

POLSKIE TOWARZYSTWO MATEMATYCZNE

Założone w 1919 roku

Członek-założyciel Europejskiego Towarzystwa Matematycznego

ODDZIAŁ W KRAKOWIE

SEMINARIUM ZASTOSOWAŃ MATEMATYKI

**W dniu 7 kwietnia 2020 (WTOREK) o godz. 17.00,
w pawilonie B-1, I p., w sali H 24,
Wydział EAIiB AGH-Kraków, Al. Mickiewicza 30**

Mgr inż. Wojciech Turlej

Doktorant wdrożeniowy Szkoły Doktorskiej AGH oraz firmy Aptiv.

wygłosi odczyt pt.

**WYKORZYSTANIE METOD
OPTYMALIZACJI W PLANOWANIU
TRAJEKTORII ROBOTÓW MOBILNYCH
ORAZ SAMOCHODÓW OSOBOWYCH**

Zapraszamy

*W. Mitkowski-
Przew. Kom. Zastosowań
oraz P. Skruch*

Wykorzystanie metod optymalizacji w planowaniu trajektorii robotów mobilnych oraz samochodów osobowych

Wojciech Turlej

Plan prezentacji

1. Wprowadzenie, pojęcia podstawowe, przykład prostego problemu generacji trajektorii.
2. Przykłady metod optymalizacyjnych.
3. Wyzwania w formułowaniu problemów optymalizacji dla generacji trajektorii.
4. Zastosowania metod optymalizacji w zaawansowanych systemach wspomagania kierowcy i ich testowaniu.

Wprowadzenie - optymalizacja

Zadanie optymalizacji polega na poszukiwaniu ekstremum zadanej funkcji celu.

Istnieje wiele kryteriów według których można podzielić problemy optymalizacyjne. Na potrzeby tej prezentacji skupimy się na dwóch podstawowych podziałach.

Ze względu na ciągłość funkcji celu, problemy optymalizacyjne możemy podzielić na:

- ciągłe,
- dyskretne,
- hybrydowe.

Ze względu na występowanie ograniczeń możemy wyróżnić problemy:

- z ograniczeniami,
- bez ograniczeń.

W zagadnieniach generacji trajektorii na potrzeby robotyki/pojazdów autonomicznych, najczęściej używa się problemów ciągłych z ograniczeniami. Generowana trajektoria opisana jest ciągłymi parametrami, natomiast ograniczenia wykorzystywane są typowo do wyrażenia fizycznych limitów systemu, na przykład związanych z mocą napędów bądź z maksymalnymi położeniami kontrolowanych stopni swobody systemu.

Wprowadzenie – generacja trajektorii

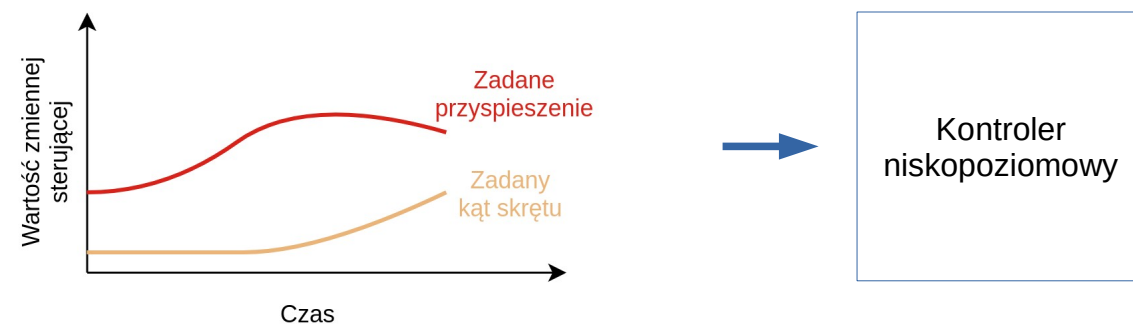
Dla systemów w robotyce i automatyce pojazdowej najczęściej generowana jest trajektoria stanu bądź trajektoria sterowania robotem/pojazdem.

Trajektoria stanu opisuje przebieg zmiennych stanu obiektu (na przykład pozycja pojazdu w przestrzeni lub konfiguracja robota) w określonym czasie

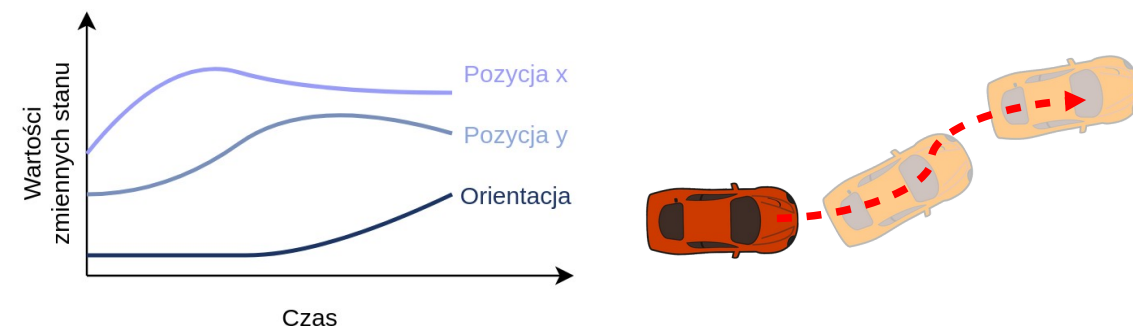
Trajektoria sterowania opisuje przebieg sygnałów sterujących pozwalających na kontrolowanie obiektu – przykładem może być zadany kąt skrętu kierownicy pojazdu lub zadany moment aktuatorów robota.

Trajektorie najczęściej opisywane są przy użyciu krzywych B-sklejanych, krzywych Béziera lub wielomianów określających zmianę wartości danych zmiennych w czasie.

Trajektoria sterowania:



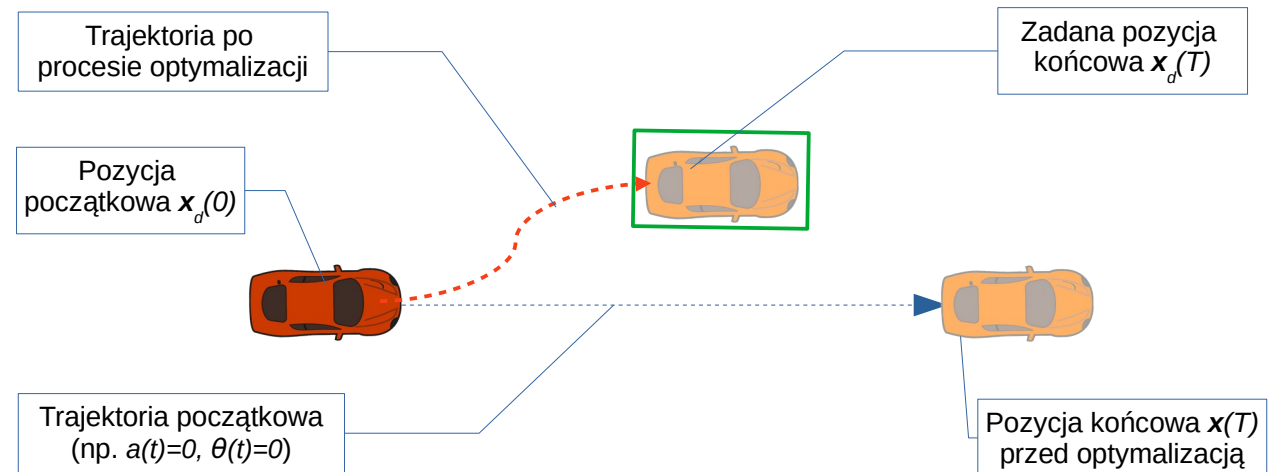
Trajektoria stanu:



Wprowadzenie – przykład problemu generacji trajektorii

Prostym przykładem omawianego problemu może być problem generacji trajektorii pojazdu z zadanym położeniem końcowym. Problemy tego typu wykorzystuje się w układach sterowania predykcyjnego, na przykład w systemach wspomaganie kierowcy utrzymujących pozycję lateralną w pasie ruchu lub odległość od pojazdu za którym sterowany pojazd podąża.

Na potrzeby przykładu możemy założyć, że dana jest pożądana pozycja końcowa pojazdu w kartezjańskim układzie współrzędnych: $\mathbf{x}(t_f)=[x_0, x_1]$ gdzie t_f jest czasem trwania generowanej trajektorii a zmiennymi sterowania są zadane przyspieszenie $a(t)$ i zadany kąt skrętu kierownicy pojazdu $\theta(t)$.



Wprowadzenie – przykład problemu generacji trajektorii

W omawianym przykładzie możemy założyć, że optymalizowanymi wartościami będzie wektor parametrów \mathbf{p} składający się z dwóch zestawów parametrów – jednego opisującego wielomian definiujący przebieg wartości zadanych kątu skrętu kierownicy pojazdu oraz drugiego opisującego przebieg zadanego przyspieszenia pojazdu.

Tak sformułowany problem pozwala na wygodną kontrolę nad własnościami generowanej trajektorii poprzez dobór składników kosztu c oraz wartości ich wag.

Składniki kosztu mogą uwzględniać:

- Komfort pasażerów - przeciążenia, pochodne przyspieszeń;
- Efektywność – zużycie paliwa, szybkość osiągnięcia celu, płynność jazdy;
- Margines bezpieczeństwa – odległość do innych pojazdów, czas do kolizji (TTC – Time To Collision)

$$\min_{\mathbf{p}} f(\mathbf{p}) = \sum_{t=0}^{T/\Delta t} \sum_{i=0}^k w_i * c_i(t) \Delta t \quad (4)$$

przy ograniczeniach: $\mathbf{h}(\mathbf{p}) \geq \mathbf{0} \quad (5)$

$$\mathbf{e}(\mathbf{p}) = \mathbf{0} \quad (6)$$

gdzie: $f(\mathbf{p})$ - koszt; $c_i(t)$ dla $i = 1..k$ - wartość minimalizowanych składników funkcji kosztu w chwili czasowej t ; w_i dla $i = 1..k$ - wagi składników funkcji kosztu; T - czas trwania generowanej trajektorii; Δt - odstęp między chwilami czasowymi w których ewaluowana jest trajektoria.

$\mathbf{h}(\mathbf{p})$ - ograniczenia nierównościowe, np. maksymalny kąt skrętu kierownicy i maksymalne przyspieszenie, ewaluowane w chwilach czasowych t

$\mathbf{e}(\mathbf{p})$ - ograniczenia równościowe, np. definicja celu jako odległość euklidesowa pomiędzy pozycją końcową pojazdu $\mathbf{x}(T)$ a zadaną pozycją końcową \mathbf{x}_d .

