

*Luca Salvatori*

# MECCANICA COMPUTAZIONALE

**Modelli numerici di frattura del calcestruzzo**

21 dicembre 2004



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Sommario

- 1) Modellazione numerica del calcestruzzo
- 2) Richiami di analisi non-lineare
- 3) Richiami di elementi finiti
- 4) Elementi finiti con non-linearità materiali
- 5) Modello di danno uniassiale
- 6) Modelli continui 3D e applicazioni



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



# 1. FRATTURA NEL CALCESTRUZZO



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori

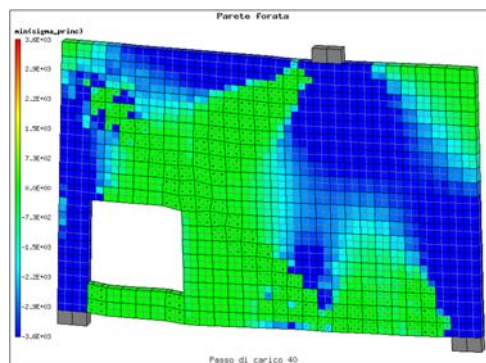


## Introduzione

Lo studio e la progettazione delle strutture in CA vengono solitamente effettuati con l'ausilio di codici agli elementi finiti, limitandosi ad analisi lineari.

In casi eccezionali (e.g. forte sisma) le strutture escono inevitabilmente dal campo lineare e si danneggiano. È necessario modellare tali situazioni per poterle controllare e garantire la sicurezza.

- La fessurazione del CLS è un fenomeno dalla fisica complessa
- Non ne esiste un modello numerico "definitivo"
- È comune anche in altri materiali dell'Ingegneria Civile (muratura, rocce, alcuni terreni)



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Fonti di non-linearità nel C.A.

calcestruzzo:

- ✓ fessurazione in trazione
- ✓ plasticità in compressione
- ✓ ritiro
- ✓ fenomeni viscosi (*fluage*)



acciaio:

- ✓ plasticità

interazione acciaio-calcestruzzo:

- ✓ trasferimento delle azioni da cls fessurato ad armature
- ✓ perdita di aderenza e sfilamento



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori

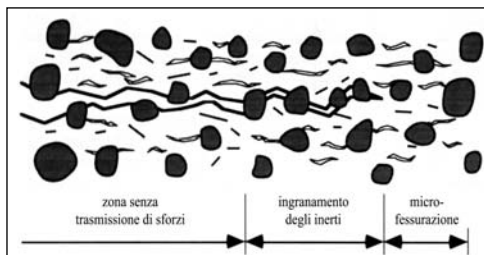


## Cause della micro-fessurazione del cls

La perdita di caratteristiche meccaniche è dovuta al proliferare di cavità e microdifetti preesistenti (già prima di ogni carico), dovuti a:

- ✓ Discontinuità della matrice cementizia (difetti in scala atomica, porosità capillare, idratazione localmente incompleta, locali eccessi d'acqua)
- ✓ Ritiro e dilatazioni termiche differenziali fra le varie fasi (matrice, inerti vari)
- ✓ Discontinuità di interfaccia matrice-inerti
- ✓ Vuoti macroscopici in fase di getto (dalle bolle d'aria a vuoti dovuti ad insufficiente vibrazione, ostacoli offerti da cassero o armature, ecc.)

In prossimità dei difetti si hanno concentrazioni tensionali che provocano l'allargamento dei difetti stessi e il loro progressivo collegarsi fino a formare macrofessure.

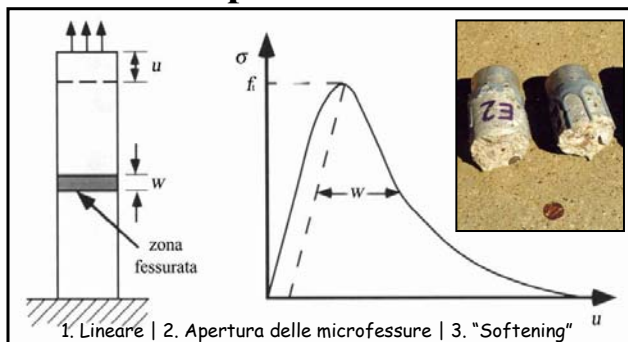


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

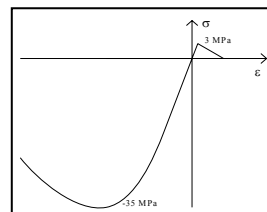
Luca Salvatori



## Prova sperimentale di trazione uniassiale



Si fa riferimento agli sforzi applicati (medi) e non agli sforzi locali che sono invece influenzati dalle disomogeneità.



GRANDEZZE CARATTERISTICHE (oltre a  $E$  e  $\nu$ ):

- Resistenza a trazione ( $f_t$ )
- Energia di fessurazione ( $G_f$ )
- Legge di softening ( $\sigma = f_w(w)$ ), per ottenere un legame con  $\varepsilon$  si divide  $w$  per una lunghezza  $h$  interessata dalle fessure

La resistenza a trazione è minore di un ordine di grandezza rispetto a quella a compressione



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori

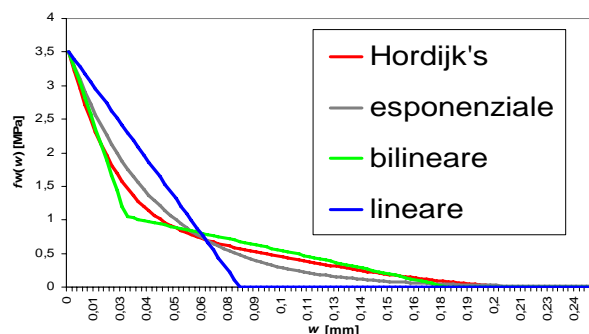


## Espressioni analitiche della legge di softening

Legge di Hordijk (segue i dati sperimentali): 
$$\sigma = f_w^{\text{Hor}}(w) = f_t \left\langle \left[ 1 + \left( \frac{c_1 w}{w_c} \right)^3 \right] \exp \left( - \frac{c_2 w}{w_c} \right) - e^{-c_2} (1 + c_1^3) \frac{w}{w_c} \right\rangle$$

Legge esponenziale: 
$$\sigma = f_w^{\text{exp}}(w) = f_t \exp \left( - \frac{w}{w_t} \right)$$

Confronto fra varie espressioni (fissati  $f_t$  e  $G_f$ )

