



**Università degli Studi di Padova**

---

DIPARTIMENTO DI INGEGNERIA CIVILE, EDILE ED AMBIENTALE  
Corso di Laurea Magistrale in Ingegneria Civile

TESI DI LAUREA MAGISTRALE  
INDIRIZZO: STRUTTURE

# **Analisi dinamiche con propagazione coesiva della frattura**

Laureando:

**Robert Damian Vochescu**

Matricola 1057079

Relatore:

**Prof. Ing. Stefano Secchi**

Controrelatore:

**Prof. Ing. Carmelo Majorana**

## **Colophon**

Questo lavoro è stato realizzato con L<sup>A</sup>T<sub>E</sub>X su Windows 8.1.

## **Contatti**

robertdamian.vochescu@gmail.com Robert Damian Vochescu

18 aprile 2015

# Dedica

Dedico questo lavoro in primo luogo alla mia famiglia e a tutti coloro che in questi anni mi hanno supportato e aiutato a raggiungere questo traguardo speciale.



# Ringraziamenti

Ringrazio i miei genitori, due persone meravigliose, senza le quali neanche avrei potuto sognare di arrivare dove sono ora e di avermi educato alla persona che sono diventato.

Ringrazio il Prof. Ing. Stefano Secchi per la disponibilità, per il tempo e per la pazienza nonché per tutte le cose che ho appreso non solo durante il percorso della tesi. È stato ed è per me una persona di assoluto riferimento.

Ringrazio i compagni di corso Enrico, Matteo, Daniele, Massimiliano, Gianluca, Filippo e Sara per aver condiviso con me una parte del percorso universitario ed averlo reso speciale.

Ringrazio Marco, che oltre ad essere un collega speciale, è anche un fantastico amico.

Ringrazio Anna, la mia dolce metà, per aver avuto così tanta pazienza con me e per avermi reso una persona migliore.

Ringrazio tutti coloro che in qualche modo mi hanno supportato in questi anni e mi hanno aiutato a crescere.



# Sommario

Oggetto del lavoro è la simulazione numerica della propagazione della frattura in solidi bidimensionali utilizzando il modello di frattura coesivo. Le soluzioni analitiche a tali problemi sono complesse e limitate, pertanto, la modellazione numerica gioca un ruolo significativo nella comprensione del comportamento dei materiali e delle strutture in cui si ha la propagazione della frattura.

Il modello numerico si basa sul classico metodo agli elementi finiti e il criterio di avanzamento e nucleazione della frattura si basa sul criterio Rankine. Rispetto alla maggior parte dei lavori presenti in letteratura dove la discontinuità all'interno del campo degli spostamenti dovuta alla frattura viene simulata attraverso il Metodo degli Elementi Finiti Esteso (XFEM), o attraverso elementi di interfaccia nel caso in esame viene utilizzata la tecnica del remeshing.

Il remeshing viene ritenuto un pregio del metodo in quanto consente un'adattività spaziale che segue l'evoluzione del fenomeno. Il percorso di frattura è modellato in maniera discreta, aggiornando continuamente il dominio mano a mano che la frattura si propaga tramite un meshatore automatico basato sulla triangolazione di Delaunay.

Per superare la singolarità del campo tensionale si utilizza una legge coesiva estrinseca e la tecnica del remeshing che adatta il dominio ad ogni avanzamento della frattura ottenendo così una naturale evoluzione della frattura come soluzione dell'algoritmo.

Nel lavoro in esame è stato ampliato il problema della propagazione della frattura dal campo quasi-statico al campo dinamico attraverso un codice di calcolo non commerciale in C++.

Si vuole studiare come l'influenza della dinamica, intesa come rapida evoluzione del sistema meccanico e quindi anche come inerzia, incida sul comportamento della frattura. Per la soluzione del problema dinamico è stato implementato l'algoritmo implicito di Newmark che ha il pregio di essere incondizionatamente stabile e di non avere dissipazione numerica.





# Abstract

Object of this work is the numerical simulation of crack propagation in two-dimensional solids using the cohesive fracture model. Analytical solutions to these problems are complex and limited, therefore, the numerical modeling plays a significant role in the understanding of the behavior of materials and structures in which occurs crack propagation.

Numerical model based, on the classical finite element method, is used in combination with the Rankine criterion for fracture nucleation and advancement. In other works reported in the literature the discontinuity in the displacement field, due to the fracture, is simulated through the Extended Finite Element Method (XFEM) or interface elements, in this case the re-meshing technique is used .

The remeshing is considered a quality of the method because it allows a spatial adaptivity that follows the evolution of the phenomenon. The fracture path is modeled discretely, continuously updating the domain as the fracture propagates through an efficiently automatic remeshing technique based on Delaunay triangulation.

To overcome the singularity of the stress field we use a extrinsic cohesive law and remeshing technique that adapt the domain at every advancement of fracture tip obtaining fracture path evolution as a natural solution of the algorithm.

In this work the numerical model of quasi-static crack propagation has been expanded to the dynamic crack propagation using a C++ numerical code.

In this work the influence of dynamic fracture evolution is investigated, meaning how rapid evolution of the mechanical system affect the behavior of the fracture. For the solution of the dynamic problem implicit Newmark algorithm is used which has the advantage of being unconditionally stable and have no numerical dissipation.



# Elenco dei simboli

$\boldsymbol{\sigma}$	Tensore delle tensioni
$\boldsymbol{\varepsilon}$	Tensore di deformazione
$\ddot{x}$	Derivata seconda incognita
$\dot{x}$	Derivata prima incognita
$\eta$	Ordinata, coordinata locale
$\Gamma_t$	Contorno caricato
$\Gamma_u$	Contorno vincolato
$\lambda$	Vettore dei moltiplicatori di Lagrange
$\mathbf{t}$	Trazioni sul contorno
$\nu$	Coefficiente di Poisson
$\rho_s$	Densità
$\boldsymbol{\sigma}$	Vettore delle tensioni
$\boldsymbol{\varepsilon}$	Vettore delle deformazioni
$\xi$	Ascisa, coordinata locale
$B$	Matrice derivate delle funzioni di forma
$C$	Matrice di smozamento
$C_{ijrs}$	Tensore di cedevolezza
$D$	Matrice di elasticità
$E$	Modulo Elastico
$f^{(\dot{\quad})}$	Vettore della forzante esterna
$f^{(\lambda)}$	Vettore dei vincoli imposti

$f^{(p)}$	Vettore delle pressioni al contorno
$f^{(u)}$	Vettore delle forze esterne
$FEM$	Finite Element Method (metodo agli elementi finiti)
$J$	Matrice jacobiana
$K$	Matrice di rigidezza
$K_1$	Matrice di Softening
$L$	Operatore differenziale
$M$	Matrice delle masse
$N$	Matrice delle funzioni di forma
$R^k$	Vettore del residuo
$u$	Spostamenti
$W$	Funzione densità di energia di deformazione
$w$	Funzioni peso
$W_D$	Lavoro delle forze di Smorzamento.
$W_E$	Lavoro di richiamo Elastico.
$W_I$	Lavoro delle forze d'Inerzia.
$W_{Ext}$	Lavoro delle forze Esterne.
$x$	Incognita
$\Phi$	Matrice Modale.
$\ddot{u}$	Spostamenti
$\Delta l$	Avanzamento Istantaneo della Frattura.
$\delta L$	Incremento di lavoro esterno
$\Delta t$	Passo temporale.
$\Delta x$	Incognita
$\dot{u}$	Spostamenti
$\mathfrak{N}$	Operatore di proiezione della mesh.
$\mathcal{M} + 1$	Mesh al passo $t + 1$ .

$\mathcal{M}$	Mesh al passo $t$ .
$\sigma_u$	Massima tensione di trazione del materiale.
$A$	Matrice di rigidità tangente
$c_d$	Velocità delle onde di dilatazione.
$c_R$	Velocità delle onde di Rayleigh.
$c_s$	Velocità delle onde di taglio.
$c_{m,l}$	Velocità delle onde meccaniche longitudinali.
$c_{m,s}$	Velocità delle onde meccaniche trasversali.
$L_e$	Lavoro esterno
$L_i$	Lavoro esterno
$w_c$	Massima apertura nel legame coesivo.



# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
<b>2</b>	<b>Meccanica della frattura</b>	<b>5</b>
2.1	Introduzione . . . . .	5
2.2	Fisica della frattura . . . . .	6
2.3	I modelli per lo studio della frattura . . . . .	8
2.3.1	Approccio continuo . . . . .	9
2.3.2	Approccio discontinuo . . . . .	11
2.3.3	Approcci combinati . . . . .	12
2.4	Il modello coesivo . . . . .	12
2.4.1	Il modello di frattura fittizio . . . . .	13
2.4.2	Tipologie di modelli coesivi . . . . .	14
2.4.3	Modello coesivo impiegato . . . . .	15
<b>3</b>	<b>Equazioni del problema strutturale</b>	<b>17</b>
3.1	Generalità . . . . .	17
3.2	Problema Elastico . . . . .	17
3.3	Condizioni al contorno . . . . .	18
3.4	Legame costitutivo . . . . .	19
<b>4</b>	<b>Applicazione FEM al problema</b>	<b>21</b>
4.1	Il metodo agli elementi finiti . . . . .	21
4.2	Applicazione al problema strutturale . . . . .	24
4.2.1	Approssimazione degli spostamenti . . . . .	24
4.2.2	Derivate . . . . .	25
4.2.3	Calcolo delle tensioni e delle deformazioni . . . . .	26
4.2.4	Forma debole del problema strutturale . . . . .	27
4.3	Condizioni al contorno . . . . .	27
4.3.1	Applicazione condizioni al contorno . . . . .	28
4.3.2	Metodo dei moltiplicatori di Lagrange . . . . .	29
4.4	Problema non lineare al contorno . . . . .	32
4.4.1	Metodo Full Newton . . . . .	32
4.5	Rigidezza negativa - Matrice di softening . . . . .	36

4.6	Scelta degli elementi finiti . . . . .	41
4.6.1	Convergenza . . . . .	41
4.6.2	Elementi finiti impiegati . . . . .	42
4.6.3	Integrazione numerica - quadratura . . . . .	43
4.7	Stato piano di deformazione . . . . .	43
4.8	Matrice di rigidezza . . . . .	44
4.9	Matrice di massa . . . . .	45
<b>5</b>	<b>Soluzione Numerica</b>	<b>47</b>
5.1	Problemi di equilibrio . . . . .	47
5.2	Discretizzazione nello spazio . . . . .	47
5.3	Discretizzazione nel tempo . . . . .	50
5.3.1	Metodi di Newmark . . . . .	52
<b>6</b>	<b>Implementazione numerica</b>	<b>57</b>
6.1	Introduzione . . . . .	57
6.2	La programmazione orientata agli oggetti . . . . .	57
6.3	Modulazione del codice di calcolo . . . . .	58
6.3.1	Main . . . . .	59
6.3.2	Soluzione del problema della frattura . . . . .	59
6.3.3	Soluzione del problema non lineare . . . . .	61
6.3.4	Soluzione del problema dinamico . . . . .	62
6.4	Nucleazione e propagazione della frattura . . . . .	63
6.5	Sdoppiamento dei nodi . . . . .	64
6.6	Generazione della mesh e remeshing . . . . .	66
6.6.1	Remeshing . . . . .	68
6.7	Proiezione delle variabili . . . . .	68
6.7.1	La mappa di proiezione consistente . . . . .	70
6.7.2	Algoritmo di proiezione . . . . .	72
6.7.3	Step di bilanciamento . . . . .	74
6.8	Ciclo full Newton . . . . .	76
6.8.1	Criteri di uscita . . . . .	77
6.9	Postprocessing . . . . .	78
<b>7</b>	<b>Applicazioni</b>	<b>79</b>
7.1	Introduzione . . . . .	79
7.2	Three point bending test . . . . .	79
7.2.1	Geometria analizzata e svolgimento delle analisi . . . . .	82
7.2.2	Commenti alle analisi svolte . . . . .	82
7.3	Ulteriori considerazioni . . . . .	83
7.4	Andamento a scatti della frattura . . . . .	88



<b>8</b>	<b>Analisi dinamiche</b>	<b>91</b>
8.1	Introduzione . . . . .	91
8.2	Ciclo di bilanciamento . . . . .	91
8.2.1	Balance step: andamento della convergenza . . . . .	92
8.2.2	Balance step: correzioni apportate . . . . .	92
8.3	Influenza dello smorzamento . . . . .	95
8.3.1	Smorzamento alla Rayleigh . . . . .	96
8.3.2	Smorzamento e frattura . . . . .	96
8.4	Analisi energetica . . . . .	98
8.5	Rilascio di energia elastica . . . . .	101
8.5.1	Analisi critica della propagazione statica . . . . .	102
8.5.2	Effetti inerziali dovuti all'apertura della frattura . . . . .	105
8.6	Algoritmo di integrazione temporale . . . . .	107
8.7	Fisica della propagazione della frattura . . . . .	107
8.7.1	Introduzione . . . . .	107
8.7.2	Velocità infinita di propagazione . . . . .	107
8.7.3	Influenza delle forze inerziali . . . . .	109
8.7.4	Comportamento stepwise . . . . .	110
8.8	Caso test . . . . .	110
8.8.1	Dipendenza dalla discretizzazione spaziale . . . . .	111
8.8.2	Dipendenza dalla discretizzazione temporale . . . . .	112
8.8.3	Dipendenza dal legame coesivo . . . . .	113
8.8.4	Andamento stepwise . . . . .	113
8.9	Legame $\Delta t - \Delta l$ con la frattura a scatti . . . . .	115
8.9.1	Velocità di propagazione . . . . .	119
8.9.2	Tempo di integrazione . . . . .	120
<b>9</b>	<b>Conclusioni</b>	<b>121</b>
<b>A</b>	<b>Implementazione</b>	<b>123</b>
A.1	Implementazione matrici . . . . .	123
A.1.1	Matrice di rigidità . . . . .	123
A.1.2	Matrice di smorzamento . . . . .	123
A.1.3	Matrice delle masse . . . . .	124
A.1.4	Matrice tangente . . . . .	124
A.2	Cicli Newton-Raphson . . . . .	125
A.2.1	Ciclo di bilanciamento . . . . .	125
A.2.2	Ciclo Newton-Raphson . . . . .	128
<b>B</b>	<b>Validazione del codice di calcolo</b>	<b>133</b>
B.1	Introduzione . . . . .	133
B.2	Codice sistema ad 1 gdl . . . . .	133
B.3	Risultati sistema ad 1 gdl . . . . .	137
B.4	Confronto con codice di calcolo commerciale . . . . .	139



# Elenco delle tabelle

7.1	Materiale e geometria del provino. . . . .	82
8.1	Materiale e geometria del provino. . . . .	92
8.2	Caratteristiche provino. . . . .	97
8.3	Parametri di smorzamento. . . . .	98
8.4	Caratteristiche provino. . . . .	112
8.5	Caratteristiche provino. . . . .	114
8.6	Materiale e geometria del provino . . . . .	114
8.7	Valori del passo di integrazione in funzione del parametro di avanzamento. . . . .	118



# Elenco delle figure

1.1	Frattura coesiva: zona di processo. . . . .	2
1.2	Storia temporale della lunghezza della frattura. . . . .	2
2.1	Rappresentazione semplificata di materiale granulare con una fessura iniziale e zona di processo (area grigia) con micro- separazioni. . . . .	6
2.2	Modi elementari di sollecitazione della fessura (da sx verso dx); apertura, scivolamento nel piano e strappo. . . . .	7
2.3	Zona plastica nell'intorno dell'apice della frattura nei mate- riali duttili. . . . .	9
2.4	Approccio continuo allo studio del fenomeno. . . . .	10
2.5	Tipologie di approcci per lo studio della frattura. . . . .	11
2.6	Rappresentazione della zona di processo nei modelli di zona coesiva. . . . .	11
2.7	Zona di processo con tensioni di richiusura secondo Hillerborg.	13
2.8	Legame costitutivo impiegato. . . . .	15
4.1	Trasformazione isoparametrica per un elemento piano a 4 nodi.	25
4.2	Interpretazione grafica del metodo di Newton-Raphson. . . . .	34
4.3	Trave doppio appoggio: confronto rigidità all'avanzare della frattura. . . . .	37
4.4	Geometria del nodo master e slave in situazione deformata e indeformata. . . . .	37
4.5	Scomposizione della geometria della frattura. . . . .	39
4.6	Alcuni elementi bidimensionali tipici con diversi gradi di in- terpolazione dei valori nodali. . . . .	41
4.7	Matrice di massa: elemento asta. . . . .	46
6.1	Diagramma di flusso che illustra i passi seguiti nella soluzione del problema della frattura. . . . .	61
6.2	Formazione di una nuova frattura. . . . .	65
6.3	Modi di avanzamento per fratture esistenti. . . . .	66
6.4	Propagazione della frattura con cambiamenti nella topologia del dominio. . . . .	67

