

BRZozowski Krzysztof, SORDYL Justyna

ZASTOSOWANIE METODY DYNAMIKI SYSTEMOWEJ DO ANALIZY ZAGADNIENI TRANSPORTOWYCH

Streszczenie

W artykule przedstawiono przykład implementacji dwóch wybranych modeli zagadnień transportowych zrealizowanych z wykorzystaniem metody dynamiki systemowej. Metoda dynamiki systemowej umożliwia formułowanie modeli symulacyjnych bez konieczności znajomości jakiegokolwiek języka programowania wyższego rzędu i samodzielnego tworzenia modelu komputerowego. Zakres możliwych zastosowań metody w zagadnieniach transportowych obejmuje całe spektrum zagadnień opisujących procesy ciągłe w czasie, od modeli łańcuchów logistycznych przez zagadnienia inżynierii ruchu, eksploatacji środków transportu po wiele innych. Tym samym metoda dynamiki systemowej jest przydatnym narzędziem, które można wykorzystać w procesie kształcenia inżynierów transportu.

WPROWADZENIE

Symulacja komputerowa z wykorzystaniem dynamiki systemowej jest jednym z użytecznych sposobów badania zachowania się złożonych systemów. Analiza złożonych systemów często wymaga podejścia interdyscyplinarnego w związku z tym, że systemy te składają się z elementów stanowiących przedmiot zainteresowań różnych dyscyplin naukowych. Ogólnie rzecz ujmując, system jest zbiorem elementów wewnętrznie spójnym o określonej strukturze oraz posiada granicę oddzielającą go od całości świata materialnego i niematerialnego [9].

Metoda dynamiki systemowej umożliwia konstrukcję modelu symulacyjnego, w którym odwzorowywane są wszystkie istotne relacje między elementami systemu typu zamkniętego a dynamika systemu jest skutkiem zachodzących w systemie sprzężeń zwrotnych. Podstawowym etapami konstrukcji modelu jest określenie granicy między systemem i jego otoczeniem oraz identyfikacja pętli sprzężeń zwrotnych, a więc łańcuchów powiązań przyczynowo-skutkowych między elementami systemu [5, 9]. Pętle sprzężeń zwrotnych należą przy tym do jednej z dwóch klas, ujemnych sprzężeń zwrotnych zwanych też równoważącymi lub dodatnich sprzężeń zwrotnych zwanych również wzmacniającymi.

Sprzężenie zwrotne ujemne opisuje sytuację propagacji skutków zmiany stanu jednego z elementów w łańcuchu relacji przyczynowych doprowadzającą do korekty początkowej zmiany stanu. Ujemne sprzężenia zwrotne opisują zatem procesy zapewniające utrzymanie stanu równowagi w systemie. W przypadku sprzężenia zwrotnego dodatniego wynikiem propagacji skutków zmiany stanu elementu jest wzmocnienie tej zmiany. Dodatnie sprzężenia zwrotne reprezentują zatem w systemie, w zależności od kierunku zachodzących zmian, procesy wzrostu lub zaniku. W wielu przypadkach rzeczywistych systemów analityczna

ocena ich dynamiki jest zadaniem trudnym ze względu na równoległe istniejące procesy równoważące i wzrostowe.

1. ZARYS METODOLOGII TWORZENIA MODELI Z WYKORZYSTANIEM METODY DYNAMIKI SYSTEMOWEJ

Implementacja modelu symulacyjnego rzeczywistego systemu z wykorzystaniem metody dynamiki systemowej wymaga dwóch podstawowych etapów działania, sformułowania modelu przyczynowo-skutkowego oraz zrealizowania jego reprezentacji w środowisku symulacyjnym. Formalnie metoda dynamiki systemowej może być klasyfikowana jako metoda umożliwiająca implementację abstrakcyjnych nieliniowych modeli dynamicznych nie wprowadzająca ponadto ograniczeń co do rzędu równań czy ich nieliniowości. Modele symulacyjne powstające z wykorzystaniem metody dynamiki systemowej tworzone są z wykorzystaniem pięciu typów elementów, w tym dwóch podstawowych (głównych) tworzących pętle sprzężeń zwrotnych: poziomów (zasobów) i strumieni (przepływów) oraz trzech elementów uzupełniających: parametrów, zmiennych egzogenicznych i zmiennych pośrednich (pomocniczych).