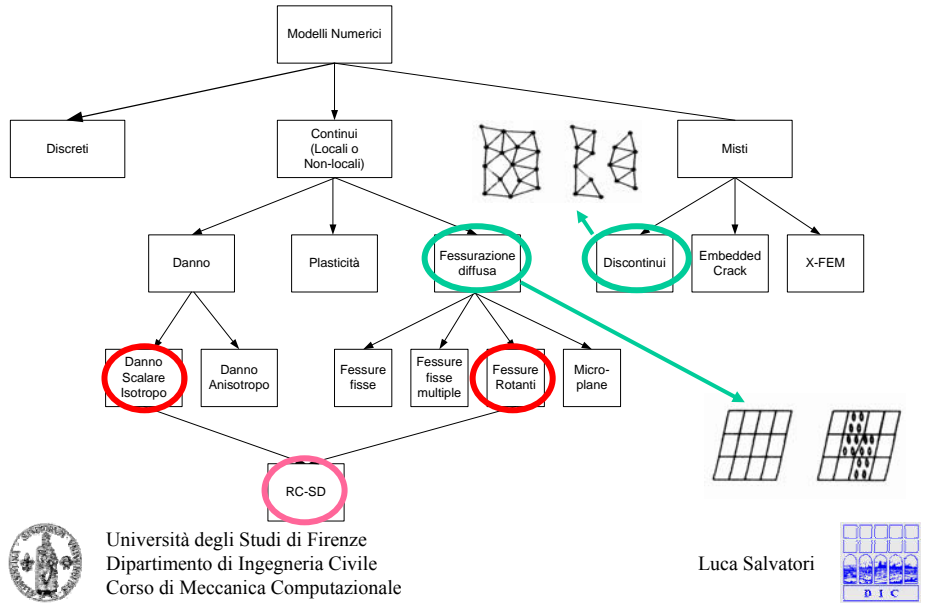


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Modelli di fessurazione

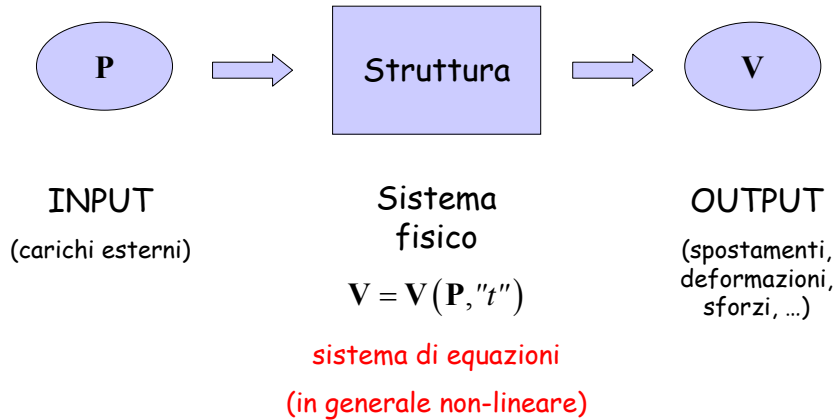


## 2. RICHIAMI DI ANALISI NON-LINEARE

Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori

## Introduzione



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Caso lineare

$$V = V(P, \cancel{t}) = K_c^{-1}P \quad \rightarrow \quad \text{система di equazioni lineari}$$

### ☺ Vantaggi:

- soluzione con metodi standard
- computazionalmente poco oneroso
- vale il "principio" di sovrapposizione degli effetti (combinazioni di carico, sovrapposizione modale, ...)

### ☹ L'approssimazione lineare non è sempre applicabile!

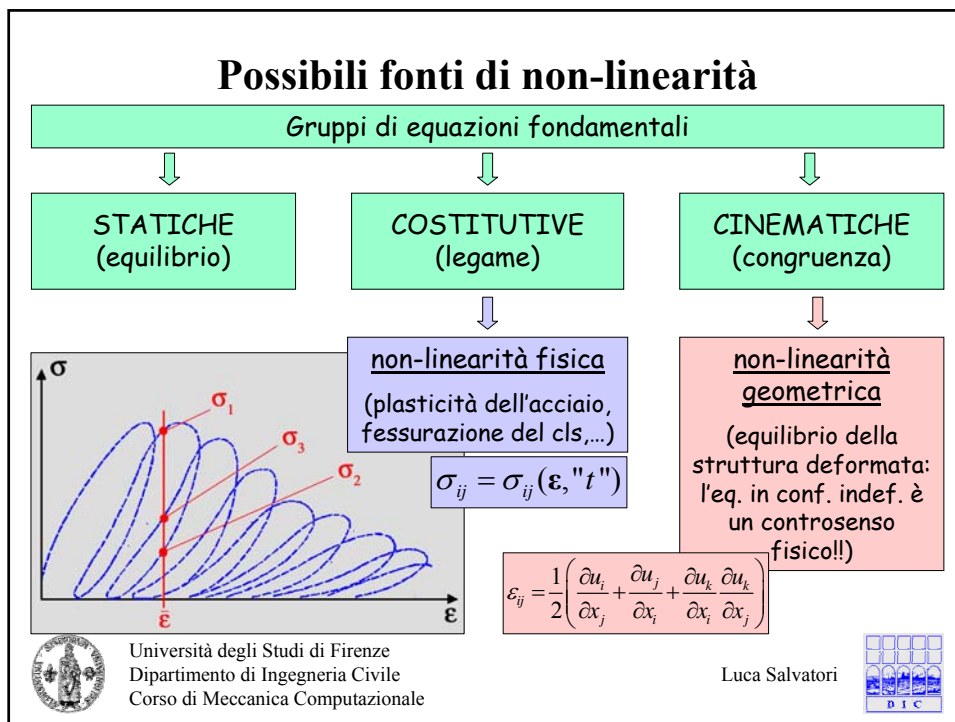


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Possibili fonti di non-linearità



## Condizione di equilibrio di un sistema discreto e linearizzazione

Immaginiamo di poter scrivere l'equilibrio in forma vettoriale come:

$$\mathbf{F}_{\text{int}}(\mathbf{V}) = \mathbf{P} \quad \text{condizione di equilibrio}$$

$\mathbf{F}_{\text{int}}$  vettore delle forze interne (dipendente dalla struttura)

$\mathbf{V}$  spostamenti nodali (incogniti)

$\mathbf{P}$  carichi esterni (dati)

Caso lineare:  $\mathbf{F}_{\text{int}}(\mathbf{V}) = \mathbf{K}_{\text{lin}} \mathbf{V}$

Caso non-lineare: linearizzazione attorno a  $\mathbf{V}_0$  in equilibrio ( $\mathbf{F}_{\text{int}}(\mathbf{V}_0) = \mathbf{P}_0$ ):

$$\mathbf{F}_{\text{int}}(\mathbf{V} + \Delta \mathbf{V}) = \mathbf{F}_{\text{int}}(\mathbf{V}_0) + \left. \frac{\partial \mathbf{F}_{\text{int}}}{\partial \mathbf{V}} \right|_{\mathbf{V}_0} \Delta \mathbf{V} + \dots = \mathbf{P}_0 + \Delta \mathbf{P} = \mathbf{P}$$

$$\mathbf{K}_T \cdot \Delta \mathbf{V} \approx \mathbf{P} - \mathbf{F}_{\text{int}} \quad \text{equilibrio linearizzato}$$

$$\mathbf{K}_T(\mathbf{V}_0) = \left. \frac{\partial \mathbf{F}_{\text{int}}}{\partial \mathbf{V}} \right|_{\mathbf{V}_0} \quad \text{matrice di rigidità tangente}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Obiettivo dell'analisi non-lineare e la procedura incrementale-iterativa

Obiettivo: tracciare la risposta carico-spostamento della struttura.