6.5	Meshatura di una regione, esempio preso da [39]. . . . .	68
6.6	Avanzamento multiplo della frattura all'interno del medesimo passo temporale. . . . .	69
6.7	Trasporto: (a) punti Gauss dalla vecchia mesh, (b) nodi dalla vecchia mesh, (c) nodi sulla nuova mesh, (d) punti Gauss sulla nuova mesh. . . . .	71
6.8	Algoritmo di proiezione delle forze nodali: da sinistra verso destra si hanno le forze nodali al tempo $n$ sulla mesh $m$ ; le forze nodali al tempo $n$ sulla mesh $m + 1$ riferite ai nodi pre- cedenti ed infine le forze nodali al tempo $n$ sulla mesh $m + 1$ riferite ai nodi di quest'ultima. . . . .	71
6.9	Fasi del funzionamento dell'algoritmo di proiezione. . . . .	74
7.1	Concetto di modello coesivo . . . . .	81
7.2	Rapporti geometrici dei provini analizzati. . . . .	82
7.3	Table impiegata nelle analisi. . . . .	83
7.4	Transizione duttile-fragile all'aumentare della dimensione del provino. . . . .	84
7.5	Transizione duttile-fragile per diverse scale del provino. . . . .	84
7.6	Avanzamento della frattura: deformata (non in scala) di alcuni istanti temporali. . . . .	85
7.7	Avanzamento della frattura: mappe di colore relative alla tensione $XX$ . (I valori sono relativi ai punti Gauss) . . . . .	86
7.8	Influenza della dimensione della mesh. . . . .	87
7.9	Costo computazionale (espresso in secondi) per le tre prove. . . . .	88
7.10	Andamento a scatti della frattura. . . . .	89
8.1	Profili di convergenza per gli step immediatamente successivi all'apertura della frattura. . . . .	93
8.2	Profili di convergenza per alcuni step dell'analisi. . . . .	93
8.3	Grafico tempo - spostamento: confronto tra soluzione ottenu- ta con e senza ciclo di bilanciamento. . . . .	94
8.4	Grafico tempo - velocità: confronto tra soluzione ottenuta con e senza ciclo di bilanciamento. . . . .	94
8.5	Grafico tempo - accelerazione: confronto tra soluzione otte- nuta con e senza ciclo di bilanciamento. . . . .	95
8.6	Andamento di $\alpha$ in funzione della lunghezza della frattura e della scelta dei modi. . . . .	97
8.7	Andamento di $\beta$ in funzione della lunghezza della frattura e della scelta dei modi. . . . .	97
8.8	Densità di energia di smorzamento ( $J/mm^2$ ) . . . . .	98
8.9	Table impiegata nelle analisi - caso test 3. . . . .	99
8.10	Andamento dei lavori: sistema conservativo. . . . .	100
8.11	Andamento dei lavori: sistema non conservativo. . . . .	100

8.12	Densità di energia di frattura ( $J/mm^2$ ) . . . . .	101
8.13	Mesh e BC del modello numerico: il percorso della frattura risulta predeterminato la frattura è "cucita" con barre cut-off. . . . .	102
8.14	Table impiegata nelle analisi del provino cucito con elementi cut-off. . . . .	102
8.15	Grafico tempo - spostamenti relativo al nodo centrale. . . . .	103
8.16	Grafico tempo - velocità relativo al nodo centrale. . . . .	103
8.17	Grafico tempo - accelerazioni relativo al nodo centrale. . . . .	103
8.18	Grafico tempo - reazioni vincolari. . . . .	104
8.19	Accelerazione AY negli istanti prima e dopo il rilascio degli elementi cut-off. . . . .	104
8.20	Grafico tempo - accelerazione AY del nodo d'apice dell'intaglio del provino durante la propagazione della frattura. Si nota come si ha un improvviso incremento dell'accelerazione. . . . .	106
8.21	Densità di energia cinetica ( $J/mm^2$ ): istanti prima e dopo l'apertura della frattura. . . . .	106
8.22	Il $\Delta l$ di cui la frattura aumenta ogni volta che viene soddisfatto il criterio di rottura. . . . .	108
8.23	Schematizzazione dell'andamento stepwise. . . . .	110
8.24	Andamento della frattura derivante da un'analisi statica. Si nota come l'andamento tenda al continuo mano a mano che diminuisce la lunghezza di avanzamento. . . . .	111
8.25	Grafico tempo - posizione della frattura per diverse dimensioni della mesh. . . . .	112
8.26	Grafico tempo - posizione della frattura per diversi tempi di integrazione. . . . .	113
8.27	Grafico tempo - posizione della frattura per vari legami coesivi. . . . .	114
8.28	Grafico tempo - posizione della frattura in cui si vince l'andamento a scatti, o stepwise della frattura. . . . .	115
8.29	Grafico tempo - posizione della frattura al variare del passo spaziale di avanzamento. Si nota come l'andamento stepwise diventi meno evidente all'aumentare del passo spaziale di avanzamento. . . . .	116
8.30	Grafico tempo - posizione della frattura al variare del passo temporale di avanzamento. Si nota come l'andamento stepwise diventi meno evidente all'aumentare del passo temporale di integrazione. . . . .	116
8.31	Grafico tempo - velocità al variare dell'inclinazione della table normalizzata con la velocità delle onde di Rayleigh. . . . .	119
B.1	Sistema ad 1 GDL. . . . .	137
B.2	Risposta del sistema - spostamenti. . . . .	138
B.3	Risposta del sistema - velocità. . . . .	138
B.4	Risposta del sistema - accelerazioni. . . . .	139

B.5	Modello in condizioni piane di deformazione impiegato. . . . .	139
B.6	Rampa impiegata per la validazione. . . . .	140
B.7	Risposta del sistema - spostamenti. . . . .	140
B.8	Risposta del sistema - velocità. . . . .	141
B.9	Risposta del sistema - accelerazioni. . . . .	141
B.10	Risposta del sistema - reazioni vincolari. . . . .	142



# Capitolo 1

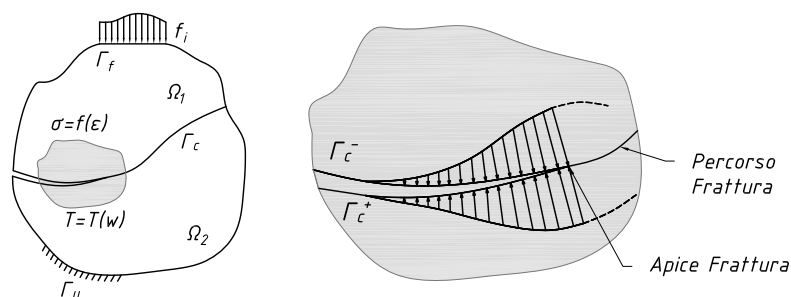
## Introduzione

L'oggetto di questa tesi è la simulazione numerica della propagazione della frattura in solidi caratterizzati da comportamento quasi-fragile. La simulazione numerica di tale comportamento richiede una diversa e specifica formulazione rispetto ai classici problemi FEM in quanto la frattura impone una discontinuità nel campo degli spostamenti. Per rispettare le condizioni di compatibilità, che la soluzione classica richiede, si deve arricchire il modello matematico-numerico, introducendo degli elementi che forniscono al modello la capacità di rappresentazione della discontinuità. Nel caso in esame la propagazione della frattura è raggiunta utilizzando un remeshing spaziale e un modello coesivo che ripristina continuità alla frattura e permette una rottura graduale.

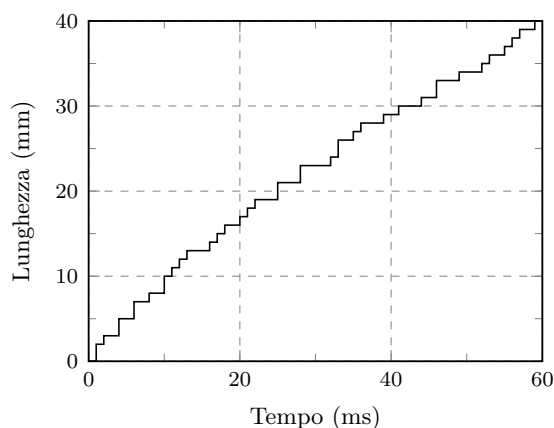
In letteratura sono presenti numerose teorie che permettono di studiare la propagazione della frattura all'interno di un solido. L'impiego di una teoria, rispetto ad un'altra dipende da molti fattori, tra i quali vi è il materiale costituente il corpo entro il quale la frattura si propaga (come verrà elencato nel capitolo 2).

Il modello che meglio si presta allo studio di problemi di propagazione di fratture in materiali quasi-fragili, come il calcestruzzo, è il *modello di fessura fittizia* introdotto da Barenblatt e successivamente modificato da Hillerborg [20], che concentra tutti i fenomeni dissipativi non lineari in una zona nell'intorno dell'apice della frattura, denominata *zona di processo*. Tale modello richiede la definizione del legame costitutivo del materiale non danneggiato e del legame trazione-separazione all'interno della zona di processo. Il materiale non danneggiato nei casi analizzati è assunto lineare, elastico ed isotropo, mentre, tra i numerosi legami trazioni-separazione presenti in letteratura, è stato impiegato un modello in cui le trazioni all'interno della zona di processo decrescono linearmente con l'apertura della frattura.

La simulazione del fenomeno avviene tramite l'impiego di un software non commerciale agli Elementi Finiti sviluppato ad-hoc per lo studio del fenomeno. La strategia impiegata per seguire l'avanzamento della frattura



**Figura 1.1:** Frattura coesiva: zona di processo.



**Figura 1.2:** Storia temporale della lunghezza della frattura.

nel solido consiste nel controllo della massima tensione di trazione principale nell'intervallo temporale corrente  $t_n$ ; qualora essa venga superata viene aperta una nuova frattura. Questo cambiamento del dominio e del contorno richiede un aggiornamento della mesh, che varia continuamente fintanto che il criterio di rottura non è soddisfatto. Quando questo accade, si ha l'aggiornamento dello step temporale (da  $t_n$  a  $t_{n+1}$ ) e, quindi, l'incremento del carico esterno. In questo modo si ha il pregio di non avere una mesh dipendenza nel percorso della frattura come può avvenire, ad esempio, impiegando elementi coesivi.

Analizzando il fenomeno senza la partecipazione delle masse, sotto condizioni quasi-statiche e plottando un grafico tempo - posizione della frattura, si ottiene un andamento della frattura graduale, o a scatti (Figura 1.2). Questo andamento rispecchia effettivamente la fisica del fenomeno ma, si ritiene, che senza la partecipazione delle masse questo andamento sia una semplice coincidenza algoritmica in quanto il codice, per come è stato concepito, prevede un avanzamento della frattura all'interno del medesimo step temporale fintanto che non viene raggiunta una situazione di equilibrio.

Si ritiene, che per una corretta simulazione del fenomeno, l'analisi più

opportuna da compiere sia un'analisi nel transitorio. L'andamento della frattura a scatti, fisicamente, presenta un'analogia con la formazione dei terremoti. Si ha un accumulo di energia elastica nei corpi seguito da un repentino movimento della faglia. Il passaggio da energia elastica a onda sismica è inerzia. Si ritiene dunque non trascurabile il contributo inerziale.

Per l'integrazione temporale, tra i vari metodi disponibili in letteratura, è stato scelto l'algoritmo di *Newmark* implicito.

Il problema principale che sorge nell'utilizzare un approccio che vede la modellazione della frattura e, quindi, una variazione del dominio, è quello del trasferimento delle variabili di interesse dal passo temporale precedente a quello corrente (in accordo con l'algoritmo di discretizzazione temporale), eventualmente su un nuovo dominio, con una nuova mesh di elementi finiti.

Si è proceduto impiegato un algoritmo di proiezione che renda le variabili *consistenti* sul nuovo dominio a cui segue uno step di bilanciamento che si occupa, tramite un ciclo Newton-Raphson di apportare le eventuali correzioni alle incognite della soluzione del dominio precedente proiettata sul nuovo dominio, prima dell'aggiornamento dei carichi esterni al passo temporale corrente.

Nel capitolo 2 sarà fornita una panoramica generale sui vari approcci allo studio della frattura. In particolare sarà esposto il modello di frattura coesivo impiegato nel presente lavoro e adatto allo studio di materiali quasi-fragili.

Nel capitolo 3 saranno presentate le equazioni differenziali che reggono il fenomeno.

Nel capitolo 4 sarà presentato brevemente il metodo risolutivo impiegato, ovvero il Metodo degli Elementi Finiti, giustificando le scelte sugli elementi finiti impiegati e descrivendo il trattamento delle condizioni al contorno e delle forzanti applicate.

Nel capitolo 5 saranno presentate le equazioni discretizzate in spazio ed in tempo per problemi agli spostamenti.

Nel capitolo 6 sarà brevemente introdotto il codice di calcolo impiegato, mettendo in evidenza la procedura impiegata per simulare l'avanzamento della frattura e presentando quanto implementato per il ciclo di bilanciamento e per l'algoritmo di integrazione temporale.

Nei capitoli 7 e 8 saranno presentate alcune applicazioni del codice di calcolo a problemi di interesse ingegneristico facendo alcune riflessioni sul ruolo delle forze inerziali nello studio del fenomeno.

Infine, nel capitolo 9 saranno esposti i risultati ottenuti, gli sviluppi futuri ed i limiti incontrati nel procedere con questo tipo di approccio.



## Capitolo 2

# Meccanica della frattura

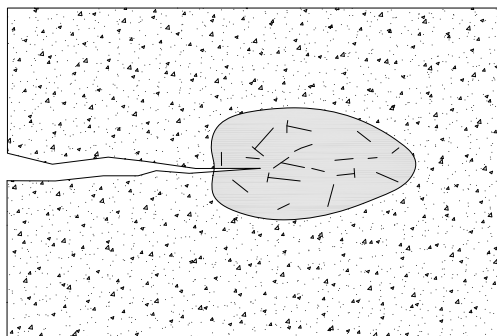
### 2.1 Introduzione

La Meccanica della Frattura è una disciplina che si sta affermando in tutti i settori più avanzati dell'ingegneria come metodologia di progetto e di verifica strutturale. In modo particolare, la Meccanica della Frattura ha come obiettivo principale quello di comprendere il comportamento meccanico dei materiali e più in generale quello degli elementi strutturali qualora essi siano soggetti a forze esterne di natura qualsiasi.

Per poter analizzare e simulare correttamente a livello macroscopico il processo di propagazione di una fessura è necessario conoscere nel modo più dettagliato possibile lo stato energetico e tensoriale nei pressi dei difetti meccanici presenti. A seconda della tipologia e comportamento del materiale considerato sono state sviluppate diverse teorie. Materiali come il calcestruzzo, tradizionalmente vengono considerati come fragili (*brittle materials*). In realtà essi appartengono alla categoria dei materiali quasi fragili (*quasi-brittle materials*). Per via della loro natura, composti di pasta e aggregati di forme e dimensioni diverse, spesso si generano al loro interno pori o microfessure che anche per piccoli valori di carico amplificano il valore dello sforzo al loro apice. Inoltre, anche da un punto di vista teorico, la descrizione del processo di frattura per questa famiglia di materiali è molto complessa e attualmente oggetto di studio. Discorso che invece non vale per materiali fragili, che fanno interamente affidamento alla Meccanica della Frattura elastica lineare (LEFM), oppure per i materiali duttili per la cui descrizione si rimanda alla teoria della plasticità.