Poziomy reprezentują najistotniejsze elementy systemu umożliwiające kompleksowy opis jego stanu w dowolnej chwili czasu a wartości chwilowe poziomów ulegają zmianom w czasie w związku z przypisanymi im strumieniami, co można przedstawić w ogólnej postaci:

$$P_{(t=T)} = P_{(t=\tau)} + \int_{\tau}^T S_{in}(t) dt - \int_{\tau}^T S_{out}(t) dt \quad (1)$$

gdzie S_{in}, S_{out} to odpowiednio strumień wejściowy i strumień wyjściowy, a $P_{(t=T)}, P_{(t=\tau)}$ są wartościami poziomu w odpowiednich momentach czasu.

Przy tym, w ogólnym przypadku zachodzi zależność strumieni od wartości poszczególnych poziomów, zmiennych pomocniczych oraz parametrów, co dla pojedynczego strumienia można przedstawić w ogólnej postaci:

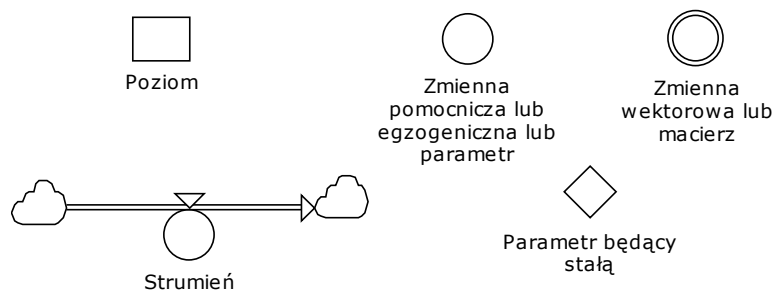
$$S_i = f(\mathbf{P}, \mathbf{Z}, \mathbf{R}) \quad (2)$$

gdzie $\mathbf{P}, \mathbf{Z}, \mathbf{R}$ to odpowiednio zbiór wartości poziomów, zmiennych pomocniczych i parametrów mających wpływ na wielkość strumienia.

Ponieważ również zmienne pomocnicze mogą zależeć od wartości poziomów w konsekwencji łatwo zauważyć, że realizacja modelu z wykorzystaniem metody dynamiki systemowej może prowadzić do skomplikowanego układu równań, a do jego rozwiązania stosowane będą techniki całkowania numerycznego, bazujące na dyskretyzacji czasu symulacji i sekwencyjnym rozwiązywaniu układu równań modelu.

Obecnie na rynku istnieje szereg środowisk wspierających wykorzystanie metody dynamiki systemowej w badaniach symulacyjnych. Wspólną cechą tych środowisk jest sposób budowy modelu komputerowego, bazujący na graficznym interfejsie użytkownika umożliwiającym implementację modelu poprzez graficzne zestawianie poszczególnych elementów podstawowych oraz ustanawianie ich połączeń zgodnie z przyjętym modelem przyczynowo-skutkowym systemu. Równania modelu są następnie automatycznie generowane na podstawie definicji poszczególnych zmiennych i utworzonej graficznej reprezentacji modelu nazywanej schematem strukturalnym.

