

Współczesne techniki heurystyczne
Sprawozdanie końcowe
Zastosowanie algorytmu rozmytego do sterowania prędkością
samochodu

Piotr Jastrzębski
Marcin Nazimek

1 Szczegółowy opis zadania

Rozważmy poruszający się samochód. W celu stworzenia modelu ruchu bierzemy pod uwagę dwie zmienne wejściowe: PRĘDKOŚĆ i ODLEGŁOŚĆ od jadącego z przodu samochodu oraz jedna zmienna określająca zmianę tego ruchu PRZYSPIESZENIE. Sterowanie odbywać będzie się poprzez zmianę prędkości. Przykładowe możliwe stany to: przyspieszenie, utrzymanie prędkości i hamowanie (ale można przyjąć więcej stanów, np. odległość jest bardzo mała, itp.).

Dla rozpatrywanego przypadku, baza reguł ma postać:

- IF odległość jest mała AND prędkość jest mała THEN utrzymaj prędkość.
- IF odległość jest mała AND prędkość jest duża THEN zredukuj prędkość.
- IF odległość jest duża AND prędkość jest mała THEN zwiększaj prędkość.
- IF odległość jest duża AND prędkość jest duża THEN utrzymaj prędkość.

Zmienne ODLEGŁOŚĆ, PRĘDKOŚĆ i PRZYSPIESZENIE to zmienne lingwistyczne, które mogą przyjmować rozmyte wartości: krótki, długi, mała, duża, utrzymaj, zredukuj i zwiększaj. Projektant ma teraz za zadanie dobrać tak parametry zbiorów rozmytych i obszarów rozważań by odpowiadały one rzeczywistości w jak najlepszym stopniu.

2 Założenia projektu

Projekt ma symulować i prezentować wykorzystanie algorytmu logiki rozmytej do sterowania prędkością samochodu w możliwie największym stopniu

odwzorowując rzeczywistość.

W przygotowanym środowisku na potrzeby prezentacji wykorzystane są dwa uproszczone modele reprezentujące samochody – pierwszy, dalej oznaczany jako A , będący samochodem, dla którego przygotowane zostanie sterowanie, oraz samochód B , który w sposób niedeterministyczny, ale zgodny z przepisami i realnymi wartościami przyspieszeń, będzie poruszał się przed A . Każdy z nich w danej chwili czasu opisany będzie prędkością bezwzględną.

Odległość między samochodami wynika wprost z różnic prędkości, a prędkość auta A modyfikowana jest przez wartość wyjścia modułu logiki rozmytej – przyspieszeniem.

2.1 Ograniczenia

Na potrzeby projektu przyjęto następujące ograniczenia:

- **Prędkość** - przyjmuje wartości z zakresu $0 \div 140 km/h$ ¹.
- **Przyspieszenie** - jest zmienne w czasie. Kwestią do rozstrzygnięcia pozostaje, czy będzie ono takie samo czy przyjmie różne wartości dla opóźnienia i przyspieszenia. Na bazie zgromadzonych informacji odpowiednie wydaje się przyjęcie wartości przyspieszenia z zakresu $\langle -9 \frac{m}{s^2}; 4 \frac{m}{s^2} \rangle$ ² i dla aktualnego stanu zaawansowania projektu takie jest wykorzystywane.
- **Odległość** - wyrażona w kilometrach dowolną nieujemną liczbą dziesiętną. Testowane jest rozwiązanie, które rozważa niejako dwa stany:
 - stan bliski – do odległości $1 km$, gdzie aktywny jest moduł logiki rozmytej
 - stan daleki – powyżej $1 km$, gdzie samochód porusza się prędkością autostradową, tak długo, aż nie na trafi w odległości $1 km$ ma poprzedzające auto. Tym samym wartość $1 km$ jest wartością rozgraniczającą niejako nieskończoność od skutecznej odległości działania modułu dystansującego.

