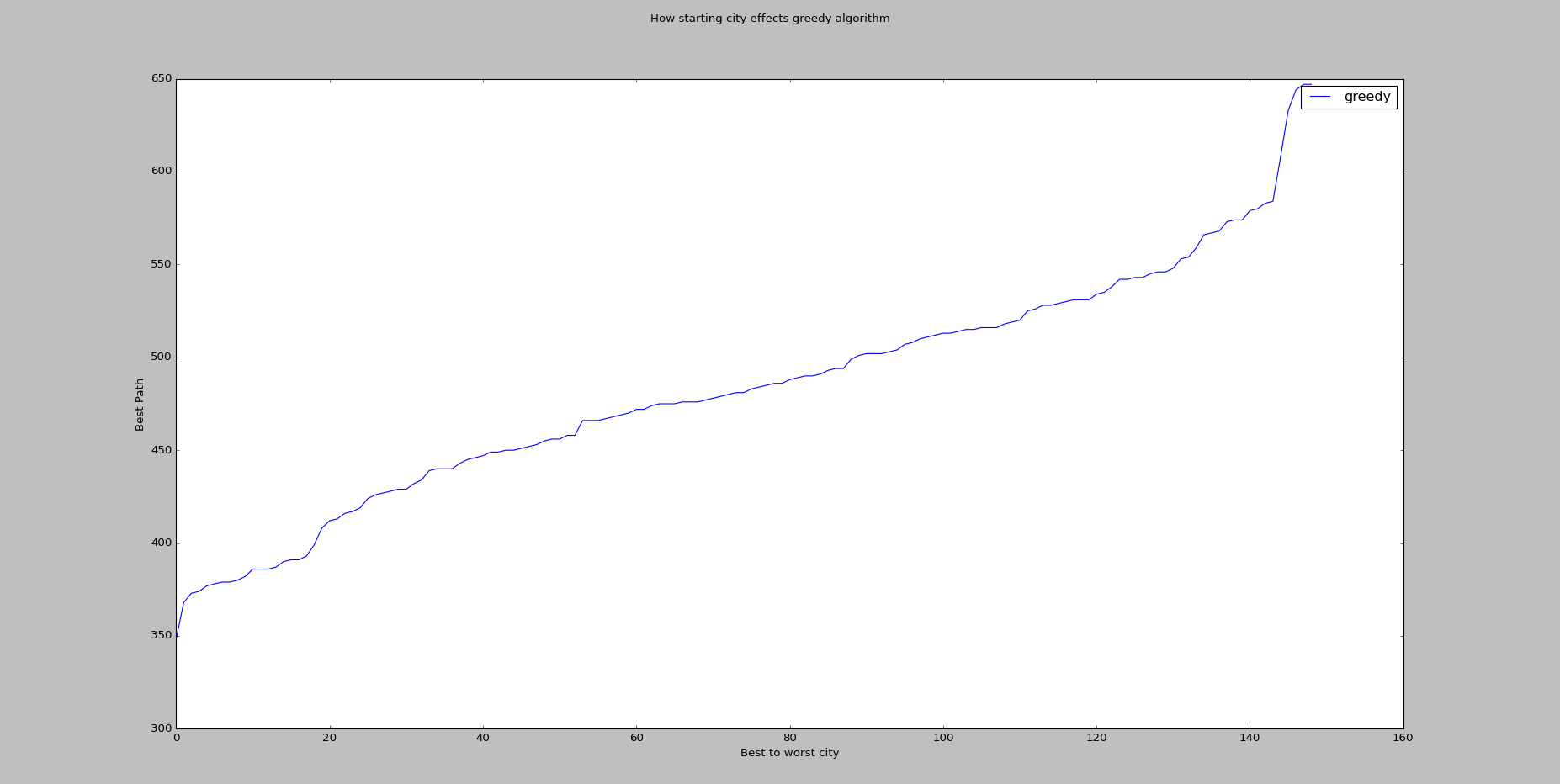
Igor Urbanowicz

4 semestr informatyka algorytmiczna

Indeks 261716

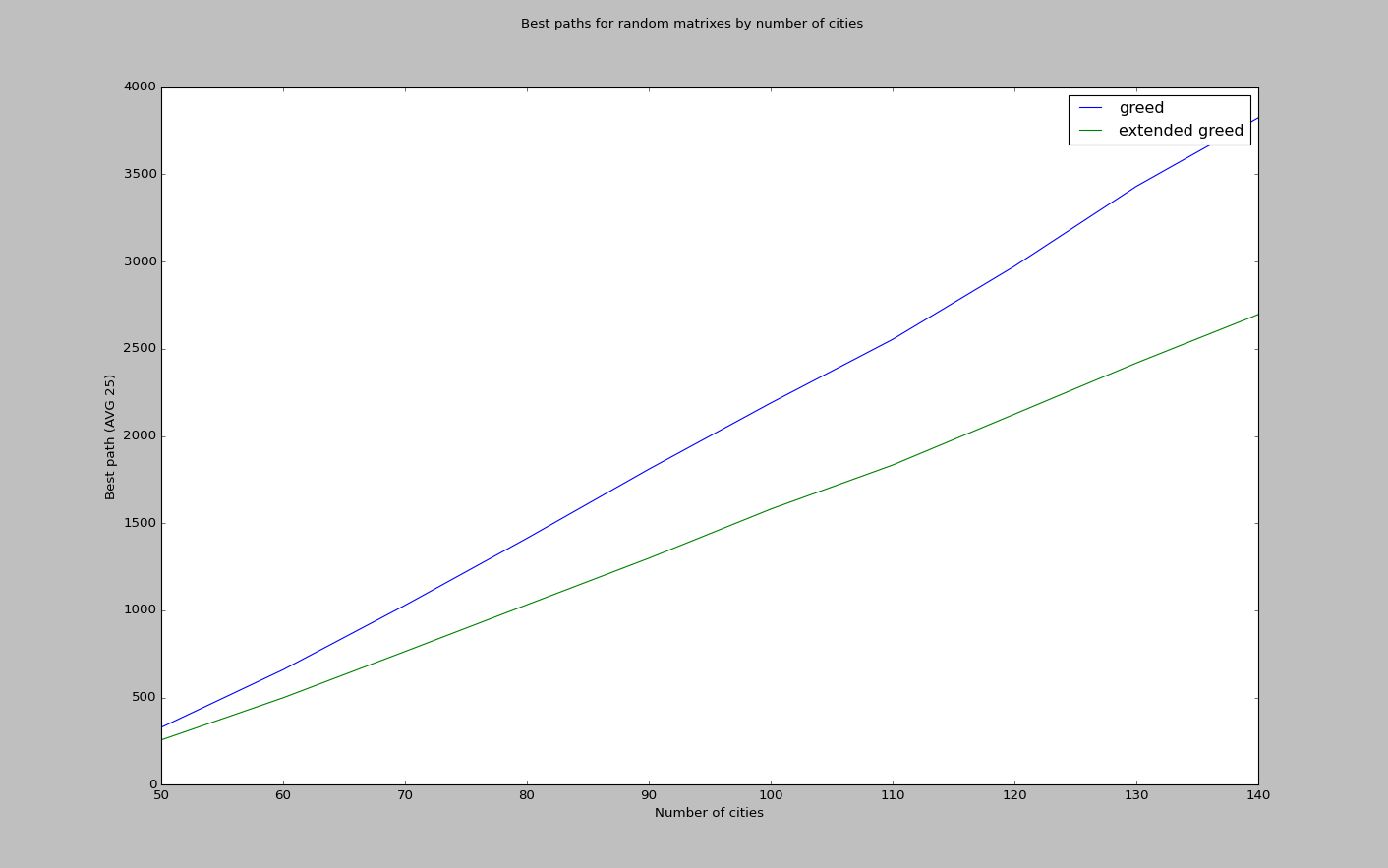
Sprawozdanie

Aby zacząć porównywanie algorytmów musimy ustalić pewne zmienne związane z algorytmami. Zwykłego greedy, który przyjmuje wierzchołek.