Elementy wykorzystywane w metodzie dynamiki systemowej do tworzenia modeli komputerowych w postaci diagramów strukturalnych w środowisku Powersim mają reprezentację graficzną jak na rys. 1:



Rys. 1 Graficzna reprezentacja podstawowych elementów modelu wykorzystywana do tworzenia schematów strukturalnych w środowisku Powersim

Źródło: opracowanie własne

Pojawiający się przy strumieniu kształt chmurki oznacza nieanalizowane źródło (koniec bez strzałki) lub ujście (koniec oznaczony strzałką). Dopiero powiązanie strumienia z poziomem lub poziomami powoduje zdefiniowanie źródła i/lub ujścia. Ponadto przy dokumentowaniu modelu zaimplementowanego w środowisku Powersim używane są między innymi następujące określenia:

- mainmodel: składnik projektu, w ramach którego dokumentowane są poszczególne elementy i jednostki,
- level: poziom,
- aux: zmienna pomocnicza,
- unit (autunit): jednostka (jednostka wyznaczona przez system),
- type (autotype): typ danych (typ wyznaczony przez system),
- inflow: strumień wejściowy do zmiennej poziom,
- outflow: strumień wyjściowy ze zmiennej poziom,
- def: definicja (dotyczy zmiennych, jednostek, zbiorów),
- init: wartość początkowa zmiennej poziom, również wartość stałej.

W dalszej części artykułu przedstawiono przykłady modeli dwóch różnych zagadnień transportowych zrealizowane przy użyciu metody dynamiki systemowej.

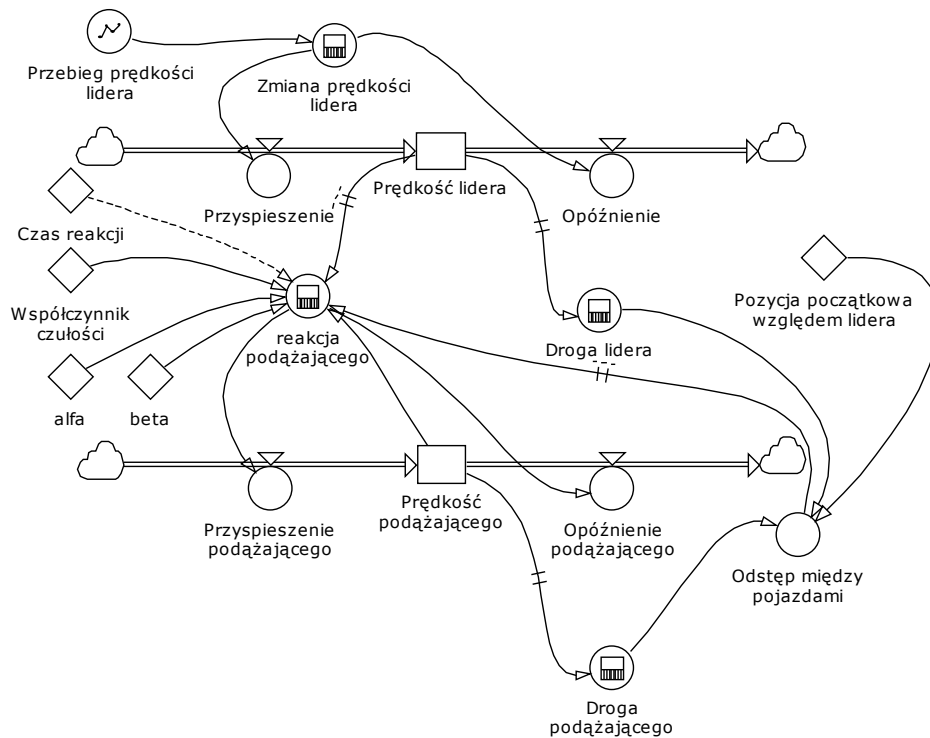
2. MODEL PODĄŻANIA POJAZDÓW ZREALIZOWANY PRZY UŻYCIU METODY DYNAMIKI SYSTEMOWEJ

Metoda dynamiki systemowej może zostać zastosowana do prowadzenia badań symulacyjnych z wykorzystaniem modeli opisujących procesy zachodzące w strumieniu ruchu. Rozpatrzmy proces podążania pojazdów w strumieniu ruchu według modelu mikroskopowego klasy bodziec–reakcja. W modelach tej klasy zakłada się, że ruch pojedynczego pojazdu w strumieniu można rozpatrywać jako ciągły proces reagowania kierowcy na pojawiające się bodźce. Rozpatrzmy zatem ruch dwóch pojazdów, pojazdu A jako prowadzącego oraz pojazdu B jako podążającego dla nieliniowego modelu GHR klasy bodziec-reakcja, który można zapisać w postaci:

