Politechnika Łódzka

Instytut Informatyki

PRACA DYPLOMOWA MAGISTERSKA

Optymalizacja struktury sieci drogowej

Structure optimization of road networks

Wydział Fizyki Technicznej, Informatyki i Matematyki Stosowanej
Promotor: dr hab. inż. Aneta Poniszewska-Marańda
Kopromotor: mgr inż. Łukasz Chomątek
Dyplomant: inż. Michał Siatkowski
Nr albumu: 186865
Kierunek: Informatyka
Specjalność: Sztuczna Inteligencja i Inżynieria Oprogramowania

Łódź 20.01.2015

Instytut Informatyki

90-924 Łódź, ul. Wólczańska 215, budynek B9

tel. 042 631 27 97, 042 632 97 57, fax 042 630 34 14 email: office@ics.p.lodz.pl

Spis treści

Rozdział 1

Wstęp.

1.1 Problematyka i zakres pracy.

Niniejsza praca obejmuje zagadnienia z zakresu inżynierii oprogramowania i sztucznej inteligencji. Głównym jej celem jest stworzenie aplikacji optymalizującej strukturę sieci drogowej.

Problemy komunikacji w dzisiejszych miastach są wszystkim znane. Zatory drogowe i korki w godzinach szczytu są chlebem powszednim. Pomimo wielu prób i sposobów, wciąż nie istnieje metoda jednoznacznie rozwiązująca tę kwestię. Bezspornie, dotyczy to wszystkich miast na świecie. Z teoretycznego punktu widzenia, jedynym rozwiązaniem jest komunikacja publiczna. Oczywistym jest jednak, że nigdy nie doprowadzimy do sytuacji, gdy wszyscy mieszkańcy zrezygnują ze swoich pojazdów. Dodatkowo, wiele osób i usług wymaga oddzielnej formy transportu. W obliczu tych faktów miasta decydują się na rozwój swojej infrastruktury drogowej. Budowa nowych tras oraz poszerzanie starych przynosi nadzieję mniejszych zatorów a co za tym idzie, szybszego przejazdu do celu. Niestety, historia pokazuje, że takie inwestycje nie zawsze przynoszą oczekiwane korzyści.

Teorii próbujących wytłumaczyć te zjawiska, jak również dowodów, które je popierają lub obalają jest wiele. Jedną z najpopularniejszych oraz taką która została wykorzystana w niektórych miastach na świecie jest paradoks Braessa[15]. Jest to twierdzenie matematyczne orzekające, że w pewnym modelu ruchu drogowego czasy podróży pojazdów mogą ulec wydłużeniu po dodaniu do sieci drogowej nowego połączenia. Ma ono również zastosowanie w przypadku sieci komputerowych oraz istnieją jego analogie dla doświadczeń fizycznych.

Moim celem jest opracowanie metody, która dla danej struktury sieci drogowej zmodyfikuje ją wykorzystując prawidłowość z powyższego paradoksu. W efekcie poprzez zamknięcie wybranych ulic w danej sieci drogowej, średni czas podróży powinien ulec skróceniu.

1.2 Metoda badawcza i cel pracy.

Studia literaturowe.

Moje badania rozpocząłem od poszukiwania źródeł traktujących o opisywanym przeze mnie problemie. Paradoks Braessa został sformułowany w 1970 roku i był od tego czasu wykorzystywany przy planowaniu przestrzeni i infrastruktury wielu miast, np:

- Korea, Seul, likwidacja m.in. estakad Cheonggyecheon,
- Niemcy, Stuttgart, likwidacja dróg zbudowanych w latach 60,
- USA, Nowy Jork, czasowe zamknięcie ulicy 42,
- USA, Winnipeg.[16]

Propozycja rozwiązania problemu.

Oczywistym rozwiązaniem problemu komunikacji mogłoby być stworzenie idealnej sieci odpowiadającej potrzebom danego miasta. Rozbudowa lub modyfikacja tej infrastruktury jest jednak kosztowna i czasochłonna. Dlatego zdecydowałem się na przetestowanie rozwiązania zaproponowanego przez Braessa. Ponieważ istnieją prace negujące lub podważające paradoks[2] , zdecydowałem by przy potwierdzaniu wyniku optymalizacji nie kierować się wyłącznie założeniami zawartymi w twierdzeniu.

