

Matlabfiler for Miniloopen

Skrevet av: Heidi Sivertsen

14th June 2006

1 Innhold

Disse filene beskriver brukemåten til filer som benyttes til å modellere og analysere Miniloopen. De er basert på modellen laget av Espen.

2 Initialisere og tune modell:

2.1 *gen_data.m*

Her ligger data angående konfigurasjonene av riser, samt data for fluids. I tillegg tuningsparameteren `data.n`, som må stiller på vha *slug_3D_linearize.m* [2.6]. Må fylles ut på forhånd, og kalles av de fleste andre filene.

2.2 *Exp_data.m*

Her ligger data angående rater og bifurkasjonspunkt for eksempelet du vil analysere. Må fylles ut på forhånd, og kalles av de fleste andre filene.

2.3 *initialize2.m*

`[x0, y, data]=initialize(z, mG_in, mL_in, P1, h1, P2)`

Regner ut:

- `x0=[mL, mG1, mG2]`
- `y`=vektor med tilstander

som gir stasjonærpunktet $dx/dt=0$ for gitte input-verdier. `y` er her ikke den vanlige måling-vektoren `[P1 P2 rho W Q]`.

2.4 *linearize.m*

`[A, B, C, D, E, F, ys]=linearize(func, t0, x0, flag, u0, d0)`

`func`: ikkelineær modell på formen: `sys=func(t,x,flag,u,d)`.

Eks: *slug_3D_lin.m* [2.5].

`x0`: initialverdier for tilstandene `x=[mL mG1 mG2]`'.

`u0`: initialverdier for input `u=[z mG_in mL_in]`. `d`: optional `t`: optional? Regner ut løsningene til ligningssystemet: $x' = Ax + Bu + Ed$ $y = Cx + Du + Fd$:

- A B C D E F
- ys: stasjonærverdien til målingene [P1 P2 rho W Q]

2.5 *slug_3D_lin.m*

[sys]=slug_3D_lin(t, x, u)

Regner ut de deriverte av tilstandene for gitte input-verdier. Finner:

- sys=dx/dt=[dmL/dt dmG1/dt dmG2/dt]
- global: Y = [P1 P2 rho W Q] for de gitte verdiene

2.6 *slug_3D_linearize.m*

Script som kjører *initialize2.m* [2.3] rundt bifurkasjonpunktet for å finne stasjonærverdier. Disse blir brukt av *linearize.m* [2.4] til å finne en lineær modell for dette operasjonspunktet. Dette punktet er akkurat stabilt, og polene (eig A) skal derfor ligge på den imaginære akse. Blir brukt til å tune modellen. Stiller på *data.n* i *gen_data.m* [2.1] til verdiene på pol-paret blir riktig. Regner ut:

- A B C D E F for bifurk.punktet.
- Polene eller eig(A) til systemet for bifurkasjonspunktet.

3 Filer tilknyttet SIMULINK

3.1 *slug_3D_sim.m*

[sys]=slug_3D(t, x, u, output, data)

Kalles av *s_slug.m* [3.2], og gir derfor enten ut de deriverte av tilstandene, dx/dt vektoren, eller verdien på målingene, y = [P1 P2 rho W Q] for de gitte input.

- sys: 1) x=[mL mG1 mG2 rho_mix] eller 2) y=[P1 P2 rho W Q] avh. av output-variabelen.

(Egentlig gjør denne det samme som *slug_3D_lin.m* [2.5], bare at her oppgis det om en heller vil ha ut målingene.)

3.2 *s_slug.m*

[sys, x0, str, ts]=s_slug(t, x, u, flag)

s-fil for SIMULINK kjøring. Benytter *slug_3D_sim.m* [3.1] til å regne ut verdier og de deriverte. Inputs kommer fra SIMULINK-simuleringen, og outputs blir sendt til SIMULINK. Initialiserer først rundt bifurkasjonspunktet, bruker derfor *initialize2.m* [2.3].

4 Filer for matlab simuleringer av systemet

4.1 *slug_3DAE.m*

[sys]= slug_3D(t,x,u,output,data)

Er beregnet til bruk for ode-løsere i matlab, og kjøres av *slug_3D_script.m* [4.2]. NB: Legger til algebraisk ligning. Den siste ligninga sjekker trykbalansen i systemet. Ellers er den lik *slug_3D_sim.m* [3.1], så den gir ut:

- sys: 1) x=[mL mG1 mG2 rho_mix] eller 2) y=[P1 P2 rho W Q] avh. av output-variabelen.
- Global: Y=P1

4.2 *slug_3D_script.m*

Skript som lineariserer rundt bifurk.punktet vha *initialize2.m* [2.3]. Legger så til rho_mix i tilstandsvektoren. Simulerer systemet ved bruk av ode-løser og *slug_3DAE.m* [4.1]. Finner x(t) og y(t) for verdiene av x0 og u som er oppgitt i skriptet.

- x(t)
- y(t)

Plotter resultatene.

5 Filer til bifurkasjonsplot

5.1 GenBif.m

Skript som regner ut bifurkasjonsplottet til simulerte verdier av systemet, gitt av *gen_data.m* [2.1]. Initialiser vha *initialize2.m* [2.3], og bruker ode-løser til å simulere systemet og finne max og min-verdier for trykkene P1 og P2 med tilhørende ventilåpning,z. Bruker *slug_3D_sim.m* [3.1] til ode-løserne. I tillegg finner den den stasjonære løsningen samt xs (den tilhørende stasjonære verdiene til tilstandene).

Lagrer mat-filene:

- *slugdata.mat*: max og min P1 og P2 med tilhørende z-verdier
- *Stationary.mat*: stasjonærverdier P1 og P2 med tilhørende z, samt xs

Plotter verdiene, samt evt eksperimentelle verdier dersom disse er lagt i tilgjengelig mat-fil spesifisert av skriptet.

Benytter seg også av funksjonene:

secant_cont_circle.m

slug_wrapper.m

Broyden_secant.m

slug_3D_sim.m [3.1]

for å regne ut stasjonærverdier. Har ikke lest de filene så nøye.