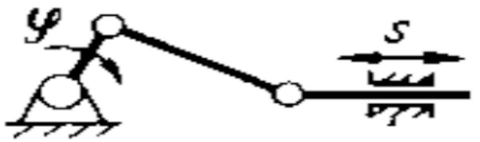
Zur Vorlesung Signale+Systeme (MFB320) - Praktikum 2
- Vereinfachung von nichtlinearen Gleichungen

von Prof. Dr. O.Wallrapp, FK06 HM - 25.10.2009 - erstellt mit Maple 11.
Hinweis: dieses File ist auch mit Maple 9 oder 10 aufrufbar.

Übung: Bewegung des Kolbens der zentrischen Schubkurbel

siehe z.B. Tabellenbuch, hier: r ist die Kurbellaenge, L die Pleuellaenge, s ist die Pos. Kolben bez. Kurbellager.



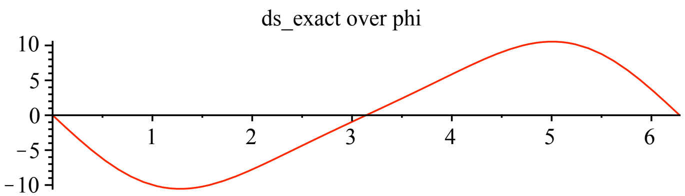

```
> restart;
> s := r*cos(phi) + L*sqrt(1 - (r/L*sin(phi))^2);
```

$$s := r\cos(\phi) + L\sqrt{1 - \frac{r^2 \sin(\phi)^2}{L^2}}$$

```
> ds := diff(s,phi);
```

$$ds := -r\sin(\phi) - \frac{r^2 \sin(\phi)\cos(\phi)}{L\sqrt{1 - \frac{r^2\sin(\phi)^2}{L^2}}}$$

```
> plot(subs(r=10, L=30, ds), phi=0..2*Pi, title="ds_exact
over phi");
```




```
> series(s, phi=0, 2);
  s1 := convert(%,polynom);
```

$$r + L + \mathrm{O}(\phi^2)$$

$$s1 := r + L$$

```
> ds1 := diff(s1,phi);
```


$$ds1 := 0$$

Diese Lösung ist total falsch!

Gehen wir eine Potenz höher

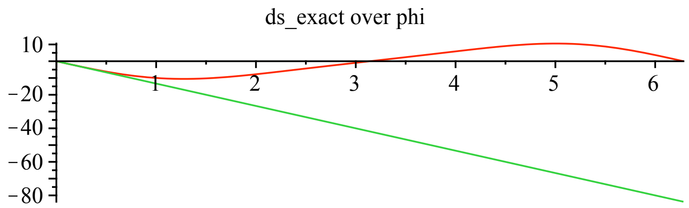
```
> series(s, phi=0, 3):
  s2 := convert(%,polynom);
```

$$s2 := r + L + \left(-\frac{1}{2}r - \frac{1}{2}\frac{r^2}{L}\right)\phi^2$$

```
> ds2 := diff(s2,phi);
```

$$ds2 := 2\left(-\frac{1}{2}r - \frac{1}{2}\frac{r^2}{L}\right)\phi$$

```
> plot([subs(r=10, L=30, ds), subs(r=10, L=30, ds2)], phi=0..2*Pi, title="ds_exact over phi");
```



Diese Lösung ist ebenfalls total falsch, weil wir um den Arbeitspunkt null linearisieren.
Loesung: die Sinusschwingung muss erhalten werden, nur die Wurzel soll weg:

```
> fz := sqrt(1-z);
```

$$fz := \sqrt{1 - z}$$

```
> series(fz, z=0, 2):
  fz1 := convert(%,polynom);
```

$$fz1 := 1 - \frac{1}{2}z$$

```
> s_simple := r*cos(phi) + L*fz1;
```

$$s\_simple := r\cos(\phi) + L\left(1 - \frac{1}{2}z\right)$$

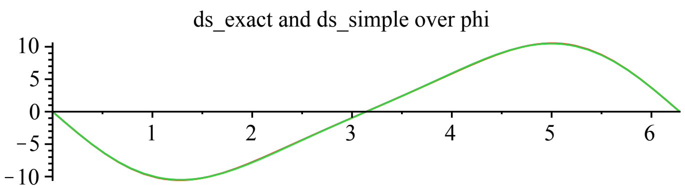
```
> s_simple := subs(z=(r/L*sin(phi))^2, %);
```

$$s\_simple := r\cos(\phi) + L\left(1 - \frac{1}{2}\frac{r^2\sin(\phi)^2}{L^2}\right)$$

```
> ds_simple := diff(s_simple, phi);
```