Opis zastosowania algorytmów genetycznych.

Ponieważ nie znalazłem żadnych przesłanek wykazujących jednoznaczną ocenę co do słuszności zamknięcia danej ulicy, zdecydowałem się losowe przeszukiwanie przestrzeni rozwiązań. Jednym z rozwiązań w przypadku takich poszukiwań są algorytmy genetyczne, na które się zdecydowałem w swojej pracy.

Przedstawienie oceny optymalizacji.

Paradoks Braessa zakłada dość oczywiste potwierdzenie swojej wiarygodności. Zdecydowałem się więc na zastosowanie zewnętrznego systemu oceny. Taką rolę spełniają systemy symulacji. System, który wybrałem działa zupełnie oddzielnie od metody twierdzenia, symulując rzeczywisty ruch na danej sieci drogowej. Wynik symulacji jest jednoznaczną wartością liczbową, przedstawiającą średni czas przejazdów wszystkich agentów biorących udział w danym scenariuszu. Zakładając stały zestaw agentów dla zmieniających się w wyżej opisany sposób sieci, dążymy oczywiście do minimalizacji średniego czasu przejazdu.

Ocena możliwości wdrożenia proponowanych rozwiązań.

Paradoks Braessa nie jest jedynym twierdzeniem traktującym o problemach komunikacyjnych miast. Wiele teorii jest opartych głownie na socjologicznych lub psychologicznych założeniach[1]. Są jednak niemniej ważne. Biorąc pod uwagę złożoność problemu, wynik otrzymany podczas eksperymentu nie może być dowodem ani decydującym głosem w decyzjach dotyczących ustalaniu rzeczywistego ruchu drogowego miasta.

1.3 Przegląd literatury w dziedzinie.

By przybliżyć temat problemów komunikacyjnych i rozwiązania zaproponowanego przez Braessa polecam pracę magisterską *Leslie Arthura Keith Bloy*[1]. W pracy zostały opisane także inne (wymieniane również przeze mnie) twierdzenia. Publikacja *Reducing the Effects of the Braess Paradox with Information Manipulation*[4] jest bardzo dobrym uzupełnieniem tematu o interesującą mnie kwestię symulacji wieloagentowych. Prezentuje ona różnice w wynikach dla przypadku losowego wyboru drogi przez agentów oraz tej wybranej przy użyciu inteligencji kolektywnej[2].

Zbiorowa praca napisana na potrzeby międzynarodowej konferencji dotyczącej technologii symulacji[4] jest pozycją, która opisuje podobny do podejmowanego przeze mnie problem. Mianowicie skupia się na modyfikacjach drogi poprzez zmiany dostępności jej składowych.

W zupełnie oddzielnej tematyce, algorytmów genetycznych, polecę bardzo łatwo dostępną pozycję, która jest wstępem do tematu[5]. Wybrałem ją głównie ze względu na prosty opis problemu oraz brak zaawansowanych przykładów, które tylko skomplikowałyby dla mnie ujęcie prostego problemu.

1.4 Układ pracy.

Tematem pracy jest optymalizacja struktury sieci drogowej, zaś za główny cel przyjęto stworzenie rozwiązania, które wykorzystując algorytmy genetyczne odnajdzie najlepsze rozwiązanie.

Rozdział 1 zawiera wstęp i cele pracy. W rozdziale 2 przybliżam istotę optymalizacji struktur sieci drogowych oraz opisuję teoretycznie wykorzystane przeze mnie rozwiązania. W rozdziale 3 wymieniam technologie, w których wykonałem aplikację oraz przybliżam zewnętrzne biblioteki, o które oparłem swoje rozwiązanie. Głównym rozdziałem pracy jest rozdział 4, w którym przedstawiam wykonaną aplikację oraz wyniki optymalizacji przykładowej sieci. W rozdziale podsumowującym 5, zawarłem wnioski oraz możliwości rozwoju pracy. Po rozdziale 5, podsumowującym, załączam spis rysunków, tabel, listingów i bibliografie, w tej kolejności.