Due sono le teorie, diffuse negli ultimi decenni, che si sono occupate dei materiali con comportamento quasi-fragile: da un lato quella degli *smearred crack models*, e dall'altro quella dei *discrete cohesive crack models*.

I *discrete cohesive crack models* hanno lo scopo di modellare la nucleazione e la propagazione di una macro fessura, contrariamente, gli *smearred crack models* si basano sull'idea che, nei materiali quasi fragili, si hanno del-



**Figura 2.1:** Rappresentazione semplificata di materiale granulare con una fessura iniziale e zona di processo (area grigia) con micro-separazioni.

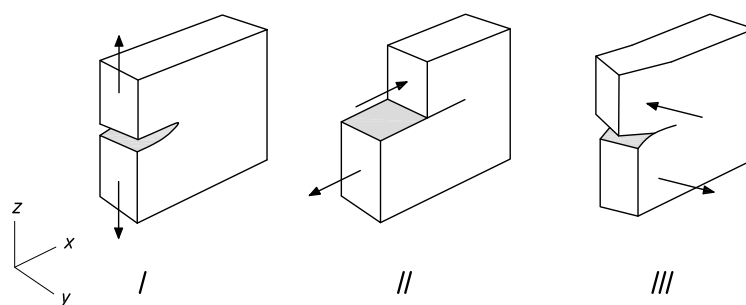
le micro fessure che, vicino alla situazione di collasso, formano una macro fessura. In questi ultimi modelli nè le micro fessure nè la macro fessura sono modellate, ma attraverso una relazione costitutiva si diminuiscono le proprietà meccaniche del materiale.

## 2.2 Fisica della frattura

Da un punto di vista meccanico, la frattura può essere descritta tramite una serie di eventi successivi. Il primo di essi consiste nella nucleazione di micro separazioni presenti in una piccola zona adiacente ad una macro-fessura esistente, denominata *zona di processo* (si veda la Figura 2.1). Le micro separazioni possono essere considerate come piccole fessure nella micro-struttura del materiale; si formano in maniera disomogenea dove localmente i legami interatomici sono relativamente deboli e le sollecitazioni alte o in presenza di imperfezioni.

La nucleazione delle micro separazioni è un processo irreversibile, che riduce la resistenza e la rigidezza del materiale. Tutto questo è accompagnato da una dissipazione di energia dovuta alle deformazioni plastiche che si sviluppano nel materiale e da rilascio di calore generato per attrito. Se i carichi esterni aumentano, le micro fratture esistenti possono crescere, formando fessure più ampie e dando luogo a concentrazioni di tensioni nel loro intorno. La coalescenza delle fessure unita al decadimento di rigidezza e resistenza del materiale porta rapidamente al collasso della struttura.

I possibili modi con cui una frattura può manifestarsi sono 3 e sono indicati in Figura 2.2. Il Modo I è detto di *apertura*, simmetrico rispetto alla fessura; il Modo II è detto di *scivolamento nel piano*, antisimmetrico rispetto all'asse  $x$ . Esiste peraltro un terzo modo elementare, relativo a stati tridimensionali: il Modo III o di *scivolamento fuori dal piano*, antisimmetrico rispetto al piano  $xz$ . Tale modo è quello caratteristico con cui si strappano i fogli di carta. I tre



**Figura 2.2:** Modi elementari di sollecitazione della fessura (da sx verso dx); apertura, scivolamento nel piano e strappo.

suddetti modi rappresentano tutti e i soli modi esistenti per sollecitare una fessura.

L'evoluzione del fenomeno, dalla micro separazione al collasso finale, è differente a seconda del tipo di materiale. Un importante fattore è rappresentato dalla natura della micro struttura del materiale, o più precisamente, dall'omogeneità del materiale e dall'ammontare di difetti iniziali. I ceramici sono un esempio di materiale tipicamente omogeneo, in cui la micro struttura è composta da atomi mutuamente legati tramite la condivisione di elettroni. La natura di questo legame è estremamente rigida e di conseguenza, la grandezza della zona di processo in questo tipo di materiali è dell'ordine delle distanze interatomiche.

Dall'altro lato si trovano i materiali eterogenei come il conglomerato cementizio composto da pasta di cemento indurita e aggregati di varie forme e dimensioni. A causa delle differenti rigidità e forze coesive dei due ingredienti, ci sono maggiori fonti per possibili nucleazioni di micro separazioni. Inoltre, a causa del ritiro, una struttura in conglomerato cementizio può contenere già micro fratture al suo interno. Appare evidente che in questo caso la zona di processo risulterà più estesa e le sue dimensioni saranno dello stesso ordine di grandezza delle dimensioni dell'elemento.

Un altro fattore importante è rappresentato dalle condizioni di carico. I solidi a struttura cristallina come i metalli mostrano un comportamento sensibilmente diverso sotto carichi monotonicamente crescenti rispetto a carichi ciclici. In questo ultimo caso, il carico che porta al collasso risulta molto inferiore rispetto al caso in cui i carichi crescano in maniera monotona; questo fenomeno è noto con il nome di *fatica* e non verrà trattato poiché esula dai contenuti di questa tesi.

Infine, anche le condizioni ambientali possono influire sul processo che porta al collasso di una struttura. Condizioni aggressive (forte umidità, presenza di acqua salata, etc.) possono accelerare il processo di frattura nei metalli per carichi relativamente bassi; questo fenomeno è noto come *stress corrosion*. Anche i materiali fibrorinforzati a matrice polimerica (FRP) sono

influenzati dalle condizioni ambientali infatti, per temperature alte, si può assistere ad un brusco decremento di resistenza unito ad uno scorrimento relativo tra matrice e fibre.

### 2.3 I modelli per lo studio della frattura

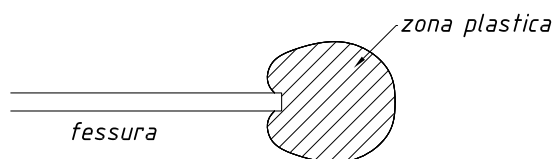
Senza entrare nel merito della micro struttura del materiale, della storia di carico e delle condizioni ambientali, il processo che porta alla frattura di un materiale può essere riassunto tramite la nucleazione, la crescita e la coalescenza di piccole fessure, o più in generale, di discontinuità nel materiale. Nella meccanica del continuo classica il materiale riempie lo spazio che occupa completamente, senza lasciare vuoti e le sue proprietà sono descritte da funzioni continue. Ovviamente questa assunzione non permette di dare informazioni sulla micro struttura del materiale con le iniziali imperfezioni, tantomeno sulla nucleazione e coalescenza delle micro separazioni presenti nella zona di processo.

Dato che sarebbe impossibile, dal punto di vista computazionale, modellare completamente la micro struttura ed ogni micro separazione del materiale, l'approccio più comune consiste nel raggruppare gli effetti di tutte le micro separazioni nella zona di processo posta nell'apice di una singola fessura dominante. Tale modello dovrebbe preferibilmente tener conto di due importanti proprietà del processo di frattura. Innanzitutto, la rigidità e la resistenza del materiale devono essere ridotte in modo realistico e in secondo luogo, questa riduzione di rigidità e resistenza deve essere accompagnata dalla corretta riduzione dell'energia interna nel materiale.

Nei primi modelli meccanici, la propagazione di una singola fessura all'interno di un solido elastico è stata simulata utilizzando il principio di minimo dell'energia. Griffith [19] suppone che una fessura, potendo essere modellata come una discontinuità all'interno di un campo di spostamenti sufficientemente regolare, può propagarsi fintanto che l'energia superficiale nella struttura eguaglia l'energia necessaria per la micro separazione nella zona di processo. L'utilizzo di un criterio energetico, piuttosto che il più intuitivo concetto di massima tensione di trazione, risiede nel fatto che in modello elastico lineare le sollecitazioni nell'intorno di un intaglio sono infinite; risulta quindi impossibile studiare il fenomeno in maniera adeguata.

la teoria di Griffith fu estesa da Irwin [23] ai materiali duttili includendo una zona plastica (Figura 2.3) all'apice della fessura. Mano a mano che i carichi applicati crescono, la zona plastica aumenta fino a quando la fessura cresce e si raggiunge una labilità. Fu anche introdotto il concetto di *fattore di intensificazione degli sforzi* e messo in relazione al tasso di rilascio di energia al fine di caratterizzare lo stato tensionale nell'intorno dell'apice della frattura





**Figura 2.3:** Zona plastica nell'intorno dell'apice della frattura nei materiali duttili.

La generalizzazione del lavoro di Griffith svolta da Rice [35] per applicazioni con solidi elasto-plastici e l'uso combinato del metodo degli elementi finiti portarono alla nascita della meccanica della frattura lineare elastica (LEFM), in grado di simulare la crescita e l'evoluzione della frattura. I limiti principali del metodo sono:

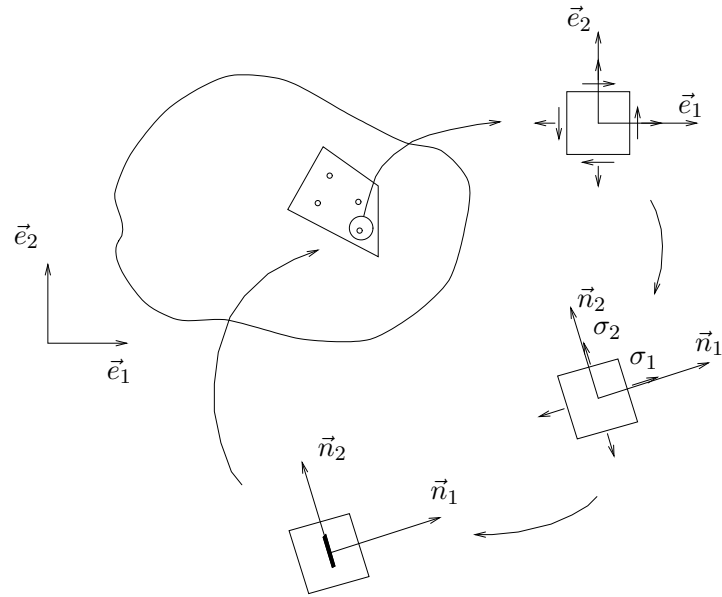
- dato che i fenomeni inerenti alla zona di processo vengono concentrati in un punto all'apice della frattura, questo metodo fornisce risultati realistici qualora effettivamente la zona di processo risulti relativamente piccola rispetto alle dimensioni del corpo;
- il metodo non consente lo sviluppo di fratture su materiali integri aventi un difetto interno dominante;
- il percorso della frattura è vincolato dalla mesh, e quindi dai lati degli elementi finiti.

In merito a questo ultimo punto, Ingraffea et al. [7] introdussero il concetto di remeshing, utilizzando i risultati del calcolo FEM come input per determinare il percorso della frattura. Questo significa che quando la frattura avanza, gli elementi in prossimità di essa hanno bisogno di un remeshing. Questa procedura limita l'applicabilità del metodo LEFM.

Con il grande successo degli elementi finiti furono sviluppate nuove tecniche per simulare la propagazione di una frattura, anche attraverso teorie non lineari. Come già affermato al 2.1, furono sviluppate due approcci fondamentali: da un lato quello degli *smeared crack models* o approcci continui, e dall'altro quella dei *discrete cohesive crack models* o approcci discontinui.

### 2.3.1 Approccio continuo

Questo tipo di approccio nasce dall'idea che i fenomeni di separazione micro meccanici in prossimità di un punto di integrazione all'interno di un elemento finito possono essere tradotti come un decadimento di rigidezza e resistenza del materiale in quel punto. Inizialmente, le proprietà del materiale in ogni punto di integrazione sono assunte lineari ed elastiche. Quando la tensione ideale in un punto di integrazione soddisfa un determinato criterio di rottura, la relazione elastica lineare tensioni-deformazioni viene cambiata. Il



**Figura 2.4:** Approccio continuo allo studio del fenomeno.

collasso della struttura viene raggiunto quando la rigidezza nel materiale è ridotta a zero. Anche se da un punto di vista globale il comportamento del modello si avvicina alla realtà, da un punto di vista strutturale non si riesce a cogliere una vera e propria fessura poichè vengono solo variate le proprietà del materiale ed il campo di spostamenti resta continuo.

Invece della creazione di una fessura nel materiale (una discontinuità nel campo di spostamenti), le deformazioni nella zona di processo sono localizzate e danno luogo ad elevati gradienti di deformazione in una piccola banda della zona di processo, che possono causare problemi numerici. Inoltre, la larghezza di questa banda deve essere almeno pari alla larghezza di un elemento specifico della mesh di elementi finiti. Chiaramente, questo impone gravi limitazioni all'applicabilità del metodo per materiali con piccole zone di processo. Per evitare questi problemi, sono stati sviluppati modelli in cui il grande gradiente di deformazione è incorporato nella relazione cinematica dell'elemento, come campo sforzo supplementare. Questi modelli sono detti *embedded discontinuity approaches*. Questi approcci migliorano la capacità deformativa dell'elemento ma spesso sono onerosi dal punto di vista computazionale poichè portano a matrici di rigidezza non simmetriche. Inoltre, non si ottiene una discontinuità vera e propria poichè le bande di fessura sono incorporate sull'elemento.

Oltre a questi limiti, gli approcci continui citati fino a qui soffrono di un'altra mancanza. Poichè il materiale è considerato omogeneo, il modello

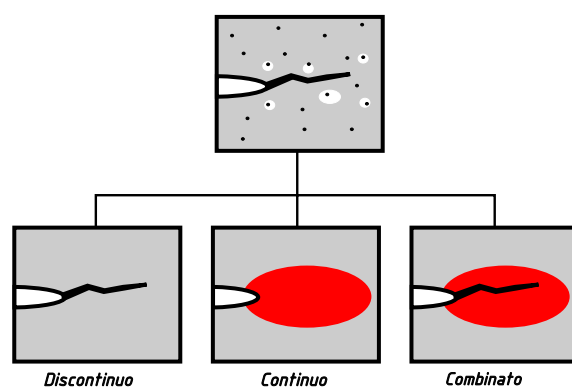


Figura 2.5: Tipologie di approcci per lo studio della frattura.

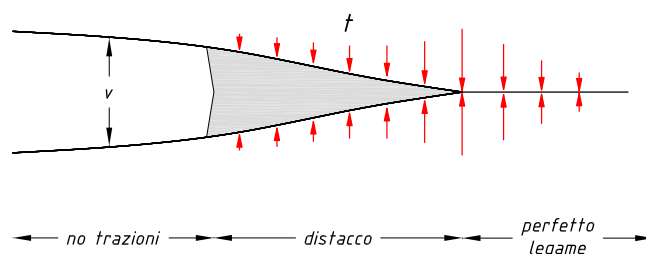


Figura 2.6: Rappresentazione della zona di processo nei modelli di zona coesiva.

non riesce a descrivere la larghezza della zona di processo e tutto il danno tende a concentrarsi in un singolo punto.

Il pregio di questi modelli è che l'implementazione numerica non richiede algoritmi adattivi e di proiezione della mesh, eccetto alcuni casi particolari che esulano dal contenuto di questo lavoro.

Sono stati messi a punto infine modelli in grado di incorporare nel continuo informazioni quali la micro struttura del materiale ma il loro utilizzo è piuttosto limitato e poco adatto alla pratica ingegneristica.

### 2.3.2 Approccio discontinuo

A questa categoria appartengono i modelli di frattura discreta. La geometria della frattura è modellata in maniera esplicita. Tra i vari modelli presenti in letteratura, uno dei più importanti è sicuramente il *modello di zona coesiva*, introdotto da Dugdale [14] e Brenblatt [4] e successivamente perfezionato da Hillerborg [20].

In questo modello, la zona di processo è concentrata in un singolo piano o in una singola linea davanti alla fessura esistente, come mostrato in Figura 2.6. Operando in questo modo, la singolarità tensionale sparisce. Matematicamente, un modello di zona coesiva è descritto da una legge costi-

tutiva coesiva che mette in relazione le trazioni coesive con l'apertura della fessura (Needleman [30]) e da una legge costitutiva classica del materiale al di fuori della zona di processo. Computazionalmente, l'approccio discontinuo comporta la modellazione della discontinuità che si viene a creare nel campo di spostamenti.

