

Na prawach rękopisu

INSTYTUT CYBERNETYKI TECHNICZNEJ
POLITECHNIKI WROCŁAWSKIEJ
Raport serii: Sprawozdania nr 3/2002

**Symulacyjne badania
prostego układu dynamicznego
(Ćwiczenie 525)**

Alicja Mazur
Marek Wnuk

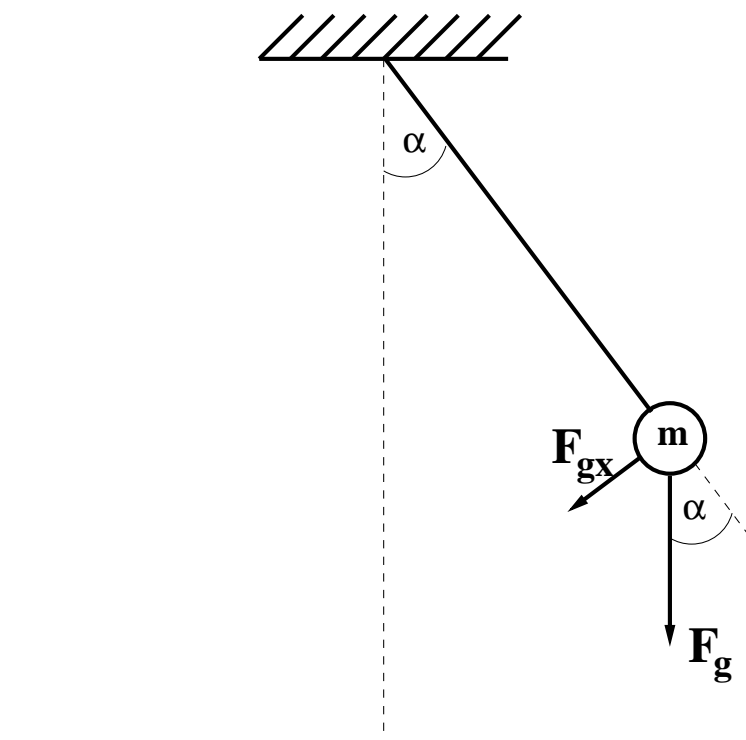
Słowa kluczowe: układ dynamiczny, symulacja

Wrocław 2002

1 Cel ćwiczenia

Celem ćwiczenia jest zaznajomienie z programem TUTSIM i nabycie umiejętności modelowania układów dynamicznych. Ponadto należy zaobserwować różnice w zachowaniu układów w przypadkach, gdy w opisie dynamiki uwzględniamy pełne, nieliniowe oddziaływania, uwzględniamy tylko pewne przybliżenia działających sił lub stosujemy uproszczenia poprzez pominięcie np. sił tarcia.

2 Matematyczne sformułowanie problemu



Rysunek 1: Wahadło matematyczne w polu grawitacyjnym Ziemi.

Rozważmy model matematycznego 1-wahadła przedstawionego na rysunku 1. Masa całego wahadła jest skupiona na końcu nieważkiej linki o długości l łączącej wahadło z punktem zaczepienia. Wahadło jest umieszczone w polu grawitacyjnym Ziemi. Stan wahadła opisuje kąt α - odchylenie wahadła od pionu. Zgodnie z II zasadą dynamiki Newtona, pod wpływem sił zewnętrznych układ zacznie się poruszać z pewnym przyspieszeniem

$$m \frac{d^2 \alpha}{dt^2} = \sum F_{zewn}. \quad (1)$$

W dalszej części rozważymy różne przypadki oddziaływań zewnętrznych działających na badany układ.

2.1 Zachowanie układu pod wpływem grawitacji

Rozważmy przypadek, gdy na układ oddziałuje jedynie siła grawitacji, patrz rys.1. Zmiana stanu układu zachodzi pod wpływem składowej siły grawitacji prostopadłej do nici