[1] np. paradoks Downsa-Thomsona[17] czy prawo Lewisa-Mogridge'a[18]

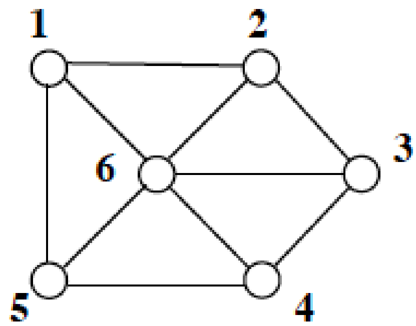
[2] ang. Collective Intelligence (COIN)

Rozdział 2

Optymalizacja struktury sieci drogowej.

2.1 Podstawowe definicje.

Definicja 1 *Grafem (nieskierowanym) nazywamy parę zbiorów* (V, E)*. Elementy zbioru* V *nazywają się wierzchołkami, natomiast elementy zbioru* E *nazywają się krawędziami. Kada krawędź jest parą wierzchołków, tzn.* $E \subseteq u, v : u, v \in V$



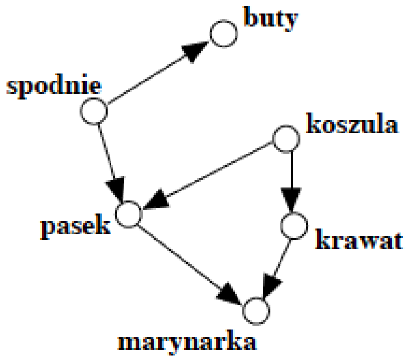
Rysunek 2.1: Przykładowy graf nieskierowany[19].

Przykładowy graf nieskierowany może zostać opisany przez zbiory:

$V = \{1, 2, 3, 4, 5, 6\}$ $\qquad E = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{4, 5\}, \{6, 5\}, \{6, 1\}, \{2, 6\}, \{3, 6\}, \{4, 6\}\}$

Definicja 2 *Grafem skierowanym nazywamy taki graf, w którym każda krawędź ma zdefiniowany początek i koniec, tzn. żeby pary były uporządkowane. Wtedy* $E \subseteq V \times V = u, v : u, v \in V$*. Krawędź* (u, v) *najłatwiej wyobrazić sobie jako strzałkę od u do v, dlatego często będziemy uznaczać ją przez* $u \to v$*.*

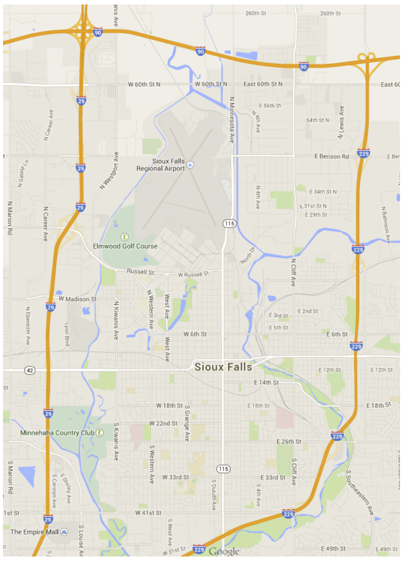
Przykładowy graf skierowany może zostać opisany przez zbiory:
$V = \{spodnie, buty, pasek, koszula, krawat, marynarka\}$
$E = \{\{spodnie \to buty\}, \{spodnie \to pasek\}, \{pasek \to koszula\}, \{koszula \to krawat\}, \{pasek \to marynarka\}, \{krawat \to marynarka\}\}$



Rysunek 2.2: Przykładowy graf skierowany[19].