Powyższy graf przedstawia długość ścieżkę wygenerowanej przez greedy w zależności od podanego startowego miasta dla 150 miast. Jest to tylko jedna macierz ale pokazuje to jak ważne dla algorytmu jest wybranie odpowiedniego miasta startowego. PRD najlepszej do najgorszej ścieżki to około 83,1395%. Jeśli natomiast weźmiemy sobie średnią PRD z 150 takich ich wynik to ok 82,2471%. Tutaj wchodzi nasz ulepszony algorytm który sprawdza drogę dla każdego miasta i zwraca najlępszą z nich. Natomiast ma on jedną dużą wagę ponieważ zwiększa tępo wzrostu naszego problemu o n, gdzie n to liczba miast. Zobaczmy sobie teraz jakie w ogóle jest to tępo wzrostu.

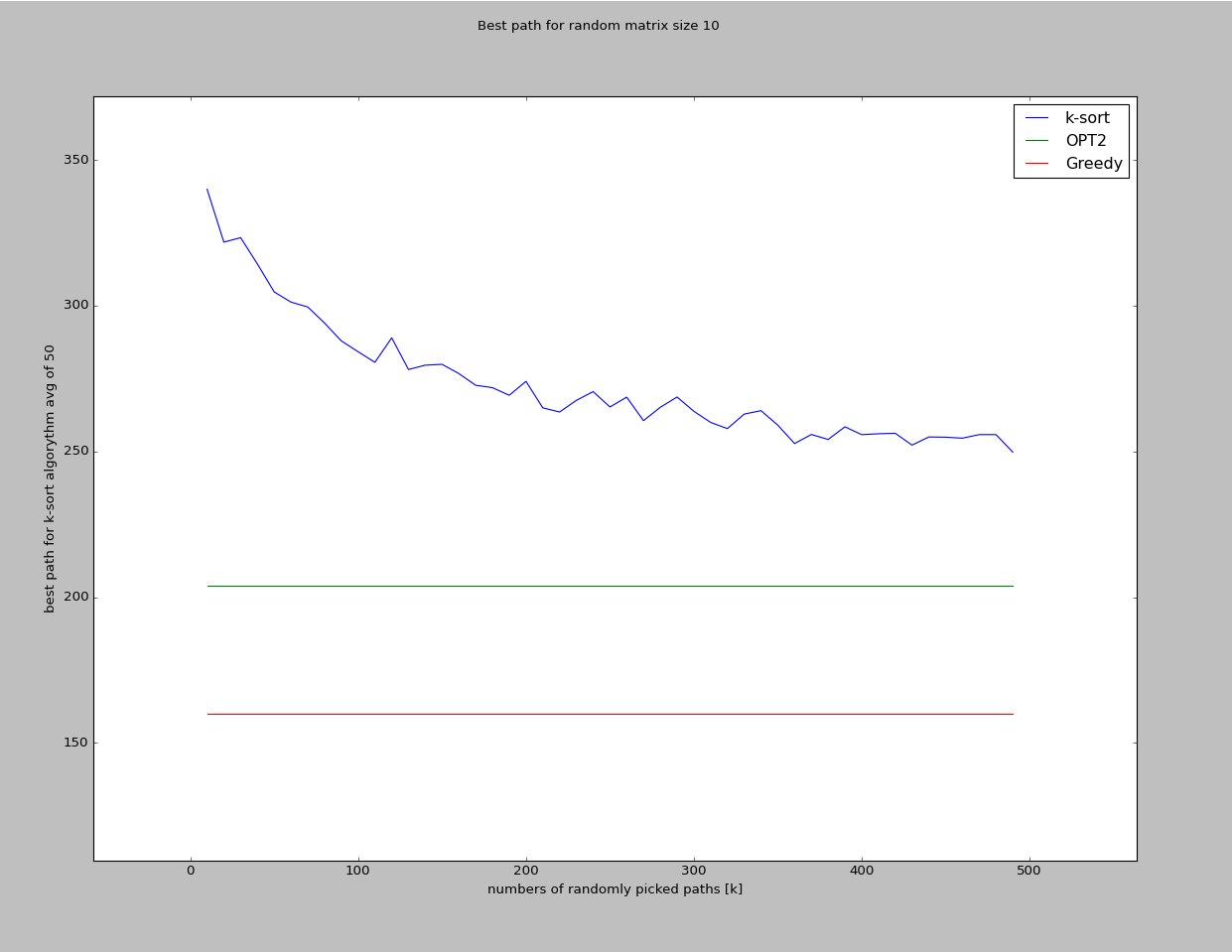
Zaczynamy od pewnego miasta x dla niego musimy wyliczyć najlepszą drogę do następnego punktu. Szukamy najmniejszej wartości pośród wszystkich innych miast liczących n (n przeszukań) i to samo musimy powtórzyć dla każdego następnego miasta (n – liczba już wcześniej przeszukanych miast) liczba przeszukanych miast to ciąg algorytmiczny od n o r= -1 więc średnia to n/2. Stałą możemy pominąć dla notacji O więc O(n^2) dla naszego zwykłego greedy oraz O(n^3) dla extended greedy.