Tramite una procedura di *remeshing* automatico è possibile simulare l'avanzamento della frattura limitando la dipendenza del percorso dalla mesh. Un'alternativa al remeshing è data dal metodo X-FEM basato sul *Partition of Unity Method*.

### 2.3.3 Approcci combinati

I due approcci appena presentati presentano pregi e difetti. Gli approcci discontinui hanno il pregio di poter modellare la propagazione della frattura, ma hanno il difetto di non catturare il meccanismo che porta a rottura il materiale, caratteristica invece posseduta dagli approcci continui.

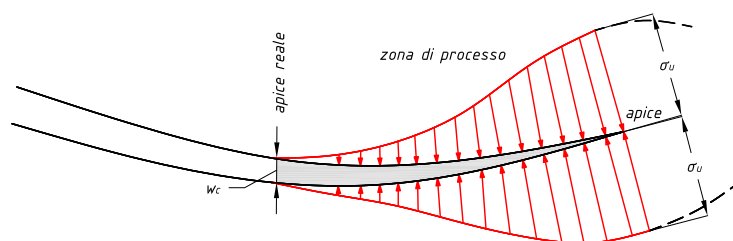
Nasce quindi l'idea di combinare i due approcci in un approccio *continuo-discontinuo* in cui la degradazione del materiale all'interno della zona di processo è modellata assumendo un comportamento di softening. Quando le proprietà di resistenza del materiale calano e ci sono le condizioni per l'apertura di una frattura, viene modellata la discontinuità che si viene a creare. La propagazione della frattura è governata dall'interazione tra questi due meccanismi.

In questo lavoro di tesi verrà utilizzato un approccio discontinuo, rimandando a [41] per ulteriori dettagli riguardanti gli approcci combinati

## 2.4 Il modello coesivo

I materiali disordinati e fragili come il calcestruzzo, le rocce, i ceramici ecc. contengono un gran numero di difetti e microfessure. Sottoponendoli ad elevati sforzi di trazione, si verifica un'interazione tra i processi di crescita delle microfessure che provoca la localizzazione della deformazione in una banda molto stretta, dove avviene la dissipazione di energia, mentre il materiale, al di fuori di questa banda, si comporta in modo elastico e lineare. Secondo i concetti della meccanica del continuo, si può affermare che in questa banda (chiamata *zona di processo*), si verifica una riduzione di capacità portante come conseguenza di un incremento di deformazione anelastica. Questo fenomeno è chiamato *incrudimento negativo*. La classica teoria della plasticità non risulta applicabile in quanto il postulato di stabilità di Drucker è violato. Le conseguenze di questa violazione, anche in assenza di effetti instabilizzanti di tipo geometrico, sono:

- possibilità di avere perdita di stabilità in controllo del carico (*snap-trough*);



**Figura 2.7:** Zona di processo con tensioni di richiusura secondo Hillerborg.

- possibilità di avere perdita di stabilità in controllo di spostamento (*snap-back*);
- possibilità di avere perdita di stabilità in controllo del carico (*snap-trough*);
- biforcazione del percorso di equilibrio;

Il modello coesivo è stato proposto da Barenblatt [4] e Dugdale [14]. Più recentemente è stato riproposto da Wnuk [46] con il nome di *Final Stretch Model*, e infine da Hillerborg [20] con la denominazione di *Fictitious Crack Model*.

Il modello coesivo permette quindi di descrivere il comportamento di materiali caratterizzati da una risposta di tipo strain-softening. I problemi di Modo I sono caratterizzati dalla conoscenza a priori della traiettoria della fessura, mentre nei problemi di Modo Misto questa rappresenta un'ulteriore incognita. In entrambi i casi, il Metodo degli Elementi Finiti rappresenta un elegante modo di affrontare problemi di propagazione di fessure nei materiali, tenendo conto di fenomeni quali l'effetto scala e la transizione duttile-fragile.

### 2.4.1 Il modello di frattura fittizio

L'ipotesi di base è che si formi, come prolungamento della fessura reale, una fessura fittizia, denominata *zona di processo*, nella quale il materiale, sebbene danneggiato, è ancora in grado di trasferire tensioni. Il punto che separa la zona libera da tensioni, denominata *fessura reale*, dalla zona di processo, è chiamato *apice della fessura reale*, mentre il punto che separa la zona di processo dal materiale integro è detto *apice della fessura fittizio*. Nella zona di processo avviene la dissipazione energetica; essa inizia a svilupparsi quando la resistenza a trazione del materiale viene superata in direzione perpendicolare alla direzione principale di trazione. In questo modo si eliminano le singolarità nello stato tensionale dovute ad una deformazione infinita.

Nella zona di processo inoltre, il materiale trasmette tensioni che sono funzioni decrescenti della discontinuità di spostamento mentre nella zona non danneggiata il materiale ha un comportamento elastico e lineare.

L'idea che sta alla base di questo approccio è quello di combinare un criterio energetico con un criterio basato sulle tensioni. La frattura viene descritta come un fenomeno di progressiva separazione lungo un'estesa porzione di materiale, detta appunto zona coesiva, nella quale agiscono trazioni all'interfaccia delle superfici di frattura.

### 2.4.2 Tipologie di modelli coesivi

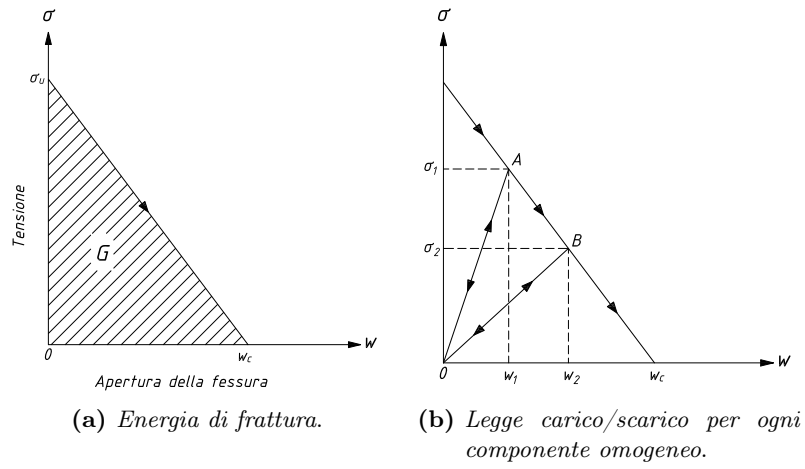
Esistono diverse tipologie di modelli coesivi. Le caratteristiche richieste generalmente per le leggi coesive, sono riassumibili nel seguito:

- La legge di separazione-trazione è indipendente da qualsiasi moto rigido del corpo;
- Il lavoro necessario per creare una nuova superficie ha un valore finito, e corrisponde al valore dell'energia di frattura, cioè, l'area sottesa alla curva della legge separazione-trazione, specifica del materiale;
- L'energia della modalità di frattura I di solito è diversa dall'energia della modalità II di frattura;
- Esiste una lunghezza tale per cui non si ha più nessuna capacità portante (lunghezza caratteristica della zona di processo);
- La trazione attraverso la superficie di frattura generalmente diminuisce fino ad annullarsi mentre la separazione aumenta sotto condizioni rammollimento, che determina una rigidità negativa;
- È ammessa l'esistenza di un potenziale per la legge coesiva, e quindi la dissipazione di energia associata a scarico/ricarico è indipendente dal potenziale.

Come si è detto la zona coesiva è zona del contorno del dominio materiale dove esistono delle discontinuità negli spostamenti. Una specifica legge coesiva è definita dalle equazioni che legano le trazioni con gli spostamenti tangenziali e normali. Esistono diversi modelli coesivi proposti in letteratura, Barenblatt [4]; Dugdale [14]; Needleman [30]; Rice and Wang [36]; Tvergaard and Hutchinson [45]; Xu and Needleman [47]; Camacho and Ortiz [10]; Geubelle and Baylor [18].

In base alle funzioni elementari utilizzate nei modelli coesivi si possono classificare come:

- Modelli multilineari;
- Modelli polinomiali;
- Modelli trigonometrici;
- Modelli esponenziali.



**Figura 2.8:** Legame costitutivo impiegato.

### 2.4.3 Modello coesivo impiegato

Le forze coesive che resistono all'apertura e allo scivolamento sulla superficie che viene a crearsi a seguito della crescita di una frattura sono assunte decrescenti linearmente con l'apertura, come si vede in Figura 2.8.





## Capitolo 3

# Equazioni del problema strutturale

### 3.1 Generalità

In questo capitolo vengono presentate brevemente le equazioni che reggono il problema strutturale nell'ipotesi di piccole deformazioni e di solido elastico, omogeneo ed isotropo.

### 3.2 Problema Elastico

Si introduce in questo capitolo il problema dell'equilibrio elastico. Esso consiste nelle sei equazioni differenziali di congruenza e in tre equazioni differenziali di equilibrio, in cui compaiono quindici incognite, ovvero tre spostamenti, sei deformazioni e sei tensioni. Si definiscono prima di tutto:

$$\boldsymbol{\sigma} = \{\sigma_x, \sigma_y, \sigma_z, \tau_{xy}, \tau_{yz}, \tau_{zx}\}^T \quad (3.1)$$

$$\boldsymbol{\varepsilon} = \{\varepsilon_x, \varepsilon_y, \varepsilon_z, \gamma_{xy}, \gamma_{yz}, \gamma_{zx}\}^T \quad (3.2)$$

che rispettivamente rappresentano il **tensore delle tensioni** e il **tensore delle deformazioni**. Essi rappresentano puntualmente le tensioni e deformazioni sui piani coordinati.

Si definisce inoltre il vettore spostamento:

$$\mathbf{u} = \{u, v, w\}^T \quad (3.3)$$

mentre i vettori velocità e accelerazione sono calcolati derivando gli spostamenti rispetto al tempo:

$$\begin{cases} \dot{\mathbf{u}} = \frac{\partial \mathbf{u}}{\partial t} \\ \ddot{\mathbf{u}} = \frac{\partial^2 \mathbf{u}}{\partial t^2} \end{cases} \quad (3.4)$$

Si possono ora definire le **equazioni indefinite di equilibrio** che definiscono gli sforzi che si sviluppano in punti del corpo assai prossimi tra loro:

$$\begin{cases} \frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + b_x = \rho \frac{\partial^2 u}{\partial t^2} \\ \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{yz}}{\partial z} + b_y = \rho \frac{\partial^2 u}{\partial t^2} \\ \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma_z}{\partial z} + b_z = \rho \frac{\partial^2 u}{\partial t^2} \end{cases} \quad (3.5)$$

Per ogni solido elastico soggetto a forze di volume  $\mathbf{b}$ , a trazioni  $\mathbf{t}$  sul contorno  $\Gamma_t$  e a spostamenti impressi  $\bar{\mathbf{u}}$  sul contorno  $\Gamma_u$  si ottengono le equazioni di congruenza e quelle di equilibrio.

Le equazioni di congruenza sono definite come:

$$\boldsymbol{\varepsilon} = \mathbf{L}\mathbf{u} \quad (3.6)$$

Le equazioni di equilibrio sono invece definite come:

$$\mathbf{L}^T \boldsymbol{\sigma} + \mathbf{b} = \rho \ddot{\mathbf{u}} \quad (3.7)$$

dove  $\mathbf{L}$  è l'operatore differenziale:

$$\mathbf{L} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \end{bmatrix} \quad (3.8)$$

Per ogni solido elastico è inoltre possibile definire una forma variazionale debole definita come **Principio dei Lavori Virtuali**:

$$\delta L_i - \delta L_e = \int_{\Omega} \delta \mathbf{u} \rho \ddot{\mathbf{u}} d\Omega + \int_{\Omega} \delta \boldsymbol{\varepsilon}^T \boldsymbol{\sigma} d\Omega - \int_{\Omega} \delta \mathbf{u}^T \mathbf{F} d\Omega - \int_{\Gamma} \delta \mathbf{u}^T \mathbf{t} d\Gamma = 0 \quad (3.9)$$

Il principio dei lavori virtuali è l'identità fondamentale nella meccanica dei corpi deformabili. Si osserva inoltre che il principio dei lavori virtuali è di validità generale, poiché risulta indipendente dalle proprietà meccaniche del materiale costituente il corpo continuo e indeformabile, quindi vale per qualsiasi legge costitutiva del materiale.

### 3.3 Condizioni al contorno

Le equazioni (3.6) e (3.7) insieme ad un'appropriata relazione costitutiva definiscono il comportamento del mezzo in condizioni sia statiche che dinamiche. Le incognite in questo sistema di equazioni sono gli spostamenti del solido  $\mathbf{u}$ .

Le condizioni al contorno imposte su queste variabili completano il problema:

- Per la parte di contorno caricato,  $\Gamma_t$ , sono specificate le trazioni totali  $\mathbf{t}$  (o in termini di tensioni totali  $\boldsymbol{\sigma} \cdot \mathbf{G}$  in cui  $\mathbf{G}$  è l'equivalente vettoriale di  $n_j$ , componente della normale al contorno);
- Per la parte rimanente,  $\Gamma_u$ , sono dati gli spostamenti impressi  $\bar{\mathbf{u}}$ .

Riassumendo, si può scrivere:

$$\begin{cases} \Gamma = \Gamma_f \cup \Gamma_u \\ \mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{G} = \bar{\mathbf{t}} \quad su \quad \Gamma = \Gamma_f \\ \mathbf{u} = \bar{\mathbf{u}} \quad su \quad \Gamma = \Gamma_u \end{cases} \quad (3.10)$$

### 3.4 Legame costitutivo

Per quanto detto precedentemente servono sei equazioni che leghino deformazioni e tensioni, ossia un legame costitutivo. Il legame costitutivo rappresenta il comportamento specifico di un materiale. Esso è dato dalle relazioni costitutive che legano sforzi e deformazioni.

In questo lavoro, al di fuori della zona di processo, viene assunto materiale a comportamento *iperelastico* o *elastico secondo Green*. Un materiale è detto iperelastico se esiste una funzione densità di energia di deformazione  $W(\boldsymbol{\varepsilon})$  da cui è possibile derivare la risposta tensionale:

$$\boldsymbol{\sigma} = \frac{\partial W}{\partial \boldsymbol{\varepsilon}} \quad \left( \sigma''_{ij} = \frac{\partial W}{\partial \varepsilon_{ij}} \right) \quad (3.11)$$

Si ha quindi:

$$\sigma_{ij} = C_{ijrs} \varepsilon_{rs} \quad (3.12)$$

in cui  $C_{ijrs}$  è la forma tensoriale di  $\mathbf{D}$ . I coefficienti elastici dipendono dall'energia di deformazione,  $W$  e possono essere espressi utilizzando le costanti di Lamé come:

$$C_{ijrs} = \frac{\partial^2 W}{\partial \varepsilon_{ij} \partial \varepsilon_{rs}} = \mu(\delta_{is} \delta_{jr} + \delta_{ir} \delta_{js}) + \lambda \delta_{ij} \delta_{rs} \delta_{ij} \quad (3.13)$$

La differenza sostanziale di un legame iperelastico rispetto ad legame elastico consiste sostanzialmente nell'esistenza di una funzione densità di energia deformazione. Infatti, relazioni del tipo:

$$\sigma_{ij} = \sigma_{ij}(\varepsilon_{kl}) \quad (3.14)$$

non implicano di per sè l'esistenza di un'energia di deformazione. L'esistenza di un potenziale della deformazione caratterizza quindi il legame in maniera

più stringente. Con maggior rigore terminologico quindi, si definisce *elastico* un legame rispettoso della 3.14, mentre l'esistenza di  $W$  dà luogo alla più ristretta categoria dei legami detti *iperelastici*. Nel caso di isotropia, esprimendo il legame in forma matriciale si ha, in termini di tensioni totali:

$$\begin{bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{bmatrix} = \begin{bmatrix} \lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\ \lambda & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\ \lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & 0 & \mu & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix} \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{bmatrix} \quad (3.15)$$

Queste relazioni mettono in luce una caratteristica importante del legame isotropo: se le deformazioni sono riferite alla terna principale, si ha  $\gamma_{ij} = 0$  e dalla 3.15 consegue che è anche  $\tau_{ij} = 0$ . In un materiale elastico-lineare ed isotropo quindi, le direzioni principali di sforzo e di deformazione coincidono. Ciò risulta verificato per ogni comportamento isotropo, anche non elastico.