2.2 Sieć drogowa w postaci grafu

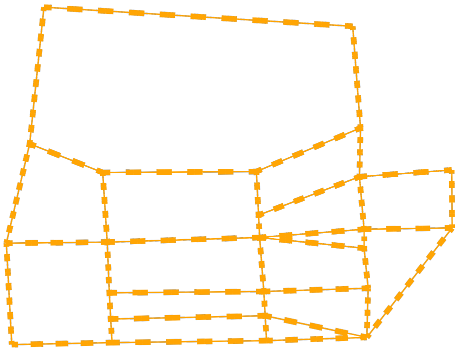
W przypadku sieci drogowej mamy oczywiście do czynienia z abstrakcyjną strukturą sieci. Przez sieć drogową rozumiemy bowiem układ dróg lub ulic np. w mieście. Podczas swoich badań posługuję się zawsze pewnym fragmentem większej sieci. Najlepiej te dane zobrazuje poniższy przykład.



Rysunek 2.3: Fragment sieci drogowej miasta Sioux Falls, Południowa Dakota.

Posługując się powyższymi definicjami, tworząc graf z pewnej sieci drogowej, przyjmujemy, że zbiorem V, wierzchołków są skrzyżowania, natomiast zbiór krawędzi E odnosi się do ulic pomiędzy tymi skrzyżowaniami. W mojej pracy posługuję się zawsze grafem skierowanym. W związku z tym, w przypadku ulic dwukierunkowych tworzę parę krawędzi z odpowiednimi kierunkami, nawet jeśli ulice nie są rozłączne w rzeczywistości.

W celu potwierdzenia wiarygodności powyższego przykładu prezentuję graf nałożony na mapę miasta.



Rysunek 2.4: Siec drogowa miasta Sioux Falls w postaci grafu.

Rysunek 2.5: Graf z dopasowaną geometrią [14].

2.3 Paradoks Braessa.

Jak już wcześniej wspomniałem, jest to twierdzenie matematyczne orzekające, że w pewnym modelu ruchu drogowego czasy podróży pojazdów mogą ulec wydłużeniu po dodaniu do sieci drogowej nowego połączenia. Autorem twierdzenia jest niemiecki matematyk Dietrich Braess[15]. Paradoks działa w oparciu o model ruchu drogowego, który ma następujące cechy:

1. Sieć drogowa składa się ze skończenie wielu węzłów i łączących je odcinków dróg
2. Po sieci porusza się skończenie wiele pojazdów, każdy z nich ma wyznaczony węzeł startowy i węzeł docelowy
3. Odcinki dróg mają przypisane sobie czasy przejazdu, przy czym czasy te mogą zależeć od liczby pokonujących dany odcinek pojazdów.
4. Układ sieci drogowej i czasy przejazdu poszczególnych odcinków są znane pojazdom
5. Celem pojazdów jest przejazd przez sieć z węzłów początkowych do docelowych po trasie złożonej z odcinków drogowych tak, by zminimalizować łączny czas ich pokonania
6. Decyzje o wyborze tras pojazdy podejmują indywidualnie i niezależnie od siebie

Przedstawię paradoks w oparciu o przykład z oryginalnego artykułu Dietricha Braessa.

2.3.1 Przykładowy wyjściowy układ drogowy.

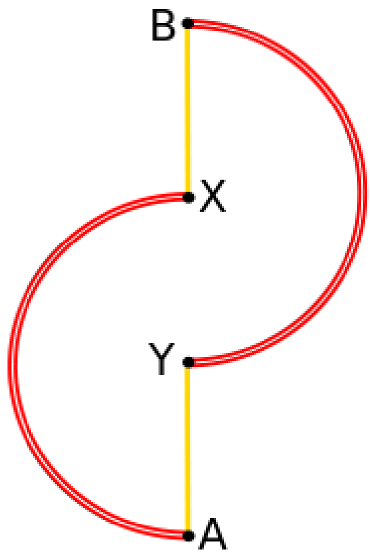
Sieć drogowa i auta

Przykład sytuacji, w której ujawnia się paradoks Braessa jest skonstruowany z czterech miast A, B, XiY. Są one połączone odcinkami drogowymi jak na rysunku i z następującymi czasami przejazdu, przy czym p oznacza gęstość ruchu w tysiącach aut.