Si assume il carico dipendente da un solo parametro scalare  $\lambda$  (moltiplicatore di carico):

$$\mathbf{P} = \lambda \mathbf{P}_{\text{ref}}$$

Come il FEM discretizza la struttura nello spazio, la **procedura incrementale** effettua la discretizzazione del percorso carico-spostamento: dato uno stato in equilibrio definito dalla coppia  $(\mathbf{V}_i, \lambda_i)$  la procedura incrementale stabilisce come si debba ricercare il successivo stato  $(\mathbf{V}_{i+1}, \lambda_{i+1})$ .

Poiché le equazioni di equilibrio sono non-lineari, ogni coppia in equilibrio è ricercata con l'ausilio di un **metodo iterativo**.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## “Ingredienti”

Procedura incrementale (criterio con cui vengono ricercate le coppie  $(\mathbf{V}, \lambda)$ ):

- Controllo di carico
- Controllo di spostamento
- “A lunghezza d'arco”

Metodo iterativo (metodo con cui si risolvono le equazioni di equilibrio):

- Rigidezza tangente aggiornata (Newton-Raphson o “N-R”)
- Rigidezza tangente non aggiornata (metodi “modificati”)
- Rigidezza elastica (o “rigidezza costante” o “rigidezza lineare”)
- “Rigidezza” secante (quasi-Newton)

Criterio di convergenza (grandezza con cui si misura l'errore e che determina la fine delle iterazioni):

- Norma dei residui
- Norma degli spostamenti
- Norma energetica



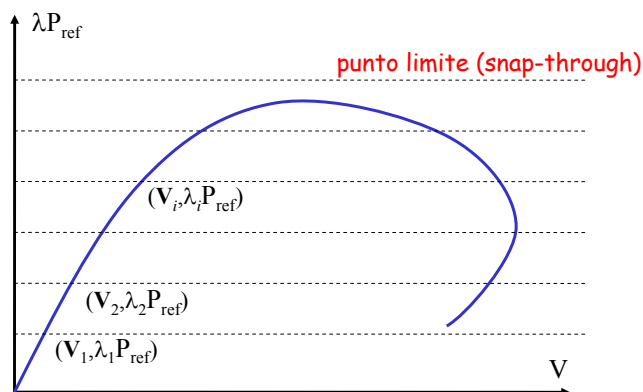
Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori





## Procedura incrementale: controllo di carico



- ⊗ corrisponde ad un modo "naturale" di applicare il carico per passi
- ⊗ impossibile seguire il percorso di equilibrio oltre il punti limite

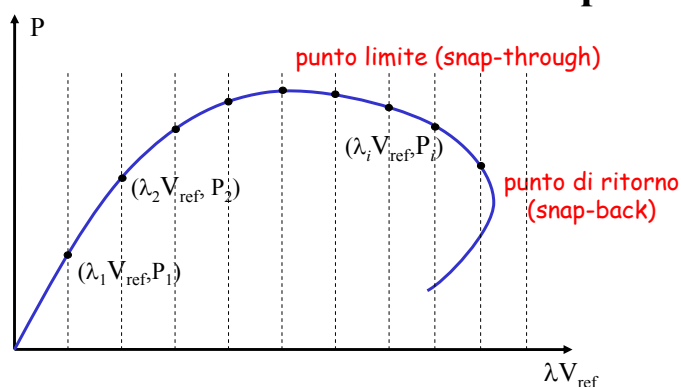


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Procedura incrementale: controllo di spostamento



Il moltiplicatore si applica agli spostamenti e si ricercano i carichi corrispondenti:

- ⊗ tutti gli spostamenti assegnati devono essere fra loro proporzionali, ciò accade solo in pochi modelli fisici (e.g. carichi concentrati in prove sperimentali).
- ⊗ consente di superare i punti limite
- ⊗ ma non i punti di ritorno

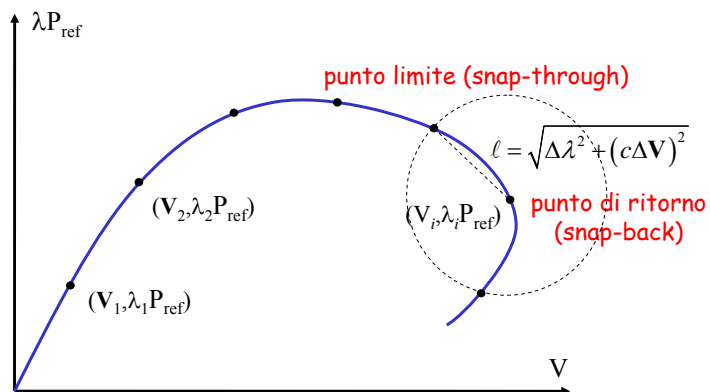


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Procedura incrementale: “a lunghezza d’arco”



⊗ non si ha il controllo né sul carico né sugli spostamenti

⊙ è possibile seguire ogni percorso di equilibrio

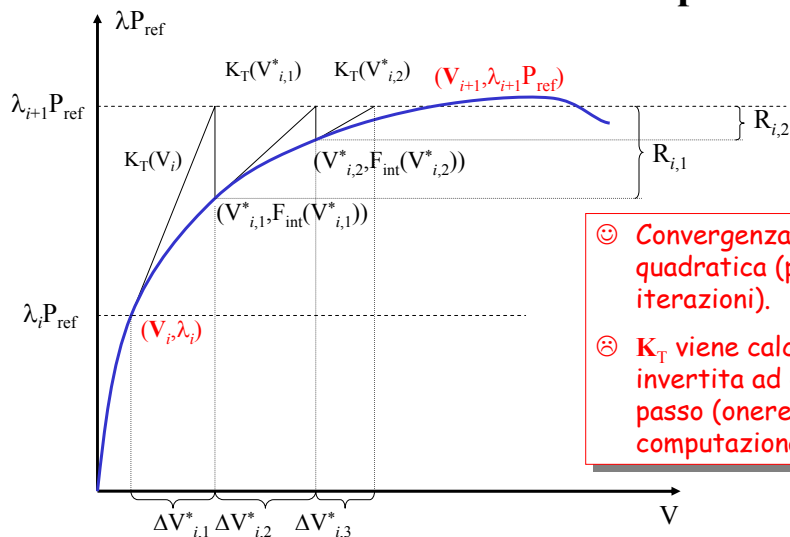


Università degli Studi di Firenze  
 Dipartimento di Ingegneria Civile  
 Corso di Meccanica Computazionale

Luca Salvatori



## Metodi iterativi: Newton-Raphson



⊙ Convergenza quadratica (poche iterazioni).