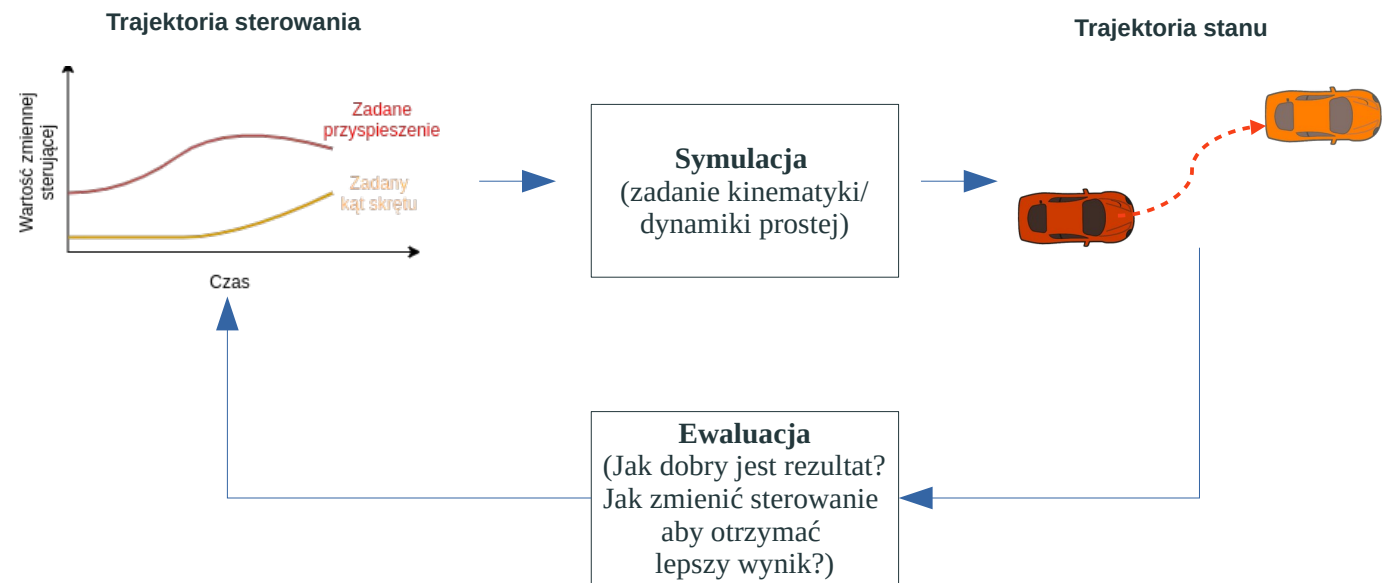
Przykład 1 – Prosty problem optymalizacji trajektorii pojazdu

Wprowadzenie – przykład problemu generacji trajektorii

Przedstawiony problem jest przykładem generacji trajektorii tzw. metodą *single shooting*, w której trajektoria stanu jest uzyskiwana na drodze symulacji obiektu poddanego danemu przebiegowi optymalizowanych zmiennych sterujących.

Jest to metoda prosta w implementacji i pozwala na łatwe narzucenie ograniczeń na zmienne sterujące, jednak przy obiektach o skomplikowanym, nieliniowym modelu może powodować trudności związane z narzucaniem ograniczeń na trajektorię stanu.

W dalszej części prezentacji zostaną omówione alternatywne metody stosowane powszechnie w generacji trajektorii.



Przykład 1 – Prosty problem optymalizacji trajektorii pojazdu – metoda *single shooting*

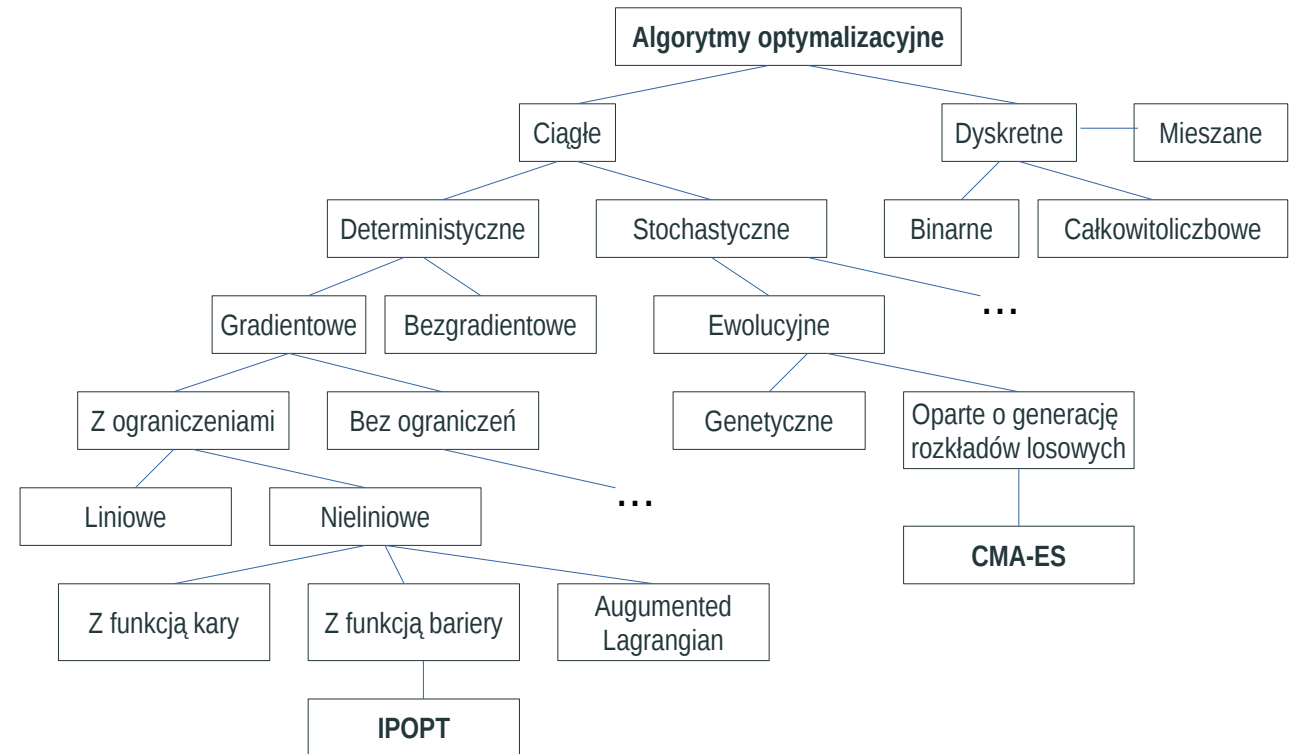
Algorytmy optymalizacyjne

Algorytmy optymalizacyjne

Rozwiązywanie problemów optymalizacyjnych jest niezwykle szerokim tematem, a sama próba klasyfikacji istniejących algorytmów jest trudnym wyzwaniem.

Ze względu na złożoność tematu ograniczymy się do krótkiego omówienia dwóch przykładów skrajnie różnych algorytmów znajdujących szerokie zastosowanie w generacji trajektorii:

- **IPOPT** – deterministyczny algorytm gradientowy dla ciągłych problemów nieliniowych z potencjalnie nieliniowymi ograniczeniami;
- **CMA-ES** – stochastyczny algorytm ewolucyjny, szczególnie skuteczny w globalnej optymalizacji problemów o dużej ilości wymiarów z wieloma lokalnymi optimami.



Fragment przykładowej taksonomii algorytmów optymalizacji

IPOPT jako przykład algorytmu gradientowego

IPOPT jest algorytmem optymalizacyjnym zaimplementowanym przez A. Wächtera [1] wykorzystującym algorytm gradientowy *L-BFGS* [2] (quasi-Newtonowska metoda optymalizacji nieliniowej bez ograniczeń z aproksymacją hesjanu) w połączeniu z metodą *interior-point* [3] wykorzystującą logarytmiczne funkcje bariery do narzucenia ograniczeń problemu optymalizacji.

[1] A. Wächter and L. T. Biegler, On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming, *Mathematical Programming* 106(1), pp. 25-57, 2006

[2] Fletcher, Roger (1987), *Practical methods of optimization* (2nd ed.), New York: John Wiley & Sons, ISBN 978-0-471-91547-8

[3] Wright, Margaret H. (2004). "The interior-point revolution in optimization: History, recent developments, and lasting consequences". *Bulletin of the American Mathematical Society*. 42: 39–57

Zalety:

- Wysoka skuteczność w wielkoskalowych problemach nieliniowych z ograniczeniami
- Możliwość zdefiniowania ograniczeń nieliniowych

Wady:

- Konieczność obliczenia gradientu funkcji celu i jakobianu ograniczeń (możliwie czasochłonne obliczenia)
- Podatność na lokalne optima
- Niska skuteczność przy zaszumionej funkcji celu/ograniczeniach

IPOPT – Obliczanie gradientu funkcji celu i jacobianu ograniczeń

Gradient funkcji celu jest wektorem wskazującym jaki wpływ ma każda z optymalizowanych zmiennych na funkcję celu w ewaluowanym punkcie. Jacobian ograniczeń jest natomiast macierzą wskazującą jaki wpływ ma każda ze zmiennych na każde ze zdefiniowanych ograniczeń problemu optymalizacji.

Dobór sposobu obliczania gradientu i jacobianu zależy od problemu optymalizacyjnego, dostępnych zasobów oraz możliwości pakietów obliczeniowych używanych do definicji problemu.

Ze względu na dużą ilość zmiennych często występującą w problemach generacji trajektorii, dobór metody wyznaczania gradientów i jacobianów może mieć kluczowy wpływ na szybkość obliczeń.

Podstawowe metody wyznaczania gradientu i jacobianu:

- **Analityczna** – umożliwia uzyskanie dokładnych wyników, ale wymaga analitycznego wyznaczenia równań gradientu i jacobianu, co zwykle nie jest możliwe przy bardziej złożonych problemach.
- **Numeryczna** – dla n-wymiarowego problemu polega na n-krotnym obliczeniu funkcji celu i ograniczeń przy dodaniu niewielkiej perturbacji do każdej z n optymalizowanych zmiennych. Prosta w implementacji ale wymagająca obliczeniowo i pozwala jedynie na aproksymację gradientu i jacobianu .
- **Automatyczne różniczkowanie** – metoda oparta o nadpisanie podstawowych operatorów matematycznych w danym języku programowania w taki sposób, aby pozwalały na akumulację wartości pochodnych zmiennych na których wykonywane są obliczenia przy użyciu twierdzenia o pochodnej funkcji złożonej. Metoda dokładna i relatywnie szybka, ale wymaga użycia odpowiedniego pakietu matematycznego lub złożonej implementacji.

CMA-ES jako przykład algorytmu ewolucyjnego

Algorytm CMA-ES (Covariance Matrix Adaptation – Evolution Strategy), czyli Strategia Ewolucji z Adaptacją Macierzy Kowariancji jest algorytmem ewolucyjnym polegającym na iteracyjnym generowaniu zbioru rozwiązań według wielowymiarowego rozkładu normalnego opisanego macierzą kowariancji Σ i wartością średnią μ . W każdej iteracji wygenerowane rozwiązania są oceniane funkcją kosztu, co pozwala na modyfikację μ i Σ w taki sposób, aby zbiór rozwiązań wygenerowany w kolejnej iteracji miał większą szansę zawierać rozwiązanie bliższe globalnemu optimum.

Ze względu na to, że generowany zbiór rozwiązań może być mniejszy od ilości zmiennych optymalizowanych, metoda dobrze się sprawdza w wielkoskalowych, nieliniowych problemach o zaszumionych gradientach funkcji celu.