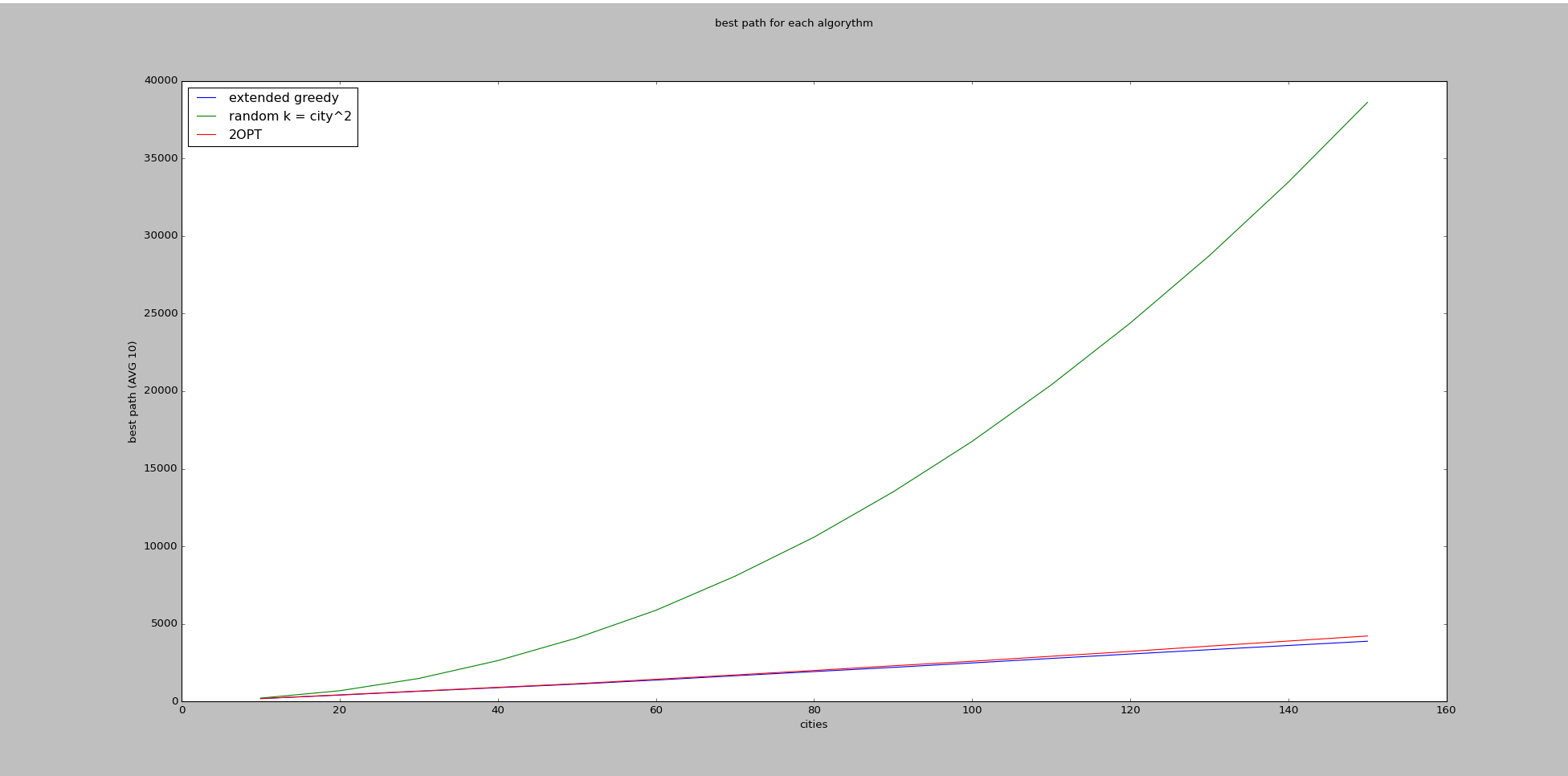
(warto zauważyć że różnica nie wynosi 80% bo nie bierzemy najgorszych przypadków dla greedy )