⊗  $K_T$  viene calcolata ed invertita ad ogni passo (onere computazionale).

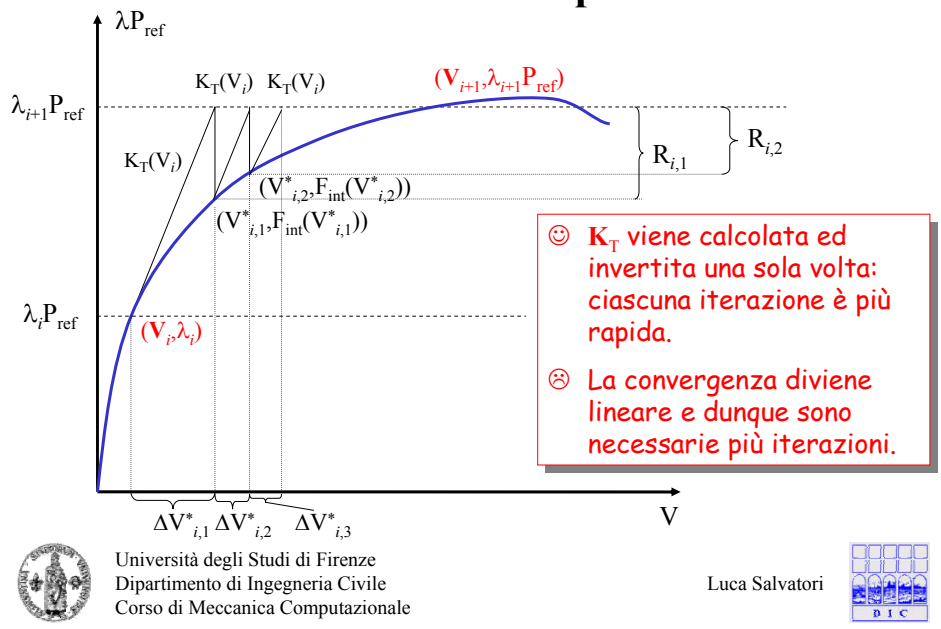


Università degli Studi di Firenze  
 Dipartimento di Ingegneria Civile  
 Corso di Meccanica Computazionale

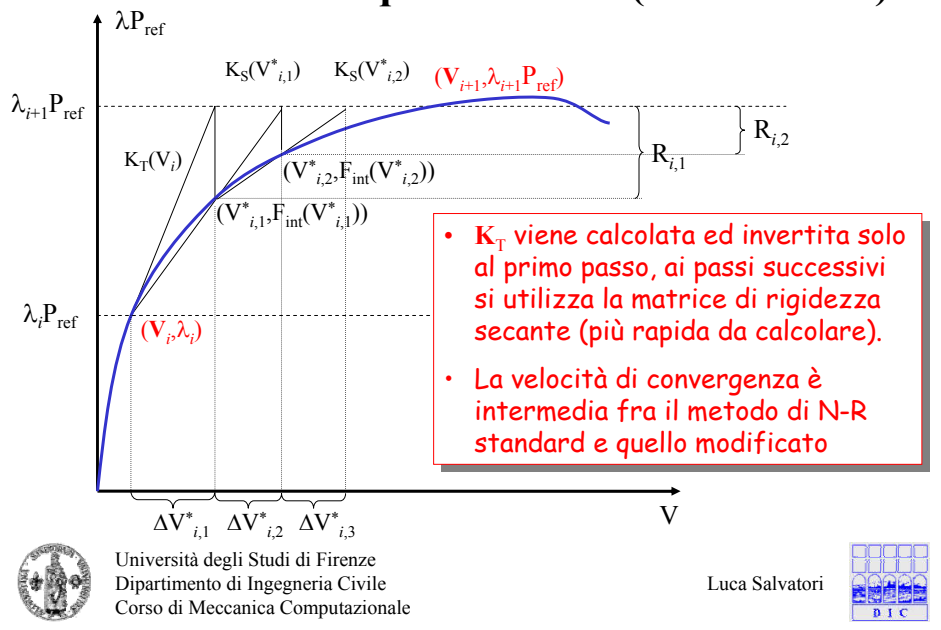
Luca Salvatori



## Metodi iterativi: Newton-Raphson modificato



## Metodi iterativi: quasi-Newton (della secante)



## Metodi iterativi: rigidezza iniziale

Il metodo della rigidezza iniziale (o elastica o costante) utilizza sempre la matrice di rigidezza elastica.

Il metodo richiede in genere molte iterazioni e presenta la convergenza più lenta.

In alcuni casi tuttavia rende l'analisi più stabile.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Criteri di convergenza

La procedura iterativa termina quando:

errore relativo o assoluto dell'iterazione corrente (calcolato dal programma)  $\delta_k = \frac{e_k}{e_{\text{ref}}} < \delta_{\text{toll}}$  precisione ricercata (fissata dall'utente)

Le possibili varianti sono:

a) nella scelta della grandezza da normare:

- 1) norma dei **residui** ( $\mathbf{R}_{i,k}$ )
- 2) norma delle correzioni degli **spostamenti** ( $\Delta \mathbf{V}_{i,k}^*$ )
- 3) norma **energetica** (prodotto di  $\mathbf{R}_{i,k}$  per  $\Delta \mathbf{V}_{i,k}^*$ )

b) nella scelta del tipo di norma (euclidea, max-abs, sum-abs, ...).

c) nella scelta della quantità di riferimento (se si considera l'errore assoluto è  $e_{\text{ref}} = 1.0$ , se si considera l'errore relativo è  $e_{\text{ref}} = e_1$ ).