Zalety:

- Wysoka skuteczność w problemach silnie nieliniowych, z wieloma optimumami lokalnymi
- Metoda bezgradientowa – pozwala na szybkie iteracje w wielkoskalowych problemach i wysoką skuteczność przy zaszumionej funkcji celu
- Stosunkowo mała ilość wywołań funkcji celu – pozwala na zachowanie szybkości przy funkcjach celu złożonych obliczeniowo

Wady:

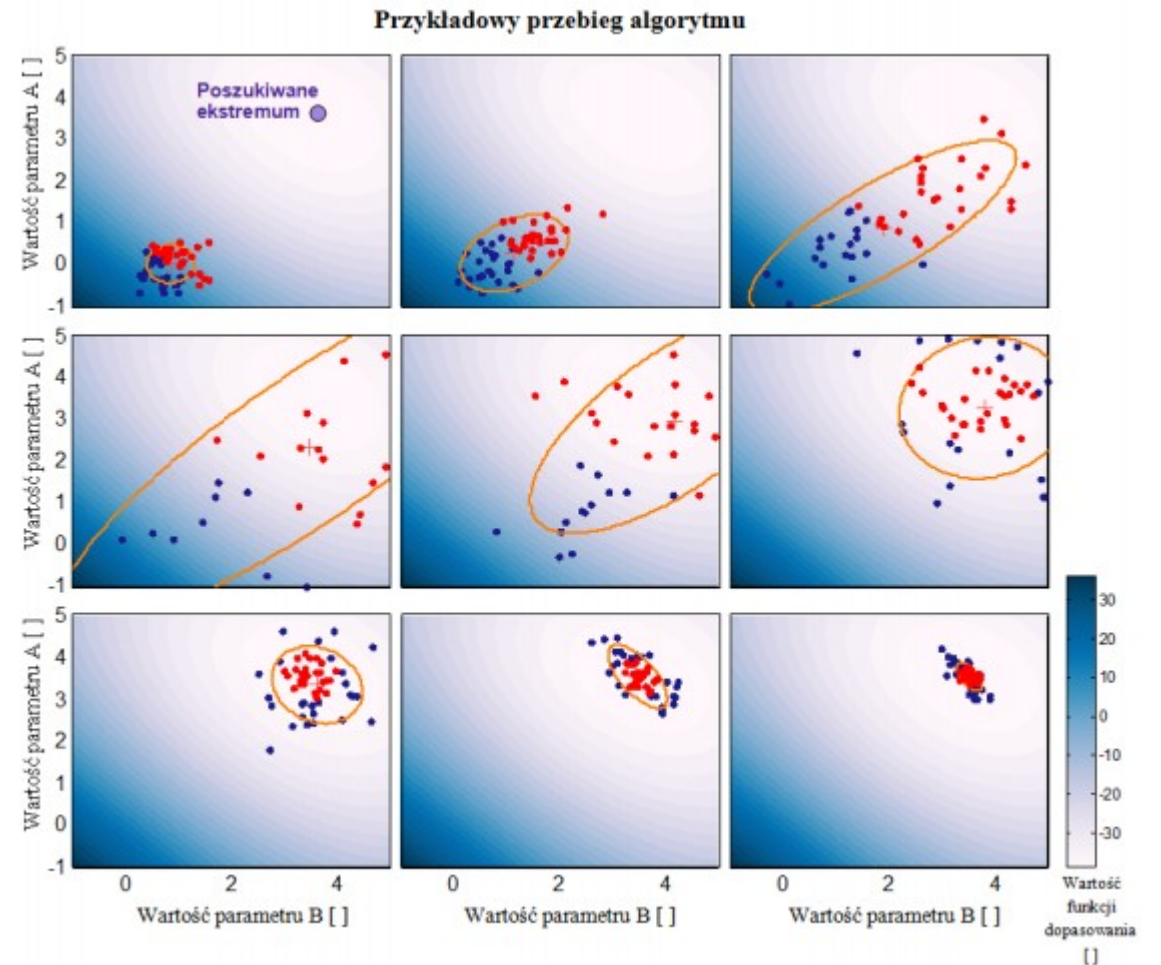
- Nie uwzględnia ograniczeń
- Często wolniejszy od metod gradientowych w problemach o gładkiej funkcji celu

CMA-ES jako przykład algorytmu ewolucyjnego

Załączony diagram przedstawia przykładowy przebieg procesu optymalizacji funkcji dwóch zmiennych przy użyciu metody CMA-ES.

W pierwszych trzech krokach wartość średnia najlepszych rozwiązań (zaznaczonych na czerwono) jest silnie oddalona od wartości średniej rozkładu, co skutkuje „rozciągnięciem” generowanego rozkładu w kierunku najlepszych obserwacji.

W kolejnych krokach najlepsze rozwiązania są już zawarte bliżej centrum generowanego rozkładu, co skutkuje „zawężeniem” przestrzeni poszukiwań poprzez odpowiednią modyfikację macierzy kowariancji.

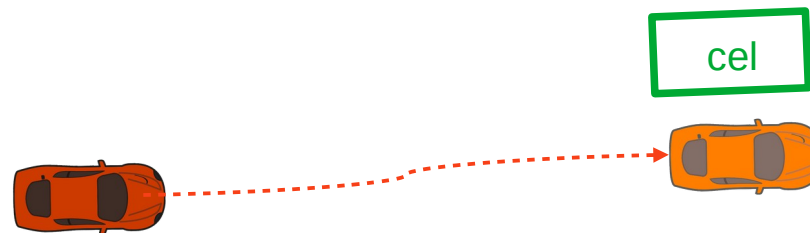


Wyzwania w formułowaniu problemów generacji trajektorii

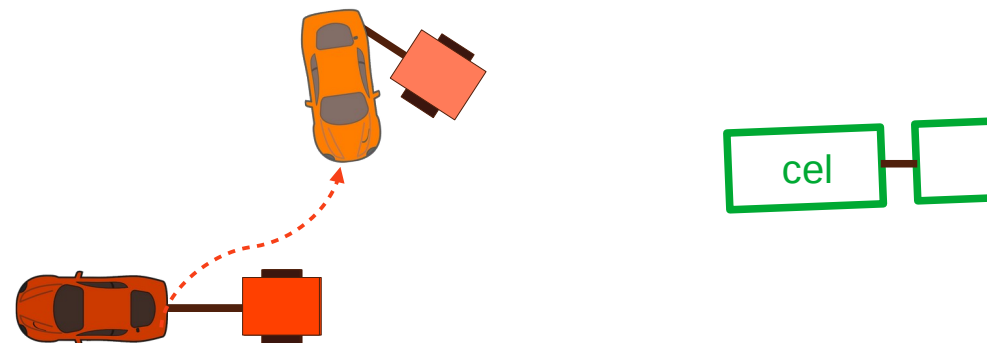
Problem 1 – obiekty nieliniowe

W przypadku silnie nieliniowych funkcji celu i/lub ograniczeń, użyteczność gradientu i jacobianu funkcji do wyznaczenia rozwiązania drastycznie spada. Jest to problem szczególnie widoczny w obiektach nieliniowych takich jak wahadła odwrócone, roboty kroczące, pojazdy wykonujące manewr cofania lub pojazdy z przyczepami.

Dla przykładu jeśli generujemy trajektorię robota kroczącego metodą *single shooting*, w której trajektoria początkowa (*initial guess*) kończy się jego przewróceniem na samym początku symulacji, jacobian ograniczenia jego pozycji końcowej nie wnosi użytecznych informacji do procesu optymalizacji (nie możemy z niego wnioskować co zrobić, aby robot „przewrócił się mniej”).



Przykład 1a: w tym przypadku możemy na podstawie jacobianu ograniczenia celu wnioskować, że lekkie zwiększenie kąta skrętu kierownicy przybliży nas do celu.

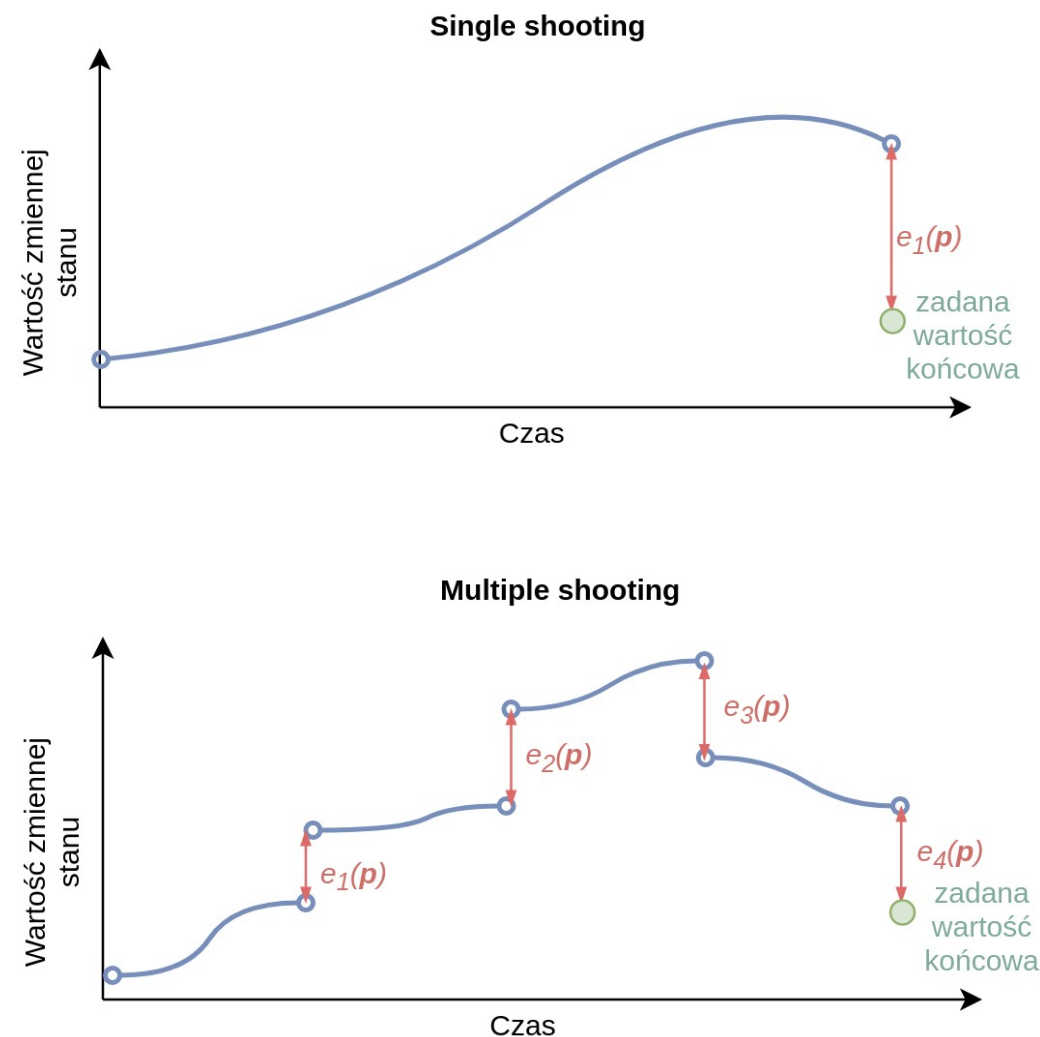


Przykład 1b: samochód cofający z przyczepą – w tym przypadku przy rozwiązaniu początkowym dalekim od celu jacobian ograniczeń nie dostarcza użytecznej informacji na temat potrzebnych modyfikacji optymalizowanych zmiennych.

Rozwiązanie problemu 1 - metoda multiple shooting

Powszechnie stosowanym rozwiązaniem w problemach nieliniowych jest użycie metody *multiple shooting*. W metodzie tej generowana trajektoria podzielona jest na wiele krótszych odcinków.

Aby zapewnić że na końcu procesu optymalizacji generowane odcinki utworzą ciągłą trajektorię, wprowadzane są dodatkowe ograniczenia równościowe ($e_1(p) - e_3(p)$ na dolnym wykresie) wymuszające ciągłość trajektorii. W zależności od rozwiązywanego problemu, możliwe jest narzucenie analogicznych ograniczeń na pochodne łączonych odcinków w punktach łączenia.