## Capitolo 4

# Applicazione FEM al problema

### 4.1 Il metodo agli elementi finiti

Si vogliono risolvere i problemi retti dalle equazioni differenziali del capitolo precedente. Essi sono detti problemi di equilibrio o BVP (*Boundary Value Problems*). Se il problema non è stazionario si dovranno definire anche dei valori iniziali. Le equazioni possono essere scritte, in forma generica, come:

$$\begin{cases} \mathbf{A}(\mathbf{u}) = \mathbf{C}(\mathbf{u}) + \mathbf{p} = 0 & \text{in } \Omega \\ \mathbf{B}(\mathbf{u}) = \mathbf{M}(\mathbf{u}) + \mathbf{q} = 0 & \text{in } \Gamma \end{cases} \quad (4.1)$$

dove  $\Omega$  e  $\Gamma$  sono rispettivamente il dominio del corpo ed il suo contorno,  $\mathbf{C}(\mathbf{M})$  è un opportuno operatore differenziale e  $\mathbf{p}$  una funzione nota indipendente dalle incognite  $\mathbf{u}$  che rappresenta l'esatta soluzione del problema di equilibrio.

Il Metodo agli Elementi Finiti (*FEM*) dà la possibilità di risolvere sistemi matematici costruiti per descrivere un problema fisico come nel caso in esame. Esso si basa sulla rappresentazione della soluzione in un sottospazio di  $\mathbb{R}^N$ , con  $N$  numeri di nodi, in cui viene discretizzato il dominio. La scelta della base del sottospazio risiede, essenzialmente, nell'ordine delle funzioni interpolanti, che consentono di estendere la soluzione trovata, dai nodi, a tutto il dominio. Così, si passa dalle incognite originarie, che sono funzioni, alle incognite del metodo, che sono valori nodali, in quanto le funzioni di interpolazione sono note a priori.

Dato che il set di equazioni (4.1) deve essere pari a zero in ogni punto del dominio, segue che:

$$\int_{\Omega} \mathbf{v}^T \mathbf{A}(\mathbf{u}) d\Omega + \int_{\Gamma} \bar{\mathbf{v}}^T \mathbf{B}(\mathbf{u}) d\Gamma = 0 \quad (4.2)$$

in cui:

$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \end{bmatrix}; \quad (4.3)$$

è un set di funzioni arbitrarie il cui numero è uguale al numero di equazioni (o componenti di  $\mathbf{u}$ ) in gioco.

La potenza di questa espressione sta nel fatto che se essa è soddisfatta per le funzioni arbitrarie  $\mathbf{v}$  e  $\bar{\mathbf{v}}$  allora le equazioni (4.1) sono soddisfatte in tutti i punti del dominio. L'equazione (4.2) può avere una soluzione esatta solo per problemi molto semplici. Per trovare la soluzione è possibile utilizzare un approccio di tipo variazionale in cui si sostituisce alla soluzione esatta  $\mathbf{u}$  una soluzione approssimata  $\bar{\mathbf{u}}$ , in questo modo la funzione sarà continua in  $x$ ,  $y$  e  $z$ :

$$\mathbf{u} \approx \bar{\mathbf{u}} = \sum_i^r N_i \mathbf{a}_i = \mathbf{N} \mathbf{a} \quad (4.4)$$

È evidente che per problemi con geometria complessa non è pensabile definire funzioni approssimante in grado di coprire l'intero dominio, pertanto si procede a una suddivisione del dominio  $\Omega$  in sottodomini detti *elementi finiti*. Questo è equivalente a passare da un numero infinito di gradi di libertà (GDL) ad un numero finito. La discretizzazione del dominio porta quindi alla generazione di nodi e di elementi finiti. I nodi, nelle applicazioni del metodo FEM, sono entità fondamentali in quanto la soluzione dell'intera struttura viene riferita ad essi (cioè ai GDL ad esso associati): per estendere i valori del campo delle incognite su tutto il corpo vengono utilizzate delle funzioni, dette *funzioni di forma*  $\mathbf{N}$ , che con la desiderata approssimazione riportano i valori nodali  $\mathbf{a}$  in ogni sottodominio. I valori di  $\mathbf{a}$  possono essere intesi come spostamenti  $\bar{\mathbf{u}}$ o nel caso di altri problemi come per esempio mezzi porosi come pressioni (dell'acqua)  $\bar{\mathbf{p}}^w$ . La scelta delle funzioni di forma però non è banale.

Gli elementi finiti sono delle entità geometriche più o meno regolari caratterizzate da un determinato numero di nodi variabile a seconda del tipo di elemento. Tali nodi possono coincidere con i vertici degli elementi, ma in alcuni casi, possono essere disposti lungo i lati degli elementi stessi o addirittura all'interno. È evidente che all'aumentare del numero di nodi aumenta il grado del polinomio utilizzato per interpolazione dei dati ai nodi e, quindi, aumenta anche la qualità dell'approssimazione.

La scelta delle cosiddette funzioni di forma, che sono generalmente polinomiali (o comunque a comportamento noto) è un altro punto fondamentale che permette di ottenere una soluzione del modello FEM più o meno vicina

alla realtà che si vuole simulare. Al fine di rappresentare correttamente il valore ai nodi, le funzioni di forma devono assumere valori unitari nel nodo considerato e valori nulli sul resto dei nodi.

Quando la soluzione approssimata (4.4) viene introdotta nelle equazioni (4.1) si ottiene chiaramente, con ovvio significato dei termini, un residuo:

$$\mathbf{R} = \mathbf{R}_\Omega + \mathbf{R}_\Gamma = \mathbf{A}(\bar{\mathbf{u}}) + \mathbf{B}(\bar{\mathbf{u}}) \quad (4.5)$$

Per quanto detto l'equazione (4.2) permette, se esistono delle funzioni  $\mathbf{v}$  e  $\bar{\mathbf{v}}$  che soddisfano l'integrale, di considerare soddisfatte le equazioni (4.1). Il concetto di base è che la soluzione numerica ricercata deve annullare mediamente il residuo. Si considerano funzioni peso che annullino l'integrale mediamente sul dominio una volta moltiplicate per i residui. Si introducono pertanto delle *funzioni peso* approssimate di  $\mathbf{v}$  e  $\bar{\mathbf{v}}$ :

$$\mathbf{v} \approx \sum_b^n \mathbf{w}_b \delta \bar{\mathbf{u}}_b \quad (4.6a)$$

$$\bar{\mathbf{v}} \approx \sum_b^n \bar{\mathbf{w}}_b \delta \bar{\mathbf{u}}_b \quad (4.6b)$$

Riassumendo si sostituisce il sistema di equazioni differenziali detto *forma forte* con una cosiddetta *forma debole*, che è la forma integrale associata al sistema iniziale di partenza mediante l'utilizzo di tecniche variazionali. La forma debole viene costruita mediante il metodo dei residui pesati. Si impone sul dominio  $\Omega$  e sul contorno  $\Gamma$  l'ortogonalità tra i residui e le funzioni peso  $\mathbf{v}$  e  $\bar{\mathbf{v}}$ . Imporre l'ortogonalità significa moltiplicare le funzioni peso per i residui e integrarle sul dominio e sul contorno uguagliandole a zero. La giusta scelta delle funzioni peso è fondamentale per la risoluzione del problema. Dopo qualche passaggio si ottiene la forma integrale che annulla, mediamente sul dominio, il residuo:

$$\int_\Omega \mathbf{w}^T \mathbf{A}(\bar{\mathbf{u}}) d\Omega + \int_\Gamma \bar{\mathbf{w}}^T \mathbf{A}(\bar{\mathbf{u}}) d\Gamma = 0 \quad (4.7)$$

Il metodo dei residui pesati permette di ottenere soluzioni approssimate di equazioni differenziali lineari e non. Anche se il principio fisico per ottenere la forma integrale non esiste/non è noto, tramite la tecnica dei residui pesati è possibile comunque ricavarla. Nel caso in esame non è necessario conoscere il *funzionale* perché il principio fisico è il *Principio dei Lavori Virtuali*:

$$\delta L_i - \delta L_e = \int_\Omega \delta \mathbf{u} \rho \ddot{\mathbf{u}} d\Omega + \int_\Omega \delta \boldsymbol{\varepsilon}^T \boldsymbol{\sigma} d\Omega - \int_\Omega \delta \mathbf{u}^T \mathbf{F} d\Omega - \int_\Gamma \delta \mathbf{u}^T \mathbf{t} d\Gamma = 0 \quad (4.8)$$

Dove con  $\boldsymbol{\varepsilon}$  si rappresentano le deformazioni, con  $\boldsymbol{\sigma}$  si rappresentano le tensioni, con  $\mathbf{F}$  le forze di volume e con  $\mathbf{t}$  le forze sul contorno.

Avendo diviso il dominio in elementi finiti, gli integrali non sono più estesi a tutto il dominio ma al singolo elemento. Di fatto si dovranno poi sommare tutti i contributi dei singoli elementi. Questa peculiarità si nota quando si parla di matrice di rigidezza e del suo assemblaggio.

A seconda di come vengono scelte le funzioni peso si hanno vari sottometodi FEM. In questo caso si utilizza la forma di *Bubnov-Galerkin* che prevede l'utilizzo delle stesse funzioni di forma per interpolare le incognite e per interpolare le funzioni peso:

$$\mathbf{w} = \bar{\mathbf{w}} = \mathbf{N} \quad (4.9)$$

Questo metodo da l'approssimazione migliore e fornisce matrici simmetriche. Un grande vantaggio è che la forma integrale in realtà è algebrica e le derivate presenti in  $\mathbf{A}$  sono di un ordine inferiore e quindi si avrà semplicità nella risoluzione, si risolve un *sistema di equazioni algebriche*. È un sistema approssimato ma ha una soluzione molto vicina e più semplice rispetto al sistema di equazioni differenziali di partenza e, cosa molto importante, è continuo. Si possono calcolare le componenti di spostamento di un qualsiasi punto a partire da quelle dei nodi tramite l'equazione  $\mathbf{N}\mathbf{a}$ .

## 4.2 Applicazione al problema strutturale

Una volta definito brevemente il metodo risolutivo impiegato per la soluzione delle equazioni differenziali, si rende necessaria una rassegna sulle scelte effettuate.

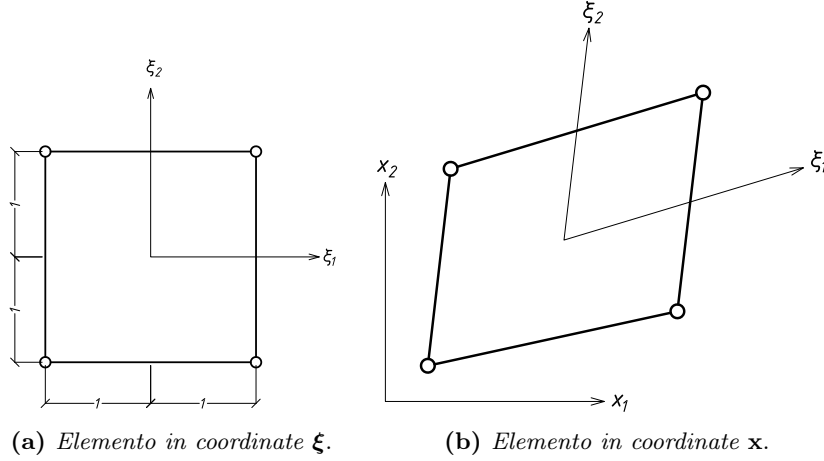
### 4.2.1 Approssimazione degli spostamenti

Da quanto stabilito nel paragrafo precedente, lo spostamento di un punto all'interno di un elemento è l'unica variabile indipendente, espresso in termini di spostamenti nodali tramite l'impiego di opportune funzioni di forma. La loro scelta è fatta in modo tale da assicurare la congruenza e garantire quindi la convergenza del metodo. L'approssimazione per gli spostamenti data dal metodo agli elementi finiti è data da:

$$\mathbf{u}(\mathbf{x}, t) \approx \hat{\mathbf{u}} = \sum_b N_b(\mathbf{x}) \bar{\mathbf{u}}_b(t) = \mathbf{N}(\mathbf{x}) \bar{\mathbf{u}}(t) \quad (4.10)$$

in cui  $N_b$  sono le funzioni di forma,  $\bar{\mathbf{u}}_b$  sono gli spostamenti nodali, dipendenti anche dal tempo, in via generale. La sommatoria è estesa al numero





**Figura 4.1:** Trasformazione isoparametrica per un elemento piano a 4 nodi.

dei nodi associati ad un elemento. Nel caso di elementi *isoparametrici*<sup>1</sup> è possibile scrivere:

$$\begin{aligned} \mathbf{u}(\boldsymbol{\xi}, t) &\approx \hat{\mathbf{u}}(\boldsymbol{\xi}, t) = \sum_b N_b(\boldsymbol{\xi}) \bar{\mathbf{u}}_b(t) = \mathbf{N}(\boldsymbol{\xi}) \bar{\mathbf{u}}(t) \\ \mathbf{x}(\boldsymbol{\xi}) &= \sum_b N_b(\boldsymbol{\xi}) \bar{\mathbf{x}}_b = \mathbf{N}(\boldsymbol{\xi}) \bar{\mathbf{x}} \end{aligned} \quad (4.11)$$

### 4.2.2 Derivate

La forma debole delle equazioni governanti l'equilibrio contiene al suo interno la derivata prima degli spostamenti. Considerando l'equazione (4.11), è possibile ricavare la derivata prima delle funzioni di forma rispetto a  $x_j$ :

$$\frac{\partial N_a}{\partial \xi_i} = \frac{\partial x_j}{\partial \xi_i} \frac{\partial N_a}{\partial x_j} \quad (4.12)$$

in forma matriciale si ha:

$$\frac{\partial N_a}{\partial \boldsymbol{\xi}} = \mathbf{J} \frac{\partial N_a}{\partial \mathbf{x}} \quad (4.13)$$

dove:

$$\frac{\partial N_a}{\partial \boldsymbol{\xi}} = \begin{bmatrix} \frac{\partial N_a}{\partial \xi_1} \\ \frac{\partial N_a}{\partial \xi_2} \\ \frac{\partial N_a}{\partial \xi_3} \end{bmatrix}; \quad \frac{\partial N_a}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial N_a}{\partial x} \\ \frac{\partial N_a}{\partial y} \\ \frac{\partial N_a}{\partial z} \end{bmatrix}; \quad \mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi_1} & \frac{\partial y}{\partial \xi_1} & \frac{\partial z}{\partial \xi_1} \\ \frac{\partial x}{\partial \xi_2} & \frac{\partial y}{\partial \xi_2} & \frac{\partial z}{\partial \xi_2} \\ \frac{\partial x}{\partial \xi_3} & \frac{\partial y}{\partial \xi_3} & \frac{\partial z}{\partial \xi_3} \end{bmatrix} \quad (4.14)$$

<sup>1</sup>se il numero di nodi in cui sono note le funzioni spostamento e geometria sono uguali; è il tipo di elemento più diffuso in quanto semplice ed efficiente.

in cui  $\mathbf{J}$  è lo Jacobiano della trasformazione tra  $x$  e  $\xi$ . Utilizzando queste relazioni, le derivate delle funzioni di forma sono date da:

$$\frac{\partial N_a}{\partial \mathbf{x}} = \mathbf{J}^{-1} \frac{\partial N_a}{\partial \xi} \quad (4.15)$$

Nel caso invece piano il tutto si riduce ad avere

### 4.2.3 Calcolo delle tensioni e delle deformazioni

Una volta definita l'approssimazione degli spostamenti con le funzioni di forma (equazione (4.10)), il campo deformativo può essere ottenuto differenziando il campo di spostamento dell'elemento:

$$\boldsymbol{\varepsilon} = \mathbf{L}\mathbf{N}\bar{\mathbf{u}} = \mathbf{B}\bar{\mathbf{u}} \quad (4.16)$$

con  $\mathbf{L}$  operatore differenziale definito come:

$$\mathbf{L} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & 0 \\ 0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial x} & 0 & \frac{\partial}{\partial z} \end{bmatrix} \quad (4.17)$$

e  $\mathbf{B}$  può essere ricavata utilizzando l'equazione (4.15):

$$\mathbf{B}^T = \begin{bmatrix} N_{b,x_1} & 0 & 0 & N_{b,x_2} & 0 & N_{b,x_3} \\ 0 & N_{b,x_2} & 0 & N_{b,x_1} & N_{b,x_3} & 0 \\ 0 & 0 & N_{b,x_3} & 0 & N_{b,x_2} & N_{b,x_1} \end{bmatrix} \quad (4.18)$$

Il campo di deformazioni così ottenuto, qualunque sia il campo di spostamenti, è sicuramente cinematicamente ammissibile, poiché sono rispettate le condizioni di compatibilità. Il problema diventa la ricerca di un campo di tensioni staticamente ammissibile, che equilibra le forze esterne applicate.