$$\frac{dv_B(t + \tau)}{dt} = \frac{\lambda v_B^\alpha(t + \tau)}{[x_A(t) - x_B(t)]^\beta} [v_A(t) - v_B(t)] \quad (3)$$

gdzie $v_A(t), v_B(t)$ są prędkościami odpowiednio pojazdu A i B w danej chwili czasu, różnica $x_A(t) - x_B(t)$ jest odległością między pojazdami A i B w danej chwili czasu, wykładniki α, β są parametrami empirycznymi, λ jest stopniem wrażliwości (czułości) na pojawiający się bodziec a τ jest czasem reakcji.

Realizacja modelu GHR przy użyciu metody dynamiki systemowej prowadzi do następującego schematu strukturalnego rozważanego modelu jak na rys. 2.



Rys. 2 Schemat strukturalny dla mikroskopowego modelu GHR

Źródło: opracowanie własne

Analizując schemat przedstawiony na rys. 2 łatwo zauważyć, że w modelu wprowadzono zmienną egzogeniczną przebieg prędkości lidera oraz jedną stałą pomocniczą czas reakcji. Ponieważ w prezentowanym modelu stała czas reakcji określa tylko czas opóźnienia, a tym samym ma znaczenie jedynie informacyjne, jej połączenie ze zmienną reakcja podążającego przedstawione jest w postaci linii przerywanej. Łatwo również zauważyć, że niektóre z połączeń zostały wyróżnione podwójnym przekreśleniem – są to połączenia definiujące relacje opóźnione w czasie. Szczegółową dokumentację modelu przedstawiono na rys. 3.

```

mainmodel Mikroskopowy model GHR {
  const alfa {
    autotype Real
    init 2
  }
  const beta {
    autotype Real
    init 2.5
  }
  const Czas reakcji {
    autotype Real
    autounit s
    init 2 <<s>>
  }
  aux Droga lidera {
    autotype Real
    autounit m
    def INTEGRATE('Prędkość lidera')
  }
  aux Droga podążającego {
    autotype Real
    autounit m
    def INTEGRATE('Prędkość podążającego')
  }
  aux Odstęp między pojazdami {
    autotype Real
    autounit m
    def 'Droga lidera'-'Droga podążającego'-'Pozycja początkowa względem lidera'
  }
  aux Opóźnienie {
    autotype Real
    autounit m/s2
    def IF('Zmiana prędkości lidera' < 0 <<m/s2>>; ABS('Zmiana prędkości lidera'); 0 <<m/s2>>)
  }
  aux Opóźnienie podążającego {
    autotype Real
    autounit m/s2
    def IF('reakcja podążającego' < 0 <<m/s2>>; ABS('reakcja podążającego');0<<m/s2>>)
  }
  const Pozycja początkowa względem lidera {
    autotype Real
    autounit m
    init -10 <<m>>
  }
  aux Przebieg prędkości lidera {
    autotype Real
    autounit m/s
    def GRAPH(TIME;0<<@s>>;2<<s>>;{22.2;20;22.2;22.2//Min:0;Max:30//}<<m/s>>)
  }
  aux Przyspieszenie {
    autotype Real
    autounit m/s2
    def IF('Zmiana prędkości lidera' > 0 <<m/s2>>; 'Zmiana prędkości lidera'; 0 <<m/s2>>)
  }
  aux Przyspieszenie podążającego {
    autotype Real
    autounit m/s2
    def IF('reakcja podążającego' > 0 <<m/s2>>;'reakcja podążającego';0<<m/s2>>)
  }
  level Prędkość lidera {
    autotype Real
    autounit m/s
    init 22.2<<m/s>>
    inflow { autodef Przyspieszenie }
    outflow { autodef Opóźnienie }
  }
  level Prędkość podążającego {
    autotype Real
    autounit m/s
    init 22.2<<m/s>>
    inflow { autodef 'Przyspieszenie podążającego' }
    outflow { autodef 'Opóźnienie podążającego' }
  }
  aux reakcja podążającego {
    autotype Real
    autounit m/s2
    def 'Współczynnik czułości' * ((1<<s/m>>*Prędkość podążającego)^alfa) * (DELAYPPL('Prędkość lidera';
    'Czas reakcji')-DELAYPPL('Prędkość podążającego';Czas reakcji)) / ((1 <<1/m>> * DELAYPPL
    ('Odstęp między pojazdami';Czas reakcji))^beta)
  }
  const Współczynnik czułości {
    autotype Real
    autounit s-1
    init 0.4<<1/s>>
  }
  aux Zmiana prędkości lidera {
    autotype Real
    autounit m/s2
    def DERIVN('Przebieg prędkości lidera';1)
  }
}
unit m {
  def ATOMIC
}