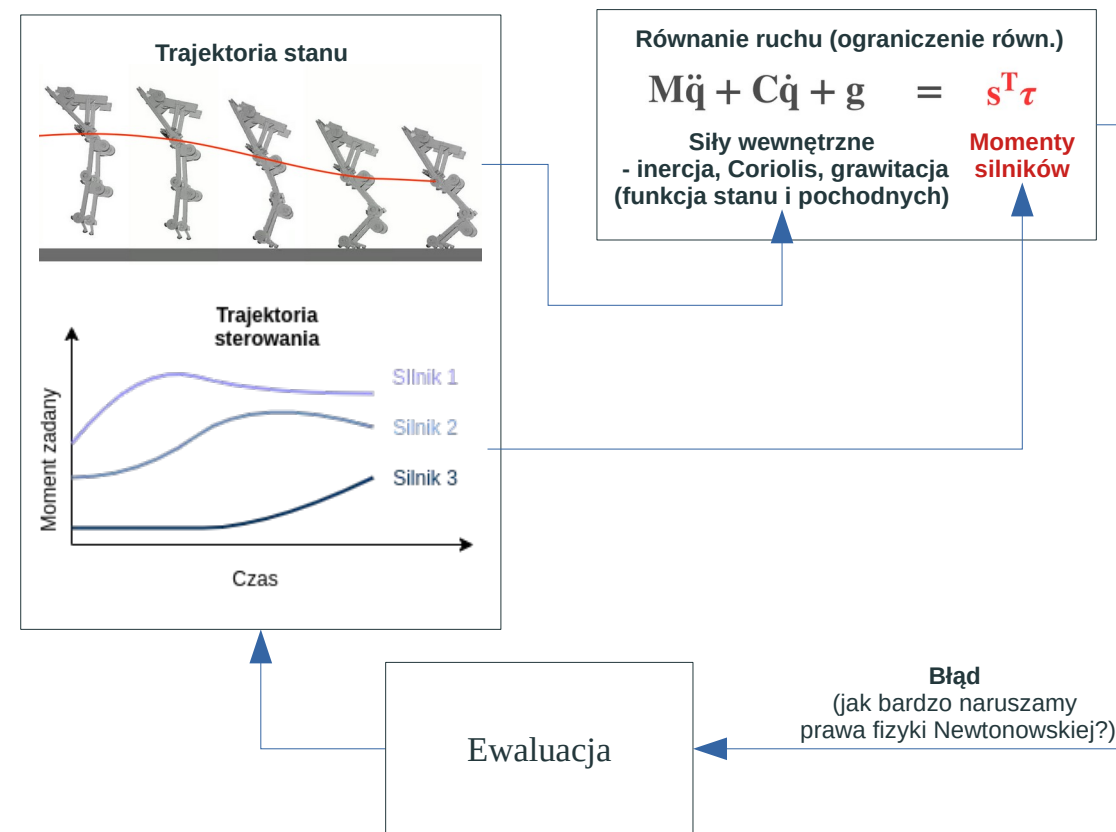


Rozwiązanie problemu 1 - metoda kolokacji

Alternatywną metodą stosowaną w generacji trajektorii dla obiektów silnie nieliniowych jest *metoda kolokacji*. W metodzie tej trajektorie stanu oraz sterowania generowane są **jednocześnie**, a więc wektor optymalizowanych parametrów p opisuje dwa zestawy trajektorii.

Aby zagwarantować, że wygenerowana trajektoria sterowania skutkować będzie przebiegiem stanu opisanym przez wygenerowaną trajektorię stanu, wprowadzany jest zestaw dodatkowych ograniczeń równościowych opartych o równanie stanu obiektu (np. równanie ruchu robota), ewaluowane w zadanych punktach czasowych.

O ile ilość wymiarów problemu oraz ilość ograniczeń równościowych drastycznie wzrasta w tej metodzie, problem z reguły jest znacznie prostszy do rozwiązania ze względu na uproszczone, słabiej skorelowane zależności między parametrami p , a wartościami ograniczeń.



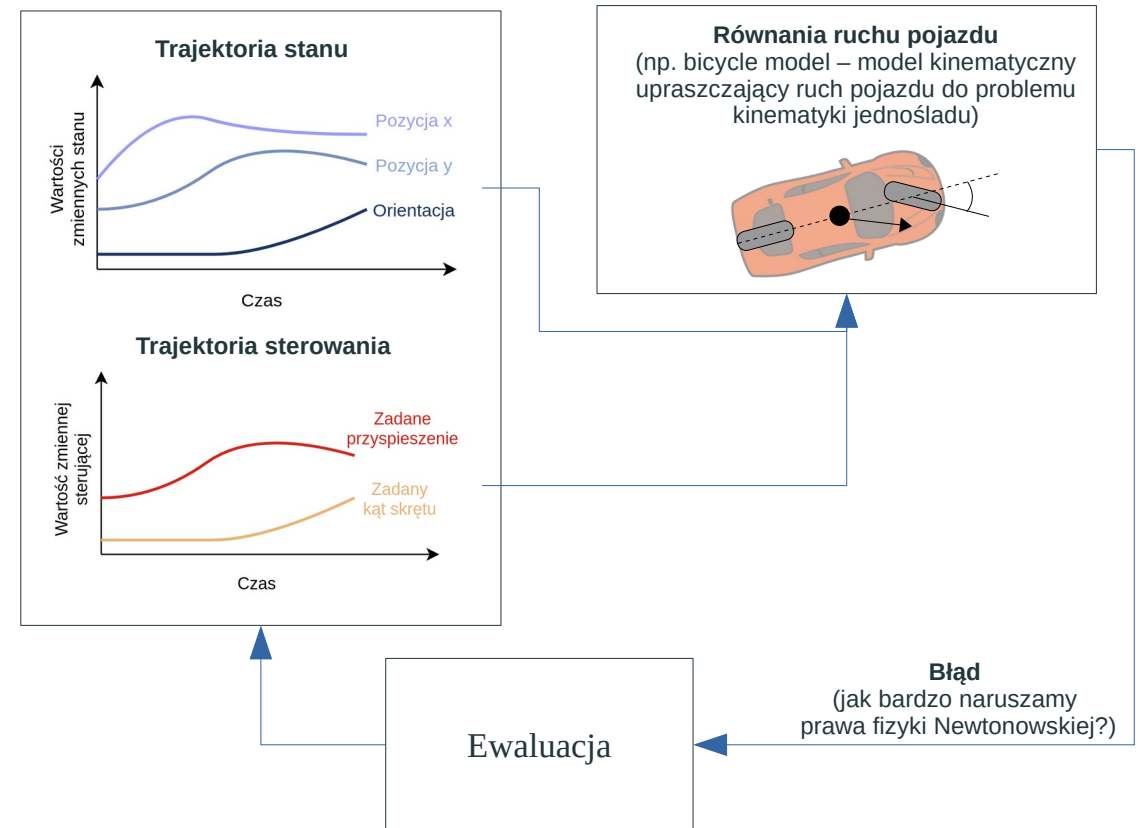
Przykład 1c: metoda kolokacji w robotyce. Ewaluacja błędu równania ruchu w równych odstępach czasowych pozwala na zbudowanie zestawu ograniczeń równościowych gwarantujących fizyczną realizowalność trajektorii.

Rozwiązanie problemu 1 - metoda kolokacji

Dodatkową zaletą metody kolokacji jest jej skuteczność w problemach z ograniczeniami zmiennych stanu i sterowania.

W większości problemów generacji trajektorii musimy narzucić ograniczenia zarówno na zmienne sterowania (np. maksymalny kąt skrętu kierownicy, maksymalne przyspieszenie), jak i zmienne stanu (np. ograniczenie pozycji pojazdu do brzegów pasa ruchu, ograniczenie równościowe celu).

W przypadku metody kolokacji, zależności pomiędzy optymalizowanymi parametrami p a wartościami zmiennych sterowania i, w szczególności, stanu są bardzo proste i słabo skorelowane, co bardzo ułatwia znalezienie rozwiązania problemu. W przypadku metod *single* i *multiple shooting*, zależność między zmiennymi stanu a parametrami p może być silnie nieliniowa, co poważnie utrudnia rozwiązanie problemu.



Przykład 1d: Metoda kolokacji w generacji trajektorii pojazdu.

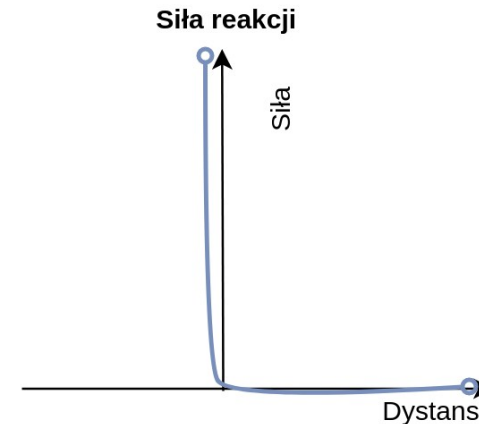
Problem 2 – skokowe zmiany dynamiki obiektu

Problemem często występującym przy generacji trajektorii dla złożonych obiektów jest występowanie silnie nieliniowych zjawisk, takich jak zmienna konfiguracja kontaktu obiektu z podłożem, czy poślizgi.

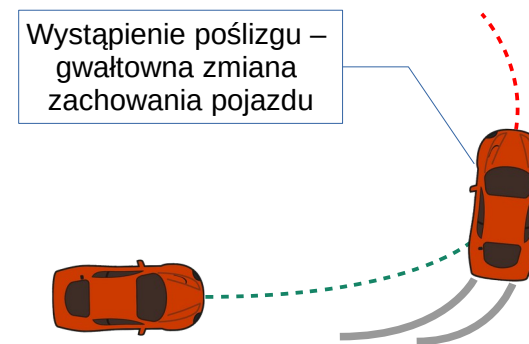
Zachowanie obiektu i jego reakcja na zmienne sterujące z reguły nie zmienia się znacząco dopóki konfiguracja (ilość/typ kontaktów, występowanie poślizgu) pozostaje niezmienną. W momencie jej zmiany konieczna jest jednak drastyczna modyfikacja wartości sterowania.

Wspomniane skokowe zmiany dynamiki mogą wprowadzać bifurkacje do systemu – na przykład drobna zmiana momentu w czasie w którym następuje zmiana konfiguracji kontaktów może doprowadzić do skrajnie różnego wyniku końcowego.

Uwzględnienie zmiennej dynamiki systemu sprawia szczególną trudność w metodzie kolokacji – przy braku typowej symulacji wprowadzenie zmian konfiguracji do równań ograniczeń może stanowić poważną trudność.



Przykład 2a – wystąpienie lub zanik kontaktu między obiektem sterowanym a podłożem lub innym obiektem. Siła reakcji może wzrosnąć od zera do tysięcy N w ułamku sekundy wraz z przemieszczeniem rzędu setnych części milimetra, powodując znaczącą zmianę zachowania obiektu.



Przykład 2b – tranzycja pomiędzy tarcie statycznym a dynamicznym (np. utrata lub odzyskanie przyczepności pojazdu) skutkuje skokową zmianą sposobu reakcji obiektu na zmienne sterujące.

