

OBELISC: Modelowanie i sterowanie oparte na oscylatorach z wykorzystaniem efektywnego uczenia neuronowego do inteligentnego obliczania sygnalizacji świetlnej w ruchu drogowym

paź 18, 2025

Wprowadzenie

Głównym celem niniejszego badania jest wprowadzenie modelowania i sterowania opartego na oscylatorach z wykorzystaniem metody OBELISC (Efficient neural Learning for Intelligent Road Traffic Signal Calculation), nowej metodologii i systemu do wspólnego modelowania, uczenia się i sterowania dynamiką przepływów ruchu w celu efektywnego obliczania czasu trwania fazy.

W bardzo dobrym przeglądzie i perspektywie, w badaniu [2] wprowadzono formalizm modelowania i sterowania ruchem opartym na oscylatorach. Pomimo dobrych podstaw matematycznych, proponowane podejście było statyczne, ponieważ usuwało całą dynamikę konwergencji i samoorganizacji oscylatorów, zastępując ją rozwiązaniem stacjonarnym. Takie podejście ma zalety na poziomie pojedynczego skrzyżowania, jak twierdzą również autorzy, ale zawiedzie w rozległych, heterogenicznych sieciach drogowych (tj. o nierównomiernej geometrii dróg, zaburzonych wzorcach ruchu itp.).

Podejście OBELISC wprowadza nowy typ nieliniowego modelu sprzężonych oscylatorów opartego na [15], wraz z nieliniowym mechanizmem sterowania, który pozwala mu uchwycić złożone wzorce przepływu i nieprzewidywalne zmiany [16] w dużych sieciach drogowych. Zapewnia to solidną kontrolę modelu opartego na oscylatorach przy dynamicznych zmianach zapotrzebowania na podstawie pomiaru lokalnych danych o ruchu. Podobne podejście oscylacyjne zostało zastosowane w [13], a później w [5], jako obszarowe sterowanie sygnałem w miejskiej sieci ruchu. Jednak ze względu na dynamikę i optymalizację o złożonych wartościach, systemy nie były w stanie uchwycić zarówno korelacji przestrzennych, jak i czasowych przy realistycznych kosztach obliczeniowych dla wdrożenia w świecie rzeczywistym.

Dodatkowo, przyczyniamy się do wydania wielowymiarowego zbioru danych o ruchu miejskim, który zawiera 59 dni rzeczywistych danych o ruchu drogowym na drogach miejskich z 8 skrzyżowań w mieście w Chinach. Zorientowany na wdrożenie w warunkach rzeczywistych i lepszą wydajność w czasie rzeczywistym przy rzeczywistych przepływach ruchu,

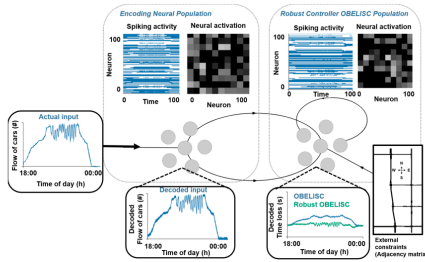
OBELISC:

- wykorzystuje wydajną implementację w impulsowych sieciach neuronowych [4],
- unika procedur optymalizacyjnych i działa w czasie rzeczywistym,
- wyróżnia się minimalizacją typowych kluczowych wskaźników wydajności ruchu, gdy czas trwania fazy jest obliczany na podstawie pomiarów zapotrzebowania w czasie rzeczywistym.

[Read more](#)

Treść kodu dla zgłoszenia ECML21 nr 394

New text



2.3 Reprezentacja, uczenie się i dynamika w sieciach neuronowych

Pojęcie fazy pozwala na bezpośrednią identyfikację stanu systemu w kategoriach zmiennej jednowymiarowej, opisanej w równaniu 1. Ułatwia to analityczne podejście do solidnej kontroli takiej dynamiki, jak pokazano w równaniu 3.

Tak złożony opis analityczny dynamiki sieciowej nie jest jednak możliwy do zastosowania w dużych, rzeczywistych zastosowaniach. Aby wdrożyć efektywną optymalizację fazy sygnalizacji świetlnej za pomocą OBELISC, reprezentacja danych, dynamika sieci oscylacyjnej oraz solidny sterownik są implementowane w ramach Neural Engineering Framework (NEF) [4].