```

Rys. 3 Dokumentacja modelu podążania pojazdów

Źródło: opracowanie własne

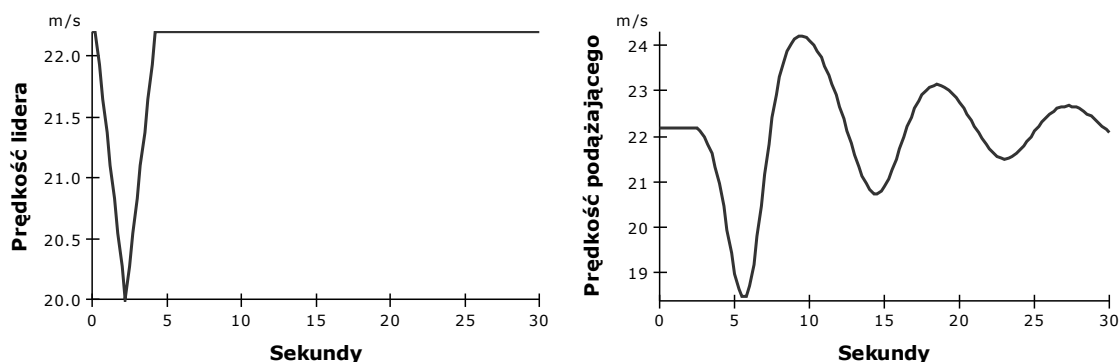
Dokumentacja zastosowanych funkcji systemowych w przedstawionej implementacji modelu podążania pojazdów została zamieszczona w tabeli 1.

Tab. 1 Dokumentacja zastosowanych funkcji systemowych

Nazwa funkcji	Działanie	Pierwszy argument	Pozostałe argumenty
Abs	moduł z wartości zmiennej lub wyrażenia	zmienna lub wyrażenie	-
Delayppl	opóźnienie przepływu informacji o wartości zmiennej	zmienna	stała definiująca czas opóźnienia
Integrate	obliczenie wartości całki	zmienna	-
If	funkcja zależna od wartości logicznej warunku	warunek	zmienna lub wyrażenie przyjmowane odpowiednio w zależności od wartości logicznej warunku
Graph	wartość wyznaczona na podstawie interpolacji funkcji dyskretnej	zmienna niezależna	wartość minimalna zmiennej niezależnej, odstęp między zdefiniowanymi wartościami zmiennej niezależnej, zdefiniowane wartości funkcji
Derivin	obliczenie pochodnej odpowiedniego rzędu	zmienna	rząd pochodnej

Źródło: opracowanie własne

Symulację procesu podążania z wykorzystaniem zdefiniowanego modelu przeprowadzono dla okresu czasu wynoszącego 30 sekund z rozdzielczością czasową 1/4 sekundy stosując do całkowania metodę Rungego-Kutty II rzędu. Rozpatrywano reakcję pojazdu podążającego na pojedynczą chwilową zmianę prędkości lidera. Na rys. 4 zamieszczono wynik symulacji przedstawiający przyjętą zmianę w czasie prędkości lidera oraz obliczony przebieg prędkości pojazdu podążającego.