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Scelta dell'algoritmo

Non è possibile definire a priori l'algoritmo "migliore",  
ma va scelto di volta in volta in base a:

- Non-linearità fisica o geometrica
- Hardening o softening
- Dimensione del problema
- Carichi applicati in singoli punti o su intere superfici
- Non-linearità improvvise o gradualì



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## 3. RICHIAMI DI ELEMENTI FINITI



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Problema elastostatico

Il problema elastostatico è caratterizzato dai tre gruppi di equazioni definiti in ogni punto del continuo

$$\left. \begin{array}{l}
 \text{Congruenza:} \quad \varepsilon_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \\
 \text{Legame:} \quad \sigma_{ij} = \frac{E}{1+\nu} \left( \varepsilon_{ij} + \frac{\nu}{1-2\nu} \varepsilon_{kk} \delta_{ij} \right) \\
 \text{Equilibrio:} \quad \frac{\partial \sigma_{ij}}{\partial x_i} + b_j = 0
 \end{array} \right\} \text{ in } V$$

Cui si aggiungono le condizioni al contorno:

$$\text{c.c. statiche} \quad \sigma_{ij} n_j = \hat{t}_i \quad \text{su } S_t$$

$$\text{c.c. cinematiche} \quad u_i = \hat{r}_i \quad \text{su } S_r$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Notazione computazionale

In meccanica computazionale si è soliti raccogliere spostamenti, deformazioni e sforzi in vettori:

$$\mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad \boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{31} \end{bmatrix} \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \gamma_{12} \\ \gamma_{23} \\ \gamma_{31} \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad \hat{\mathbf{t}} = \begin{bmatrix} \hat{t}_1 \\ \hat{t}_2 \\ \hat{t}_3 \end{bmatrix} \quad \hat{\mathbf{r}} = \begin{bmatrix} \hat{r}_1 \\ \hat{r}_2 \\ \hat{r}_3 \end{bmatrix}$$

Si noti come anche i tensori del secondo ordine di sforzi e deformazione vengano rappresentati in vettori colonna.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Problema in notazione computazionale

**congruenza:**  $\boldsymbol{\varepsilon} = \boldsymbol{\partial}_{\text{cin}} \cdot \mathbf{u}$

**legame:**  $\boldsymbol{\sigma} = \mathbf{E} \cdot \boldsymbol{\varepsilon}$

**equilibrio:**  $\boldsymbol{\partial}_{\text{sta}} \cdot \boldsymbol{\sigma} + \mathbf{b} = \mathbf{0}$

**c.c. statiche:**  $\mathbf{R}_{\text{sta}} \cdot \boldsymbol{\sigma} = \hat{\mathbf{t}}$  su  $S_t$

**c.c. cinematiche:**  $\mathbf{R}_{\text{cin}} \cdot \mathbf{u} = \hat{\mathbf{r}}$  su  $S_r$

$$\boldsymbol{\partial}_{\text{sta}}^T = \boldsymbol{\partial}_{\text{cin}} = \begin{bmatrix} \partial_1 & 0 & 0 \\ 0 & \partial_2 & 0 \\ 0 & 0 & \partial_3 \\ \partial_2 & \partial_1 & 0 \\ 0 & \partial_3 & \partial_2 \\ \partial_3 & 0 & \partial_1 \end{bmatrix}$$

$$\mathbf{E} = \frac{E}{(1+\nu)(1-2\nu)}$$

$$\begin{bmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix}$$

$$\mathbf{R}_{\text{sta}} = \begin{bmatrix} \alpha_1 & 0 & 0 & \alpha_2 & 0 & \alpha_3 \\ 0 & \alpha_2 & 0 & \alpha_1 & \alpha_3 & 0 \\ 0 & 0 & \alpha_3 & 0 & \alpha_1 & \alpha_2 \end{bmatrix}$$

$$\mathbf{R}_{\text{cin}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

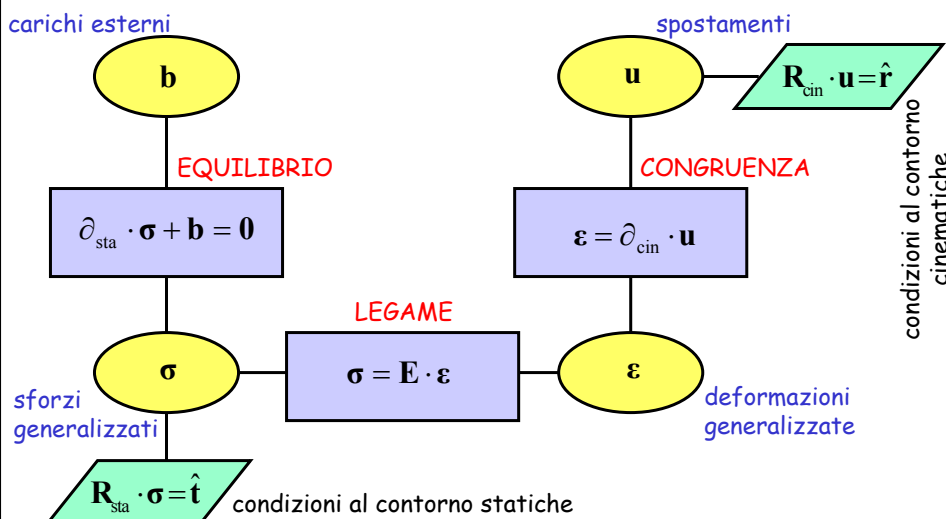


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Diagramma di Tonti



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Energia potenziale totale

$$\Pi_u^{\text{EPT}} = \frac{1}{2} \int_V \boldsymbol{\varepsilon}_u^T \cdot \boldsymbol{\sigma}_u dV - \int_V \mathbf{u}^T \cdot \mathbf{b} dV - \int_{S_t} \mathbf{u}^T \cdot \hat{\mathbf{t}} dA$$

Il pedice "u" indica che si tratta di un funzionale in cui la variabile indipendente sono gli spostamenti (deformazioni e tensioni sono assunti in funzione di questi tramite congruenza e legame). Nella formulazione agli spostamenti del FEM è questo il funzionale che utilizzeremo.

È possibile scrivere funzionali analoghi dove le variabili indipendenti sono gli sforzi (metodo delle forze):

- energia potenziale complementare

Per problemi specifici si ricorre inoltre alle formulazioni miste con l'assunzione di più variabili indipendenti:

- potenziale di Hellinger-Reissner (spostamenti e sforzi)
- potenziale di Veubeke-Hu-Washizu (spostamenti, deformazioni e sforzi)



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Principio dei lavori virtuali (PLV)

- **Definizione di spostamento virtuale:** parte al primo ordine di una variazione di configurazione arbitraria (virtuale) compatibile con i vincoli.
- **Teorema dei lavori virtuali:** la variazione di energia potenziale totale prodotta da spostamenti virtuali e deformazioni virtuali fra loro congruenti è nulla:

$$\delta \Pi_u^{\text{EPT}} = \int_V \delta \boldsymbol{\varepsilon}^T \cdot \boldsymbol{\sigma} dV - \int_V \delta \mathbf{u}^T \cdot \mathbf{b} dV - \int_{S_t} \delta \mathbf{u}^T \cdot \hat{\mathbf{t}} dA = 0$$

se e solo se forze e tensioni sono in equilibrio.

N.B.: Data la sua importanza, si è soliti riferirsi a tale teorema (cioè enunciato dimostrabile!), come "Principio" dei Lavori Virtuali (PLV).