NEF oferuje systematyczną metodę „kompilowania” dynamiki wysokiego poziomu, takiej jak równania różniczkowe zwyczajne (ODE), w wagi połączeń synaptycznych między populacjami neuronów impulsowych o efektywnych możliwościach uczenia się.

Reprezentacja danych o przepływie ruchu.

W NEF populacje neuronowe reprezentują sygnały zmienne w czasie, takie jak dane o przepływie ruchu, poprzez swoją aktywność impulsową. Takie sygnały sterują populacjami neuronów w oparciu o krzywą strojenia każdego neuronu, która opisuje, jak często dany neuron będzie się aktywował w funkcji sygnału wejściowego (patrz rys. 5 – Kodowanie populacji neuronów, panel górny).

Rolą reprezentacji (tj. uzupełnionej o parę operacji kodowania/dekodowania) jest zapewnienie rozproszonej wersji sygnału wejściowego o wartościach rzeczywistych. Zasadniczo, korzystając z tej reprezentacji, możemy oszacować sygnał wejściowy pierwotnie zakodowany poprzez dekodowanie wzoru impulsów (patrz rys. 5 – Kodowanie populacji neuronów). Wagi dekodowania są określane poprzez minimalizację kwadratów różnicy między zdekodowanym oszacowaniem a rzeczywistym sygnałem wejściowym i uwzględniają proces uczenia się wag³.

Uczenie się dowolnych funkcji danych o przepływie.

Operacje kodowania i dekodowania na reprezentacjach populacji neuronów NEF pozwalają nam kodować sygnały przepływu ruchu w czasie i dekodować transformacje (tj. funkcje matematyczne) tych sygnałów.

W rzeczywistości NEF pozwala nam dekodować dowolne transformacje sygnału wejściowego. W naszym przypadku prawa strona równania 1 zawiera nieliniową kombinację członów, z których na przykład sinus względnej różnicy faz $\sin(\theta_j(t) - \theta_i(t))$ jest dekodowany jako transformacja sinusoidalna z populacji kodującej różnicę faz $\theta_j(t) - \theta_i(t)$. Ta sama zasada dotyczy odpornej kontrolowanej dynamiki w równaniu 2 i jest przedstawiona na rys. 5. Ten proces określa, w jaki sposób możemy dekodować ciągi impulsów, aby obliczyć liniowe i nieliniowe transformacje różnych sygnałów zakodowanych w populacji neuronów. Zasadniczo zapewnia to sposób uczenia się wag połączeń neuronowych w celu obliczenia funkcji między populacjami (np. iloczynu populacji kodującej sprzężenie sąsiedztwa przestrzennego A_{ij} i populacji kodującej transformację sinusoidalną różnicy faz $\theta_j(t) - \theta_i(t)$).

Dynamika sieci oscylatorów ruchu w sieciach neuronowych.

Zasadniczo NEF automatycznie tłumaczy standardowe opisy układów dynamicznych na opisy zgodne z dynamiką neuronową. Wykorzystując rozproszoną reprezentację neuronową danych o ruchu i ucząc się dowolnych funkcji danych o ruchu, możemy teraz opisać połączoną implementację dynamiki sieci oscylatorów i kontrolera trybu przesuwne.

Rysunek 5 przedstawia szczegóły implementacji wysokiego poziomu. Implementacja neuronowa przedstawiona na rys. 5 jest powiązana z każdym oscylatorem i w sieci. Każdy oscylator jest zasilany danymi o przepływie ruchu $k_i(t)$ odpowiadającymi kierunkowi, który kontroluje. Dane o wartościach rzeczywistych są następnie kodowane w rozproszonym wzorze w populacji neuronów kodujących. Ten proces kodowania jest widoczny na panelach „Aktywność impulsowa” i „Aktywność impulsowa” na rys. 5, gdzie każdy neuron koduje dane wejściowe w ciągu impulsów o modulowanej częstotliwości (Aktywność impulsowa). Aktywacja czasowa każdego z neuronów kodujących względem siebie jest zilustrowana na panelu „Aktywacja neuronowa”. Jak widać na rys. 5 – panele lewy i dolny lewy – zdekodowany przepływ ruchu samochodów jest zaszumioną wersją rzeczywistego sygnału wejściowego (intuicyjnie, większa liczba neuronów zapewni lepszą rekonstrukcję, ale większy koszt obliczeniowy).