Per materiali elastici lineari il campo tensionale è legato a quello deformativo mediante la matrice del legame costitutivo  $\mathbf{D}$ , chiamata anche matrice di elasticità:

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon} = \mathbf{D}\mathbf{B}\bar{\mathbf{u}} \quad (4.19)$$

A seconda del legame tensioni-deformazioni è possibile modificare la relazione (4.19).

#### 4.2.4 Forma debole del problema strutturale

Sostituendo le relazioni ottenute fino qua nell'equazione di equilibrio scritta in forma debole (4.8) si ottiene:

$$\delta \bar{\Pi}_{eq} = \delta \bar{\mathbf{u}}^T \left[ \int_{\Omega_e} \mathbf{N}^T \rho \mathbf{N} d\Omega_e \ddot{\mathbf{u}} + \int_{\Omega_e} \mathbf{B}^T \boldsymbol{\sigma} d\Omega - \int_{\Omega_e} \mathbf{N}^T \mathbf{F} d\Omega - \int_{\Gamma_e} \mathbf{N}^T \mathbf{t} d\Gamma \right] \quad (4.20)$$

in cui tutti gli integrali sono estesi al dominio (o contorno) dell'elemento finito. Eseguendo la sommatoria estesa a tutti gli elementi e omettendo il termine  $\delta \bar{\mathbf{u}}$  poichè arbitrario, si ottiene:

$$\mathbf{M} \ddot{\mathbf{u}} + \mathbf{P}(\boldsymbol{\sigma}) = \mathbf{f} \quad (4.21)$$

in cui:

$$\mathbf{M} = \sum_e \mathbf{M}^{(e)}; \quad \mathbf{P} = \sum_e \mathbf{P}^{(e)}; \quad \mathbf{f} = \sum_e \mathbf{f}^{(e)} \quad (4.22)$$

Il termine  $\mathbf{P}$  è inteso come divergenza del tensore degli sforzi e, nel caso di materiale elastico e lineare, è possibile esprimerlo come:

$$\mathbf{P} = \mathbf{B}^T \mathbf{D} \mathbf{B} = \mathbf{K} \bar{\mathbf{u}} \quad (4.23)$$

in cui  $\mathbf{K}$  è la matrice di rigidezza dell'elemento.

Si ottiene quindi, con ovvio significato dei termini:

$$\mathbf{M} \ddot{\mathbf{u}} + \mathbf{K} \bar{\mathbf{u}} = \mathbf{f} \quad (4.24)$$

Una volta scritto il sistema, assemblata la matrice di rigidezza e di massa di tutti gli elementi finiti ed il termine noto si risolve il sistema, senza entrare nei dettagli, per trovare il vettore incognito  $\bar{\mathbf{u}}$ . Quando visto vale in particolare per risolvere il problema in campo dinamico lineare. Il metodo FEM può essere esteso a molti campi della fisica, ed a problemi più complessi come il problema accoppiato e la dinamica non lineare.

### 4.3 Condizioni al contorno

Dopo aver assemblato la matrice di rigidezza degli elementi si può scrivere l'equazione:

$$\mathbf{K} \bar{\mathbf{u}} = \mathbf{f} \quad (4.25)$$

Questa equazione non è direttamente risolvibile nelle incognite  $\mathbf{u}$  poichè la matrice  $\mathbf{K}$  risulta singolare e quindi non invertibile. Il significato fisico, nell'ambito della meccanica dei solidi, è che la struttura così caricata è libera di muoversi di moto rigido, quindi non è vincolata. Si devono quindi specificare un numero di gradi di vincolo, che dipendono dalla fisica del problema. L'imposizione di una determinata condizione al contorno definisce una soluzione particolare del problema. Tali condizioni devono essere ovviamente rispettate dalla soluzione. Esistono due tipologie di condizioni al contorno:

- Condizioni al contorno *geometriche* o *essenziali* o *condizioni di Dirichlet*. Vengono dette tali in quanto vengono imposte in maniera esplicita nella determinazione dello spazio funzionale in cui il problema è posto;
- Condizioni al contorno *libere* o *naturali* o *condizioni di Neumann*. Sono dette naturali, in quanto vengono soddisfatte implicitamente dalla soluzione del problema.

Utilizzando un principio di tipo variazionale per derivare le equazioni del problema, si devono specificare solo le condizioni al contorno *essenziali* mentre le condizioni al contorno *naturali* saranno soddisfatte implicitamente nella procedura risolutiva.

#### 4.3.1 Applicazione condizioni al contorno

Vengono riportate le definizioni e l'approccio definito in [9]. L'applicazione delle condizioni al contorno del modello discretizzato del problema avviene modificando in modo opportuno le equazioni algebriche che si ottengono al termine del processo di assemblaggio delle matrici della struttura. Tale sistema modificato è quello che, una volta risolto, fornirà la soluzione del problema numerico.

I metodi con i quali tenere conto delle condizioni al contorno sono vari ma in questo contesto ne vengono elencati solo tre, i più utilizzati:

- *Metodo di eliminazione righe-colonne*: Il procedimento si traduce semplicemente nella eliminazione della riga e della colonna corrispondente ai gradi di libertà vincolati, che pertanto non saranno più contenuti nella matrice di rigidezza finale.
- *Metodo Master-Slave*: I gradi di libertà coinvolti in ogni condizione di vincolo sono separati in g.d.l. master e g.d.l. slave. Viene scritta un'equazione che lega il nodo master a quello slave.
- *Metodo Penalty*: Questo metodo impone le condizioni di vincolo introducendo degli elementi strutturali fittizi (penalty elements) le cui caratteristiche elastiche dipendono da un parametro numerico chiamato peso; le condizioni di vincolo imposte con questo metodo risultano approssimate e solo al tendere del parametro peso ad un valore infinito vengono esattamente rispettate.
- *Metodo dei Moltiplicatori di Lagrange* Per ogni condizione di vincolo una nuova incognita viene aggiunta al modello. Fisicamente tali nuove incognite rappresentano le forze di vincolo che, applicate al sistema non vincolato, lo costringono al rispetto delle condizioni al contorno assegnate. Il soddisfacimento delle condizioni di vincolo si ottiene,

*nell'ambito della scrittura variazionale del problema*, aggiungendo un termine al funzionale il cui contributo risulta esattamente nullo solo quando le condizioni di vincolo sono esattamente rispettate.

### 4.3.2 Metodo dei moltiplicatori di Lagrange

Solitamente nei problemi strutturali semplici il metodo di più immediato utilizzo è il metodo di eliminazione righe-colonne. La sua applicabilità a casi generici e con numerose condizioni di vincolo che seguono un percorso (come nel caso in esame sono utilizzate rampe di spostamento) risulta proibitiva e non si presta quindi alla implementazione in codici ad elementi finiti di uso generale. In questi casi vengono adottati diversi metodi tra cui il *Metodo dei Moltiplicatori di Lagrange* molto usato soprattutto nell'ambito della programmazione orientata agli oggetti. Il metodo dei Moltiplicatori di Lagrange permette di applicare qualsiasi tipo di condizione al contorno di Dirichlet. A causa della variazione continua del dominio come conseguenza della propagazione della frattura, anche il contorno e le relative condizioni meccaniche cambiano. Come già detto solo le condizioni al contorno essenziali devono essere imposte. A questo fine, la parte di contorno dove le condizioni al contorno imposte non cambiano durante l'analisi viene dapprima rilevata e le condizioni relative sono imposte. Per il contorno rimanente, aggiornato con le forze di richiusura, ad ogni passo temporale, le condizioni al contorno sono imposte per mezzo del metodo dei moltiplicatori di Lagrange. Questo richiede una modifica del sistema che risolve il problema con un insieme di equazioni che esprimono le condizioni di vincolo e cambiano *dinamicamente* con il procedere della soluzione. Ciò non solo permette di rappresentare le fessure in movimento, ma anche di applicare carichi variabili esterni (dipendenti da una *time-table*).

Si vuole applicare la seguente condizione al contorno di Dirichlet:

$$\mathbf{x} = \tilde{\mathbf{x}} \quad \text{su } \Gamma_x \quad (4.26)$$

che viene convertita nella seguente forma con i moltiplicatori di Lagrange:

$$L_e = \int_{\Omega} \boldsymbol{\lambda}^T (\mathbf{x} - \tilde{\mathbf{x}}) d\Omega \quad (4.27)$$

Si definisce ora il relativo contributo virtuale dei moltiplicatori di Lagrange che andrà aggiunto al lavoro virtuale dell'equazione eqrefeq:plv. Il *Principio dei Lavori Virtuali* offre il vantaggio di chiamare in causa solamente le forze attive ma richiede una condizione: gli spostamenti virtuali imposti, anche se possono essere qualunque (virtuali vuol dire *arbitrari*), devono essere congruenti cioè compatibili con i vincoli e non devono ledere la continuità della struttura. Il contributo, di lavoro virtuale interno è dato sia dalle variazioni virtuali di spostamenti sia dalla variazione virtuale dei moltiplicatori (che sono incognite aggiuntive):

$$\delta L_e = \int_{\Omega} \delta \mathbf{x}^T \boldsymbol{\lambda} d\Omega + \int_{\Omega} \delta \boldsymbol{\lambda}^T \mathbf{x} d\Omega \quad (4.28)$$

La discretizzazione della formulazione debole porta alla formulazione

$$\delta L_e = \int_{\Omega} \mathbf{C}^T \boldsymbol{\lambda} d\Omega + \int_{\Omega} \mathbf{C} \mathbf{x} d\Omega \quad (4.29)$$

In cui  $\mathbf{C}$  è una matrice di coefficienti che può definita in diversi modi. Si può fare riferimento ad una formulazione a constraint (o link) in cui i coefficienti sono derivati da equazioni di constraint (*vincoli*). L'equazione di constraint può essere descritta come:

$$\mathbf{X}_{(n_s, k_s)} = \boldsymbol{\lambda}_1(n_1) \mathbf{X}_{(n_1, k_1)} + \boldsymbol{\lambda}_2(n_2) \mathbf{X}_{(n_2, k_2)} + \dots + \boldsymbol{\lambda}_i(n_i) \mathbf{X}_{(n_i, k_i)} \quad (4.30)$$

- $\mathbf{X}_{(n_i, k_i)}$  è lo spostamento del D.o.F. k-esimo del nodo specificato da  $n_i$ ;
- $\boldsymbol{\lambda}_i(n_i)$  è il moltiplicatore;
- $(n_s, k_s)$  si riferiscono al nodo slave.

I contributi da inserire nella sottomatrice  $\mathbf{C}$  (e simmetricamente in  $\mathbf{C}^T$ ) derivano da semplici equazioni. Ad esempio se si vuole impostare uno spostamento del nodo pari ad 1 in direzione  $x$  pari a un valore assegnato (ad esempio 5) l'equazione si pone semplicemente come:

$$1 \cdot u_x(1) = 5 \quad (4.31)$$

$$\Phi = 1 \cdot u_x(1) - 5 \quad (4.32)$$

$$\mathbf{C} = \frac{\partial \Phi}{\partial \mathbf{u}} \quad (4.33)$$

da qui il valore 1 nella matrice  $\mathbf{C}$  (e simmetrica) e il valore 5 nel termine noto.

Un approccio più sofisticato e forse più elegante è quello di implementare i constraint tramite interpolazione con funzioni di forma. Anche qui esistono diversi sotto-metodi tra cui quello di usare delle funzioni di forma (che coincidono con quelle utilizzare per approssimare  $\mathbf{u}$ ) per approssimare le condizioni al contorno. Queste funzioni di forma agiscono solo sul contorno. L'espressione così per  $\mathbf{C}$  diventa:

$$\mathbf{C} = \int_{\Omega} (\mathbf{N}^\lambda)^T (\mathbf{N}^\lambda) d\Omega = \int_{\Omega} (\mathbf{N}^u)^T (\mathbf{N}^u) d\Omega \quad (4.34)$$

Nel caso in esame si è utilizzato il primo approccio mentre il secondo può essere utilizzato in situazioni più complesse (contatto).

Che si usi il primo od il secondo approccio, la soluzione del sistema con i moltiplicatori di Lagrange porta a una formulazione in cui le matrici sono

composte da matrici piccole e dense. Questa formulazione è nota come *block-matrix formulation*. Le dimensioni di un singolo blocco sono dettate dalle dimensioni del corpo e dai D.o.F.<sup>2</sup>:

$$[\mathbf{A}] = \begin{bmatrix} \mathbf{K} & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{u}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{(u)} \\ \mathbf{f}^{(\lambda)} \end{bmatrix} \quad (4.35)$$

Con:

- Con  $\mathbf{K}$  matrice di rigidità del sistema;
- Con  $\mathbf{C}$  generica matrice  $m \times n$  che contiene i coefficienti di vincolo, in particolare è la matrice Jacobiana dei constraints. La matrice Jacobiana è ottenuta differenziando le equazioni di constraint rispetto alle coordinate;
- Con  $\bar{\mathbf{u}}$  vettore degli spostamenti incogniti;
- Con  $\boldsymbol{\lambda}$  vettore dei moltiplicatori incogniti;
- Con  $\mathbf{f}$  vettore dei termini noti;
- Con  $\mathbf{f}^{(\lambda)}$  vettore dei vincoli imposti mediante formulazione lagrangiana (detti anche *constraints*).

La matrice di rigidità  $\mathbf{K}$  risulta così *delimitata* da  $\mathbf{C}$  e  $\mathbf{C}^T$ . La risoluzione del sistema fornisce  $\mathbf{u}$  e  $\boldsymbol{\lambda}$ . L'ultimo termine può essere interpretato nel modo seguente: un vincolo rimosso può essere rimpiazzato da un sistema di forze caratterizzato da  $\boldsymbol{\lambda}$  moltiplicato per i coefficienti di vincolo. Più precisamente, le forze vincolari sono  $-\mathbf{C}^T \boldsymbol{\lambda}$ .

Nonostante la matrice  $\mathbf{A}$  sia ancora simmetrica non è più definita positiva ma semidefinita positiva e perciò non può essere risolta con i metodi standard di risoluzione. Le equazioni di *constraints* aggiungono delle entrate nulle sulla diagonale. In altre situazioni inoltre la matrice  $\mathbf{A}$  può anche perdere la sua simmetria.

A discapito dei vantaggi il principale svantaggio della formulazione dei moltiplicatori di Lagrange è che il sistema da risolvere aumenta perché anche i moltiplicatori stessi sono incognite del problema.

### Metodo dei Moltiplicatori di Lagrange

Con riferimento a 5.1 la soluzione del sistema discretizzato con i moltiplicatori di Lagrange si modifica in (il punto rappresenta la derivata temporale):

$$\begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\bar{\mathbf{u}}} \\ \ddot{\boldsymbol{\lambda}} \end{bmatrix} + \begin{bmatrix} \mathbf{K} & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{u}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{(u)} \\ \mathbf{f}^{(\lambda)} \end{bmatrix} \quad (4.36)$$

<sup>2</sup>Questa formulazione verrà impiegata in modo ricorrente nel corso di questo lavoro.

Con:

$$\mathbf{f}^{(\lambda)} = \int_{\Omega} \mathbf{N}^{\lambda} \bar{\mathbf{t}} \, d\Omega \quad (4.37)$$

- Con  $\mathbf{K}$  matrice di rigidezza del sistema;
- Con  $\mathbf{C}$  matrice dei moltiplicatori di Lagrange;
- Con  $\bar{\mathbf{u}}$  vettore degli spostamenti incogniti;
- Con  $\boldsymbol{\lambda}$  vettore dei moltiplicatori di Lagrange;
- Con  $\mathbf{f}^{(u)}$  vettore dei termini noti;
- Con  $\mathbf{f}^{(\lambda)}$  vettore delle reazioni dei vincoli imposti mediante formulazione lagrangiana (detti anche *constraints*).