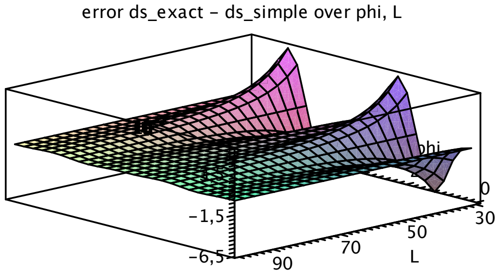
$$ds\_simple := -r\sin(\phi) - \frac{r^2\sin(\phi)\cos(\phi)}{L}$$

```
> plot([subs(r=10, L=30, ds), subs(r=10, L=30, ds_simple)],
phi=0..2*Pi, title="ds_exact and ds_simple over phi");
```



not bad!!!

```
> plot3d(subs(r=10, ds-ds_simple), phi=0..2*Pi, L=30..100,
  title="error ds_exact - ds_simple over phi, L");
```



Merke: Je groesser r/L desto geringer der Fehler.

Aufgabe 1: Kruemmung des Balkens.

Die Krümmung eines Balkens, vgl. Techn.Mechanik, Kap 2, ergibt sich aus

$$\kappa = \frac{w'\,u'^2 - w''\,s^{*2}}{s^*\,s'^3}, \quad s^* = \sqrt{1+u'^2}, \quad s' = \sqrt{1+u'^2+w'^2}$$

Verwende die Variablen: w' = ws, w" = Wss, u' = us, s' = ss, s* = s_star.

```
> reatart:
```

Lösung für a): lineare Form fuer w = u = 0 ,

```
> kappa := (ws*us^2 - wss*s_star^2) / (s_star * ss^3);
```

$$\kappa := \frac{ws\,us^2 - wss\,(1+us^2)}{\sqrt{1+us^2}\,(1+us^2+ws^2)^{3/2}}$$

```
> s_star := sqrt(1 + us^2);
```

$$s\_star := \sqrt{1+us^2}$$

```
> ss := sqrt(1 + us^2 + ws^2);
```

$$ss := \sqrt{1+us^2+ws^2}$$

```
> kappa;
```

$$\frac{ws\,us^2 - wss\,(1+us^2)}{\sqrt{1+us^2}\,(1+us^2+ws^2)^{3/2}}$$

```
> series(kappa, wss=0, 2 ):
  erg := convert(%,polynom);
```


$$erg := \frac{ws\,us^2}{\sqrt{1+us^2}\,(1+us^2+ws^2)^{3/2}} + \frac{(-1-us^2)\,wss}{\sqrt{1+us^2}\,(1+us^2+ws^2)^{3/2}}$$

```
> series(erg, ws=0, 2 ):
  erg := convert(%,polynom);
```

$$erg := \frac{(-1-us^2)\,wss}{(1+us^2)^2} + \frac{us^2\,ws}{(1+us^2)^2}$$

```
> series(erg, us=0, 2 ):
  erg := convert(%,polynom);
```

$$erg := -wss$$

```
>
```

Lösung für b): quadratische Form

```
> kappa;
```

$$\frac{ws\,us^2 - wss\,(1+us^2)}{\sqrt{1+us^2}\,(1+us^2+ws^2)^{3/2}}$$

```
> series(kappa, ws=0, 3 ):
  erg_b := convert(%,polynom);
```

$$erg\_b := -\frac{wss}{1+us^2} + \frac{us^2\,ws}{(1+us^2)^2} + \frac{3}{2}\,\frac{wss\,ws^2}{(1+us^2)^2}$$

```
> series(erg_b, us=0, 3 ):
  erg_b := convert(%,polynom);
```