Autostrady:
AX, $t_{AX}(p) = 50 + p$ min
YB, $t_{YB}(p) = 50 + p$ min

Drogi lokalne:
AY, $t_{AY}(p) = 10p$ min
XB, $t_{XB}(p) = 10p$ min

Aut jest 6000 i wszystkie mają za zadanie przejechać trasę z A do B.



Rysunek 2.6: Wyjściowy układ drogowy

Analiza równowagi Nasha

Każde auto musi zdecydować się na wybór trasy albo AXB albo AYB.

Równowaga Nasha to taka sytuacja, w której każdy z samochodów spowoduje wydłużenie swojego czasu jazdy, zmieniając decyzję co do wyboru trasy przy niezmienionych decyzjach pozostałych aut.

Jeśli p i q to liczby aut w tysiącach pokonujących odpowiednio trasy AXB i AYB, otrzymujmy równania:

$$p + q = 6$$
$$t_{AX}(p) + t_{XB}(p) = t_{AY}(q) + t_{YB}(q)$$
$$50 + p + 10p = 10q + 50 + q$$

rozwiązaniem jest $p = q = 3$. Przy tej gęstości ruchu pokonanie obu dostępnych tras zabiera $50 + 3 + 30 = 83$ minuty.

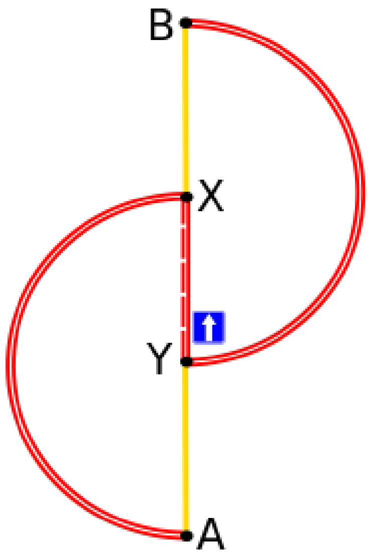
2.3.2 Przykładowy, uzupełniony układ drogowy.

Sieć drogowa i auta

Do wyjściowego układu drogowego dodana zostaje autostrada:

Autostrady:
YX, $t_{YX}(p) = 10 + p$ min

Aut jest nadal 6000 i wszystkie mają za zadanie przejechać trasę z A do B.



Rysunek 2.7: Uzupełniony układ drogowy

Analiza równowagi Nasha

Jeśli p, q i r to liczby aut w tysiącach pokonujących odpowiednio trasy AXB, AYB i AYXB, otrzymujmy równania:

$$p + q + r = 6$$
$$t_{AX}(p) + t_{XB}(p + r) = t_{AY}(q + r) + t_{YB}(q) = t_{AY}(q + r) + t_{YX}(r) + t_{XB}(p + r)$$
$$50 + p + 10(p + r) = 10(q + r) + 50 + q = 10(q + r) + 10 + r + 10(p + r)$$

rozwiązaniem jest $p = q = r = 2$. Czas przejazdu każdej z tych dróg wynosi wówczas $50 + 2 + 10(2 + 2) = 92$ minuty.

2.3.3 Wyjaśnienie intuicyjne.

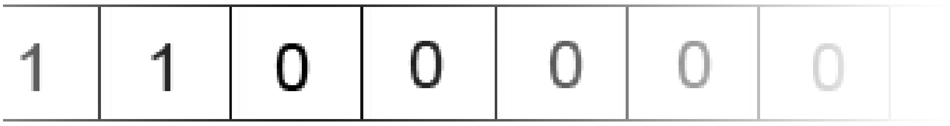
Wąskim gardłem systemu są drogi lokalne, na których czas przejazdu bardzo szybko wzrasta wraz z intensywnością ruchu. Po pojawieniu się dodatkowej drogi dostępna staje się nowa trasa, prowadząca oprócz nowego skrótu YX tylko drogami lokalnymi.

Z perspektywy całości systemu nowy odcinek drogowy odciąża ruch na autostradach, gdzie jest to mało odczuwalne, a w zamian jeszcze bardziej zagęszcza ruch na drogach lokalnych, powodując wydłużenie czasu podróży.