#### 4.4 Problema non lineare al contorno

In natura, a rigore, non esistono comportamenti lineari. Ingegneristicamente parlando però si possono utilizzare comportamenti lineari o approssimati come tali laddove il comportamento non lineare è trascurabile. Le fonti di non linearità strutturali possono essere:

- Non linearità geometriche (Geometrical Nonlinearities): l'effetto di grandi spostamenti su la configurazione geometrica complessiva della struttura.
- Non linearità per Materiale (Material Nonlinearities): comportamento materiale è non lineare;
- Non linearità di contorno (Boundary Nonlinearities), cioè quando le forze applicate e gli spostamenti ai contorni dipendono dalle deformazioni del sistema.

##### 4.4.1 Metodo Full Newton

La versatilità del metodo degli elementi finiti in ambito lineare suggerisce il desiderio di estendere la formulazione anche a problemi non lineari. Alla formulazione del problema, si premette l'illustrazione del procedimento da cui traggono origine le tecniche più comunemente utilizzate per la sua soluzione. Si consideri il seguente sistema di equazioni non lineare:

$$\mathbf{f} = \mathbf{P}(\mathbf{u}) \quad (4.38)$$

Esso può equivalenemente scriversi:

$$\boldsymbol{\Psi}(\mathbf{u}) = \mathbf{P}(\mathbf{u}) - \mathbf{f} \quad (4.39)$$



in cui  $\mathbf{f}$  e  $\mathbf{u}$  indicano i vettori delle forze e degli spostamenti nodali (nel caso lineare è  $\mathbf{P} = \mathbf{K}\mathbf{u}$ ). Si supponga che le componenti di  $\mathbf{P}$  siano funzioni differenziabili di  $\mathbf{u}$  e si immagini di conoscere una soluzione approssimata  $\mathbf{u}_{n+1}^i$  dalla (4.39) in cui il pedice  $n + 1$  rappresenta il tempo corrente e l'apice  $i$  rappresenta il numero dell'iterazione. Sviluppando questa relazione in serie di Taylor nell'intorno di  $\mathbf{u}_{n+1}^i$  e troncando al termine del primo ordine si ottiene:

$$\Psi(\mathbf{u}_{n+1}^{i+1}) \approx \Psi(\mathbf{u}_{n+1}^i) + \left( \frac{\partial \Psi}{\partial \mathbf{u}} \right)_{n+1}^i (\mathbf{u}_{n+1}^{i+1} - \mathbf{u}_{n+1}^i) = \mathbf{0} \quad (4.40)$$

Questo è un sistema lineare che consente il calcolo di un valore aggiornato  $\mathbf{u}_{n+1}^{i+1}$  per gli spostamenti nodali e si configura come elemento di un processo iterativo per la soluzione della (4.39). Si introduce la matrice di rigidità tangente:

$$\mathbf{K}_T(\mathbf{u}) = \left( \frac{\partial \Psi}{\partial \mathbf{u}} \right) = \left( \frac{\partial \mathbf{P}}{\partial \mathbf{u}} \right) \quad (4.41)$$

Si ponga, per semplicità di scrittura:

$$\mathbf{P}_{n+1}^i = \mathbf{P}(\mathbf{u}_{n+1}^i) \quad \Phi_{n+1}^i = \Phi(\mathbf{u}_{n+1}^i) \quad \mathbf{K}_T = \mathbf{K}_T(\mathbf{u}_{n+1}^i) \quad (4.42)$$

$$\Delta \mathbf{u}_{n+1}^i = \mathbf{u}_{n+1}^{i+1} - \mathbf{u}_{n+1}^i \quad (4.43)$$

La soluzione del sistema (4.40) allora si scrive, simbolicamente:

$$\Delta \mathbf{u}_{n+1}^i = -(\mathbf{K}_T)^{-1} \Psi_{n+1}^i \quad (4.44)$$

Il processo viene iterato aggiornando ogni volta le quantità (4.41) e (4.42) e termina quando il vettore:

$$\Phi_{n+1}^i = \Psi_{n+1}^i - \mathbf{f}_{n+1} \quad (4.45)$$

diviene sufficientemente piccolo, quando cioè risulta  $\|\Psi_{n+1}^1\| \leq \alpha$ , dove  $\alpha$  è un'assegnata tolleranza. Una serie di approssimazioni successive porta a calcolare la soluzione desiderata al passo  $n + 1$ :

$$\mathbf{u}_{n+1}^{i+1} = \mathbf{u}_n + \delta \Delta \mathbf{u}_{n+1}^i \quad (4.46)$$

in cui:

$$\delta \Delta \mathbf{u}_{n+1}^i = \sum_{k=1}^i \Delta \mathbf{u}_{n+1}^k \quad (4.47)$$

Il procedimento descritto è noto come *metodo di Newton-Raphson*. La sua applicazione può rivelarsi onerosa in quanto ogni iterazione richiede di ridefinire la matrice tangente e quindi risolvere un diverso sistema di equazioni lineari. Ciò potrebbe essere evitato, nei domini in cui non varia il numero

**Figura 4.2:** Interpretazione grafica del metodo di Newton-Raphson.

dei nodi/di elementi, ricorrendo ad una versione modificata del metodo, che consente nell'assumere, indipendentemente dall'iterazione corrente:

$$\mathbf{K}_T^i = \mathbf{K}_T^0 \quad (4.48)$$

Tuttavia, nel caso in esame non è possibile ricorrere allo schema modificato in quanto il dominio è in continua evoluzione (anche nel medesimo step temporale) e quindi la matrice tangente deve essere inevitabilmente ricalcolata ad ogni iterazione.

La convergenza del metodo non è garantita in ogni circostanza, ma normalmente si verifica se il vettore di partenza non è si discosta molto dalla soluzione. Ciò può consigliare di suddividere la storia di carico in intervalli di ampiezza non eccessiva, di modo che la prima iterazione rappresenti già una ragionevole approssimazione del valore finale.

### Full Newton

Per l'applicazione del algoritmo iterativo è necessario calcolare lo Jacobiano presentato in 4.4.1. Si definisce allora il problema nella seguente forma (le forze non lineari si includono, con i relativi contributi, nella matrice di rigidezza  $\mathbf{K}$  e nel termine noto  $\mathbf{f}^{(u)}$ ):

$$\begin{cases} \mathbf{R}_1(\bar{\mathbf{u}}, \boldsymbol{\lambda}) = \mathbf{M}\ddot{\bar{\mathbf{u}}} + \mathbf{K}\bar{\mathbf{u}} + \mathbf{C}^T\boldsymbol{\lambda} - \mathbf{f}^{(u)} = 0 \\ \mathbf{R}_2(\bar{\mathbf{u}}, \boldsymbol{\lambda}) = \mathbf{C}\bar{\mathbf{u}} - \mathbf{f}^{(\lambda)} = 0 \end{cases}$$

- $\mathbf{R}_1$  esprime l'equilibrio. È il vettore del residuo, esprime le forze non equilibrate (*off-balance forces*). È lineare in  $\boldsymbol{\lambda}$ ;
- $\mathbf{R}_2$  esprime il rispetto dei vincoli. È lineare in  $\bar{\mathbf{u}}$ ;
- $\mathbf{M}\ddot{\bar{\mathbf{u}}}$  forze d'inerzia;
- $\mathbf{K}\bar{\mathbf{u}}$  sono le forze interne;
- $\mathbf{C}^T\boldsymbol{\lambda}$  sono le reazioni dovute ai *constraints*, termine che viene aggiunto nel termine delle forze interne. Quindi può contenere sia le reazioni vincolari che le reazioni di *link* (esempio: *rigid link*);

Si riporta di seguito, a titolo esemplificativo, la procedura per il caso statico. Per quanti riguarda i problemi dipendenti dal tempo si rimanda al capitolo dedicato.

$$\begin{cases} \frac{\partial}{\partial \bar{\mathbf{u}}} \mathbf{R}_1 = \mathbf{K} \\ \frac{\partial}{\partial \lambda} \mathbf{R}_1 = \mathbf{C}^T \\ \frac{\partial}{\partial \bar{\mathbf{u}}} \mathbf{R}_2 = \mathbf{C} \\ \frac{\partial}{\partial \lambda} \mathbf{R}_2 = 0 \end{cases}$$

Espresso in forma matriciale:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial}{\partial \bar{\mathbf{u}}} \mathbf{R}_1 & \frac{\partial}{\partial \lambda} \mathbf{R}_1 \\ \frac{\partial}{\partial \bar{\mathbf{u}}} \mathbf{R}_2 & \frac{\partial}{\partial \lambda} \mathbf{R}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{K}(u) & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \quad (4.49)$$

Prima di procedere ulteriormente è necessaria una considerazione. Lo Jacobiano  $\mathbf{J}$  include la matrice singolare  $\mathbf{K}$  per cui bisogna applicare le condizioni al contorno per eliminare moti rigidi. Per applicare le condizioni al contorno di tipo cinematico (*constraint*) un approccio comune è di esprimere gli  $m$  D.o.F. dipendenti (*slave*) in funzione di altri D.o.F. indipendenti (*master*). Il vettore degli spostamenti è quindi suddiviso in tre sotto-vettori: *slave* D.o.F., *master* D.o.F. e D.o.F. *non vincolati* (o non coinvolti dai constraints). Eliminando i *slave* D.o.F. dal vettore delle incognite (spostamenti) la matrice  $\mathbf{K}$  è non-singolare. Un approccio molto simile è impiegato per la tecnica dei moltiplicatori di Lagrange.

La chiave dell'sistema (4.49) è che  $\mathbf{K}$  è l'unico blocco che varia nello Jacobiano. Il resto dei blocchi è costante. Applicando la (4.49) si ottiene:

$$\begin{bmatrix} \mathbf{K}(\mathbf{u}_{t+1}^{i+1}) & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{u}_{t+1}^{i+1} \\ \Delta \lambda_{t+1}^{i+1} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1(\mathbf{u}_{t+1}^{i+1}, \Delta \lambda_{t+1}^{i+1}) \\ \mathbf{f}^{(\lambda)}_{t+1}^{i+1} \end{bmatrix} \quad (4.50)$$

dove:

$$\begin{bmatrix} \mathbf{R}_1(\mathbf{u}_{t+1}^{i+1}, \Delta \lambda_{t+1}^{i+1}) \\ \mathbf{f}^{(\lambda)}_{t+1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{(u)} - \mathbf{K}(\Delta \mathbf{u}_{t+1}^i) \mathbf{u}_{t+1}^i - \mathbf{C}^T \lambda_{t+1}^i \\ \mathbf{f}^{(\lambda)}_{t+1} \end{bmatrix} \quad (4.51)$$

Un altro comportamento di cui è necessario tenere conto, nel codice di calcolo impiegato, è che i moltiplicatori di Lagrange vengono applicati nuovamente ad ogni iterazione applicando nuovamente i carichi esterni imposti tramite i moltiplicatori e quindi incrementando ad ogni iterazione il carico, il quale, nel ciclo deve restare costante e aumenta soltanto in base alla time-table. Non tenendo conto di questo comportamento la soluzione del sistema non converge. Quindi dato che i moltiplicatori di Lagrange vengono utilizzati per applicare le condizioni al contorno dei carichi esterni (in particolare, spostamenti impressi) allora è sufficiente applicarli alla prima iterazione altrimenti bisogna tenerne conto tramite il termine  $\mathbf{C}^T \lambda$ .

Si vuole dare un breve spiegazione matematica ed una interpretazione ingegneristica. Si è detto che la matrice  $\mathbf{K}$  è singolare perché non tiene conto dei *constraints* ma la matrice  $\mathbf{J}$ , di cui  $\mathbf{K}$  è il blocco di nord-est, è regolare, perché le matrici  $\mathbf{C}$  e  $\mathbf{C}^T$  tengono in conto sia i *constraints* che la variazione delle reazioni dei *constraint*. Si osserva che se i *constraints* sono lineari, come di fatto lo sono nel caso in esame, il vettore  $\mathbf{R}_2$  è nullo (nei limiti della tolleranza):

$$\mathbf{C}\bar{\mathbf{u}} - \mathbf{f}^{(\lambda)} = 0 \quad (4.52)$$

Questo avviene perché la condizione di Dirichlet viene perfettamente soddisfatta dato che la condizione al contorno cinematica (spostamento impresso) resta identica in tutti gli step e quindi l'incremento  $\Delta\lambda_{t+1}^{i+1}$  è nullo [31]. Dato che le reazioni dei *constraints* sono lineari (si nota che il termine  $\mathbf{C}^T\lambda$  è lineare in  $\lambda$ ) e tenendo conto di quanto appena detto, il sistema (4.50) si può riscrivere come:

$$\begin{bmatrix} \mathbf{K}(\mathbf{u}_{t+1}^{i+1}) & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_{t+1}^{i+1} \\ \lambda_{t+1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{(u)} - \mathbf{K}(\Delta\mathbf{u}_{t+1}^i)\mathbf{u}_{t+1}^i \\ \mathbf{f}^{(\lambda)}_{t+1} \end{bmatrix} \quad (4.53)$$

Dove  $\lambda_{t+1}$  viene aggiornato solo alla prima iterazione del ciclo Newton-Raphson. Nel caso in esame si è utilizzato questo secondo approccio. Alla prima iterazione si ha:

$$\begin{bmatrix} \mathbf{K}(\mathbf{u}_{t+1}^0) & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_{t+1}^0 \\ \lambda_{t+1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{(u)} - \mathbf{K}(\Delta\mathbf{u}_t)\mathbf{u}_t \\ \mathbf{f}^{(\lambda)}_{t+1} \end{bmatrix} \quad (4.54)$$

Alla seconda iterazione si ha:

$$\begin{bmatrix} \mathbf{K}(\mathbf{u}_{t+1}^1) & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{u}_{t+1}^1 \\ \lambda_{t+1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^{(u)} - \mathbf{K}(\Delta\mathbf{u}_{t+1}^0)\mathbf{u}_{t+1}^0 \\ 0 \end{bmatrix} \quad (4.55)$$

## 4.5 Rigidezza negativa - Matrice di softening

La *rigidezza negativa*, espressa in termini numerici dalla *matrice di softening*, esprime la perdita di rigidezza del solido dovuta alla presenza delle forze coesive nella zona di processo.

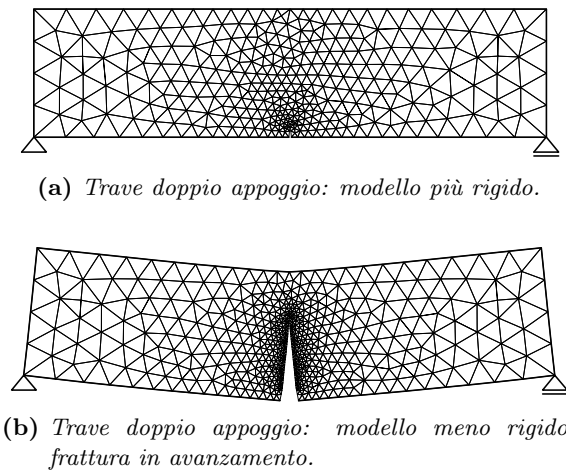
La presenza di queste forze e la loro variazione portano un contributo di rigidezza, oltre che di forze di contorno, negativo.