$$erg\_b := -wss + \frac{3}{2}\,wss\,ws^2 + (-3\,wss\,ws^2 + ws + wss)\,us^2$$

Die Reihenfolge der Entwicklung spielt keine Rolle.

```
> series(kappa, us=0, 3 ):
  erg_b := convert(%,polynom);
```

$$erg\_b := -\frac{wss}{(1+ws^2)^{3/2}} + \left(\frac{3}{2}\,\frac{wss}{(1+ws^2)^{5/2}} + \frac{-\frac{1}{2}\,wss + ws}{(1+ws^2)^{3/2}}\right)us^2$$

```
> series(erg_b, ws=0, 3 ):
  erg_b := convert(%,polynom);
```

$$erg\_b := -wss + wss\,us^2 + ws\,us^2 + \left(\frac{3}{2}\,wss - 3\,wss\,us^2\right)ws^2$$

Man könnte auch eine Procedur schreiben !

Aufgabe 2: Linearsierung der Glg. des Fliehkraftpendels

```
> restart:
```

Gegeben sind die Gleichungen eines Fliehkraftpendels, siehe Vorlesung S&S, Prof. Unterseer, S.22, 39.

phi_dot = ome (1)
phi_dotdot = -d/m ome - g/L sind(phi) + Omega^2 sin(phi) cos(phi) (2)
Omega_dot = M0/nu/Theta_M cos(phi) - ML /nu/Theta_M (3)
Der Eingang sei ML(t), der Ausgang phi(t).

a) Bestimme die stationären Lösungen für eine Last ML0. Beispiel Omega0 = 13 rad/s

b) Linearisiere die Gleichungen für einen stationären Gleichgewichtszustand bei phi > 0

Lösung a) Bestimme die stationären Lösungen für eine Last ML0. Beispiel Omega0 = 13 rad/s

```
> param1 := {L=0.1, g=9.81, M0=1, m=0.05, d=0.02, nu=1,
  Theta_M=10};
```

$$param1 := \{L=0.1, g=9.81, M0=1, m=0.05, d=0.02, \nu=1, \Theta M=10\}$$

```
> f1 := ome;        #Eigendyn 1
> f2 := -d/m*ome - g/L*sin(phi) + Omega^2*sin(phi)*cos(phi);
  #Eigendyn 2
> f3 := M0/nu/Theta_M*cos(phi); #Eigendyn 3
```

$$f1 := ome$$

$$f2 := -\frac{d\,ome}{m} - \frac{g\sin(\phi)}{L} + \Omega^2\sin(\phi)\cos(\phi)$$

$$f3 := \frac{M0\cos(\phi)}{\nu\,\Theta M}$$

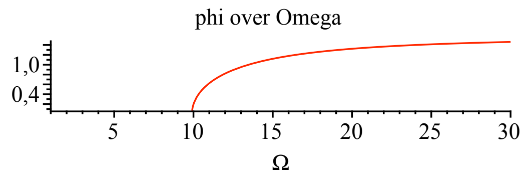
```
>
```

Statische Gleichgewicht: om_dot = 0, phi_dot = ome = 0, Omega_dot = 0

```
> ergphi := solve(subs(ome=0, f2), phi);   # Lösung phi_0
```

$$ergphi := 0, \arccos\left(\frac{g}{\Omega^2 L}\right)$$

```
> plot(subs(param1,ergphi[2]),Omega= 1..30, title=["phi over
  Omega"]);
```




```
> Omega_krit := solve(ergphi[2] = 0, Omega)[1]; # krit.
  Winkelgeschw.
```

$$\Omega krit := \frac{\sqrt{L\,g}}{L}$$

```
> subs(param1,%);
```

$$9.904544412$$

Wir wählen ein Beispiel: Omega0 = 13 rad/s für 0 < phi0 > Pi/2

```
> Omega_0 := 13;   # Beispiel
```

$$\Omega 0 := 13$$

```
> phi_0 := evalf(subs(param1,Omega=Omega_0, ergphi[2]));
  evalf(%*180/Pi);    # in Grad
```