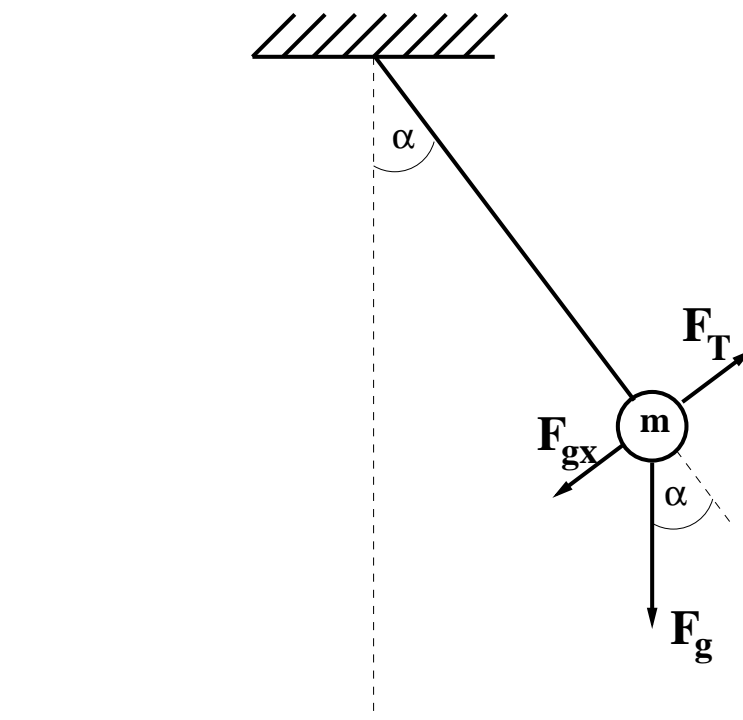
$$\frac{F_{gx}}{F_g} = \sin \alpha. \quad (2)$$

Napiszmy równanie ruchu wahadła matematycznego dla tego przypadku

$$m \frac{d^2 \alpha}{dt^2} = -F_g \sin \alpha = -mgl \sin \alpha. \quad (3)$$

Powyższe równanie jest nieliniowym równaniem różniczkowym, dla którego nie istnieje rozwiązanie analityczne. Siła przyłożona do kulki ma znak ujemny, gdyż przeciwdziała ona wychyleniu kulki od pionu.

2.2 Zachowanie układu pod wpływem grawitacji i sił tarcia wiskotycznego



Rysunek 2: Wahadło matematyczne w polu grawitacyjnym Ziemi z uwzględnionym tarciem.

Rozważmy z kolei przypadek przedstawiony na rys.2, gdy na układ oddziałuje siła grawitacji oraz tarcie wiskotyczne, które ma postać

$$F_T = F_{TW} \frac{d\alpha}{dt},$$

gdzie stały współczynnik F_{TW} zależy od ośrodka, w którym ciało się porusza. Zmiana stanu układu zachodzi pod wpływem składowej siły grawitacji prostopadłej do nici oraz siły tarcia

$$F_{zewn} = -F_g \sin \alpha + F_{TW} \frac{d\alpha}{dt}. \quad (4)$$

Napiszmy równanie ruchu wahadła matematycznego dla tego przypadku

$$m \frac{d^2\alpha}{dt^2} = -mgl \sin \alpha + F_{TW} \frac{d\alpha}{dt}. \quad (5)$$

Równanie (5) jest również nieliniowym równaniem różniczkowym, dla którego nie istnieje rozwiązanie analityczne. Ponadto, w rzeczywistych przypadkach współczynnik F_{TW} ma wartości ujemne (tarcie wpływa hamująco na ruch wahadła).

2.3 Zachowanie układu dla małych wartości kąta wychylenia

Obydwa powyższe przypadki opisane równaniami (3) i (5) są nieliniowymi równaniami różniczkowymi (ze względu na postać składowej siły grawitacji), dla których nie istnieją rozwiązania analityczne, czyli dokładne. Można jednak uzyskać przybliżone rozwiązanie obu przypadków, gdy przyjmie się założenie, że kąty wychylenia α będą niewielkie, gdyż dla małych wartości kątów zachodzi zależność

$$\sin \alpha \approx \alpha.$$

Wówczas równania nieliniowe (3) i (5) przyjmują postać przybliżoną

$$m \frac{d^2\alpha}{dt^2} = -mgl\alpha, \quad (6)$$

oraz dla przypadku z tarciem (gdzie współczynnik F_{TW} ma wartość nieco większą niż rzeczywista ale przydatną do celów symulacyjnych)

$$m \frac{d^2\alpha}{dt^2} = -mgl\alpha + F_{TW} \frac{d\alpha}{dt}, \quad (7)$$

dla której istnieje rozwiązanie analityczne (dokładne). W oparciu o teorię przedstawioną w [3], napiszmy postać ogólną rozwiązania analitycznego. Dla przypadku samej grawitacji (równanie przybliżone (6)) rozwiązanie analityczne ma postać ogólną

$$\alpha(t) = a \cdot \sin(\sqrt{gl} \cdot t) + b \cdot \cos(\sqrt{gl} \cdot t), \quad (8)$$