Zakodowane dane o przepływie ruchu są następnie przekazywane do rzeczywistej połączonej dynamiki (tj. sieci oscylatorów i kontrolera trybu ślizgowego) w populacji kontrolera Robust Controller OBELISC. Ta populacja neuronów ma połączenie rekurencyjne, które implementuje dynamikę strony wyższej równania 2.

Dokładniej, ta populacja dzieli równanie 2 w kategoriach i realizuje każde mnożenie, funkcję nieliniową i sumowanie w oddzielnych, spójnych populacjach. Zasadniczo, populacja kodująca częstotliwość oscylacji ω_i będzie połączona poprzez funkcję sumy z populacją kodującą sumę ograniczeń zewnętrznych (np. A_{ij} i F_i) wających różnice fazowe $\theta_j(t) - \theta_i(t)$, obie dekodowane z oddzielnych populacji neuronowych implementujących funkcje iloczynowe i sinusoidalne.

Operacje te zaimplementowane w neuronach odpowiadają równaniu $A_{ij} \sin(\theta_j(t) - \theta_i(t)) + F_i \sin(\theta(t) - \theta_i(t))$. Aby zwizualizować korzyści płynące ze sterowania w trybie ślizgowym w ogólnej dynamice, obliczyliśmy również stratę czasu, jako prostą metrykę, w populacji Robust Controller OBELISC. Jak wspomniano wcześniej, sterownik w trybie ślizgowym stanowi kompromis między wydajnością a aktywnością sterowania (tj. lepszą wydajność pod względem straty czasu przy szybszych zmianach prawa sterowania).

Generalnie widać to na rys. 5 – dolny prawy panel, pomiędzy godziną 18:00 a 24:00, gdzie dynamika sieci oscylatorów OBELISC działa płynniej, ale gorzej pod względem optymalizacji straty czasu, podczas gdy solidna sieć OBELISC (tj. dynamika zawierająca regularyzację trybu ślizgowego) poprawia stratę czasu kosztem oscylacji o wysokiej częstotliwości i niskiej amplitudzie.

Eksperymenty i wyniki

Oceniane systemy to:

- BASELINE – zoptymalizowane statyczne planowanie ruchu, które wykorzystuje wstępnie zapisane plany czasowe obliczone offline na podstawie danych historycznych w świecie rzeczywistym.

- MILP – implementacja optymalizacji planu fazowego w programowaniu liniowym mieszanym całkowitoliczbowym, inspirowana [14].

- OSCILLATOR – podstawowa implementacja sieci oscylatorów Kuramoto [15] dla każdego kierunku w sieci drogowej.

– OBELISC – wykorzystuje podstawowy model oscylatora Kuramoto z [15] i uwzględnia zewnętrzne odniesienie dla czasu cyklu F , modulacji przepływu k i wagi topologii przestrzennej A . Wykorzystaliśmy dwie implementacje, jedną wykorzystującą bazowy solver równań różniczkowych (OBELISC ODE), a drugą wykorzystującą impulsowe sieci neuronowe NEF (OBELISC NEF).

– Robust OBELISC – rozszerza podstawowy OBELISC o regularyzujące prawo sterowania trybem ślizgowym u. Robust OBELISC, podobnie jak OBELISC, ma dwie wersje: Robust OBELISC ODE i Robust OBELISC NEF.

Ocena czasu wykonania.

Pod względem czasu wykonania metody adaptacyjne zapewniają różne poziomy wydajności, głównie ze względu na wykorzystywane typy modelowania i optymalizacji.