$$\phi 0 := 0.9514864169$$

$$54.51615592$$

Weiter muss in (3) das Moment ML_0 gleich M0 cos(phi) sein.

```
> ML_0 := subs(phi=ergphi[2], M0*cos(phi));    # krit.
  Winkelgeschw.
```



```
  ML_0 := evalf(subs(param1,Omega=Omega_0, ML_0));
```

$$ML\_0 := M0\cos\left(\arccos\left(\frac{g}{\Omega^2 L}\right)\right)$$

$$ML\_0 := 0.5804733728$$

Wenn also ML_0 = 0.58 Nm anliegt, dann dreht sich das System mit Omega_0 = 13 rad/s und das Pendel erhaelt eine Stellung bei phi = 54,5 Grad.

Lösung b)
Linearisierung die drei Differentialgleichungen um eine Lage, z.B. Ω0 =13 rad/s für kleine Änderungen Delta_phi, Delta_ome und Delta_Omega.
Wir nutzen hier die parteille Ableitung!

```
> param_0 := [phi = phi_0, ome = 0, Omega = Omega_0];
```

$$param\_0 := [\phi = 0.9514864169, ome = 0, \Omega = 13]$$

Wir bilden die Ableitung von f1, f2, f3 nach phi.

```
> f1_lin_phi := diff(f1, phi);
```

$$f1\_lin\_phi := 0$$

```
> f2_lin_phi := diff(f2, phi);
  f2_lin_phi := evalf(subs(param1, param_0, f2_lin_phi));
```

$$f2\_lin\_phi := -\frac{g\cos(\phi)}{L} + \Omega^2\cos(\phi)^2 - \Omega^2\sin(\phi)^2$$

$$f2\_lin\_phi := -112.0555621$$

```
> f3_lin_phi := diff(f3, phi);
  f3_lin_phi := evalf(subs(param1, param_0, f3_lin_phi));
```

$$f3\_lin\_phi := -\frac{M0\sin(\phi)}{\nu\,\Theta M}$$

$$f3\_lin\_phi := -0.08142792295$$

und so weiter !
Einfacher ist eine for Schleife. die Ergebnisse schreiben wir in eine Matrix A des LZI-Systems 3. Ordnung
xdot = A . x + B . u
mit x als Zustandsvektor x := [phi, ome, Omega]; und u als Eingangsvektor u = Delta_ML(t)

```
> with(LinearAlgebra):
  A :=  ZeroMatrix(3, compact=false);
```

$$A := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Wir schreiben nun eine "for" Schleife, mit x =

```
> x := [phi, ome, Omega];
```

$$x := [\phi, ome, \Omega]$$

```
> for i from 1 to 3 do
     A[1,i] := diff(f1, x[i]);
     A[2,i] := diff(f2, x[i]);
     A[3,i] := diff(f3, x[i]);
  od:
  A;
```

$$\left[\begin{array}{ccc} 0 & 1 & 0 \\ -\frac{g\cos(\phi)}{L}+\Omega^2\cos(\phi)^2-\Omega^2\sin(\phi)^2 & -\frac{d}{m} & 2\,\Omega\sin(\phi)\cos(\phi) \\ -\frac{M0\sin(\phi)}{\nu\,\Theta M} & 0 & 0 \end{array}\right]$$

Hier sezten wir nun die Paramter und die stationaeren WErte ein:

```
> A := evalf(subs(param1, param_0, A));
```

$$A := \left[\begin{array}{ccc} 0. & 1. & 0. \\ -112.0555621 & -0.4000000000 & 12.28935268 \\ -0.08142792295 & 0. & 0. \end{array}\right]$$

```
> B := Vector([ 0 , 0, -1/nu/Theta_M ]); #  input-matrix
```

$$B := \left[\begin{array}{c} 0 \\ 0 \\ -\frac{1}{\nu\,\Theta M} \end{array}\right]$$

```
> B := evalf(subs(param1, param_0, B));
```

$$B := \left[\begin{array}{c} 0. \\ 0. \\ -0.1000000000 \end{array}\right]$$

Somit liegt das lineare System vor: xdot = A . x + B . u, wo u = Delta_ML

```
>
```

End of Prakt 2