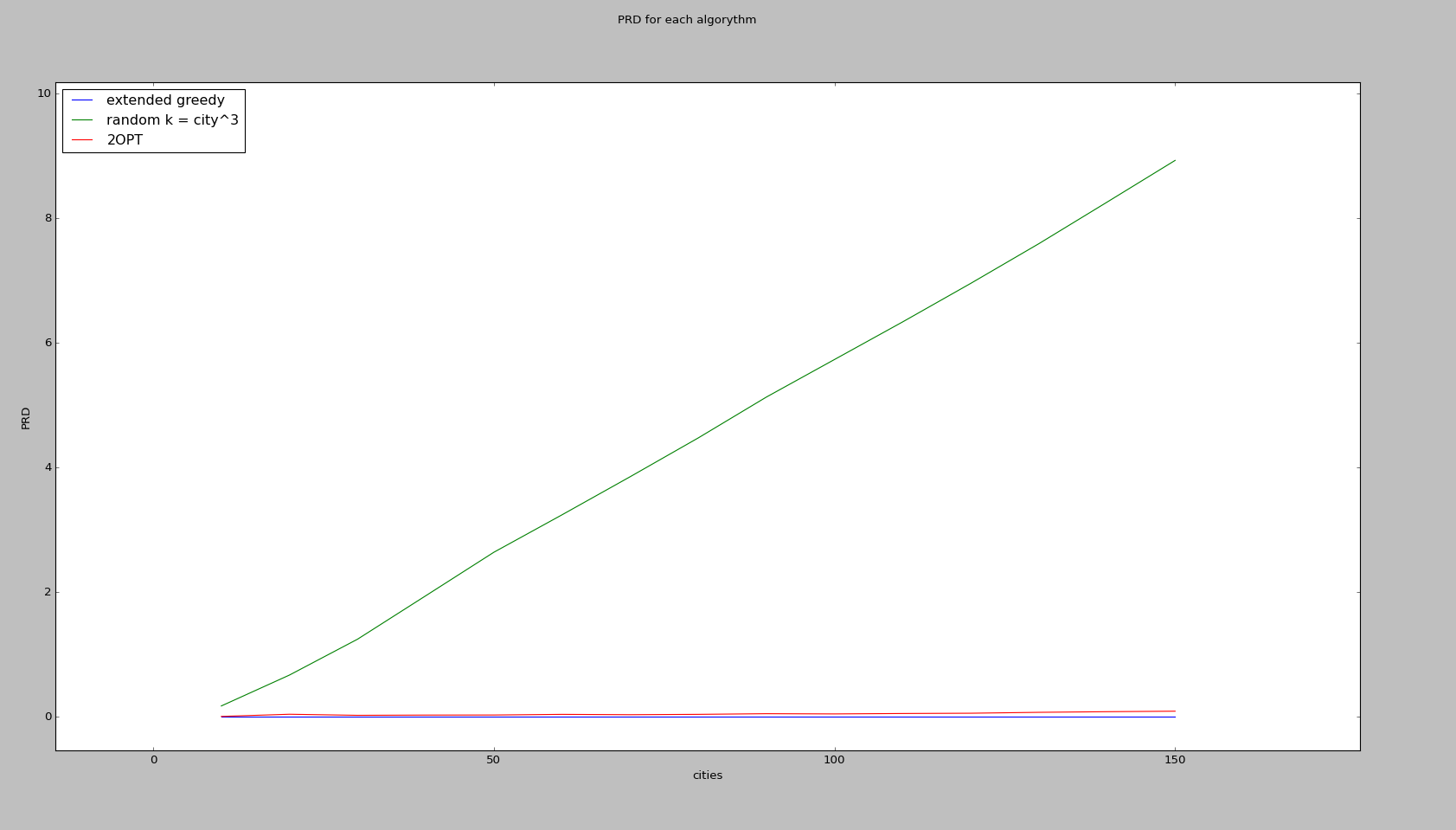
Pomyślmy o OPT2, dla niego musimy znaleźć 2 zmienne k oraz j dla których będziemy przestawiać ścieżkę. K przeszukujemy n razy a j zachowuje się jak nasze pozostałe miasta w greedy więc wiemy już że daje nam to n^2. Jednak invert ścieżki, który stosujemy w OPT2 działa liniowo, więc nasze finałowe O to O(n^3).