- L'annullarsi della variazione dell'energia potenziale totale è dunque equivalente all'equilibrio e ci si riferisce ad essa come "formulazione integrale dell'equilibrio" (o equilibrio in forma debole). Nel seguito utilizzeremo questa equivalenza:

"PLV" + "congruenza" → "equilibrio"



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori





## Discretizzazione della congruenza

Dall'interpolazione degli spostamenti:

$$\mathbf{u}(\mathbf{x}) = \mathbf{\Omega}(\mathbf{x}) \cdot \mathbf{v}$$

Applicando la congruenza si ha:

$$\boldsymbol{\varepsilon}(\mathbf{x}) = \partial_{\text{cin}} \cdot (\mathbf{\Omega}(\mathbf{x}) \cdot \mathbf{v}) = (\partial_{\text{cin}} \cdot \mathbf{\Omega}(\mathbf{x})) \cdot \mathbf{v} = \mathbf{B}(\mathbf{x}) \cdot \mathbf{v}$$

La matrice:

$$\mathbf{B}(\mathbf{x}) = \partial_{\text{cin}} \cdot \mathbf{\Omega}(\mathbf{x})$$

contiene le derivate delle funzioni di forma lungo gli spostamenti ed è detta **matrice delle funzioni di forma per le deformazioni**.

L'interpolazione delle deformazioni è dunque:

$$\boldsymbol{\varepsilon}(\mathbf{x}) = \mathbf{B}(\mathbf{x}) \cdot \mathbf{v}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Discretizzazione del legame

Considerando le equazioni di legame:

$$\boldsymbol{\sigma}(\mathbf{x}) = \mathbf{E} \cdot \boldsymbol{\varepsilon}(\mathbf{x}) = \mathbf{E} \cdot \mathbf{B}(\mathbf{x}) \cdot \mathbf{v}$$

si ottiene l'interpolazione per gli sforzi:

$$\boldsymbol{\sigma}(\mathbf{x}) = \mathbf{E} \cdot \mathbf{B}(\mathbf{x}) \cdot \mathbf{v}$$

Talvolta si definisce la matrice:

$$\mathbf{S}(\mathbf{x}) = \mathbf{E} \cdot \mathbf{B}(\mathbf{x})$$

delle funzioni di forma per gli sforzi.



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Discretizzazione del PLV (1/2)

Anziché sfruttare direttamente le equazioni di equilibrio puntuale (forma forte), applichiamo il principio dei lavori virtuali, cioè l'equilibrio in forma integrale (forma debole).

Spostamenti e deformazioni virtuali sono dovuti esclusivamente a spostamenti nodali virtuali (una volta scelte, le funzioni di forma sono fissate).

Consideriamo spostamenti e deformazioni virtuali congruenti:

$$\begin{cases} \delta \mathbf{u} = \mathbf{\Omega} \cdot \delta \mathbf{v} \\ \delta \boldsymbol{\varepsilon} = \mathbf{B} \cdot \delta \mathbf{v} \end{cases}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Discretizzazione del PLV (2/2)

Dal PLV:

$$\delta \Pi_{\text{EPT}}^u = \int_V \delta \boldsymbol{\varepsilon}^T \cdot \boldsymbol{\sigma} dV - \int_V \delta \mathbf{u}^T \cdot \mathbf{b} dV - \int_{S_t} \delta \mathbf{u}^T \cdot \hat{\mathbf{t}} dA = 0$$

sostituendo le quantità discretizzate si ottiene:

$$\int_V \delta \mathbf{v}^T \cdot \mathbf{B}(\mathbf{x})^T \cdot \mathbf{E} \cdot \mathbf{B}(\mathbf{x}) \cdot \mathbf{v} dV - \int_V \delta \mathbf{v}^T \cdot \mathbf{\Omega}(\mathbf{x})^T \cdot \mathbf{b} dV - \int_{S_t} \delta \mathbf{v}^T \cdot \mathbf{\Omega}(\mathbf{x})^T \cdot \hat{\mathbf{t}} dA = 0$$

portando fuori dagli integrali gli spostamenti nodali (che non dipendono dalla posizione):

$$\delta \mathbf{v}^T \cdot \left[ \left( \int_V \mathbf{B}^T \cdot \mathbf{E} \cdot \mathbf{B} dV \right) \cdot \mathbf{v} - \int_V \mathbf{\Omega}(\mathbf{x})^T \cdot \mathbf{b} dV - \int_{S_t} \mathbf{\Omega}(\mathbf{x})^T \cdot \hat{\mathbf{t}} dA \right] = 0$$

che conduce alla forma discretizzata del PLV:

$$\delta \mathbf{v}^T \cdot [\mathbf{k} \cdot \mathbf{v} - \mathbf{p}] = 0$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Equilibrio discretizzato

Si definiscono dunque la **matrice di rigidezza** dell'elemento:

$$\mathbf{k} = \int_V \mathbf{B}^T \cdot \mathbf{E} \cdot \mathbf{B} dV$$

e il **vettore dei carichi** nodali dell'elemento:

$$\mathbf{p} = \int_V \boldsymbol{\Omega}(\mathbf{x})^T \cdot \mathbf{b} dV + \int_{S_t} \boldsymbol{\Omega}(\mathbf{x})^T \cdot \hat{\mathbf{t}} dA$$

Dalla forma discretizzata del PLV:

$$\delta \mathbf{v}^T \cdot [\mathbf{k} \cdot \mathbf{v} - \mathbf{p}] = 0$$

tenendo conto dell'arbitrarietà degli spostamenti virtuali, si ottengono le **equazioni di rigidezza**, ovvero l'equilibrio discretizzato:

$$\mathbf{k} \cdot \mathbf{v} = \mathbf{p}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## 4. NON-LINEARITÀ FISICHE



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Equazioni costitutive

**congruenza:**  $\boldsymbol{\varepsilon} = \partial_{\text{cin}} \cdot \mathbf{u}$

**legame:**  $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon}, t)$

**equilibrio:**  $\partial_{\text{sta}} \cdot \boldsymbol{\sigma} + \mathbf{b} = \mathbf{0}$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## Ingredienti

1. Interpolazione degli spostamenti:

$$\mathbf{u}(\mathbf{x}) = \boldsymbol{\Omega}(\mathbf{x}) \cdot \mathbf{v}$$

2. Cinematica lineare:

$$\boldsymbol{\varepsilon}(\mathbf{x}) = \partial_{\text{cin}} \cdot \mathbf{u}(\mathbf{x}) \Rightarrow \boldsymbol{\varepsilon}(\mathbf{x}) = \mathbf{B}(\mathbf{x}) \cdot \mathbf{v}$$

