

## Додаток А

### Набір підпрограм автоматизованого форматування графіків

Для автоматизованого форматування графіків використовуємо наданий набір підпрограм, які оформлені у вигляді програмного сценарію, що зберігається у файлі FormatCharts.sce, що має такий зміст:

```
function FormatCharts(titlex, titley, titlec, LEGEND, pos)
    xgrid(0);
    a=get("current_axes");
    a.font_size=3;
    a.x_label.font_size=4;
    a.y_label.font_size=4;
    title(titlec, 'fontsize', 4);
    xlabel(titlex);
    ylabel(titley);
    if length(LEGEND)>0 then
        legend(LEGEND, pos).font_size=4
    end
endfunction

function SetAxesBounds(xmin, xmax, ymin, ymax)
    a=get("current_axes");
    a.data_bounds = [xmin, ymin ; xmax, ymax];
endfunction

function SetXAxesBounds(xmin, xmax)
    a=get("current_axes");
    a.data_bounds = [xmin, a.data_bounds(1,2); xmax, a.data_bounds(2,2)];
endfunction

function SetYAxesBounds(ymin, ymax)
    a=get("current_axes");
    a.data_bounds = [a.data_bounds(1,1), ymin ; a.data_bounds(2,1), ymax];
endfunction
```

## Додаток Б

### Програма-сценарій №1

Для автоматизованого виконання досліджень щодо впливу величин  $\Omega$  кругової (циклічної) частота коливального руху ланки маніпулятора на кут повороту (3.27), точне значення кутової швидкості (3.28) та вимірювану напругу (3.25) розроблена програма-сценарій, що збережена у файлі model1.sci та має зміст такого вигляду:

```
loadXcosLibs(); loadScicos(); exec('FormatCharts.sce',-1);
importXcosDiagram("model1.zcos")
function res=simulation(w, dt)
    typeof(scs_m); scs_m.props.context;
    Context.W=w; Context.dt=dt;
    scicos_simulate(scs_m,Context);
    res=RES
endfunction

titlex="$t, \mathrm{s}$"; titlec=""; dt=0.001;
W=0.3; res=simulation(W,dt);
t=res.time; fi=res.values(:,1); w=res.values(:,2); U=res.values(:,3);
LEGEND=["$\Omega=" + string(W) + "\mathrm{s}^{-1}$"];
show_window(1); plot(t,fi,"k-", "linewidth",2);
show_window(2); plot(t,w,"k-", "linewidth",2);
show_window(3); plot(t,U,"k-", "linewidth",2);
W=0.5; res=simulation(W,dt);
t=res.time; fi=res.values(:,1); w=res.values(:,2); U=res.values(:,3);
LEGEND=[LEGEND, "$\Omega=" + string(W) + "\mathrm{s}^{-1}$"];
scf(1); plot(t,fi,"k--", "linewidth",2);
scf(2); plot(t,w,"k--", "linewidth",2);
scf(3); plot(t,U,"k--", "linewidth",2);
W=0.7; res=simulation(W,dt);
t=res.time; fi=res.values(:,1); w=res.values(:,2); U=res.values(:,3);
LEGEND=[LEGEND, "$\Omega=" + string(W) + "\mathrm{s}^{-1}$"];
scf(1); plot(t,fi,"k:", "linewidth",2);
titley="$\varphi(t)$";
FormatCharts(titlex,titley,titlec,LEGEND,4); xsave("model1_res1.scg");
scf(2); plot(t,w,"k:", "linewidth",2);
titley="$\omega(t), \mathrm{s}^{-1}$";
FormatCharts(titlex,titley,titlec,LEGEND,4); xsave("model1_res2.scg");
scf(3); plot(t,U,"k:", "linewidth",2);
titley="$U(t), \mathrm{V}$";
FormatCharts(titlex,titley,titlec,LEGEND,4); xsave("model1_res3.scg");
```

