On the direct solution of critical equilibrium states

Anders Eriksson[1], Reijo Kouhia[2], Jari Mäkinen[2]

[1] KTH Mechanics

[2] TUT, Department of Mechanical Engineering and Industrial Systems

5 June 2015

Outline

1 Introduction

2 Stability eigenvalue problem

3 Solution algorithm for non-linear eigenproblem

4 Mixed strategy

5 Concluding remarks

1 Introduction

2 Stability eigenvalue problem

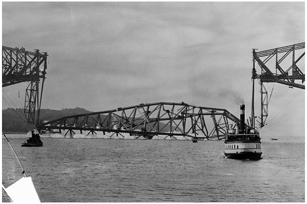
3 Solution algorithm for non-linear eigenproblem

4 Mixed strategy

5 Concluding remarks

Introduction

- Definition of criticality
- Some Algorithms
- A proposal
- Example



Stability eigenvalue problem

Definition for a critical state: Find displacements $\boldsymbol{q}_{\rm cr}$, critical load $\lambda_{\rm cr}$ and the corresponding eigenmode $\boldsymbol{\phi}$ such, that

$$\begin{cases} \boldsymbol{f}'(\boldsymbol{q}_{\rm cr}, \lambda_{\rm cr})\boldsymbol{\phi} & = \boldsymbol{0} \\ \boldsymbol{f}(\boldsymbol{q}_{\rm cr}, \lambda_{\rm cr}) & = \boldsymbol{0} \end{cases} \tag{1}$$

where $\boldsymbol{f}' = \partial \boldsymbol{f} / \partial \boldsymbol{q}$. *The non-linear* mapping $\boldsymbol{f}$ defines the equilibrium path in the displacement $\boldsymbol{q}$ and load parameter λ space:

$$\boldsymbol{f}(\boldsymbol{q}, \lambda) \equiv \boldsymbol{r}(\boldsymbol{q}) - \lambda \boldsymbol{p}_r(\boldsymbol{q}) = \boldsymbol{0}$$

and constitutes the balance between internal- and external forces.

System (1) is a non-linear eigenvalue problem, which is
HARD TO SOLVE!

Solution algorithm for non-linear eigenproblem

Extended system:

$$\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{q}, \boldsymbol{\phi}, \boldsymbol{\lambda}) = \begin{cases} \hat{\boldsymbol{f}}(\boldsymbol{q}, \boldsymbol{\lambda}) \equiv \boldsymbol{f}(\boldsymbol{q}, \lambda) + \boldsymbol{f}_0(\boldsymbol{q}, \boldsymbol{\lambda}) = \boldsymbol{0} \\ \boldsymbol{h}(\boldsymbol{q}, \boldsymbol{\phi}, \boldsymbol{\lambda}) \equiv \boldsymbol{f}'(\boldsymbol{q}, \boldsymbol{\lambda})\boldsymbol{\phi} + \boldsymbol{h}_0(\boldsymbol{\phi}, \boldsymbol{\lambda}) = \boldsymbol{0} \\ \boldsymbol{c}(\boldsymbol{q}, \boldsymbol{\phi}, \boldsymbol{\lambda}) = \boldsymbol{0}. \end{cases}$$

Newton's linearisation step results in

$$\boldsymbol{A}\delta\boldsymbol{x} = -\boldsymbol{g},$$

where

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{K}_f & \boldsymbol{0} & \boldsymbol{P} \\ \boldsymbol{Z} & \boldsymbol{K}_h & \boldsymbol{N} \\ \boldsymbol{C}_q & \boldsymbol{C}_\phi & \boldsymbol{C}_\lambda \end{bmatrix}, \quad \delta\boldsymbol{x} = \begin{Bmatrix} \delta\boldsymbol{q} \\ \delta\boldsymbol{\phi} \\ \delta\boldsymbol{\lambda} \end{Bmatrix}, \quad \boldsymbol{g} = \begin{Bmatrix} \hat{\boldsymbol{f}} \\ \boldsymbol{h} \\ \boldsymbol{c} \end{Bmatrix}.$$

Block elimination

The solution vector is partitioned as

$$\delta \boldsymbol{q} = \boldsymbol{q}_f + \boldsymbol{Q}_p \delta \boldsymbol{\lambda},$$
$$\delta \boldsymbol{\phi} = \boldsymbol{\phi}_h + \boldsymbol{\Phi}_n \delta \boldsymbol{\lambda}.$$

Vectors $\boldsymbol{q}_f, \boldsymbol{\phi}_h$ and the $n \times p$ matrices $\boldsymbol{Q}_p, \boldsymbol{\Phi}_n$ solved from

$$\boldsymbol{K}_f \boldsymbol{q}_f = -\hat{\boldsymbol{f}}, \qquad \boldsymbol{K}_f \boldsymbol{Q}_p = -\boldsymbol{P},$$
$$\boldsymbol{K}_h \boldsymbol{\phi}_h = -\boldsymbol{h} - \boldsymbol{Z} \boldsymbol{q}_{\mathrm{f}}, \qquad \boldsymbol{K}_h \boldsymbol{\Phi}_n = -\boldsymbol{N} - \boldsymbol{Z} \boldsymbol{Q}_p,$$

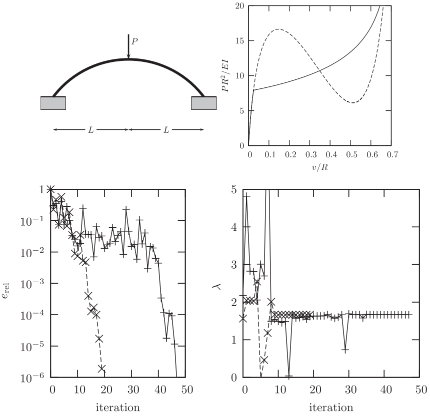
and for the control parameters

$$\delta \boldsymbol{\lambda} = -(\boldsymbol{C}_\lambda + \boldsymbol{C}_q \boldsymbol{Q}_p + \boldsymbol{C}_\phi \boldsymbol{\Phi}_n)^{-1} (\boldsymbol{c} + \boldsymbol{C}_q \boldsymbol{q}_f + \boldsymbol{C}_\phi \boldsymbol{\phi}_h).$$

Suitable strategy if direct linear solver is used.

Problems with the direct method

Since Newton's method is only *locally convergent* the method might fail if started from the unloaded, undeformed state.



1 Introduction

2 Stability eigenvalue problem

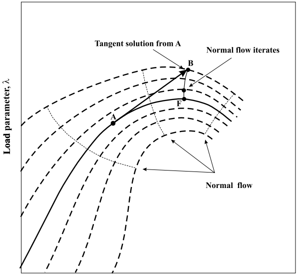
3 Solution algorithm for non-linear eigenproblem

4 Mixed strategy

5 Concluding remarks

Mixed strategy

1. Compute crude approximation to the lowest critical load.
2. Use orthogonal trajectory method (normal flow) to get a nearby point on the equilibrium path.
3. Use the extended system started from the computed equilibrium point.



1 Introduction

2 Stability eigenvalue problem

3 Solution algorithm for non-linear eigenproblem

4 Mixed strategy

5 Concluding remarks

Concluding remarks

An algorithm for direct critical point search which is belived to increase the robustness of the basic direct solution algorithm has been proposed.

Thank you for your attention!