a po zróżniczkowaniu otrzymujemy wyrażenie na prędkość

$$\dot{\alpha}(t) = a \cdot \sqrt{gl} \cdot \cos(\sqrt{gl} \cdot t) - b \cdot \sqrt{gl} \cdot \sin(\sqrt{gl} \cdot t), \quad (9)$$

gdzie a i b są stałymi zależnymi od warunków początkowych. Przyjmijmy, że w chwili początkowej kulka spoczywała w pozycji odchyłonej od pionu o kąt α równy 1 rad (tj. około 57 stopni). To oznacza, że warunki początkowe wynosiły

$$\alpha(0) = 1, \quad \dot{\alpha}(0) = 0.$$

Dla takich warunków początkowych wyliczamy wartości stałych a i b

$$\alpha(0) = a \cdot 0 + b \cdot 1 = b = 1.$$

Z warunku początkowego na prędkość zmiany kąta wyliczamy

$$\dot{\alpha}(0) = a \cdot \sqrt{gl} = 0$$

czyli rozwiązanie szczególne wynosi

$$\alpha(t) = \cos(\sqrt{gl} \cdot t).$$

Postać ogólną rozwiązania analitycznego dla przypadku grawitacji i tarcia (równanie przybliżone (7)) ma postać

$$\alpha(t) = a \cdot e^{-\frac{F_{TW}}{2}} \cdot \sin\left(\sqrt{\frac{4gl - F_{TW}^2}{4}} \cdot t\right) + b \cdot e^{-\frac{F_{TW}}{2}} \cdot \cos\left(\sqrt{\frac{4gl - F_{TW}^2}{4}} \cdot t\right). \quad (10)$$

Po zróżniczkowaniu i otrzymaniu wyrażenia na prędkość, obliczamy stałe a i b dla tych samych warunków początkowych, co poprzednio.

$$\alpha(0) = a = 1,$$

$$\dot{\alpha}(0) = -\frac{F_{TW}}{2} + b\sqrt{\frac{4gl - F_{TW}^2}{4}} = 0,$$

czyli rozwiązanie szczególne wynosi

$$\alpha(t) = e^{-\frac{F_{TW}}{2}} \cdot \sin\left(\sqrt{\frac{4gl - F_{TW}^2}{4}} \cdot t\right) + \frac{F_{TW}}{\sqrt{4gl - F_{TW}^2}} \cdot e^{-\frac{F_{TW}}{2}} \cdot \cos\left(\sqrt{\frac{4gl - F_{TW}^2}{4}} \cdot t\right). \quad (11)$$

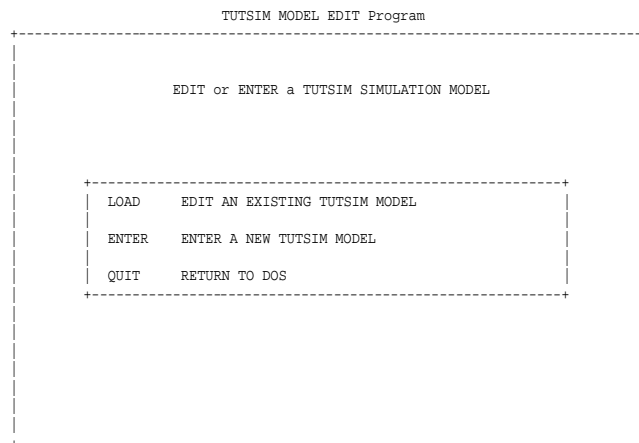
Dla celów symulacji przyjmujemy wartości stałych równe

$$g = 9.81 \left[\frac{m}{s^2}\right], \quad l = 1 [m], \quad F_{TW} = 0.7.$$

3 Badania symulacyjne

3.1 Uruchomienie programu TUTSIM

Uruchomienie programu TUTSIM służącego do symulacji układów dynamicznych należy rozpocząć od uruchomienia (poprzez wciśnięcie klawisza ENTER) programu **tutsim.exe**. W następnej chwili na ekranie pojawi się okienko informacyjne z napisem TUTSIM. Po krótkiej chwili pierwsze okienko zniknie automatycznie a następnie pojawi się okienko z informacjami o wersji TUTSIM-a, licencji i dacie rejestracji oraz o procesie ładowania niezbędnych podprogramów. Następnie pojawi się automatycznie LOAD menu z części TUTEDIT (rys. 3).