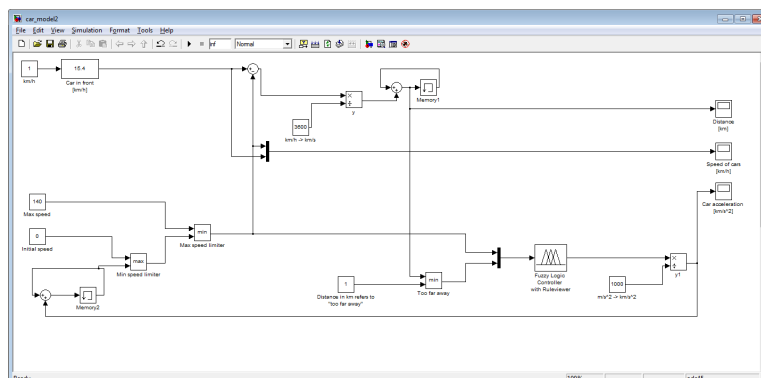
Każde przekroczenie wartości 0 świadczy o najechaniu na samochód poprzedzający i w końcowej wersji projektu nie ma prawa wystąpić.

2.2 Stan początkowy

Przed rozpoczęciem symulacji stan układu (wartości poszczególnych parametrów) mogą być modyfikowane. Ustalane mogą być zarówno prędkość początkowa samochodu A oraz B jak i odległość między nimi. Celem symulacji jest tak sterować parametrami samochodu A , aby zachowywał on

¹Założono, że ruch odbywa się na autostradzie w Polsce.

²Wartości prawdziwe dla suchej nawierzchni.



Rysunek 1: Środowisko symulacji ruchu samochodów

stałą odległość od poprzedzającego go samochodu nie powodując kolizji (najeżdżania). Aktualnie zaimplementowana funkcjonalność pozwala na dowolne sterowanie prędkością samochodu poprzedzającego.

3 Implementacja środowiska

W celu prezentacji działania algorytmu wykorzystywane są funkcjonalności zawarte w programie *Matlab R2011a* wraz z rozszerzeniem *Fuzzy Logic Toolbox*. Moduł ten zapewnia funkcje, narzędzia graficzne i elementy *Simulink* dla systemów bazujących na logice rozmytej. W zaproponowanym rozwiązaniu bezwzględne prędkości aut prezentowane są na tym samym wykresie pozwalając obserwować proces reakcji samochodu sterowanego modułem logiki rozmytej. Zrzut ekranu przedstawiający środowisko symulacji przedstawiono na rysunku 1.

4 Implementacja logiki rozmytej

Obok schematu *Simulink* symulującego środowisko najważniejszą częścią układu jest moduł odpowiedzialny za implementację logiki rozmytej. Moduł ten (Fuzzy Logic Controller) został zdefiniowany w następujący sposób:

4.1 Fuzzyfikacja (rozmycie)

Rozmywanie polega na określeniu stopnia przynależności danej wartości wielkości wejściowej do każdego z odpowiadających jej zbiorów rozmytych pokrywających zakres możliwych wartości wejściowych. Operacja ta sprowadza się do obliczania wartości funkcji. Wartości wejściowe opisywane są funkcją trójkątną o wzorze:

$$f(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (1)$$

- Prędkość (*speed*) - określona jest funkcją trójkątną o parametrach:
 - prędkość niska (*low*) ($a = -140, b = 0, c = 140$)
 - prędkość wysoka (*high*) ($a = 0, b = 140, c = 280$)
- Odległość (*distance*) - określona również funkcją trójkątną, gdzie:
 - odległość mała (*short*) ($a = -1, b = 0, c = 1$)
 - odległość duża (*long*) ($a = 0.5, b = 1, c = 2$)

4.2 Zastosowanie operatorów logiki rozmytej do określenia stopnia, w jakim spełniona jest przesłanka w każdej z reguł

Jak widać w punkcie 1 w bazie reguł znajduje się jedynie funkcja logicznej koniunkcji (*AND*) warunków. Funkcja dla koniunkcji ustalona została jako wartość minimalna (*min*) elementów składowych poprzednika implikacji.

4.3 Zastosowanie metody implikacji

Funkcja implikacja to również funkcja minimalnej wartości - *min*.