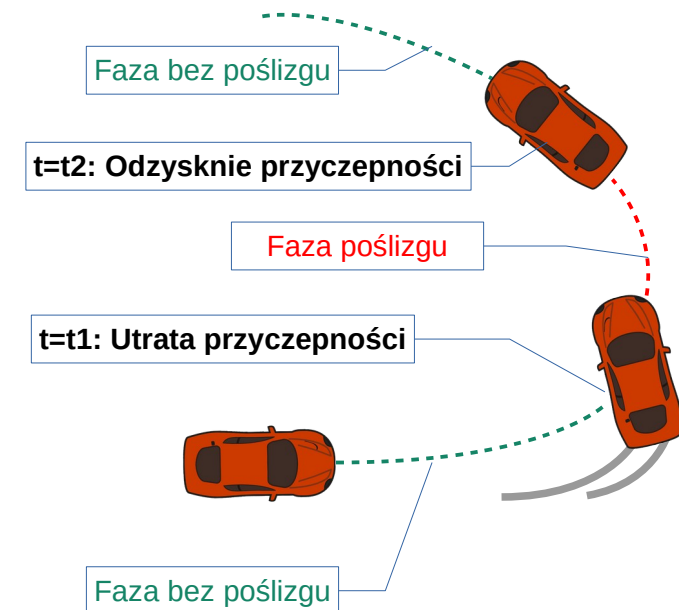
Rozwiązanie problemu 2 – predefiniowane fazy ruchu

Jednym z rozwiązań umożliwiających planowanie ruchu z występowaniem faz o zmiennej dynamice jest podzielenie trajektorii na krótsze odcinki, w czasie trwania których konfiguracja pozostaje niezmienna.

Przykładowo dla pojazdu na śliskiej powierzchni możemy podzielić planowany ruch na fazy w których występuje poślizg oraz fazy w których przyczepność jest zachowana. Każda z faz może być opisana odmiennymi równaniami, a ciągłość trajektorii może zostać wymuszona przez dodatkowe ograniczenia równościowe, podobnie jak w metodzie *multiple shooting*.

Czas trwania poszczególnych faz może zostać uwzględniony w problemie optymalizacji poprzez dodanie momentów czasowych zmiany konfiguracji (t_1 , t_2) do zestawu parametrów p .

Dodatkowe ograniczenia nierównościowe wymuszają niezmiennosc konfiguracji w czasie trwania danej fazy – na przykład w czasie fazy poślizgu, stosunek stycznej składowej siły kontaktu z podłożem do składowej normalnej musi być wystarczająco wysoki, aby utrzymać występowanie tarcia dynamicznego.

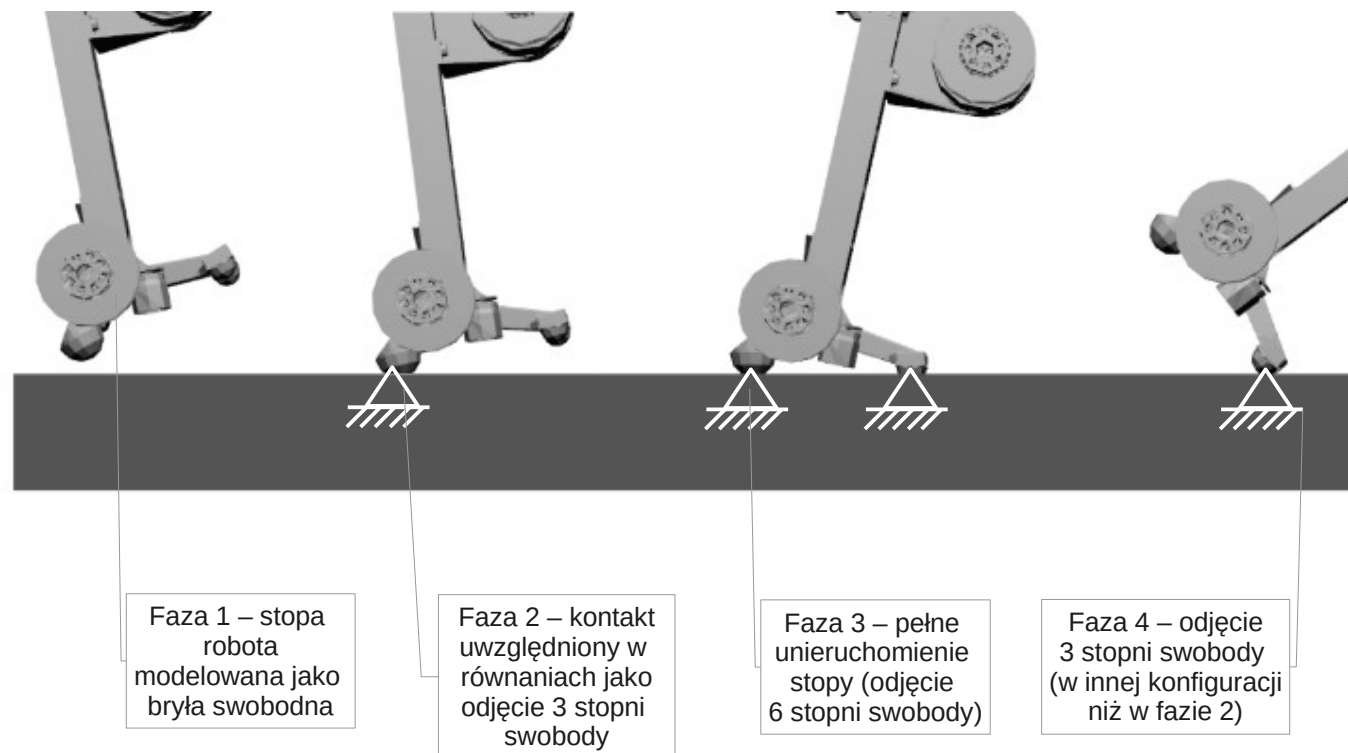


Przykład 2b: predefiniowane fazy poślizgu i przyczepności

Rozwiązanie problemu 2 – predefiniowane fazy ruchu

Opisane rozwiązanie z predefiniowanymi fazami jest proste i skuteczne w przypadku małej ilości faz ruchu, których ilość jest znana. W wielu przypadkach określenie optymalnej ilości i kolejności faz nie jest jednak możliwe.

Dobrym przykładem są roboty kroczące – w ich przypadku, każdy rodzaj kontaktu z podłożem stanowi inną fazę. Predefiniując kolejność występowania faz narzucamy ilość kroków jakie robot musi wykonać oraz sposób ich wykonywania (np. chód vs bieg dla robotów dwunożnych), potencjalnie wymuszając suboptymalne rozwiązanie problemu.



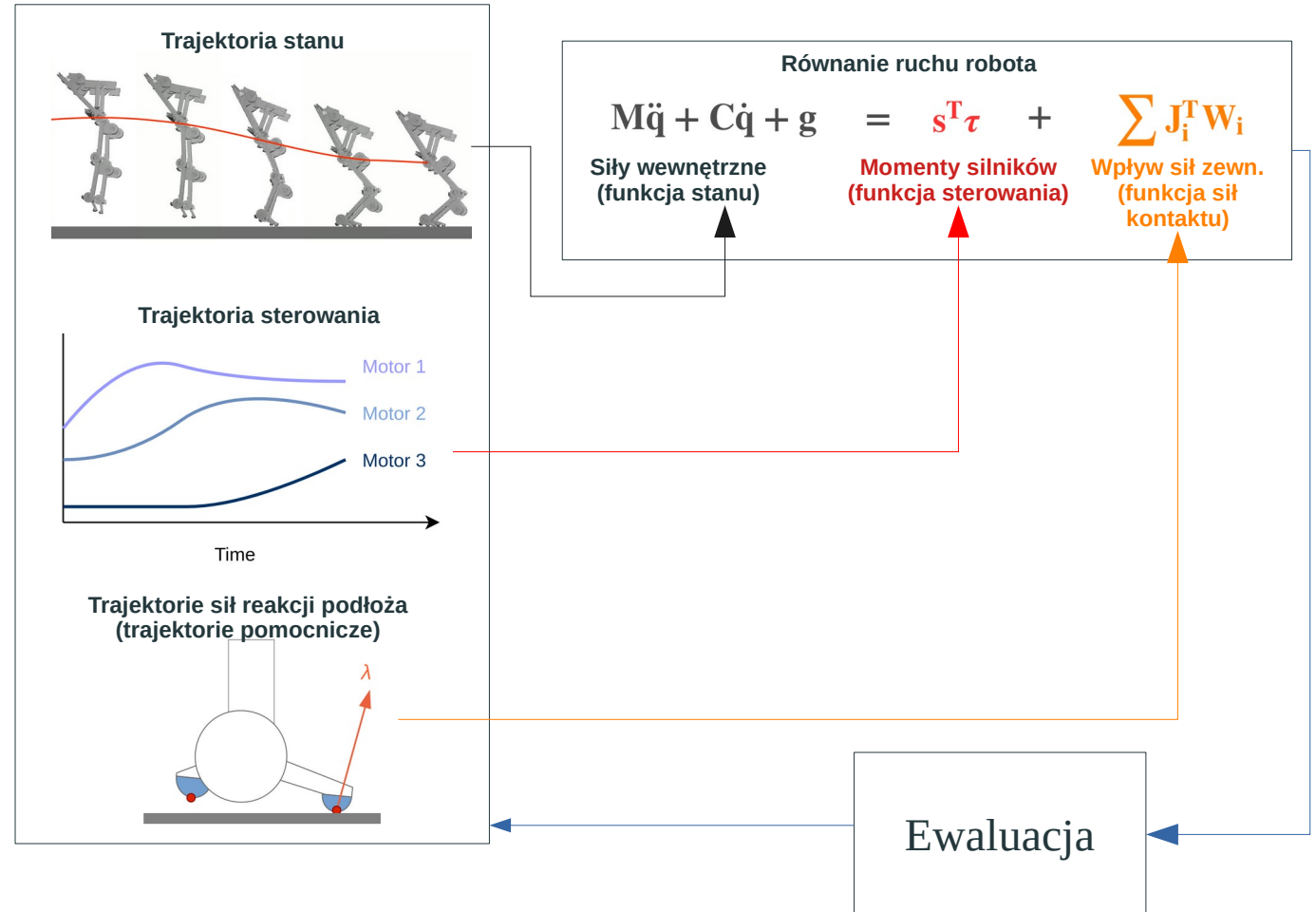
Przykład 2c: Modelowanie kontaktu z podłożem przy użyciu modeli o różnej ilości stopni swobody.

Rozwiązanie problemu 2 – rozszerzenie modelu i optymalizacja trajektorii pomocniczych

Alternatywnym rozwiązaniem problemu faz o zmiennej dynamice jest sformułowanie problemu w sposób, który opisuje zachowanie obiektu we wszystkich fazach przy pomocy jednego zestawu równań.

W przypadku występowania w modelu zmiennych silnie wrażliwych na drobne zmiany trajektorii stanu i/lub sterowania, dobre efekty może przynieść opisanie ich trajektoriami pomocniczymi, których parametry zostaną włączone do optymalizowanego zestawu parametrów p .

Przedstawimy to podejście na podstawie sił kontaktu, które ulegają drastycznym zmianom przy niewielkich modyfikacjach trajektorii stanu, będąc częstym źródłem bifurkacji modelu.



Rozwiązanie problemu 2 – rozszerzenie modelu i optymalizacja trajektorii pomocniczych

Dla zapewnienia, że generowane trajektorie pomocnicze są zgodne z prawami fizyki, należy wprowadzić dodatkowe ograniczenia problemu optymalizacji, które zapobiegą nierealistycznym zachowaniom – takim jak przykładanie niezerowej siły reakcji, kiedy kontakt nie występuje.