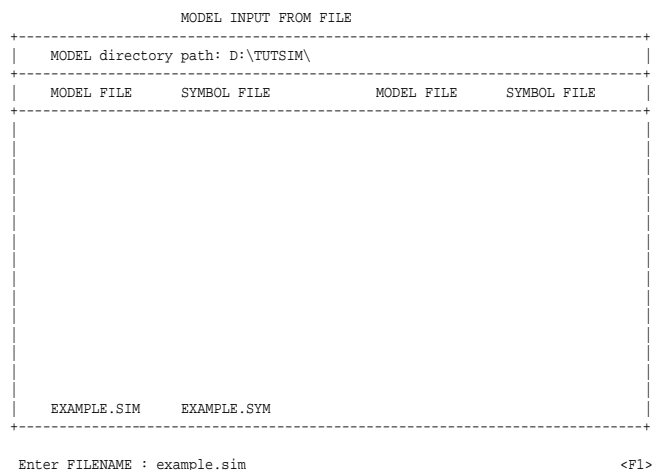


Rysunek 3: Menu LOAD programu **tutedit.exe**

To okienko pozwala na załadowanie istniejącego modelu (opcja **LOAD**), wprowadzenie nowego, własnego programu (opcja **ENTER**) lub powrót do systemu DOS (opcja **QUIT**). Żądaną opcję należy wybrać kursorem a następnie wcisnąć klawisz ENTER.

3.2 Przykładowe zadanie - **example.sim**

Pracę z programem TUTESIM rozpoczniemy od uruchomienia przykładowego, istniejącego modelu. W tym celu należy z LOAD MENU wybrać opcję **LOAD** a następnie podać nazwę wybranego pliku (**example.sim**) i potwierdzić poprzez wciśnięcie klawisza ENTER (rys. 4).



Rysunek 4: Ekran ładowania programu (LOAD)

Po wybraniu modelu pojawi się nowe okienko, a mianowicie TUTEDIT MAIN MENU (rys. 5). Okienko to umożliwi nam poprawienie modelu, jego zapamiętanie na dysku oraz uruchomienie (opcja **QUIT - START CALCULATIONS**).

```

TUTEDIT MAIN MENU
+-----+
|       |
| EDIT   CHANGE MODEL STRUCTURE |
|       |
| SAVE   SAVE MODEL TO FILE     |
|       |
| UTIL   MODEL LIST UTILITIES   |
|       |
| CONCAT ADD A SECOND MODEL     |
|       |
| BACK   BACK TO LOAD MENU      |
|       |
| QUIT   START CALCULATIONS     |
|       |
+-----+

```

Rysunek 5: Menu programu **tutedit.exe**

Ustawienie kursora na opcji **QUIT** oraz wciśnięcie klawisza ENTER powoduje przejście do części obliczeniowej programu TUTSIM i wyświetlenie okienka TUTSIM CALCULATION CONTROL PANEL.

```

----- TUTSIM CALCULATION CONTROL PANEL -----
+-----+
| COMMANDS | OUTPUT of RESULTS | UTILITIES |
+-----+
| Start Simulation | Graphic to Display | Graphics Mode: LINE |
| Proceed after interrupt | Numeric to Display | X/Y centerlines: OFF |
| Change Parameter (Temporary) | Numeric to Printer | Output Interval: 0 |
| Multi Run start | Binary to File | Signal Interval: 0 |
| Estimate Parameters | ASCII to File | Iteration Count: 0 |
| Delete Graphics | MATLAB to File | Real Time Step: OFF |
| Graphics Reconstruction | PrintPlot to Display | Random Noise: RESET |
| Hardcopy of Graphics | No output of results | Display Labels: OFF |
| TUTSIM Title Change | | Add Old Graphic: OFF |
| Filename Change DAT function | | |
| Verify Results | | |
| Initial Values Replace | | |
| Reset Real Time I/O | | |
| QUIT TUTCALC to CHANGE MODEL | | |
+-----+
+-----+
| On 80x87 | <<< Press <Enter> to INTERRUPT during Calculations >>> |
+-----+

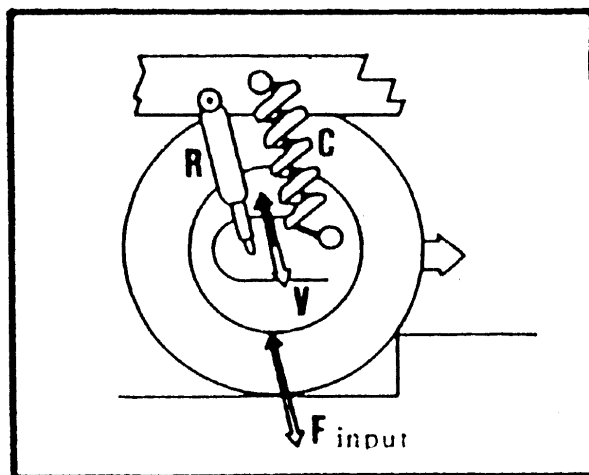
```

Rysunek 6: Ekran sterowania obliczeniami programu **tutsim.exe**

Dwukrotne wciśnięcie klawisza ENTER spowoduje wybranie domyślnych (podświetlonych) opcji i uruchomienie obliczeń.