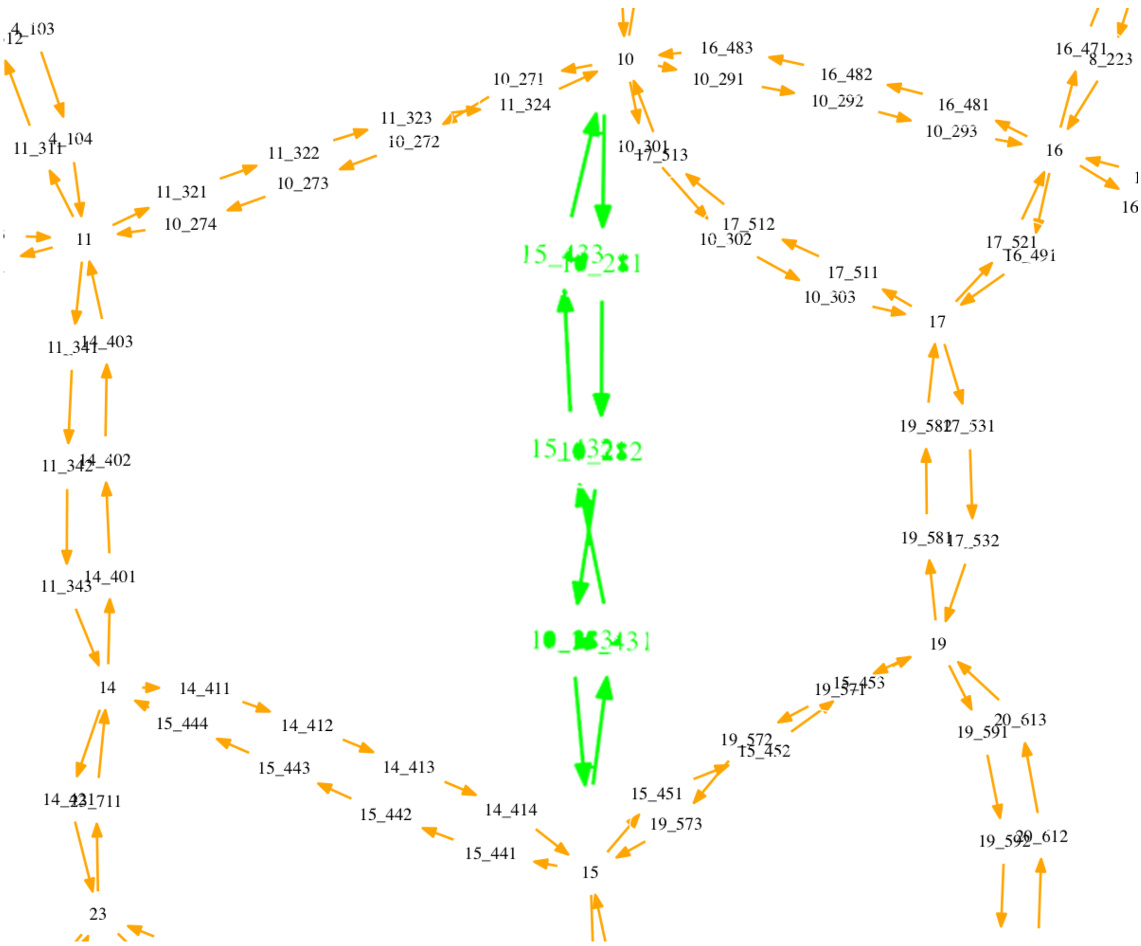
2.4 Składowa silnie spójna.

2.5 Punkty artykulacji grafu.

2.6 Klasyczny algorytm genetyczny.



Rysunek 2.8: Fragment sieci w postaci tablicy binarnej



Rysunek 2.9: Fragment sieci w postaci grafu

Rozdział 3

Technologie i metody użyte.

3.1 Symulator transportu.

Rysunek 3.1: Logo symulatora transportu MATSim [6]

3.2 Algorytmy genetyczne.

Najlepszego rozwiązania będę poszukiwał wykorzystując klasyczny algorytm genetyczny.