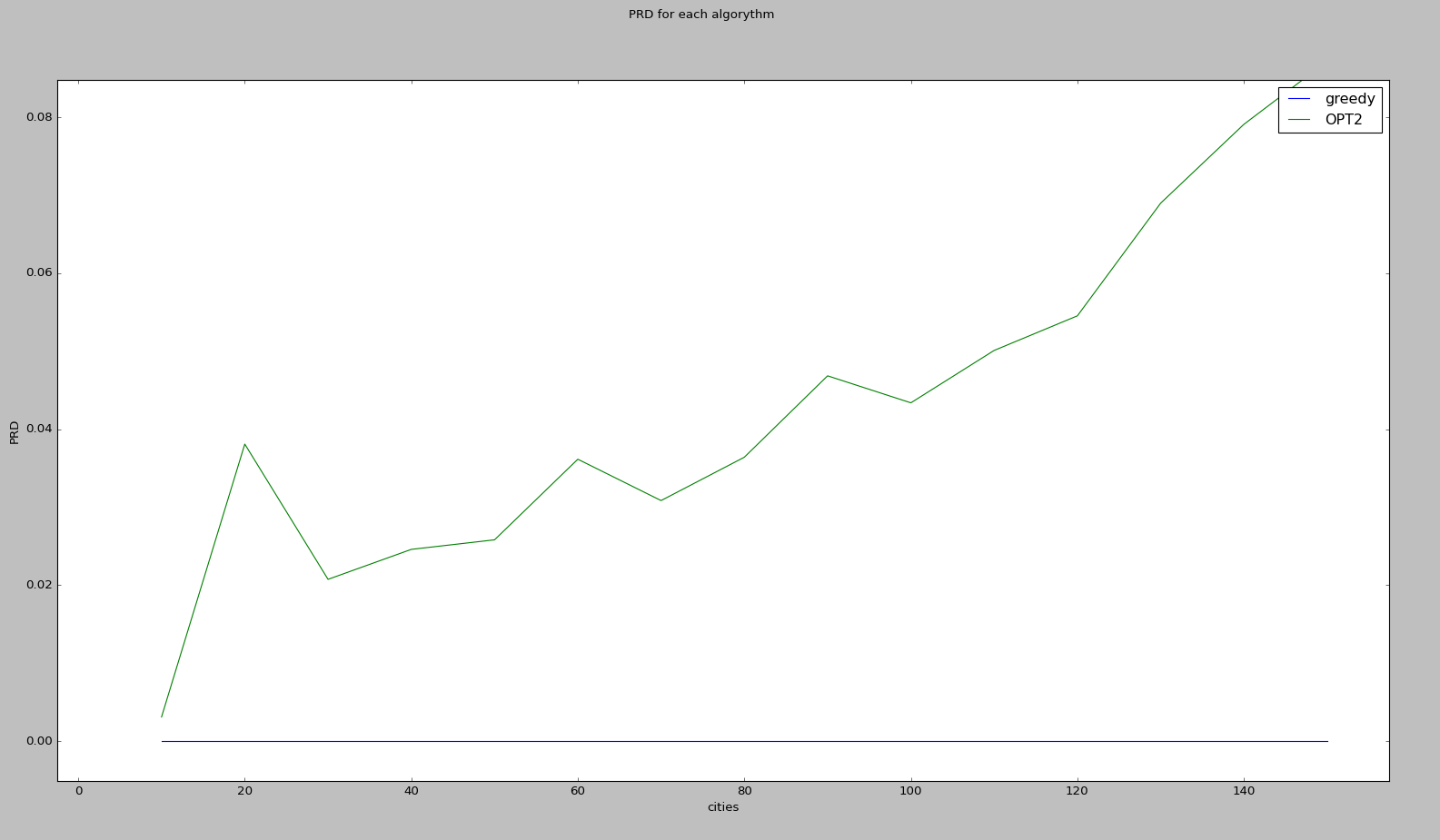
Teraz zobaczmy czas dla K-random. Dla k-random jest tylko jedna funkcja, która zbiera realnie czas i jest to tworzenie nowej ścieżki co dzieje się liniowo. Reszta czasu będzie zależna od k który tam wpiszemy. O(k\*n)