Przykładowe zadanie ma na celu zaznajomienie użytkownika z programowaniem w TUTSIM-ie. Postawione zadanie obejmuje symulację zachowania układu przedstawionego na rys.7, prezentującego napędzane koło z amortyzatorem, opisane równaniem

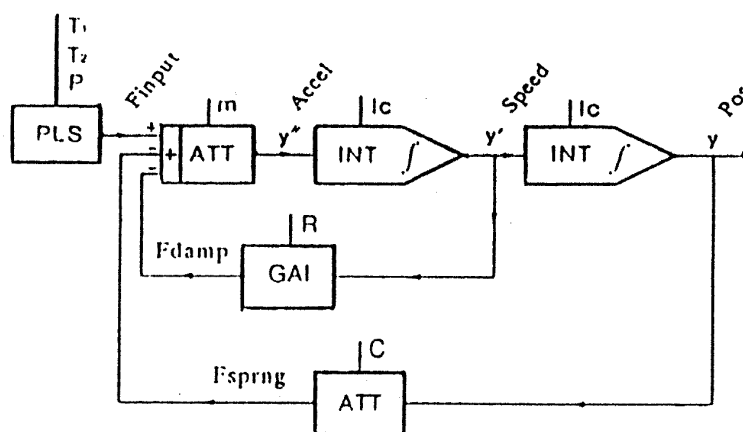
$$M\ddot{y} = F_{input} - R\dot{y} - \frac{1}{C}y, \quad (12)$$



Rysunek 7: Napędzane koło - układ dynamiczny z przykładowego zadania example.sim

gdzie symbolem M oznaczono masę koła, y jest położeniem koła, F_{input} jest zewnętrzną siłą sterującą przyłożoną do osi koła, zaś R i C są to stałe współczynniki. Prawidłowy proces przygotowania równania do zamodelowania w TUTSIM-ie wymaga przedstawienia opisu układu (12) w postaci równania na najwyższą pochodną zmiennej badanej ze współczynnikiem 1, tzn. że dla równania (12) musimy zapisać

$$\ddot{y} = \frac{1}{M} \left(F_{input} - R\dot{y} - \frac{1}{C}y \right). \quad (13)$$



Rysunek 8: Schemat układu dynamicznego przedstawionego w przykładzie example.sim

Schemat przedstawiający bloki użyte do realizacji układu w TUTSIM-ie został pokazany na rys.8. Jak widać, funkcję y i jej pierwszą pochodną \dot{y} uzyskuje się poprzez kolejne całkowanie sygnałów

będących reprezentacją prawej strony równania (13)(dwa kolejne bloki INT). Kompletny zestaw bloków realizujących równanie (13) opisanych przy użyciu TUTSIM-a został pokazany na rys.9.

```

Model File: D:\ZAJECIA\TUTSIM\example.MOD
System Date:  2 /23 /2002
System Time: 11 :  8

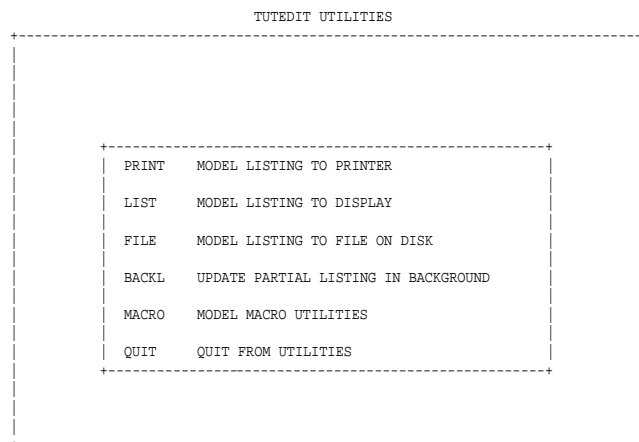
Accel=ATT[Finput-Fdamp-Fsprng]           ;Acceleration
    Attenuation           : 1.00000E+1
FINPUT=PLOT[Finput]                     ;PLOT INPUT SIGNAL
    PLOT number          : 1.00000
    Minimum               :-1.00000
    Maximum               : 9.00000
Fdamp=GAI[Speed]                        ;Damper
    Gain                  : 5.00000E-1
Finput=PLS[]                             ;Input Step
    Start time           : 1.00000E+1
    End time             : 2.00000E+2
    Signal value         : 1.00000
Fsprng=ATT[Pos]                          ;Spring
    Attenuation           : 1.00000E+1
POS=PLOT[Pos]                            ;PLOT POSITION
    PLOT number          : 2.00000
    Minimum               : 0.000000
    Maximum               : 2.00000E+1
Pos=INT[Speed]                           ;Position of Mass
    Initial value        : 0.000000
SPEED=PLOT[Speed]                        ;PLOT VELOCITY
    PLOT number          : 3.00000
    Minimum               :-1.00000
    Maximum               : 1.00000
Speed=INT[Accel]                          ;Velocity of Mass
    Initial value        : 0.000000
t=TIME[]                                  ;General Timing
    Time step DELTA     : 5.00000E-1
    End time            : 2.00000E+2