Rysunek 3.2: Logo biblioteki Apache Commons Math [7]

3.3 Technologie i metodologie programistyczne.

3.4 Badane miasto przykładowe: Siouxfalls

Rysunek 3.3: Logo Java[8]

Rysunek 3.4: Logo IDE Eclipse[9]

Rysunek 3.5: Logo Python[10]

Rysunek 3.6: Logo PyDev[11]

Rozdział 4

Opis projektu.

Ta część pracy może być podzielona na więcej rozdziałów, np kiedy autor chce w szczególności podkreślić któryś z etapów projektu. W zależności od tematu i celów pracy, pewne sekcje można dodać (np. przy projektowaniu sieci, instalacji i konfiguracji serwerów usług sieciowych), inne zaś pominąć.

4.1 Dane wejściowe.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<config>
   <project>
      <name>siouxfalls</name>
      <output-dir>../output/</output-dir>
      <threads>4</threads>
      <log-level>INFO</log-level>
      <python-path>/usr/local/bin/python</python-path>
   </project>
   <scenario>
      <config>../siouxfalls/config.xml</config>
      <network>../siouxfalls/network.xml</network>
      <population>../siouxfalls/population.xml</population>
      <facilities>../siouxfalls/facilities.xml</facilities>
      <iterations>50</iterations>
   </scenario>
   <genetics>
      <population-size>35</population-size>
      <max-generations>150</max-generations>
      <elitism-rate>0.15</elitism-rate>
      <crossover-rate>1.0</crossover-rate>
      <mutation-rate>0.1</mutation-rate>
      <tournament-arity>2</tournament-arity>
   </genetics>
</config>
```

4.2 Wdrożenie.

pewnie brakuje tekstu

4.3 Wyniki.

Rysunek 4.1: Logo Trisquel[12]

Rozdział 5

Podsumowanie.

5.1 Dyskusja wyników.

5.2 Perspektywy dalszych badań w dziedzinie.

5.3 Struktura projektu.

Spis rysunków

Spis tablic

Spis listingów

Bibliografia

[1] Leslie Arthur Keith Bloy, *An investigation into Braess' paradox*, 02/2007

[2] Rric Pas and Shari Principio *Braess' paradox: Some new insights*, April 1996

[3] Wataru Nanya, Hiroshi Kitada, Azusa Hara, Yukiko Wakita, Tatsuhiro Tamaki, and Eisuke Kita *Road Network Optimization for Increasing Traffic Flow* Int. Conference on Simulation Technology, JSST 2013.

[4] Ana L. C. Bazzan and Franziska Klügl *Reducing the Effects of the Braess Paradox with Information Manipulation*

[5] Mitchell Melanie *An Introduction to Genetic Algorithms* First MIT Press paperback edition, 1998

[6] http://matsim.org

[7] http://commons.apache.org/proper/commons-math

[8] http://www.java.com/pl/

[9] https://eclipse.org

[10] http://pl.python.org

[11] http://pydev.org

[12] https://trisquel.info

[13] M. Rieser, C. Dobler, T. Dubernet, D. Grether, A. Horni, G. Lammel, R. Waraich, M. Zilske, Kay W. Axhausen, Kai Nagel *MATSim User Guide* updated September 12, 2014

[14] A. Chakirov *Enriched Sioux Falls Scenario with Dynamic Demand* MATSim User Meeting, Zurich/Singapore, June 2013.

[15] http://pl.wikipedia.org/wiki/Paradoks_Braessa

[16] http://urbnews.pl/paradoks-braessa/

[17] http://pl.wikipedia.org/wiki/Paradoks_Downsa-Thomsona

[18] http://pl.wikipedia.org/wiki/Prawo_Lewisa-Mogridge'a

[19] http://www.mimuw.edu.pl/~kowalik/asd/2003/wyklady/grafy.pdf

Załączniki

1. Załącznik nr 1
2. Załącznik nr 2
3. Załącznik nr 3

Abstract

The purpose of the present bachelor thesis was to create an internet application with an integrated recommender system based on music resources. My work covered two main fields: creating the application as well as building a recommender system and testing it efficiency.