Przedstawiony problem optymalizacyjny z pozoru wydaje się być trudniejszy – ilość ograniczeń oraz wymiarów problemu jest drastycznie większa.

W praktyce jednak jacobian ograniczeń w tak postawionym problemie jest macierzą rzadką, zredukowane są drastycznie nieliniowe zachowania modelu oraz korelacje między ograniczeniami. Wszystkie te własności pozwalają na relatywnie szybką generację trajektorii nawet dla bardzo skomplikowanych, z pozoru chaotycznych systemów.

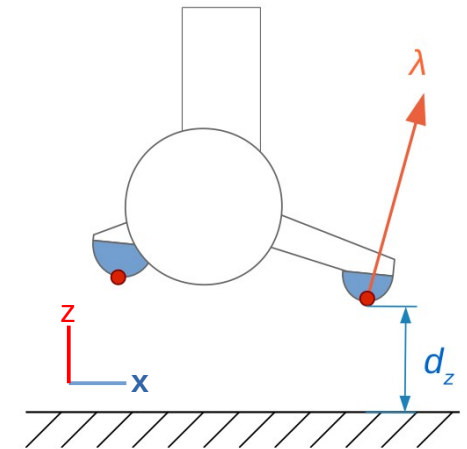
Dodatkowe ograniczenia:

$$\mathbf{d}_z(\mathbf{q}) \geq \mathbf{0} \quad (\text{zapobiega penetracji podłoża})$$

$$\lambda_z \geq \mathbf{0} \quad (\text{zapobiega sile reakcji skierowanej w podłoże})$$

$$|\lambda_x| \leq \lambda_z \sigma \quad (\text{ogranicza siły styczne})$$

$$\lambda_z \mathbf{d}_z + \lambda_z |\dot{\mathbf{d}}_x| = \mathbf{0} \quad (\text{zapobiega występowaniu siły kiedy nie ma kontaktu z podłożem, wyklucza ruch styczny do podłoża kiedy kontakt występuje})$$



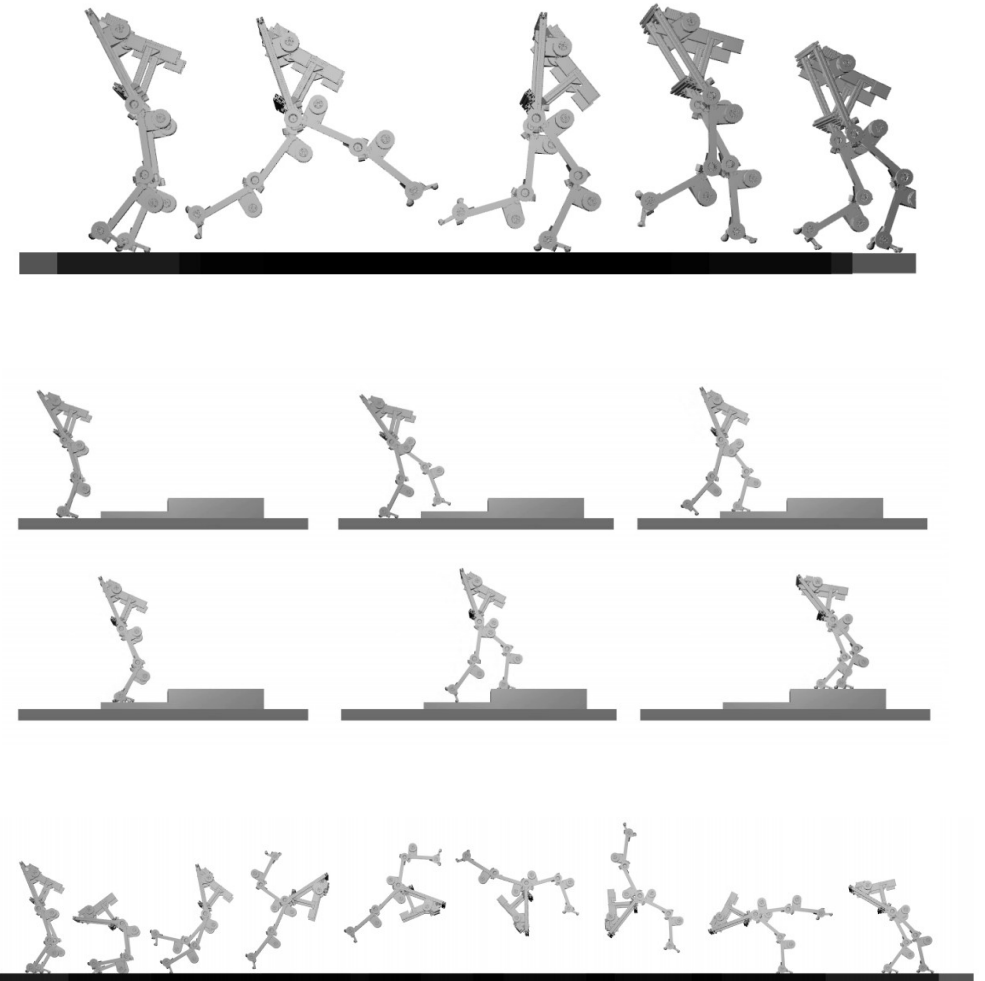
Przykłady

Użycie opisanych metod pozwala na wygenerowanie efektywnych trajektorii nawet dla silnie nieliniowych obiektów skłonnych do chaotycznych zachowań.

Po prawej zaprezentowane zostały przykłady wykorzystania wspomnianych metod w dziedzinie robotyki. Przedstawiają one trajektorie wygenerowane dla robota kroczącego z wiskoelastycznymi elementami napędowymi. Mimo minimalistycznych trajektorii początkowych (w większości przypadków statyczna poza robota z zerowymi siłami), algorytm IPOPT był w stanie wygenerować fizycznie realizowalne trajektorie realizujące złożone cele takie jak dynamiczny chód robota, bieg, skoki, wchodzenie na przeszkody, czy nawet salta [1-2].

[1] Wojciech Turlej, Optimization-based trajectory generation for compliant bipedal robots, master thesis (2017)

[2] Werner, Alexander, Wojciech Turlej, and Christian Ott. "Generation of locomotion trajectories for series elastic and viscoelastic bipedal robots." 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2017.



Przykłady: użycie opisanych metod do wygenerowania dynamicznych ruchów robota kroczącego. Od góry: bieg, wchodzenie na przeszkody, salta. Przykłady wideo dostępne na stronie: https://youtu.be/K_dTCOosIYk

Wyzwania w formułowaniu problemów generacji trajektorii

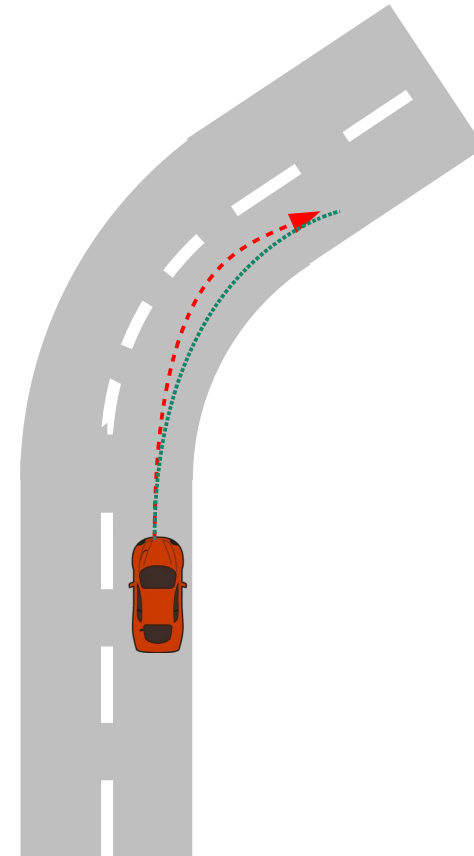
Sterowanie predykcyjne – realizacja zadanych trajektorii

Jednym z najprostszych i najpowszechniejszych zastosowań metod optymalizacyjnych w zaawansowanych systemach wspomagania kierowcy jest wykorzystanie sterowania predykcyjnego do realizacji zadanych trajektorii ruchu pojazdu.

Sterowanie predykcyjne polega na powtarzanym periodycznie rozwiązywaniu problemu generacji trajektorii sterowania, która skutkować będzie bliskim nadążaniem za zadaną trajektorią stanu.

W problemie takim minimalizowana jest różnica pomiędzy trajektorią zadaną a otrzymaną, a ograniczenia problemu optymalizacji odwzorowują ograniczenia fizyczne systemu wynikające z osiągnięć danego pojazdu, jego geometrii i innych parametrów.

Metoda ta jest powszechnie wykorzystywana w niskopoziomowych kontrolerach współpracujących z zaawansowanymi algorytmami generującymi pożądaną ścieżkę ruchu pojazdu.



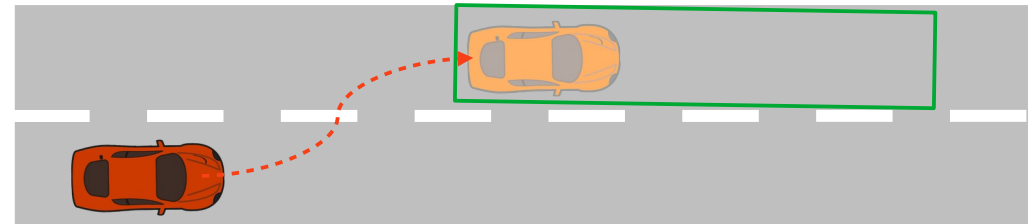
Przykład 1: Prosty problem śledzenia predefiniowanej trajektorii ruchu przy użyciu sterowania predykcyjnego.

Realizacja wysokopoziomowych manewrów

O ile sterowanie predykcyjne jest powszechnie używane w niskopoziomowych kontrolerach, zalety wykorzystania metod optymalizacyjnych do generacji trajektorii są szczególnie widoczne w przypadku ogólniej zdefiniowanych problemów, takich jak planowanie realizacji manewrów.

Przykładowy problem: **jak zrealizować manewr** (np. zmiany pasa) w sposób **efektywny** (ekonomiczny, komfortowy, płynny, szybki)?

O ile sam problem realizacji manewru jest stosunkowo prosty (może być np. sformułowany jako zestaw ograniczeń nierównościowych wymuszający osiągnięcie danej pozycji po zadanym czasie), możliwość definicji dodatkowych ograniczeń i składników funkcji kosztu umożliwia wygodną kontrolę nad wieloma aspektami generowanego ruchu.



Przykład 2: Prosty problem realizacji manewru zmiany pasa.