Rys. 4 Wynik symulacji dla mikroskopowego modelu GHR

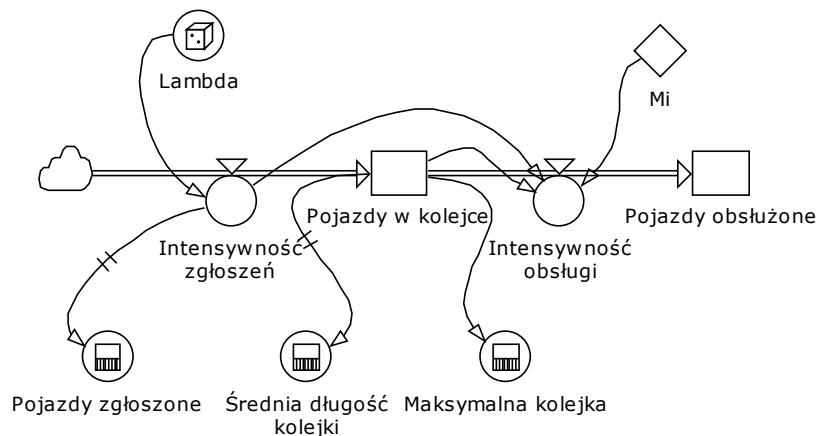
Źródło: opracowanie własne

Jak łatwo zauważyć przyjęty w analizowanym modelu czas reakcji oraz współczynnik czułości prowadzi do uzyskania rozwiązania o charakterze oscylacyjnym. W wyniku realizacji procesu podążania odległość między pojazdami oscyluje ze zmniejszającą się amplitudą, aby z czasem zaniknąć.

Metoda dynamiki systemowej pozwala także na implementację modeli symulacyjnych, których dynamika jest wynikiem procesów losowych. Przykład takiego modelu przedstawiono w kolejnym rozdziale.

3. MODEL OBSŁUGI MASOWEJ TYPU M/D/1 ZREALIZOWANY PRZY UŻYCIU METODY DYNAMIKI SYSTEMOWEJ

W niniejszym rozdziale metoda dynamiki systemowej zostanie zastosowana do implementacji modelu systemu masowej obsługi, w którym w wyniku nierównomierności zgłoszeń dochodzi do powstawania kolejek. Rozważany jest model systemu, w którym intensywność zgłoszeń ma rozkład wykładniczy przy ustalonej intensywności obsługi. Model tego typu w klasyfikacji Kendalla jest przedstawiany jako model M/D/1. Przyjęto ponadto, że średnia intensywność zgłoszeń jest mniejsza niż intensywność obsługi. Model takiego systemu pozwala analizować sytuację, w której w wyniku nierównomierności ruchu dochodzi do powstawania kolejek pomimo nominalnie wystarczającej przepustowości elementu infrastruktury transportowej. Schemat strukturalny rozważanego modelu przedstawiono na rys. 5.



Rys. 5 Schemat strukturalny dla modelu obsługi pojazdów typu M/D/1
Źródło: opracowanie własne

Do analizy rozważanego procesu przyjęto, że intensywność strumienia zgłoszeń ma rozkład wykładniczy, co wymaga zastosowania funkcji generującej liczbę losową o takim rozkładzie. W tym celu wykorzystano funkcję `EXPRND`, której parametrem jest oczekiwana wartość średnia. Z kolei obliczenie maksymalnej kolejki wymaga zastosowania funkcji `RUNMAX` zwracającej wartość maksymalną zmiennej podanej jako jej argument w trakcie symulacji. Wartość ta jest zaokrąglana do najbliższej liczby całkowitej poprzez zastosowanie funkcji `ROUND`. W implementacji wykorzystana została także funkcja `MIN` odpowiadająca za użycie najmniejszej wartości spośród podanych jako jej argumenty. Szczegółowy opis przyjętych danych i zastosowanych jednostek przedstawiono na rys. 6.