```

Rysunek 9: Program w języku TUTSIM realizujący układ dynamiczny z przykładu example.sim

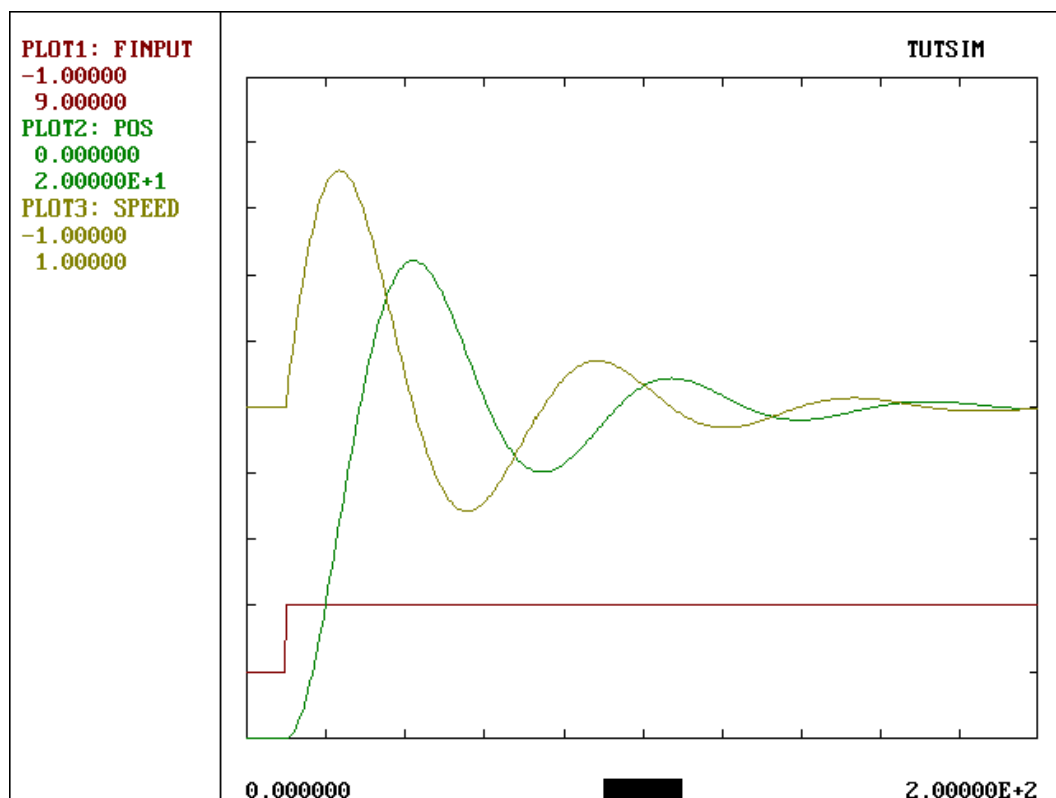
Zawartość programu w postaci pliku znakowego można uzyskać wybierając z funkcji pomocniczych

(UTIL – rys. 10) edytora (rys. 5) opcję **FILE**.



Rysunek 10: Ekran funkcji pomocniczych **tutedit.exe**

Po zakończeniu obliczeń wyniki zostaną wyświetlone w postaci graficznej jako wykresy (rys. 11).