Przykładowe składniki kosztu:

- przeciążenia (komfort pasażerów)
- estymowane zużycie paliwa
- odległości od innych pojazdów (zwiększenie marginesu bezpieczeństwa)

Przykładowe ograniczenia:

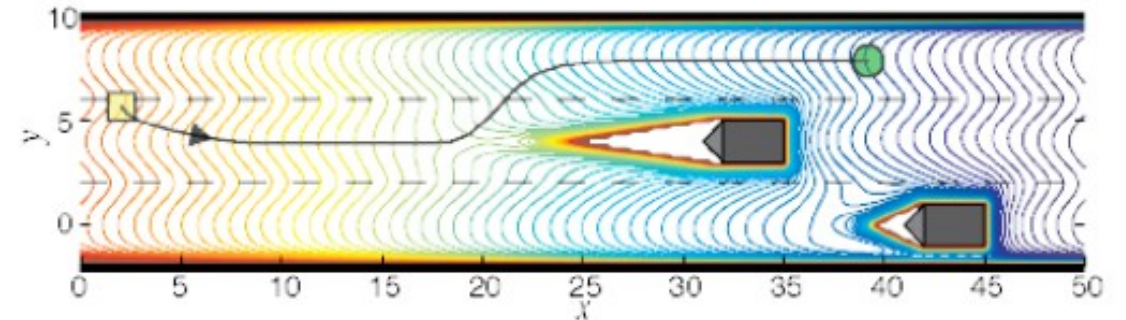
- prędkość przyspieszenie, kąt skrętu kierownicy, pochodne tych wartości
- maksymalna pozycja lateralna (ograniczenie przestrzeni jazdy do pasów ruchu)

Zapewnienie bezpieczeństwa

Istotnym problemem jest generacja trajektorii z uwzględnieniem aspektów bezpieczeństwa. Jednym z proponowanych rozwiązań jest sformułowanie problemu, w którym funkcja kosztu oparta jest o sztuczne pola potencjału.

Przestrzeń w której manewr może zostać wykonany opisana jest za pomocą pola potencjału określającego koszt przebywania w danej pozycji. Pole takie jest modyfikowane w zależności od pozycji i prędkości innych pojazdów w taki sposób, że przebywanie w ich pobliżu jest karane, w szczególności przy dużej różnicy prędkości. Dodatkowo pole potencjału może penalizować przebywanie pomiędzy dwoma pasami a także opuszczanie dostępnych pasów ruchu.

Tak sformułowany problem jest relatywnie prosty w implementacji i pozwala na łatwe uwzględnienie wielu aspektów wpływających na bezpieczeństwo planowanych trajektorii. Nie uwzględnia on jednak potencjalnych przyszłych manewrów innych pojazdów, a zmieniające się pole potencjału może doprowadzić do nieprzewidzianych suboptymalnych bądź wręcz niebezpiecznych zachowań.



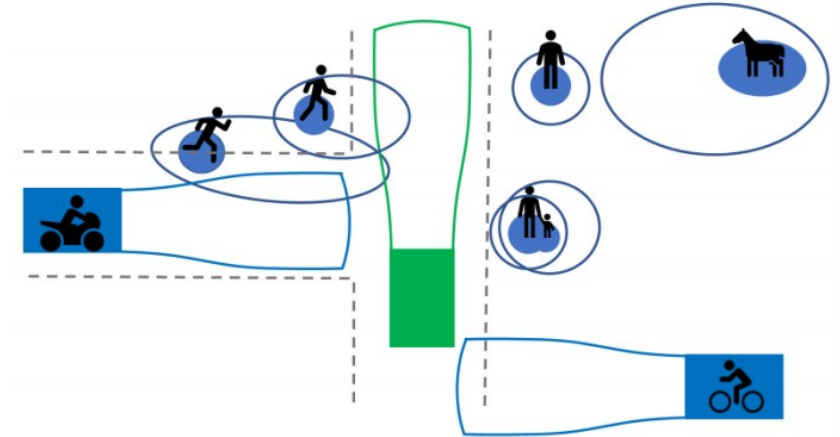
Wolf, Michael & W. Burdick, Joel. (2008). **Artificial potential functions for highway driving with collision avoidance.** 3731 - 3736. 10.1109/ROBOT.2008.4543783.

Zapewnienie bezpieczeństwa – modele bezpieczeństwa

Obecnie bardzo dużą popularność zyskuje podejście do problemu oparte o modele matematyczne formalizujące prawne jak i niepisane, zasady bezpieczeństwa standardowo przestrzegane przez kierowców.

Przykładami takich modeli są **Responsibility Sensitive Safety [1]** (RSS, model bezpieczeństwa zależny od odpowiedzialności) zaproponowany przez firmę MobilEye oraz **Safety Force Field [2]** (SFF, pole siłowe bezpieczeństwa) opracowany przez firmę Nvidia.

Oba modele opierają się o założenia dotyczące najgorszego możliwego scenariusza rozwoju sytuacji, przy założeniu że inne pojazdy będą przestrzegać tych samych podstawowych zasad bezpieczeństwa (dla ludzkich kierowców – zasady te są związane z prawami ruchu drogowego oraz zdrowym rozsądkiem).



Przykład koncepcji modelu bezpieczeństwa pozwalającego na obliczenie obszarów w których obiekty (wliczając kontrolowany pojazd) mogą się znaleźć w przypadku sytuacji niebezpiecznej. (Źródło: [2])

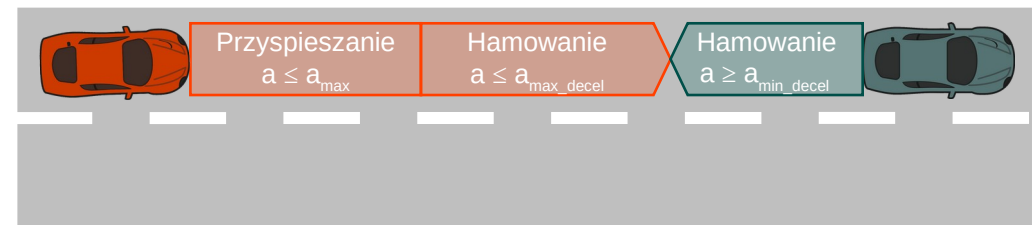
[1] Shalev-Shwartz, Shai, Shaked Shammah, and Amnon Shashua. "On a formal model of safe and scalable self-driving cars." *arXiv preprint arXiv:1708.06374* (2017).

[2] David Nistér, Hon-Leung Lee, Julia Ng, Yizhou Wang. "An Introduction to the Safety Force Field" NVIDIA, March 2019

Zapewnienie bezpieczeństwa – model RSS

Model bezpieczeństwa **RSS** opiera się na definicjach:

- **bezpiecznego dystansu** – model określa jak policzyć w danej sytuacji minimalny dystans (poprzeczny i wzdłużny) jaki powinien być zachowany do innych pojazdów;
- **niebezpiecznej sytuacji** – która występuje gdy dystans nie jest zachowany;
- **prawidłowej odpowiedzi** na niebezpieczną sytuację – w większości przypadków prawidłową odpowiedzią jest trajektoria hamowania spełniająca ograniczenia narzucone przez model;
- **odpowiedzialności** – model jest sformułowany w taki sposób, że dla każdej w pełni zarejestrowanej sytuacji drogowej zakończonej kolizją, możliwe jest wskazanie jednego lub więcej uczestnika ruchu odpowiedzialnego za kolizję.



Przykład: Obliczanie dystansu bezpiecznego w modelu RSS w prostej sytuacji nadążania za innym pojazdem.

Najgorszym zakładanym scenariuszem jest agresywne zahamowanie przez samochód zielony z przyspieszeniem większym lub równym minimalnemu przyspieszeniu hamowania a_{min_decel} (przykładowa wartość -7 m/s^2).

Samochód czerwony w najgorszym przypadku będzie przyspieszał z $a \leq a_{max}$ przez czas reakcji δ (np. 1 s), po którym dopiero zacznie hamować z maksymalnym przyspieszeniem a_{max_decel} (np. 4 m/s^2).

Bezpieczny dystans jest obliczany jako minimalny dystans, który pozwoli uniknąć kolizji w tym przypadku, nawet jeśli oba samochody będą musiały zahamować do pełnego zatrzymania.

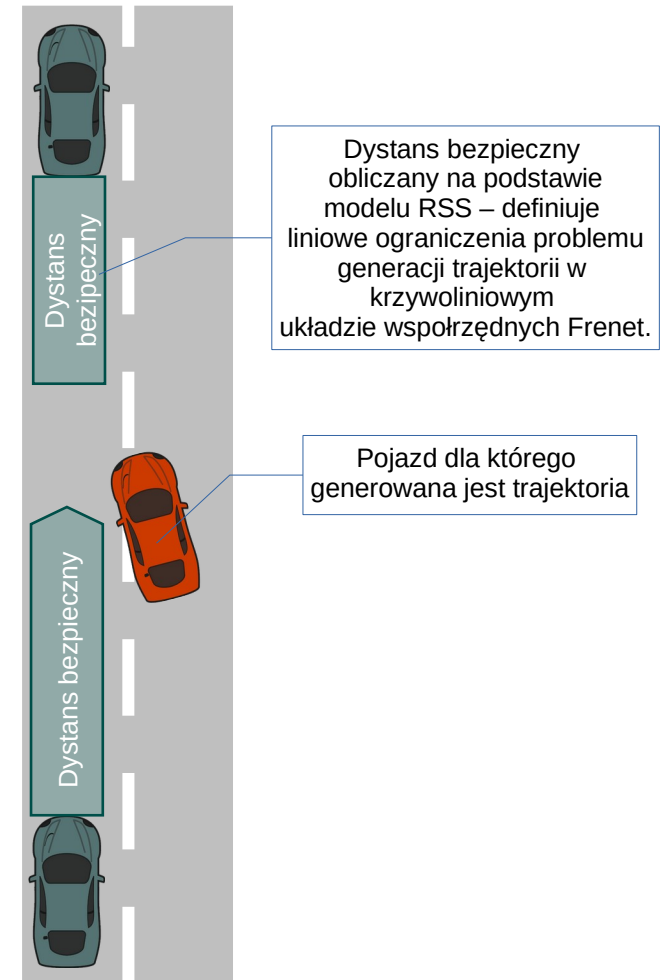
Parametry a_{min_decel} , a_{max} , a_{max_decel} , δ podlegają kalibracji (np. na podstawie obserwacji typowych zachowań kierowców) i mogą być dynamicznie zmieniane w zależności do warunków opogodowych, klasy pojazdu i innych czynników.

Zapewnienie bezpieczeństwa

Implementacja metody generacji trajektorii przy wykorzystaniu bezpieczeństwa RSS:

Model RSS cechuje się prostotą i transparentnością umożliwiającą jego szybką implementację. Zakładając, że trajektoria stanu jest planowana we współrzędnych krzywoliniowych związanych z krzywizną drogi (układ Frenet), dystanse bezpieczne mogą zostać uwzględnione w problemie optymalizacyjnym jako liniowe ograniczenia nierównościowe.

Tak sformułowany problem może zostać rozwiązany bardzo szybko i przy niskim nakładzie mocy obliczeniowej.



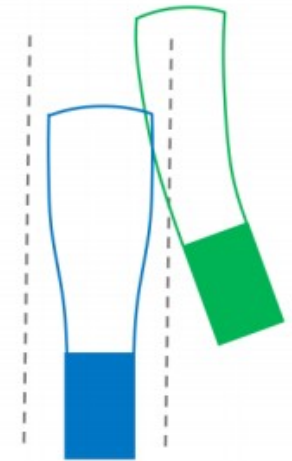
Przykład 3: Dystanse bezpieczne w modelu RSS jako zestaw liniowych ograniczeń nierównościowych modelu generacji trajektorii.

Zapewnienie bezpieczeństwa – model SFF

Model bezpieczeństwa SFF zaproponowany przez Nvidię zakłada bardziej złożone podejście do tematu wyznaczania obszarów bezpiecznych oraz prawidłowej odpowiedzi. W modelu SFF bezpieczne dystanse zastąpione są zmiennymi w czasie obszarami rezerwowanymi na wykonanie manewrów bezpieczeństwa wyznaczanymi na podstawie m.in. zakładanej kinematyki pojazdu.