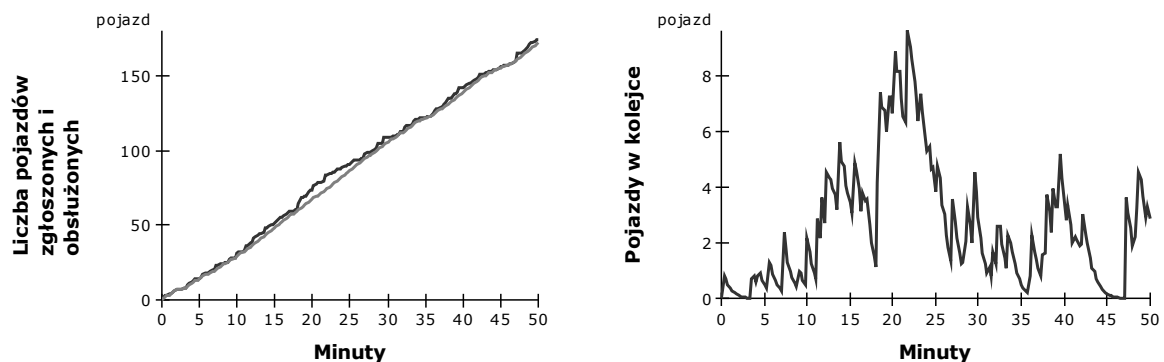
Symulację procesów obsługi z wykorzystaniem zaimplementowanego modelu o losowym strumieniu zgłoszeń przeprowadzono dla okresu czasu wynoszącego 50 minut z rozdzielczością czasową jednej minuty. Na rys. 7 zamieszczono przykład wyniku symulacji przedstawiający zmiany w czasie liczby pojazdów oczekujących w kolejce oraz liczby zgłoszonych pojazdów (pogrubiona linia) i obsłużonych do danej chwili czasu.

```

mainmodel Model obsługi pojazdów typu M_D_1 {
  aux Intensywność obsługi {
    autotype Real
    autounit pojazd/minuta
    def MIN(Mi;'Pojazdy w kolejce'/1<<minuta>> + 'Intensywność zgłoszeń')
  }
  aux Intensywność zgłoszeń {
    autotype Real
    autounit pojazd/minuta
    def Lambda
  }
  aux Lambda {
    autotype Real
    autounit pojazd/minuta
    def EXPRND(3)* 1<<pojazd/minuta>>
  }
  aux Maksymalna kolejka {
    autotype Real
    autounit pojazd
    def ROUND(RUNMAX('Pojazdy w kolejce'))
  }
  const Mi {
    autotype Real
    autounit pojazd/minuta
    init 4 <<pojazd/minuta>>
  }
  level Pojazdy obsłużone {
    autotype Real
    autounit pojazd
    init 0<<pojazd>>
    inflow { autodef 'Intensywność obsługi' }
  }
  level Pojazdy w kolejce {
    autotype Real
    autounit pojazd
    init 0<<pojazd>>
    inflow { autodef 'Intensywność zgłoszeń' }
    outflow { autodef 'Intensywność obsługi' }
  }
  aux Pojazdy zgłoszone {
    autotype Real
    autounit pojazd
    def INTEGRATE('Intensywność zgłoszeń')
  }
  aux Średnia długość kolejki {
    autotype Real
    autounit pojazd
    def INTEGRATE('Pojazdy w kolejce')/(STOPTIME - STARTTIME)
  }
}
unit pojazd {
  def ATOMIC
}

```

Rys. 6 Dokumentacja modelu obsługi pojazdów typu M/D/1
Źródło: opracowanie własne



Rys. 7 Przykład wyniku symulacji dla modelu obsługi pojazdów typu M/D/1
Źródło: opracowanie własne

Oczywiście w konsekwencji wykorzystania generatora liczb losowych, przedstawione wyniki dotyczące powstającej kolejki oczekujących pojazdów po ponownym uruchomieniu symulacji będą inne. Z tej przyczyny dopiero wielokrotne powtórzenie symulacji pozwala na

sformułowanie wniosków odnośnie średniej oczekiwanej kolejki pojazdów w rozpatrywanym systemie obsługi. Wykorzystane w niniejszej pracy środowisko Powersim udostępnia również dodatkowe narzędzia pozwalające przeprowadzać wielokrotną symulację.