4.4 Agregacja wszystkich wyjść

Funkcja agregująca wszystkie wyjścia bierze pod uwagę wartość maksymalną (*max*) spośród wszystkich następników reguł. Funkcje wyjść zdefiniowane są jako:

- Przyspieszenie (*acceleration*) - określone funkcjami o różnym przebiegu:
 - zmniejsz (*decrease*) ($a = 0.15, b = 0.6, c = -9$) (f. dzwonowa, jak we wzorze 2)
 - utrzymuj (*maintain*) ($a = -0.4, b = 0, c = 0.4$) (f. trójkątna, jak we wzorze 1)
 - zwiększ (*increase*) ($a = 0, b = 1.5, c = 4, d = 10$) (f. trapezoidalna, jak we wzorze 3)

4.5 Schemat wnioskowania rozmytego

Wykorzystany został schemat wnioskowania rozmytego *Mamdani*, który to charakteryzuje się łączeniem następników każdej reguły poprzez operator agregacji a następnie przeprowadzaniem defuzyfikacji w celu otrzymania odpowiedzi systemu. Toolbox logiki rozmytej pozwalał także na wybranie modelu typu *Sugeno*, w którym następnik każdej reguły jest liniową kombinacją wejść systemu. Wyjście systemu jest ważoną kombinacją liniową następników reguł definiujących system.

4.6 Defuzyfikacja (wyostrzanie)

Wartość wyostrzana jest wyznaczana funkcją *centroid* - jest to punkt na osi argumentów będący przecięciem linii równowagi powierzchni pod krzywą z osią argumentów. Wzór funkcji składowych: trójkątnej przedstawiono jako 1, dzwonowej jako 2, a funkcji trapezoidalnej jako 3.

$$f(x) = \frac{1}{1 + \left|\frac{x-c}{a}\right|^{2b}} \quad (2)$$

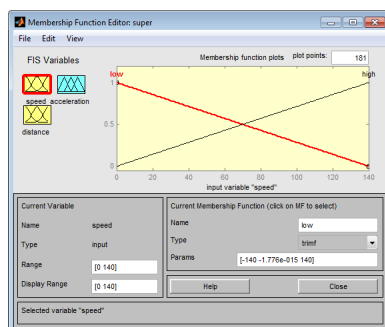
$$f(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0, & c \leq x \end{cases} \quad (3)$$

5 Wyniki, testy poprawności i jakości algorytmów

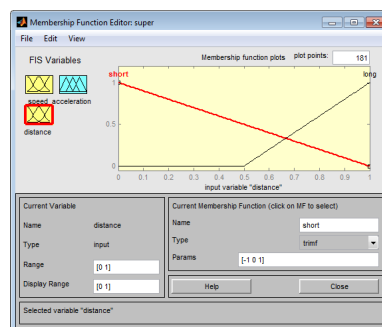
Proces testowania polegał na wielokrotnym uruchamianiu symulacji w module *Simulink*. Każda symulacja trwała przynajmniej minutę i realizowała oddzielny scenariusz odzwierciedlający rzeczywistą sytuację na drodze (korek, pościg za samochodem albo normalny ruch uliczny). Wynikiem koniecznym zaliczenia testów było nieosiągnięcie ujemnej wartości odległości między samochodami, co sugerowałoby wystąpienie kolizji. Wyniki potwierdzające skuteczność sterowania można zobaczyć na rysunkach 3, 4 oraz 5.

6 Wnioski

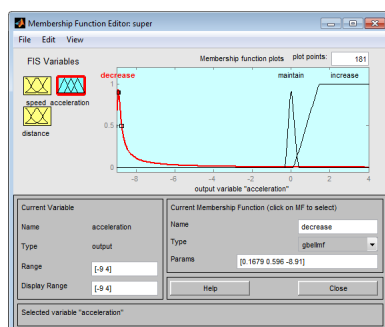
Wyniki, a szczególnie wykresy zgromadzone podczas pracy algorytmu, przywołują na myśl sterowanie z wykorzystaniem regulatorów PID. Analiza wykresów pozwala założyć, że moduł sterowania prędkością samochodu działa prawidłowo i spełnia zasady bezpieczeństwa nie dopuszczając do wystąpienia sytuacji najechania.