## Додаток В

### Набір підпрограм-сценаріїв

Для виконання обчислень кутової швидкості за формулою (3.26), в якій похідна вимірюваної (змодельованої) у вигляді за (2.24), (2.25) напруги має визначатися за однією із формул наближеного диференціювання (2.35), (2.47) або (2.63), а також для обчислення похибки визначення кутової швидкості за формулою (3.33) розроблено набір підпрограм-сценаріїв, які збережені у файлі AV\_comp.sce, що має такий вигляд:

```
function res=AV_comp1(t, U)
    dt=t(2)-t(1)
    N=length(U);
    dudt=zeros(N-1,1);
    dudt=(U(2:N)-U(1:N-1))/dt;
    W=alpha_max/Us*dudt
    res=[t(2:N),W]
endfunction
```

```
function res=AV_comp2(t, U)
    dt=t(2)-t(1)
    N=length(U);
    dudt=zeros(N-2,1);
    dudt=(3*U(3:N)-4*U(2:N-1)+U(1:N-2))/2/dt;
    W=alpha_max/Us*dudt
    res=[t(3:N),W]
endfunction
```

```
function res=AV_comp3(t, U)
    dt=t(2)-t(1)
    N=length(U);
    dudt=zeros(N-3,1);
    dudt=(11*U(4:N)-18*U(3:N-1)+9*U(2:N-2)-2*U(1:N-3))/6/dt;
    W=alpha_max/Us*dudt
    res=[t(4:N),W]
endfunction
```

```
function res=AV_error(w, W)
    Nw=length(w); NW=length(W); n=Nw-NW;
    res=max(abs(w(1+n:Nw)-cw));
endfunction
```

## Додаток Г

### Програма-сценарій №2

Дослідження впливу вигляду формули наближеного диференціювання та кроку дискретизації на похибку визначення кутової швидкості при відсутності збурень даних є досить громіздкою задачею, отже для цього розроблена програма-сценарій, яка збережена у файлі model1R1.sci, який має такий вигляд:

```
clear; clc;
loadXcosLibs(); loadScicos(); exec('FormatCharts.sce',-1);
importXcosDiagram("model1.zcos"); exec('comp_AV.sce',-1);

alpha_max=300*%pi/180;
Us=5;

function res=simulation(w, dt)
    typeof(scs_m); scs_m.props.context;
    Context.W=w; Context.dt=dt;
    scicos_simulate(scs_m,Context);
    res=RES
endfunction

titlex="$\Delta t, \mathrm{s}$"; titlec="";
dt=1; W=0.3; Dt=[]; eps=[];
LEGEND=["1", "2", "3"];
for i=1:5
    res=simulation(W,dt);
    t=res.time; w=res.values(:,2); U=res.values(:,3);
    res=AV_comp1(t,U); ct=res(:,1); cw=res(:,2);
    Dt=[Dt,dt];
    eps=[eps,AV_error(w,cw)];
    dt=dt/2;
    disp(i);
end
show_window(1); plot(Dt,eps,"ko","linewidth",2);
dt=1; W=0.3; Dt=[]; eps=[];
for i=1:5
    res=simulation(W,dt);
    t=res.time; w=res.values(:,2); U=res.values(:,3);
    res=AV_comp2(t,U); ct=res(:,1); cw=res(:,2);
    Dt=[Dt,dt];
    eps=[eps,AV_error(w,cw)];
    dt=dt/2;
    disp(i);
end
```

```

show_window(1); plot(Dt,eps,"kd","linewidth",2);
dt=1; W=0.3; Dt=[]; eps=[];
for i=1:5
    res=simulation(W,dt);
    t=res.time; w=res.values(:,2); U=res.values(:,3);
    res=AV_comp3(t,U); ct=res(:,1); cw=res(:,2);
    Dt=[Dt,dt];
    eps=[eps,AV_error(w,cw)];
    dt=dt/2;
    disp(i);
end
show_window(1); plot(Dt,eps,"k^","linewidth",2);
SetXAxesBounds(0,1.1);
SetYAxesBounds(-0.02,0.12);
titley="\$\varepsilon(\Delta t),\mathrm{s}^{-1}$";
FormatCharts(titlex,titley,titlec,LEGEND,2);
xsave("model1R1_res1.scg");

```

Зрозуміло, що в цій програмі-сценарії використовуються підпрограми, що наведені у Додатку В.