Rysunek 11: Wynik przykładowej symulacji

4 Zadania do samodzielnego wykonania

Aby wykonać ćwiczenie należy zamodelować dokładne (nieliniowe) i przybliżone (liniowe) równania opisujące zachowanie pojedynczego wahadła oraz porównać przebiegi trajektorii na płaszczyźnie fazowej. Wykres na płaszczyźnie fazowej jest jednym z najczęściej wykorzystywanych narzędzi w automatyce służących do badania zachowania i stabilności układów dynamicznych. Wykres taki przypomina płaszczyznę kartezjańską, przy czym na osi OX zaznaczana jest prędkość, natomiast na osi OY zaznaczane jest położenie odpowiadające tej prędkości.

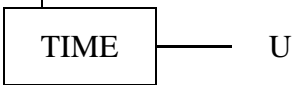
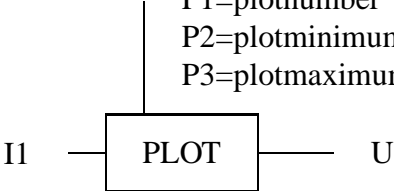
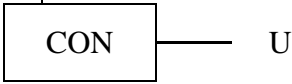
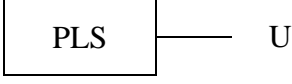
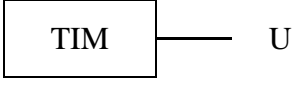
Modelowanie układu możemy rozpocząć dopiero po powrocie do okienka LOAD w menu TUTEDIT. Zgodnie z listą dostępnych bloków przedstawioną w rozdziale 5, należy zamodelować równania wahadła dokładne i przybliżone. Wszystkie parametry równania należy wprowadzić jako konkretne wartości liczbowe (TUTSIM nie potrafi przetwarzać wartości symbolicznych).

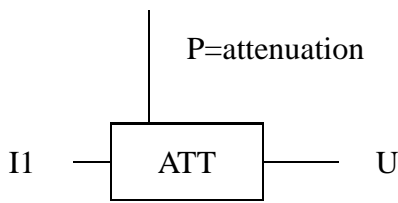


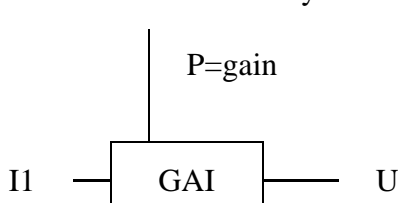

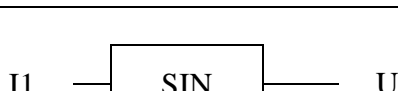
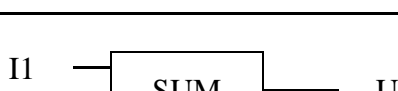
Badania symulacyjne należy przeprowadzić dla przypadku, gdy jedyną działającą siłą zewnętrzną jest grawitacja oraz dla przypadku, gdy oprócz grawitacji działa siła tarcia. Najpierw należy zamodelować równania (3) i (6), narysować wykresy fazowe dla obu przypadków i zauważyć różnicę. Następnie należy zamodelować równania dla przypadku z tarciami (5) i (7) i również sporządzić dla nich wykresy fazowe. Czy tym razem wystąpiły różnice? Na koniec należy porównać wykres fazowy przypadków bez tarcia i z tarciami i wyciągnąć wnioski.

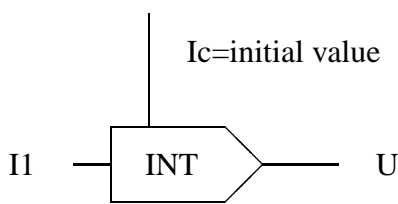
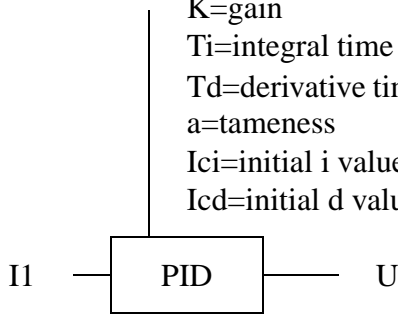
UWAGA: Aby uzyskać wykres fazowy, należy jako blok PLOT wprowadzić prędkość układu i nadać parametrowi *plotnumber* wartość 0. Położenie należy również wprowadzić jako kolejny blok PLOT z numerem 1. Wówczas wykres nie będzie zależał w sposób jawny od czasu - otrzymamy wykres fazowy.

5 Zestaw podstawowych bloków TUTSIM-a

Zestawienie wszystkich bloków dostępnych w programie **tutsim** można znaleźć w [1]. Poniżej podano opis wybranych bloków (użytych w przykładzie i niezbędnych do wykonania ćwiczenia).