3. Legame non-lineare:

$$\boldsymbol{\sigma} = \tilde{\boldsymbol{\sigma}}(\boldsymbol{\varepsilon}, t) \Rightarrow \boldsymbol{\sigma}(\mathbf{x}, \mathbf{v}, t) = \tilde{\boldsymbol{\sigma}}(\mathbf{x}, \mathbf{B}(\mathbf{x}) \mathbf{v}, t)$$

4. Equilibrio (come PLV):

$$\int_V \delta \boldsymbol{\varepsilon}^T \cdot \boldsymbol{\sigma} dV = \int_V \delta \mathbf{u}^T \cdot \mathbf{b} dV + \int_{S_t} \delta \mathbf{u}^T \cdot \hat{\mathbf{t}} dA$$

$$\delta \mathbf{u} = \boldsymbol{\Omega} \cdot \delta \mathbf{v} \quad \delta \boldsymbol{\varepsilon} = \mathbf{B} \cdot \delta \mathbf{v}$$

$$\cancel{\delta \mathbf{v}^T} \cdot \int_V \mathbf{B}(\mathbf{x})^T \cdot \tilde{\boldsymbol{\sigma}}(\mathbf{x}, \mathbf{B}(\mathbf{x}) \mathbf{v}, t) dV = \cancel{\delta \mathbf{v}^T} \cdot \int_V \boldsymbol{\Omega}(\mathbf{x})^T \cdot \mathbf{b} dV + \cancel{\delta \mathbf{v}^T} \cdot \int_{S_t} \boldsymbol{\Omega}(\mathbf{x})^T \cdot \hat{\mathbf{t}} dA$$

$$\mathbf{F}_{\text{int}}(\mathbf{v}, t) = \mathbf{P}$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



$$\mathbf{F}_{\text{int}}(\mathbf{v}, t) = \int_V \mathbf{B}(\mathbf{x})^T \cdot \tilde{\boldsymbol{\sigma}}(\mathbf{x}, \mathbf{B}(\mathbf{x})\mathbf{v}, t) dV$$

### 5. Linearizzazione dell'equilibrio

$$\mathbf{F}_{\text{int}}(\mathbf{v}_0 + \Delta\mathbf{v}) \approx \mathbf{F}_{\text{int}}(\mathbf{v}_0) + \left. \frac{\partial \mathbf{F}_{\text{int}}}{\partial \mathbf{v}} \right|_{\mathbf{v}_0} \Delta\mathbf{v}$$

$$\mathbf{K}_T(\mathbf{v}_0) \cdot \Delta\mathbf{v} \approx \mathbf{P} - \mathbf{F}_{\text{int}}(\mathbf{v}_0)$$

$$\begin{aligned} \mathbf{K}_T &= \frac{\partial}{\partial \mathbf{v}} \mathbf{F}_{\text{int}} = \frac{\partial}{\partial \mathbf{v}} \int_V \mathbf{B}^T \cdot \tilde{\boldsymbol{\sigma}}(\boldsymbol{\varepsilon}, t) dV = \int_V \mathbf{B}^T \cdot \frac{\partial}{\partial \mathbf{v}} \tilde{\boldsymbol{\sigma}}(\boldsymbol{\varepsilon}, t) dV = \\ &= \int_V \mathbf{B}^T \cdot \frac{\partial \tilde{\boldsymbol{\sigma}}}{\partial \boldsymbol{\varepsilon}} \cdot \frac{\partial \boldsymbol{\varepsilon}}{\partial \mathbf{v}} dV = \int_V \mathbf{B}(\mathbf{x})^T \cdot \mathbf{D}_T(\mathbf{x}) \cdot \mathbf{B}(\mathbf{x}) dV \end{aligned} \quad \mathbf{D}_T = \frac{\partial \tilde{\boldsymbol{\sigma}}}{\partial \boldsymbol{\varepsilon}}$$

### 6. Integrazione numerica

$$\mathbf{K}_T = \int_V \mathbf{B}^T(\mathbf{x}) \mathbf{D}(\mathbf{x}) \mathbf{B}(\mathbf{x}) d\mathbf{x} = \sum_{i,j,k=1}^{N_G} w^{i,N_G} w^{j,N_G} w^{k,N_G} \det(\mathbf{J}(\boldsymbol{\xi}^{ijk,N_G})) \mathbf{B}^T(\boldsymbol{\xi}^{ijk,N_G}) \mathbf{D}(\boldsymbol{\xi}^{ijk,N_G}) \mathbf{B}(\boldsymbol{\xi}^{ijk,N_G})$$

$$\mathbf{F}_{\text{int}} = \int_V \mathbf{B}^T(\mathbf{x}) \boldsymbol{\sigma}(\mathbf{x}) d\mathbf{x} = \sum_{i,j,k=1}^{N_G} w^{i,N_G} w^{j,N_G} w^{k,N_G} \det(\mathbf{J}(\boldsymbol{\xi}^{ijk,N_G})) \mathbf{B}^T(\boldsymbol{\xi}^{ijk,N_G}) \boldsymbol{\sigma}(\boldsymbol{\xi}^{ijk,N_G})$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



## 5. MODELLI DI DANNO

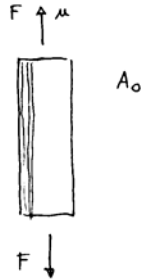


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



### CASO UNIASSIALE



fascio di fibre parallele indipendenti  
 linee con rottura perfettamente fragile  
 (ma allungamento e rottura diverso in ciascuna fibra)

si rimpicciolisce proporzionalmente e l'area  $A$  passa da  $A_0$  a  $0$

$$A_d = A_0 - A \quad \text{area danneggiata}$$

$$\omega = \frac{A_d}{A_0} = 1 - \frac{A}{A_0} \quad \text{indice di danneggiamento } 0 \rightarrow 1$$



Università degli Studi di Firenze  
 Dipartimento di Ingegneria Civile  
 Corso di Meccanica Computazionale

Luca Salvatori



$$\bar{\sigma} = \frac{F}{A} = E_d \epsilon$$

sforzo efficace (agisce sulle microstrutture del materiale)

$$\sigma = \frac{F}{A_0}$$

sforzo nominale (medio, ricerca sull'equilibrio globale)

$$F = A E_d \epsilon$$

$$\sigma = \frac{A}{A_0} E_d \epsilon = \underbrace{(1-\omega)}_{E(\epsilon)} E_d \epsilon$$

$E(\epsilon)$  non lineare

$\omega = f(\epsilon)$  descrive la perdita di caratteristiche meccaniche del materiale

Nel caso di rotture esponenziali:

$$(1-\omega) E_d \epsilon = \sigma = \begin{cases} E_d \epsilon & \epsilon \leq \epsilon_0 \\ f_t \exp\left(-\frac{\epsilon - \epsilon_0}{\epsilon_f}\right) & \epsilon > \epsilon_0 \end{cases}$$

$$\omega = \begin{cases} 0 & \text{per } \epsilon \leq \epsilon_0 \\ 1 - \frac{\epsilon_0}{\epsilon} \exp\left(-\frac{\epsilon - \epsilon_0}{\epsilon_f}\right) & \epsilon > \epsilon_0 \end{cases}$$