## Додаток Д

### Програма-сценарій №3

Для автоматизованого виконання досліджень щодо впливу величин  $\Omega$  кругової (циклічної) частоті коливального руху ланки маніпулятора на вимірювану напругу (3.35) з урахуванням похибок вимірювань розроблена програма-сценарій, що збережена у файлі model2.sci та має зміст такого вигляду:

```
loadXcosLibs(); loadScicos(); exec('FormatCharts.sce',-1);

importXcosDiagram("model2.zcos")

function res=simulation(w, dt)
    typeof(scs_m); scs_m.props.context;
    Context.W=w; Context.dt=dt;
    scicos_simulate(scs_m,Context);
    res=RES
endfunction

titlex="$t, \mathrm{s}$"; titlec=""; dt=0.001;

W=0.3; res=simulation(W,dt); t=res.time; U=res.values(:,3);
LEGEND=["$\Omega=" + string(W) + "\mathrm{s}^{-1}$"];
show_window(1); plot(t,U,"k-", "linewidth",2);
W=0.5; res=simulation(W,dt); t=res.time; U=res.values(:,3);
LEGEND=[LEGEND, "$\Omega=" + string(W) + "\mathrm{s}^{-1}$"];
scf(1); plot(t,U,"k-", "linewidth",2);
W=0.7; res=simulation(W,dt); t=res.time; U=res.values(:,3);
LEGEND=[LEGEND, "$\Omega=" + string(W) + "\mathrm{s}^{-1}$"];
scf(1); plot(t,U,"k:", "linewidth",2);
titley="$U(t), \mathrm{V}$";
SetXAxesBounds(0,0.2);
SetYAxesBounds(2.48,2.62);
FormatCharts(titlex,titley,titlec,LEGEND,4); xsave("model2.scg");
```

## Додаток Е

### Програма-сценарій №4

Дослідження впливу вигляду формули наближеного диференціювання та кроку дискретизації на похибку визначення кутової швидкості з урахуванням похибок вимірювань є досить громіздкою задачею, отже для цього розроблена програма-сценарій, яка збережена у файлі model2R1.sci, який має такий вигляд:

```
clear; clc;
loadXcosLibs(); loadScicos(); exec('FormatCharts.sce',-1);
importXcosDiagram("model2.zcos"); exec('comp_AV.sce',-1);

alpha_max=300*%pi/180;
Us=5;

function res=simulation(w, dt)
    typeof(scs_m); scs_m.props.context;
    Context.W=w; Context.dt=dt;
    scicos_simulate(scs_m,Context);
    res=RES
endfunction

titlex="$\Delta t, \mathrm{s}$"; titlec="";
dt=1; W=0.3; Dt=[]; eps=[];
LEGEND=["1", "2", "3"];
for i=1:5
    res=simulation(W,dt);
    t=res.time; w=res.values(:,2); U=res.values(:,3);
    res=AV_comp1(t,U); ct=res(:,1); cw=res(:,2);
    Dt=[Dt,dt];
    eps=[eps,AV_error(w,cw)];
    dt=dt/2;
    disp(i);
end
show_window(1); plot(Dt,eps,"ko","linewidth",2);
dt=1; W=0.3; Dt=[]; eps=[];
for i=1:5
    res=simulation(W,dt);
    t=res.time; w=res.values(:,2); U=res.values(:,3);
    res=AV_comp2(t,U); ct=res(:,1); cw=res(:,2);
    Dt=[Dt,dt];
    eps=[eps,AV_error(w,cw)];
    dt=dt/2;
    disp(i);
end
```

```

show_window(1); plot(Dt,eps,"kd","linewidth",2);
dt=1; W=0.3; Dt=[]; eps=[];
for i=1:5
    res=simulation(W,dt);
    t=res.time; w=res.values(:,2); U=res.values(:,3);
    res=AV_comp3(t,U); ct=res(:,1); cw=res(:,2);
    Dt=[Dt,dt];
    eps=[eps,AV_error(w,cw)];
    dt=dt/2;
    disp(i);
end
show_window(1); plot(Dt,eps,"k^","linewidth",2);
SetXAxesBounds(0,1.1);
//SetYAxesBounds(-0.02,0.12);
titley="$\varepsilon(\Delta t), \mathrm{s}^{-1}$";
FormatCharts(titlex,titley,titlec,LEGEND,1);
xsave("model2R1_res1.scg");

```

Зрозуміло, що в цій програмі-сценарії використовуються підпрограми, що наведені у Додатку В.

**Додаток Ж**  
Демонстраційний матеріал