(droga dla różnych k na tej samej macierzy porównanie greedy i OPT2, widać że początkowa zmiana k dała nam dużo, ale później zwiększając k najlepsza ścieżka pozostaje podobna)

Teraz chciałbym porównać nasz 3 algorytmy, aby był to miarodajne chciałbym dać wszytkim algorytmom ten sam czas 0 więc wybieram OPT2, extended greedy oraz k-random dla k =n^2. Wszystkie z nich powinny mieć O(n^3). Teraz o tym jak wygląda przeprowadzony test. Będziemy liczyć najlepszą drogę na losowo generowanych macierzach symetrycznych gdzie najmniejsza odległość to 1 a najdłuższa to 100. Naszymi punktami odniesienia będą liczby miast w danej macierzy zaczynając od 10 a kończąc na 150 i przesuwając się o 10 z każdym punktem. Dla każdego punktu odniesienia test wykonamy 12 razy a wynik odpowiednio uśrednimy. (te same lub bardzo podobne metody zostały zastosowane dla wszystkich poprzednich wykresów)

(najlepsza droga na naszych algorytmów)



(PRD bez k-random)

Biorąc wszystko pod uwagę algorytm entended greedy ora OPT2 wydają się dużo bardziej efektywne w rozwiązywaniu problemu komiwojażera niż k-random przy podobnych O. między OPT2 oraz greedy, greedy wydaje się oddawać minimalnie lepszą ścieżkę ale różnice są na poziome 0-10%. Najlepszą ścieżkę oddaje oczywiście OPT2 ze ścieżką uzyskaną przez greedy, ale O rośnie do n^6.