LINIA BAZOWA jest pomijana, ponieważ jest to jedynie statyczna, zoptymalizowana alokacja planów dla rzeczywistego natężenia ruchu w SUMMER-MUSTARD, zasadniczo proste wywołanie wartości z tabeli. Zmierzyliśmy czas potrzebny każdemu z ocenianych systemów adaptacyjnych na oszacowanie czasu trwania fazy po próbie sensorycznej (tj. jednym sensorycznym odczycie danych o przepływie ruchu).

Jak wspomniano, każdy system wykorzystuje inne podejście obliczeniowe: MILP wykorzystuje solver implementujący algorytm rozgałęzień i ograniczeń oparty na LP, OSCILLATOR wykorzystuje solver równania różniczkowego Runge-Kutty 45, natomiast OBELISC można zaimplementować za pomocą solvera równania różniczkowego Runge-Kutty 45 lub połączonych populacji impulsowych sieci neuronowych NEF. Wyniki przedstawiono w tabeli 2, gdzie uwzględniono średnią wartość dla całego zakresu warunków ruchu (normalnych i zakłóceń).

Konfiguracja eksperymentalna naszych eksperymentów wykorzystywała 3 maszyny, każda z 24 rdzeniami procesora i 132 GB pamięci RAM, oraz Apache Flink do przetwarzania strumieniowego i zarządzania klastrem. Zgodnie z oczekiwaniami, na poziomie optymalizacji pojedynczego skrzyżowania, podejścia solvera równania różniczkowego (tj. MILP, OSCILLATOR i OBELISC (ODE)) mieszczą się w tym samym zakresie, zapewniając nową wartość czasu trwania fazy po 50 ms. Na poziomie regionu, biorąc pod uwagę wszystkie 8 krzyżowań, czas wykonania wzrasta o rząd wielkości, a MILP wyprzedza OSCILLATOR i OBELISC (ODE) ze względu na wydajność optymalizacji ograniczeń MILP w dużej skali oraz podobne obliczenia OSCILLATOR i OBELISC (ODE).

Najszybszym podejściem, zarówno na poziomie pojedynczego krzyżowania, jak i regionów, jest implementacja neuronowa NEF w OBELISC. Z ponad 80% poprawą czasu wykonania, zarówno na poziomie pojedynczego krzyżowania, jak i regionalnym, OBELISC (NEF) wyróżnia się dzięki wydajnemu podłożu obliczeniowemu i uczeniu się.

Dyskusja

Tradycyjnie optymalizacja czasu trwania fazy dla skoordynowanej sygnalizacji świetlnej opiera się na średnim czasie przejazdu między skrzyżowaniami i średnim natężeniu ruchu na każdym skrzyżowaniu.

Modelowanie.

Nasze badanie wprowadza kompleksowy system modelowania, sterowania i uczenia się do optymalizacji czasu trwania fazy ruchu drogowego, mający zastosowanie do dowolnego układu, skali i architektury ruchu drogowego (np. liczby pasów ruchu w każdym kierunku itp.).

Dokładniej, wykorzystując model dynamiki przepływu ruchu oparty na oscylatorach [15] w dużych sieciach dróg z sygnalizacją świetlną, system wykorzystuje okresową naturę cyklicznej fazy sygnalizacji świetlnej, podobną do [1,5] – nazywanej w naszych eksperymentach OSCYLATOREM. OBELISC wykracza poza OSCILLATOR, uwzględniając ważne zaburzenie zewnętrzne (np. wagę odniesienia czasu cyklu F), modulację przepływu k oraz wagę topologii przestrzennej A. Takie podejście do modelowania dostosowuje się do nieprzewidywalnych zakłóceń w przepływach ruchu (np. wypadków, zmian tras, niekorzystnych warunków pogodowych) do pewnego stopnia, w którym dynamika zakłócenia nie zakłóca samoorganizacji sprzężonych oscylatorów.

Solidne sterowanie.