W przypadku przecięcia zarezerwowanych obszarów czasoprzestrzeni, każdy pojazd powinien wykonać predefiniowany manewr bezpieczeństwa (np. hamowanie z określonymi parametrami, podobnie jak w RSS). W momencie rozpoczęcia manewru rezerwowany obszar ulega zamrożeniu, a pojazd w trakcie manewru z definicji nie wykracza poza zarezerwowany obszar.

Pojazd może jednak wykonać również tzw. manewr równie dobry, czyli inny manewr który nie powoduje zwiększenia pola przecięcia obszarów zarezerwowanych.



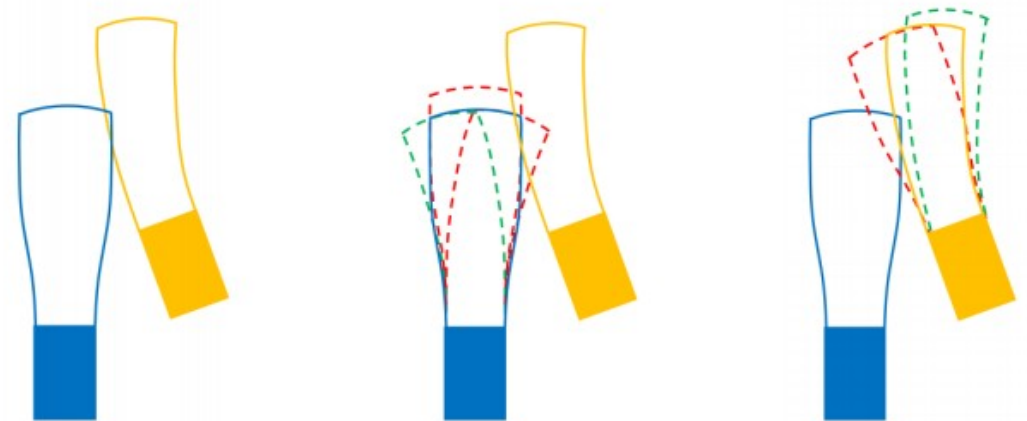
Przykład 4: Obszary rezerwowane dwóch pojazdów. W sytuacji przecięcia obszarów pojazdy muszą wykonać manewr bezpieczeństwa w trakcie którego nie wykrócą poza swoje obszary rezerwowane lub inny manewr który nie spowoduje zwiększenia części wspólnej obszarów.

Źródło: David Nistér, Hon-Leung Lee, Julia Ng, Yizhou Wang. “An Introduction to the Safety Force Field” NVIDIA, March 2019

Zapewnienie bezpieczeństwa – model SFF

Założenie o możliwości wykonania dowolnego manewru bezpieczeństwa nie zwiększającego pola przecięcia obszarów rezerwowanych umożliwia sformułowanie problemu optymalizacji, który w przypadku wystąpienia sytuacji niebezpiecznej wygeneruje optymalny (pod wybranymi względami) manewr bezpieczeństwa. Możliwość kształtowania trajektorii poprzez dodatkowe ograniczenia i składniki kosztu pozwala nie tylko na uniknięcie kolizji, ale również na maksymalizację marginesu bezpieczeństwa oraz zachowanie względnego komfortu i efektywności ruchu.

Podstawową wadą modelu SFF jest nieliniowość rezerwowanych obszarów, co znacząco zwiększa trudność rozwiązania problemu optymalizacyjnego oraz złożoność obliczeniowa problemu wyznaczenia obszarów rezerwowanych.



Analiza perturbacji. Sformułowanie problemu optymalizacji który minimalizuje pole przecięcia obszarów rezerwowanych pozwala na znalezienie bardziej efektywnych rozwiązań sytuacji niebezpiecznej – np. dla żółtego pojazdu przyspieszenie i zwiększenie kąta skrętu zamiast hamowania.

Źródło: David Nistér, Hon-Leung Lee, Julia Ng, Yizhou Wang. “An Introduction to the Safety Force Field” NVIDIA, March 2019

Generacja scenariuszy testowych, aktywne poszukiwanie błędów w systemie

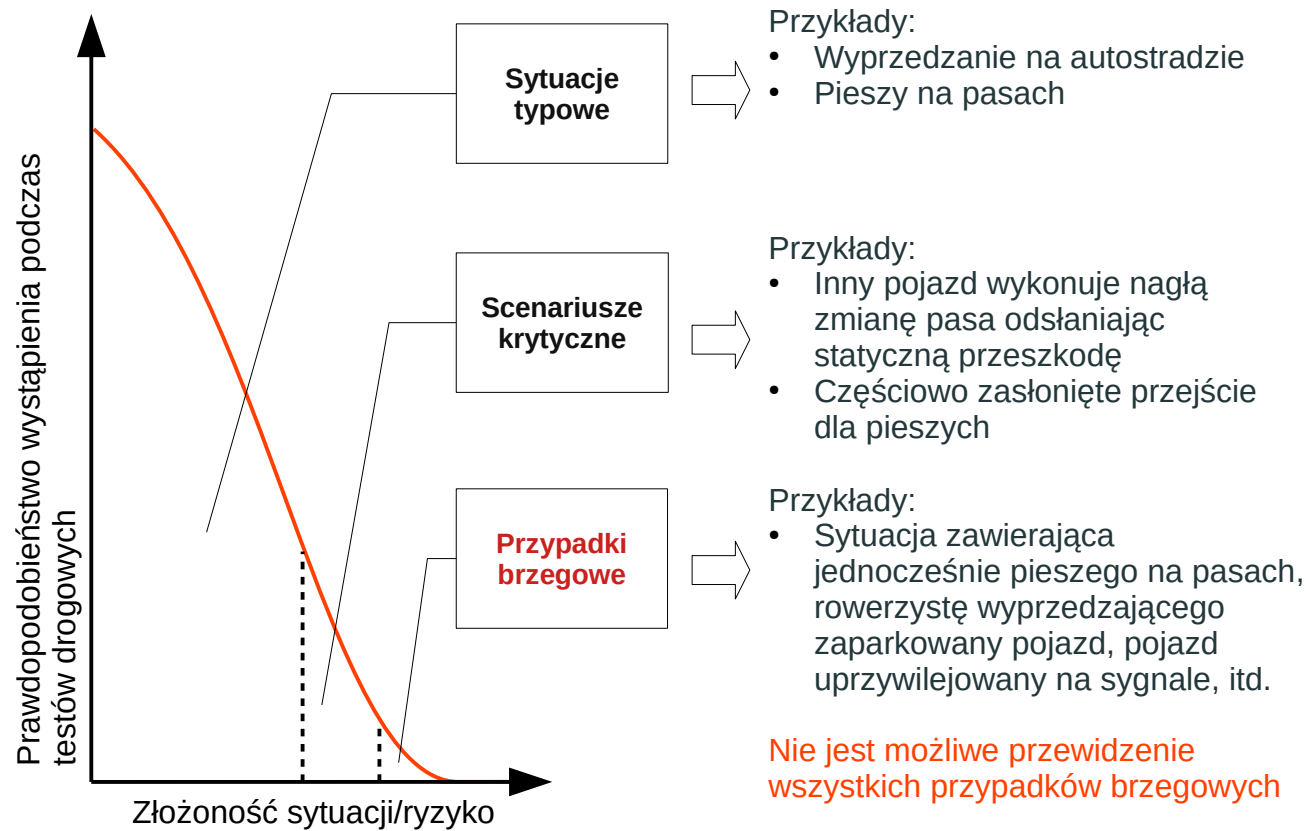
Poważnym problemem w testowaniu systemów jazdy autonomicznej jest eksploracja scenariuszy krytycznych i przypadków brzegowych. Dla poznania słabości projektowanego systemu, bardzo ważne jest przetestowanie jego działania w możliwie największej liczbie trudnych i skomplikowanych sytuacji. Prawdopodobieństwo napotkania takich sytuacji podczas testów drogowych jest jednak niskie.

Jednym z proponowanych rozwiązań pozwalających na automatyczne wygenerowanie w symulacji przypadków brzegowych dla celów testowych jest optymalizacja trajektorii pojazdów zaobserwowanych w czasie testów drogowych dla zwiększenia krytyczności zarejestrowanej sytuacji [1].

Istnieją również rozwiązania formułujące problemy optymalizacyjne ukierunkowane bezpośrednio na wygenerowanie trajektorii innych użytkowników ruchu, które mają za zadanie doprowadzić do kolizji w symulacji, odsłaniając potencjalne słabości testowanego systemu. [2]

[1] Althoff, Matthias, and Sebastian Lutz. "Automatic generation of safety-critical test scenarios for collision avoidance of road vehicles." 2018 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2018.

[2] Abeyirigoonawardena, Yasasa, Florian Shkurti, and Gregory Dudek. "Generating adversarial driving scenarios in high-fidelity simulators." 2019 International Conference on Robotics and Automation (ICRA). IEEE, 2019.

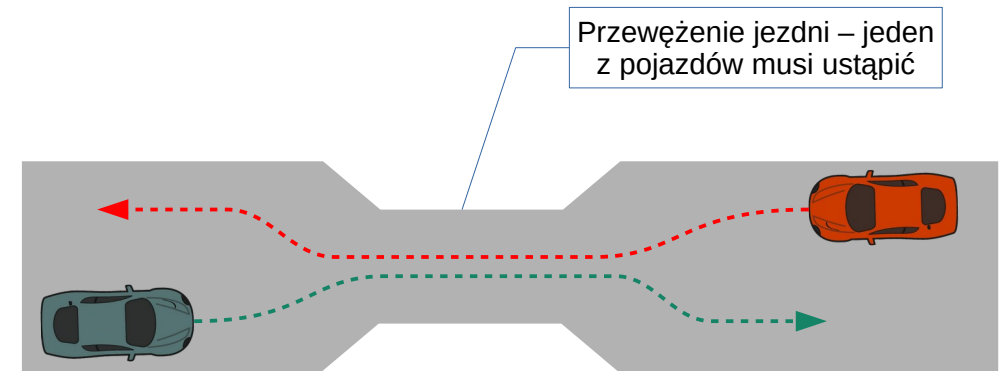


Generacja optymalnego zachowania w sytuacjach spornych

Innym problemem w którym optymalizacja trajektorii znajduje zastosowanie jest problem rozwiązywania sytuacji spornych, wymagających ustąpienia pierwszeństwa, gdzie nie jest ono narzucone prawnie (patrz Przykład 5).

W takich sytuacjach problem optymalizacyjny może zostać sformułowany dla całej sceny (generowane są trajektorie obu pojazdów jednocześnie, jak również minimalizowany jest koszt dla obu pojazdów) aby otrzymać rozwiązanie kwestii spornej optymalne z globalnego punktu widzenia. Pojazd sterowany podąża za wygenerowaną dla siebie trajektorią, przyjmując hipotezę, że kierowca drugiego pojazdu intuicyjnie również podejmie optymalną decyzję.

Naturalnym zastosowaniem dla takiego rozwiązania jest periodycznie wywoływany problem sterowania predykcyjnego, więc jeśli drugi pojazd wybierze jednak suboptymalne rozwiązanie sytuacji spornej, zachowanie pojazdu sterowanego zostanie skorygowane.



Przykład 5: Sytuacja sporna w której pierwszeństwo nie jest określone. Problem optymalizacyjny jest formułowany dla obu pojazdów dla znalezienia globalnie optymalnego rozwiązania sytuacji.