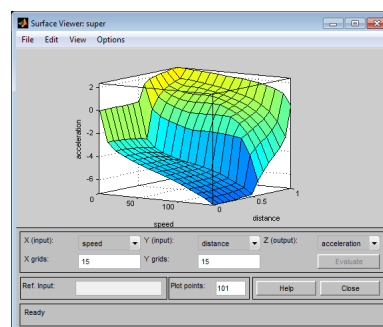
(a) Funkcja wejściowa prędkości



(b) Funkcje wejściowa odległości

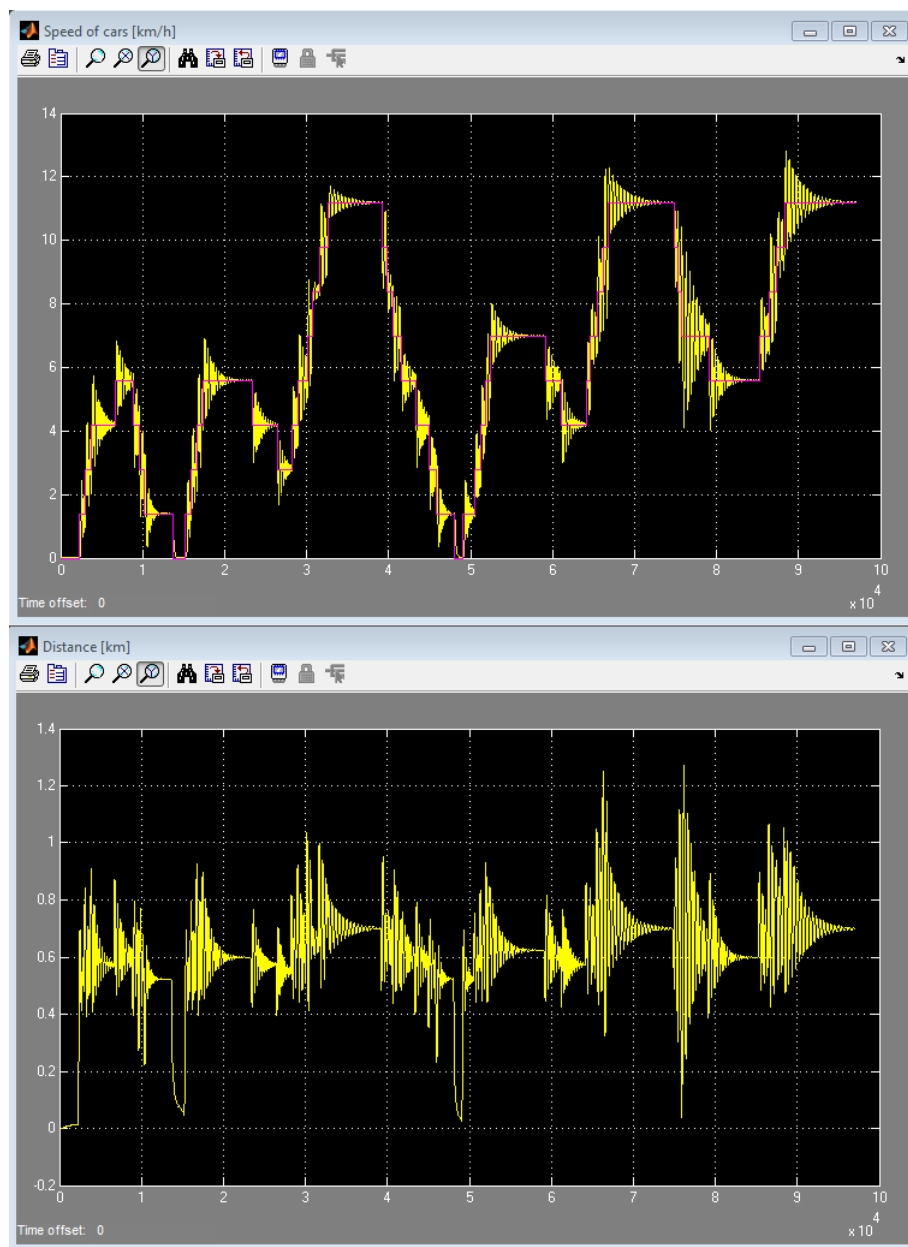


(c) Funkcja wyjściowa

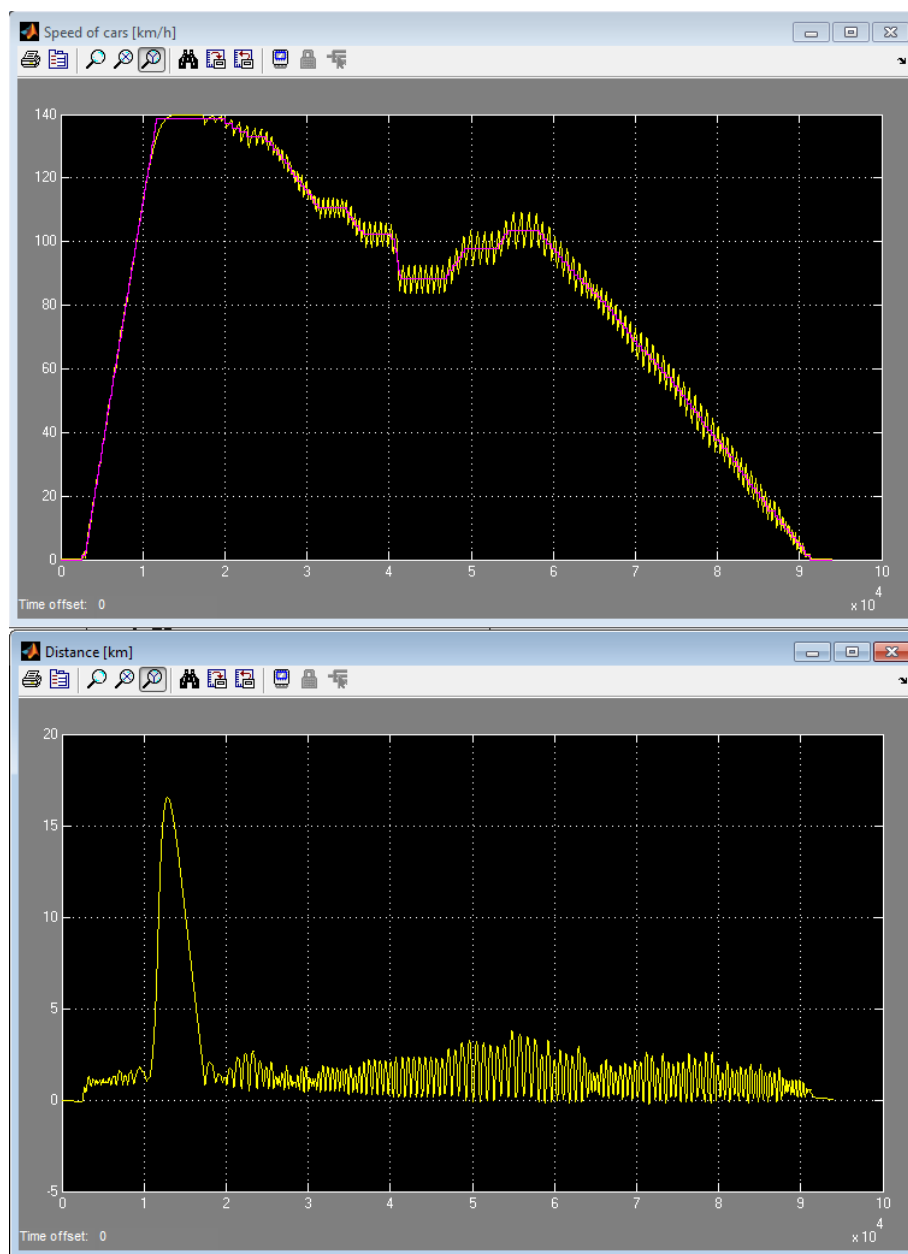


(d) Płaszczyzna funkcji wyjściowej

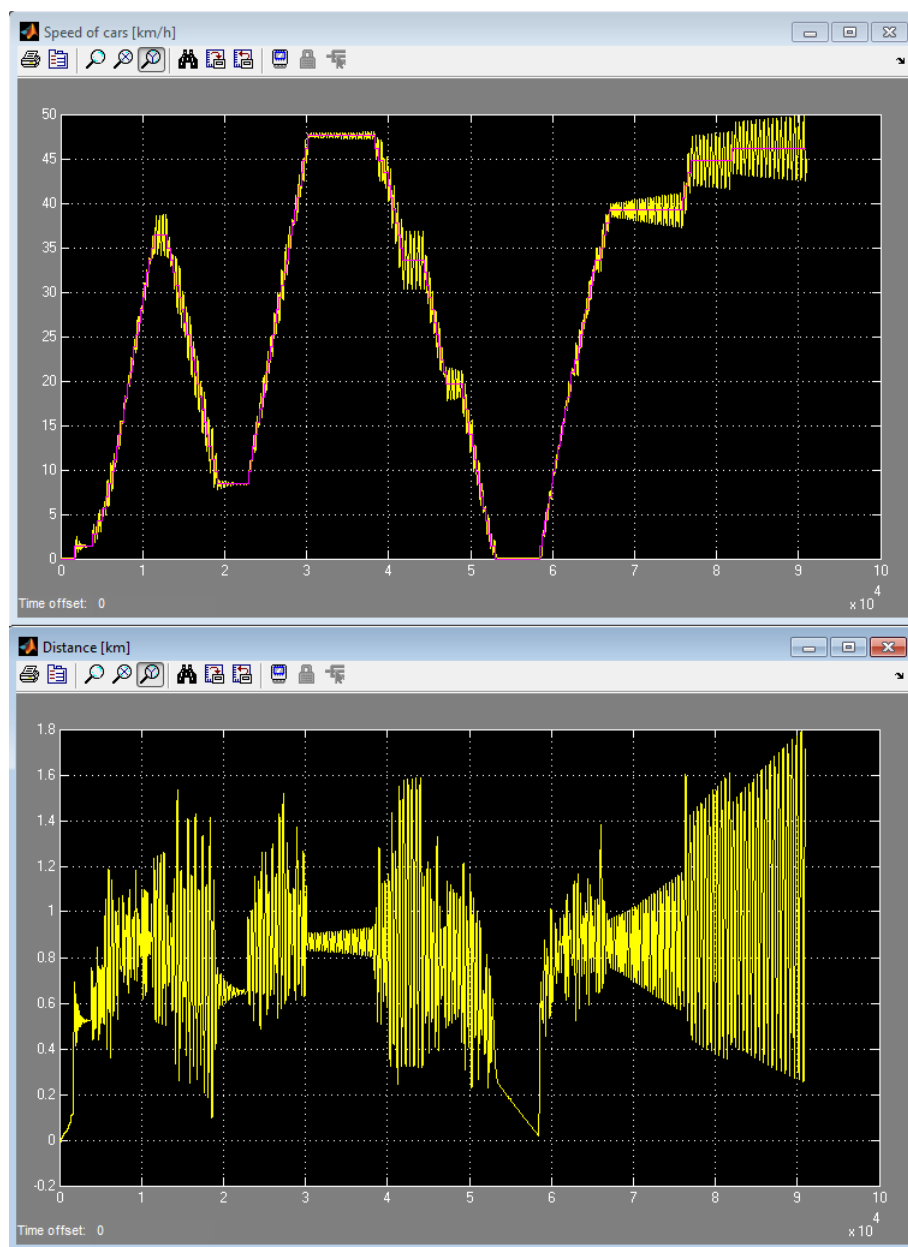
Rysunek 2: Porównanie przykładowych kształtów funkcji oraz płaszczyzny wartości przyspieszenia w zależności od prędkości i odległości.



Rysunek 3: Test 1 - korek uliczny



Rysunek 4: Test 2 - pościg za szybkim samochodem



Rysunek 5: Test 3 - normalny ruch miejski

Literatura

- [1] *Fuzzy Logic Toolbox User's Guide*
http://www.mathworks.com/help/pdf_doc/fuzzy/fuzzy.pdf
- [2] Ross T., *Fuzzy Logic with Engineering Applications*
- [3] Rykaczewski K., *Systemy rozmyte i ich zastosowania*
<http://math.uni.lodz.pl/~fulmanp/zajecia/ssn/materialy/duszek.pdf>
- [4] Mamat M., Ghani N. M., *Fuzzy Logic Controller on Automated Car Braking System*, IEEE International Conference on Control and Automation, 2009. ICCA 2009.
- [5] Khodayari, A., Kazemi, R., Ghaffari, A., Braunstingl, R., *Design of an improved fuzzy logic based model for prediction of car following behavior*, IEEE International Conference on Mechatronics (ICM), 2011
- [6] Sato, T., Akamatsu, M., Pengjun Zheng, McDonald, M., *Comparison of car following behavior between UK and Japan*, ICCAS-SICE, 2009
- [7] *Simulating Dynamic Systems*
<http://www.mathworks.com/help/simulink/ug/simulating-dynamic-systems.html>