W rzeczywistości „stromie pochodne” przepływów ruchu nie pozwalają OBELISC i OSCILLATOR na zbieżność do najlepszej wartości czasu trwania fazy. Aby osiągnąć wysoką wydajność (tj. minimalizację metryk, takich jak strata czasu lub maksymalizacja średniej prędkości), uzupełniliśmy OBELISC o kontroler o trybie przesuwym. Taki odporny sterownik „popycha” zaburzoną dynamikę pod wpływem zakłóceń w kierunku dynamiki, która „napędza” sieć sprzężonych oscylatorów w kierunku fazy optymalnej. W ten sposób system Robust OBELISC jest w stanie rozwiązywać lokalne i globalne problemy z dynamiką ruchu, wykorzystując sprzężenie między różnymi oscylatorami opisującymi okresowość ruchu w przypadku zakłóceń.

Zaproponowany w [14] system – nazywany w naszych eksperymentach MILP – wykorzystywał dokładne techniki programowania matematycznego (tj. mieszane programowanie liniowe całkowitoliczbowe) do optymalizacji sterowania sygnalizacją świetlną i wykazał jedynie ograniczone możliwości adaptacyjne.

Obliczenia i uczenie się.

W rzeczywistych ograniczeniach sterowania ruchem, OBELISC i Robust OBELISC nie mogą zostać zaimplementowane poprzez proste całkowanie równań różniczkowych zwyczajnych. Aby złagodzić typowe problemy z konwergencją, stabilnością i odpornością integracji równań różniczkowych zwyczajnych, zaimplementowaliśmy OBELISC w wydajnych sieciach neuronowych z impulsami, wykorzystując NEF. Zasadniczo, wykorzystując rozproszone reprezentacje danych o przepływie ruchu, ucząc się dowolnych funkcji z danych i „kompilując” równania różniczkowe (ODE) w populacjach neuronowych, uzyskaliśmy wydajne i elastyczne implementacje OBELISC (tj. OBELISC NEF i Robust OBELISC NEF).

Taki wybór zapewnił wyraźną przewagę nad implementacją MILP z [14], która sformułowała optymalizację fazową w ciągły problem optymalizacji bez zmiennych całkowitych poprzez modelowanie przepływu ruchu jako sinusoidalnego. System rozwiązał wypukłą relaksację problemu niewypukłego, wykorzystując redukcję dekompozycji drzewa i losowe zaokrąglanie, aby odzyskać rozwiązanie niemal globalne. Biorąc pod uwagę złożoność wyrażania dynamiki systemu, MILP dobrze wypadł w symulacjach, jednak brakuje możliwości adaptacji do nagłych zmian w sytuacji drogowej (patrz ranking strat czasu, prędkości i czasu oczekiwania na rys. 6).

Wnioski

Sterowanie ruchem drogowym to wielowymiarowy problem, który należy optymalizować w warunkach głębokiej niepewności. Modelowanie dynamiki ruchu drogowego ma fundamentalne znaczenie dla sterowania ruchem drogowym. Dążąc do uchwycenia okresowej natury ruchu drogowego, proponujemy OBELISC, system wykorzystujący sieć oscylatorów rejestrujących interakcje przestrzenne i czasowe między różnymi punktami w sieci drogowej. Aby adaptacyjnie radzić sobie z nieoczekiwanymi zakłóceniami w przepływie ruchu, OBELISC

został rozszerzony o kontroler o trybie ślizgowym, który wzmacnia jego możliwości adaptacyjne w kierunku globalnego konsensusu w przypadku zakłóceń o dużej skali.

System został wdrożony jako lekki system uczący się, który wykorzystuje interakcje sprzężeń między różnymi kontrolowanymi oscylatorami. Nasza obszerna ocena systemu w oparciu o dane rzeczywiste i najnowocześniejsze metody pokazuje zalety OBELISC. Od uchwycenia okresowej dynamiki fazowania ruchu drogowego, przez osadzanie korelacji przestrzennej między przepływem ruchu w jego wymiarach czasowych, aż po solidną adaptację do nieoczekiwanych zakłóceń w ruchu, OBELISC wyróżnia się jako elastyczne rozwiązanie do obliczania czasu trwania fazy. Wreszcie, korzystając z efektywnego uczenia się i obliczeń w impulsowych sieciach neuronowych, OBELISC jest solidnym kandydatem do rzeczywistego wdrożenia w świecie rzeczywistym.