Nazwa	Funkcja	Symbol
TIME	definiuje krok obliczeń <i>DELTA</i> (<i>timestep</i>) i horyzont czasowy obliczeń <i>END TIME</i> (<i>endtime</i>). Musi wystąpić w każdym modelu (w razie jej braku <i>DELTA</i> i <i>END TIME</i> zostaną przyjęte jako równe 1.0. Na wyjściu <i>U</i> jest dostępna wartość bieżąca czasu symulacji. Parametry: <i>P1</i> = <i>timestep</i> , <i>P2</i> = <i>endtime</i> .	<p>Parametry:</p> <p>P1=timestep P2=endtime</p> 
PLOT	służy do specyfikowania wyjść <i>X</i> i Y_n dla wizualizacji wyników symulacji. <i>Plotnumber</i> wybiera Y_n ($1 \leq n \leq 8$); $n = 0$ wybiera <i>X</i> . Brak specyfikacji <i>X</i> daje wykres w funkcji czasu od 0 do <i>END TIME</i> . Wejście <i>I1</i> daje wartość <i>X</i> lub Y_n , a wyjście <i>U</i> jest mu równe.	<p>Parametry:</p> <p>P1=plotnumber P2=plotminimum P3=plotmaximum</p> 
CON	służy do zadawania stałej wartości <i>constant value</i> .	<p>Parametry:</p> <p>P1=constant value</p> 
PLS	służy do zadawania funkcji impulsowej lub skokowej: $U(t) = P \Leftrightarrow T1 < t < T2$.	<p>Parametry:</p> <p>T1=start time T2=end time P=pulse amplitude</p> 
TIM	dostarcza bieżącą wartość czasu symulacji.	

Nazwa	Funkcja	Symbol
ATT	łumienie (<i>attenuation</i>): $U(t) = \frac{I_1(t)}{P}$.	<p>Parametry:</p> <p>P=attenuation</p> 
COS	funkcja kosinus: $U(t) = \cos(I_1(t))$.	
EXP	funkcja wykładnicza: $U(t) = e^{I_1(t)}$.	
GAI	wzmocnienie (<i>gain</i>): $U(t) = P \cdot I_1(t)$.	<p>Parametry:</p> <p>P=gain</p> 
MUL	mnożenie: $U(t) = I_1(t) \cdot I_2(t)$.	
SIN	funkcja sinus: $U(t) = \sin(I_1(t))$.	
SUM	dodawanie: $U(t) = I_1(t) + I_2(t)$.	

Nazwa	Funkcja	Symbol
INT	<p>blok całkujący: $U = \int I_1 dt$. Algorytm całkowania Adamsa-Bashforta:</p> $U(0) = I_c$ $U(1) = I_c + 1.5 \cdot \Delta$ $U(t) = U(t - \Delta) + 0.5 \cdot \Delta \cdot (3 \cdot I_1(t - \Delta) - I_1(t - 2 \cdot \Delta))$	<p>Parametry: Ic=initial value</p> 
PID	<p>sterownik PID. $U(t)$ jest związane z $I_1(t)$ przez transmitancję:</p> $K_{pid}(s) = K \cdot \left(\frac{1}{sT_i} + \frac{1 + sT_d}{1 + a \cdot sT_d} \right)$ <p>z ograniczeniami:</p> $T_i > T_d ; a < 1$ <p>(przy $a = 1$ lub $T_d = 0$ sterownik redukuje się do PI).</p>	<p>Parametry: K=gain Ti=integral time Td=derivative time a=tameness Ici=initial i value Icd=initial d value</p> 

Bibliografia

- [1] TUTSIM for IBM's PC Version 7.0 - podręcznik użytkownika, 1992.
- [2] R. Resnick, D. Halliday: *Fizyka*, tom 1, PWN, Warszawa, 1980, wyd.VI.
- [3] M. Gewert, Z. Skoczylas: *Równania różniczkowe zwyczajne: teoria, przykłady, zadania*, z serii: *Matematyka dla studentów politechnik*, Oficyna Wydawnicza GiS, Wrocław, 2000.

dr inż. Alicja Mazur
dr inż. Marek Wnuk
Instytut Cybernetyki Technicznej
Politechniki Wrocławskiej
ul. Janiszewskiego 11/17
50-372 Wrocław

Niniejszy raport otrzymują:

1. OINT - 1 egz.
2. Biblioteka Wydziału Elektroniki - 3 egz.
3. Laboratorium Robotyki (010/C-3) - 3 egz.
4. Autorzy - 2 egz.

Razem : 9 egz.

Raport wpłynął do redakcji I-6
w lutym 2002 roku.