PODSUMOWANIE

Przedstawione w artykule implementacje modeli stanowią tylko przykład możliwych zastosowań metody dynamiki systemowej w badaniach symulacyjnych związanych z zagadnieniami transportowymi. W szczególności interdyscyplinarność zagadnień związanych z funkcjonowaniem systemów transportowych jest istotną przesłanką do zastosowania metody dynamiki systemowej przy tworzeniu modeli symulacyjnych systemów transportowych lub ich wybranych elementów. Analizując modele prezentowane w literaturze znaleźć można przykłady zastosowań modeli implementowanych z użyciem metody dynamiki systemowej do analiz:

- procesów obsługi [8],
- kosztów zewnętrznych transportu [1,3],
- wpływu polityki transportowej na system transportowy [2,4,10].

Warto tutaj podkreślić, że implementację modeli przedstawionych zarówno w niniejszej pracy jak i w pracach przywołanych powyżej można zrealizować w dowolnym środowisku wspierającym wykorzystanie metody dynamiki systemowej choć może to wymagać wprowadzenia pewnych zmian (np. związanych z innymi nazwami funkcji wbudowanych w danym środowisku). Spośród najbardziej popularnych wskazać można środowisko Vensim, które jest dostępne w nieograniczonej czasowo i nieodpłatnej wersji edukacyjnej. Środowisko to jest jednocześnie wystarczająco szczegółowo opisane w ogólnie dostępnych pozycjach monograficznych [6, 7].

BIBLIOGRAFIA

1. Chen C., Shao L., Xu, L. Shang J., *A case study predicting environmental impacts of urban transport planning in China*. Environ Monit Assess 2009, 157: 169–177.
2. Goh Y.M., Love P.E.D., *Methodological application of system dynamics for evaluating traffic safety Policy*. Safety Science 2012, 50: 1594–1605.
3. Han J., Hayashi Y., *A system dynamics model of CO₂ mitigation in China's inter-city passenger transport*. Transportation Research Part D 2008, 13: 298–305.
4. Hang W., Li X., *Application of system dynamics for evaluating truck weight regulations*. Transport Policy 2010, 17: 240–250.
5. Hannon B., Ruth M., *Dynamic modeling*. Springer-Verlag, New York 2001.
6. Krupa K., *Modelowanie, symulacja i prognozowanie. Systemy ciągłe*. WNT, Warszawa 2008.
7. Łatuszyńska M., *Symulacja komputerowa dynamiki systemów*. Wydawnictwo PWSZ, Gorzów Wielkopolski 2008.
8. Manataki I.E., Zografos K.G., *A generic system dynamics based tool for airport terminal performance analysis*. Transportation Research Part C 2009, 17: 428–443.
9. Tarajkowski J. (red.naukowy), *Elementy dynamiki systemów*. Wydawnictwo Akademii Ekonomicznej w Poznaniu, Poznań 2008.
10. Wang J., Lu H., Peng H., *System dynamics model of urban transportation system and its application*. Journal of Transportation Systems Engineering and Information Technology 2008, 8(3): 83-89.

AN APPLICATION OF THE SYSTEM DYNAMICS TO ANALYSIS OF TRANSPORTATION ISSUES

Abstract

In the paper are presented two examples of computational models concerning selected transportation issues. The system dynamics method was applied for model implementation. This method has many advantages against other techniques of computational modeling. First of all skills in programming languages are not required in order to formulate computational models. Moreover, the range of method application covers many continuous issues in time like: logistics chains, traffic engineering tasks, maintenance of transport means. At last the system dynamic method can be an effective tool which supports education of transportation engineers.

Autorzy:

dr hab. inż. **Krzysztof Brzozowski** – Akademia Techniczno-Humanistyczna w Bielsku-Białej
mgr inż. **Justyna Sordyl** – Akademia Techniczno-Humanistyczna w Bielsku-Białej