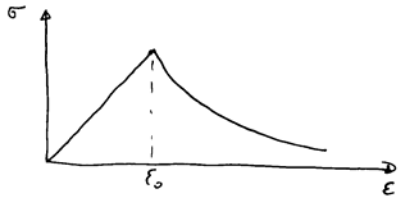
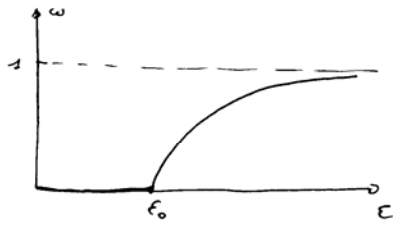
t.c.  $\int_0^\infty \sigma' d\omega = G_f$   
 garantisce la consistenza energetica



Università degli Studi di Firenze  
 Dipartimento di Ingegneria Civile  
 Corso di Meccanica Computazionale

Luca Salvatori





Questo nel caso di carico monotonicamente crescente. Se il pealino viene scaricato il materiale non "si ripara" !  $\Rightarrow$   $w$  non può decrescere



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori

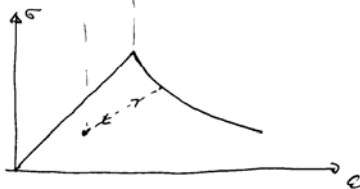
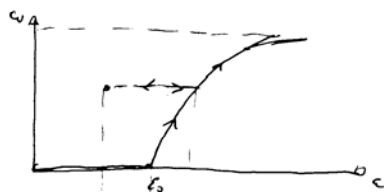


Si può allora:

$$w = f(k)$$

con

$$k(t) = \max_{\tau \leq t} \varepsilon(\tau)$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



Estensione 3D :

MODELLO A DANNO SCALARE ISOTROPO

$$\underline{\underline{\sigma}} = (1-\omega) \underline{\underline{D}} \underline{\underline{\varepsilon}}$$

$$\underline{\underline{D}} = \frac{E}{(1-\nu)(1-2\nu)} \begin{pmatrix} 1-\nu & \nu & \nu & 0 & 0 & 0 \\ \nu & 1-\nu & \nu & 0 & 0 & 0 \\ \nu & \nu & 1-\nu & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1-2\nu}{2} \end{pmatrix}$$

$$\omega = f(k)$$

$$k(t) = \max_{\tau \leq t} \text{Equivalent}(\tau)$$

↳ è necessario, per usare le stesse funzioni scalari di rischio, passare dalla deformazione tensoriale  $\underline{\underline{\varepsilon}}$  ad un valore scalare equivalente:

$$\text{Equivalent} = N(\underline{\underline{\varepsilon}})$$



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



Sette naturali di  $N$ : norma euclidea:

$$N(\underline{\underline{\varepsilon}}) = \|\underline{\underline{\varepsilon}}\|_2 = \sqrt{\sum_{i=1}^6 \varepsilon_i^2}$$

in tal modo si ottiene però un analogo comportamento in trazione e compressione (cosa indesiderata per il d.d.), si possono allora utilizzare espressioni del tipo:

$$N(\underline{\underline{\varepsilon}}) = \sqrt{\sum_{I=1}^3 \langle \varepsilon_I \rangle^2}$$

essendo  $\varepsilon_I$  l' $I$ -esima deformazione principale e avendo indicato con le parentesi angolari l'operatore "parte positiva di":

$$\langle \varepsilon \rangle = \max(0, \varepsilon)$$



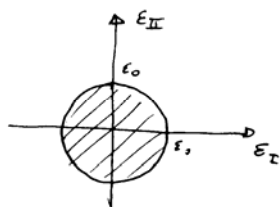
Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori

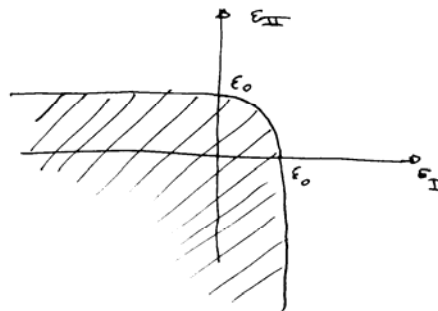




Nello spazio delle deformazioni principali (nel caso 2D) le superfici di rottura  $\epsilon_{equivalente} = \epsilon_0$  sono:



norma euclidea



norma "delle parti positive"



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



Problema del modello a danno scalare isotropo (iSDM)  
"isotropic Scalar Damage Model":

l'isotropia del modello non rispecchia il comportamento del materiale fessurato (le fessure hanno orientazione ben determinata), si ha un irrealistico grado delle caratteristiche meccaniche anche nelle direzioni ~~perpendicolar~~ parallele alle fessure.

D'altra parte mi tratta di un modello numericamente robusto e che conduce a risultati apprezzabili in molte simulazioni:

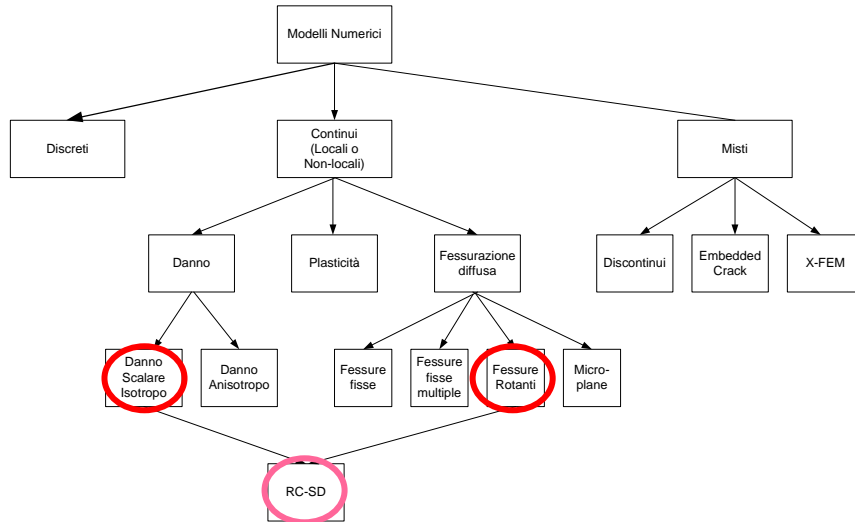


Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori



# Modelli di fessurazione



Università degli Studi di Firenze  
Dipartimento di Ingegneria Civile  
Corso di Meccanica Computazionale

Luca Salvatori