Il contributo virtuale e discretizzato di queste forze è:

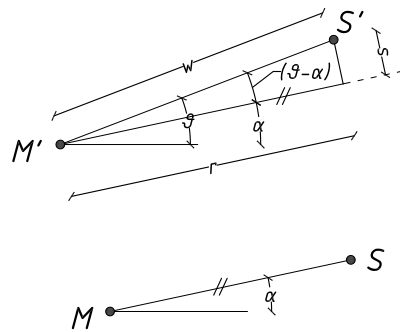
$$\int_{\Gamma_c} (\mathbf{N}^u)^T \mathbf{c} \, d\Gamma \quad (4.56)$$

La continua variazione di queste forze rende il problema non lineare. Il loro contributo alla rigidezza è dato da:

$$\mathbf{J}(\mathbf{u}_{t+1}^{i+1}) = \left[ \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right]_{t+1}^{i+1} = \left[ \mathbf{M} \frac{\partial \ddot{\mathbf{u}}}{\partial \mathbf{u}} + \mathbf{K} + \mathbf{D} \frac{\partial \dot{\mathbf{u}}}{\partial \mathbf{u}} - \frac{\partial \mathbf{f}^{(c)}}{\partial \mathbf{u}} \right]_{t+1}^{i+1} \quad (4.57)$$



**Figura 4.3:** Trave doppio appoggio: confronto rigidezze all'avanzare della frattura.



**Figura 4.4:** Geometria del nodo master e slave in situazione deformata e indeformata.

La matrice di softening è fornita allora da:

$$\mathbf{K1} = - \frac{\partial \mathbf{f}^{(c)}}{\partial \mathbf{u}} \quad (4.58)$$

Le forze di richiusura si sviluppano tra due nodi che sono stati sdoppiati dalla mesh precedente e appartengono ai due lati apposti della frattura. Per convenzione in questo lavoro, questi due nodi vengono chiamati *nodo slave* e *nodo master* ed appartengono ai due lati opposti della fessura. La forza di richiusura dipende come già detto in 2.4, dall'apertura  $w$  della frattura (Fratture di Modo 1).

Prima di ricavare i termini che compongono la matrice di rigidezza, si forniscono alcune definizioni utili a comprendere i passaggi che seguiranno. Si consideri la Figura 4.4. Si nota come ci siano due configurazioni, una deformata e una indeformata. Si definiscono:

- $(M, S)$  denominazione coppia dei nodi nel sistema indeformato;
- $(M', S')$  denominazione coppia dei nodi nel sistema deformato;
- $(x_M, y_M)$  è la coordinata del *nodo master*;
- $(x_S, y_S)$  è la coordinata del *nodo slave*;
- $(dx_M, dy_M)$  è l'incremento di spostamento dovuto alla deformazione del nodo *nodo master*;
- $(dx_S, dy_S)$  è l'incremento di spostamento dovuto alla deformazione del nodo *nodo slave*;
- $w$  è l'apertura attuale della frattura (misurata come distanza tra i due nodi);
- $r$  è l'apertura ortogonale alla frattura;
- $s$  è l'apertura parallela alla frattura, o scorrimento;
- $\alpha$  è l'angolo che si forma rispetto al riferimento con la congiungente  $(M, S)$ , quindi nel sistema indeformato;
- $\vartheta$  è l'angolo che si forma rispetto al riferimento con la congiungente  $(M', S')$ , quindi nel sistema deformato.

Le forze di richiusura possono essere espresse in funzione dell'apertura  $w$  tramite il legame coesivo. Si ha:

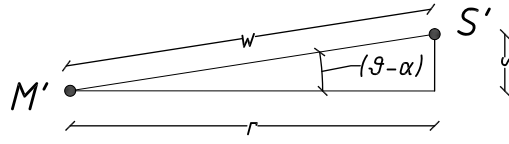
$$F(w) = A\sigma_u \frac{w}{w_c} \quad (4.59)$$

dove:

- $A$  è l'area di influenza della tensione su cui viene calcolata la forza, pari a metà elemento finito da un lato della frattura e metà dall'altro;
- $\sigma_u$  è la tensione calcolata sui bordi della frattura;
- $w$  è l'apertura attuale della frattura (misurata come distanza tra i due nodi);
- $w_c$  è l'apertura massima della frattura (misurata come distanza tra i due nodi).

ed in particolare con riferimento al suddivisione del problema tra Modo I e II si può definire:

$$\begin{cases} r = w \cdot \cos(\vartheta - \alpha) \\ s = w \cdot \sin(\vartheta - \alpha) \end{cases} \quad (4.60)$$



**Figura 4.5:** Scomposizione della geometria della frattura.

con  $r$  ed  $s$  come in Figura 4.5 con  $\theta$  e  $\alpha$ . Nel seguito verrà fatto riferimento solo ad  $r$  essendo identico il procedimento per  $s$ .

La (4.59) si riscrive ora come:

$$F(w) = A\sigma_u \frac{w}{w_c} = A\sigma_c \frac{r}{w_c} \frac{1}{\cos(\vartheta - \sigma)} \quad (4.61)$$

La matrice di softening è una matrice 4x4, fornisce il contributo di una coppia di nodi master-slave ( $S$  sta per slave ed  $M$  per master) e si presenta nella forma (la forma seguente vale solo per Modo I) :

$$\mathbf{K1} = \begin{bmatrix} S_x & S_y & M_x & M_y \\ K1_{11} & K1_{12} & K1_{13} & K1_{14} \\ K1_{21} & K1_{22} & K1_{23} & K1_{24} \\ K1_{31} & K1_{32} & K1_{33} & K1_{34} \\ K1_{41} & K1_{42} & K1_{43} & K1_{44} \end{bmatrix} \begin{matrix} S_x \\ S_y \\ M_x \\ M_y \end{matrix} \quad (4.62)$$

I contributi della matrice di softening sono dati dalle derivate di  $F$ , nelle componenti  $x$  ed  $y$  sia del nodo slave che del nodo master, rispetto alle coordinate di entrambi i nodi:  $x_S$ ,  $y_S$ ,  $x_M$  e  $y_M$ . Si hanno quindi quattro forze  $F_x^{(M)}$ ,  $F_y^{(M)}$ ,  $F_x^{(S)}$  e  $F_y^{(S)}$  che vanno derivate rispetto a  $x_S$ ,  $y_S$ ,  $x_M$  e  $y_M$  ottenendo così i 16 contributi della matrice.

Le forze sono date da:

$$\begin{aligned} F_{M_x} &= F(w) \cdot \cos(\alpha) \\ F_{M_y} &= F(w) \cdot \sin(\alpha) \\ F_{S_x} &= -F(w) \cdot \cos(\alpha) \\ F_{S_y} &= -F(w) \cdot \sin(\alpha) \end{aligned}$$

Si riporta per intero un esempio:

$$\frac{\partial F^{(S)}}{\partial x_S} = \frac{\partial F^{(S)}}{\partial r} \frac{\partial r}{\partial x_S} = -\cos(\alpha) \cdot \frac{\sigma_u A}{w_c} \left[ \frac{\partial w}{\partial x_S} \cdot \cos(\vartheta - \alpha) + w \cdot \frac{\partial(\cos(\vartheta - \alpha))}{\partial x_S} \right]$$

Per poter procedere si esprime l'apertura  $w$  come:

$$\begin{cases} \Delta x = x_S - x_M + dx_S - dx_M \\ \Delta y = y_S - y_M + dy_S - dy_M \\ w = \sqrt{(\Delta x)^2 + (\Delta y)^2} \end{cases} \quad (4.63)$$

Senza entrare nel dettaglio, si ottengono i seguenti passaggi intermedi:

$$\begin{aligned} \frac{\partial w}{x_M} &= -\frac{\Delta x}{w}; & \frac{\partial \vartheta}{x_M} &= +\frac{\Delta y}{w} \\ \frac{\partial w}{y_M} &= -\frac{\Delta y}{w}; & \frac{\partial \vartheta}{y_M} &= -\frac{\Delta x}{w} \\ \frac{\partial w}{x_S} &= +\frac{\Delta x}{w}; & \frac{\partial \vartheta}{x_S} &= -\frac{\Delta y}{w} \\ \frac{\partial w}{y_S} &= +\frac{\Delta y}{w}; & \frac{\partial \vartheta}{y_S} &= +\frac{\Delta x}{w} \end{aligned}$$

Sviluppando i calcoli, e ricordando la (4.63) si ottiene:

$$\frac{\partial F_x^{(S)}}{\partial x_S} = \frac{\partial F_x^{(S)}}{\partial r} \frac{\partial r}{\partial x_S} = -\cos(\alpha) \cdot F_c \cdot \left[ +\frac{\Delta x}{w} \cdot \cos(\vartheta - \alpha) - \frac{\Delta y}{w} \cdot \sin(\vartheta - \alpha) \right] \quad (4.64)$$

e con  $F_c$  si intende:

$$F_c = \frac{\partial F^{(S)}}{\partial r} = \frac{\partial}{\partial r} \quad (4.65)$$

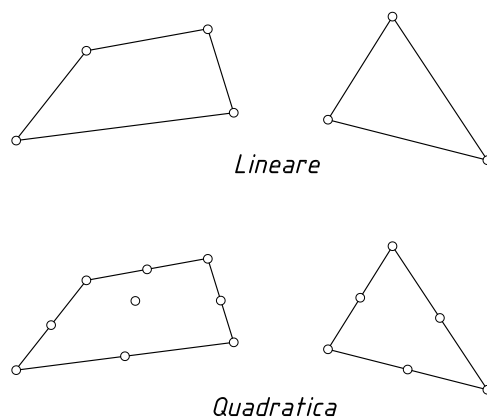
Sulla base della (4.64) si possono ottenere tutti i contributi della matrice di softening. Si riporta per semplicità solo la prima riga della matrice essendo le altre ottenute in modo simile:

$$\begin{aligned} K1_{11} &= \frac{\partial F_x^{(S)}}{\partial x_S} = -\cos(\alpha) \cdot F_c \cdot \left[ +\frac{\Delta x}{w} \cdot \cos(\vartheta - \alpha) - \frac{\Delta y}{w} \cdot \sin(\vartheta - \alpha) \right] \\ K1_{12} &= \frac{\partial F_y^{(S)}}{\partial y_S} = -\cos(\alpha) \cdot F_c \cdot \left[ +\frac{\Delta x}{w} \cdot \cos(\vartheta - \alpha) + \frac{\Delta y}{w} \cdot \sin(\vartheta - \alpha) \right] \\ K1_{13} &= \frac{\partial F_x^{(S)}}{\partial x_M} = -\cos(\alpha) \cdot F_c \cdot \left[ -\frac{\Delta x}{w} \cdot \cos(\vartheta - \alpha) + \frac{\Delta y}{w} \cdot \sin(\vartheta - \alpha) \right] \\ K1_{14} &= \frac{\partial F_y^{(S)}}{\partial y_M} = -\cos(\alpha) \cdot F_c \cdot \left[ -\frac{\Delta x}{w} \cdot \cos(\vartheta - \alpha) - \frac{\Delta y}{w} \cdot \sin(\vartheta - \alpha) \right] \end{aligned}$$

La matrice è  $\mathbf{K1}$  è una matrice 4x4 e viene ricalcolata ad ogni variazione della zona di processo quindi ad ogni iterazione del ciclo Newton-Raphson. Le forze coesive quindi danno un contributo sia in termini di rigidezza attraverso la matrice  $\mathbf{K1}$  che in termini di forze esterne da sommare al termine noto. I due contributi diventano quindi:

$$\begin{cases} (\mathbf{K} - \mathbf{K1}) & \text{matrice di rigidezza effettiva} \\ (\mathbf{f}^{(u)} + \mathbf{f}^{(c)}) & \text{termine noto effettivo} \end{cases} \quad (4.66)$$





**Figura 4.6:** Alcuni elementi bidimensionali tipici con diversi gradi di interpolazione dei valori nodali.

## 4.6 Scelta degli elementi finiti

Gli elementi finiti utilizzati sono elementi finiti *isoparametrici* in cui il numero di nodi in cui sono note le funzioni spostamento e geometria sono uguali. È il tipo di elemento più diffuso in quanto semplice ed efficiente. Le funzioni di forma devono sottostare al principio che, all'aumentare del grado di raffinamento della mesh, la soluzione converga alla soluzione esatta. Si devono pertanto soddisfare le condizioni minime:

- Devono essere regolari (almeno  $C^1$ ) all'interno dell'elemento;
- Devono essere continue ( $C^0$ ) attraverso il contorno degli elementi;
- Devono essere complete.

Elementi finiti che soddisfano le prime due condizioni sono spesso chiamati elementi a continuità  $C^0$ . Le prime due condizioni garantiscono che le derivate prime delle funzioni di forma più *povere* siano continue nelle interfacce tra gli elementi.

### 4.6.1 Convergenza

La discretizzazione spaziale agli elementi finiti viene implementata esprimendo le incognite principali (spostamenti e/o pressioni) in termini di valori nodali mediante le funzioni di forma.

Le condizioni di convergenza che si devono rispettare sono:

- L'accuratezza raggiunta dipende principalmente dal numero di elementi finiti usati e dalla loro distribuzione nel dominio. L'accuratezza raggiunta può essere misurata attraverso un'opportuna norma, ed è

importante che questa cresca con continuità, al crescere del numero di elementi finiti utilizzati nella discretizzazione. Questa condizione chiamata *convergenza uniforme*, si realizza, se l'elemento usato è di tipo *compatibile e completo*. In questi casi l'elemento è comunemente chiamato *conforme*. Nel caso degli elementi finiti isoparametrici, le coordinate e gli spostamenti lungo un lato dipendono solamente dalle coordinate e dagli spostamenti dei nodi posti lungo tale lato. Se quindi due elementi attigui usano le stesse funzioni forma la compatibilità è garantita.

- La seconda condizione dipende dalla natura delle funzioni forma. Nella forma debole delle equazioni che governano il problema sono presenti derivate spaziali di primo ordine quindi sono sufficienti funzioni di forma di continuità  $C^0$ .

#### 4.6.2 Elementi finiti impiegati

Nel caso in esame, sono stati impiegati elementi finiti bidimensionali triangolari.

Le funzioni di forma utilizzate in tutti i casi analizzati sono quelle lineari. Si tratta delle cosiddette *coordinate d'area*  $\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_3$  :

$$\begin{cases} N_1 = L_1 = 1 - \xi - \eta \\ N_2 = L_2 = \xi \\ N_3 = L_3 = \eta \end{cases}$$

È necessario analizzare nel dettaglio la formulazione dell'elemento per poter definire nel dettaglio le varie matrici impiegate nell'analisi agli elementi finiti.

Le matrici influenzate dalle funzioni di forma sono, per il caso piano:

- $\mathbf{N}$
- $\mathbf{B} = \mathbf{LN}$

Esplicitamente sono:

$$\mathbf{N} = \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix} \quad (4.67)$$

$$\mathbf{B} = \begin{bmatrix} \frac{\partial}{\partial \xi} & 0 \\ 0 & \frac{\partial}{\partial \eta} \\ \frac{\partial}{\partial \xi} & \frac{\partial}{\partial \eta} \end{bmatrix} \mathbf{J}^{-1} \begin{bmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 \end{bmatrix} \quad (4.68)$$

Con  $\mathbf{J}$  lo Jacobiano della trasformazione:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} \quad (4.69)$$

con  $\mathbf{L}$  operatore differenziale,  $\mathbf{N}$  matrice delle funzioni di forma dell'elemento e  $\mathbf{J}$  lo Jacobiano della trasformazione.

Si osserva come negli elementi finiti triangolari in esame la matrice  $\mathbf{B}$  sia costante. Questo implica che il campo di deformazione all'interno dell'elemento è costante. A volte gli elementi finiti triangolari lineari vengono anche definiti come *constant strain elements*. In realtà la deformazione e la tensione variano in tutti i punti del dominio quindi utilizzando questi triangoli si deve utilizzare mesh molto fine laddove si hanno elevati gradienti di deformazione e di tensione.

### 4.6.3 Integrazione numerica - quadratura

Gli integrali di area e di volume presenti nelle formule fino a qui presentati sono risolti numericamente tramite le *formule di quadratura di Gauss*. Considerando un integrale generico esso si può approssimare come:

$$\int_A \mathbf{F} dA = \int_A \mathbf{F} d\xi d\eta \approx \sum_{k=1}^n \mathbf{w}_k \mathbf{F}_k \quad (4.70)$$

dove  $\mathbf{w}_k$  è il peso del punto Gauss k-esimo e  $\mathbf{F}_k$  è la funzione calcolata nel punto di integrazione considerato  $(\xi, \eta)$ . È necessaria una breve nota sulla scelta del numero di punti di Gauss per l'integrazione numerica in quanto essi devono essere scelti in numero tale da poter ottenere una matrice di rango completo, quindi *esatta*. L'uso di elementi *sottointeintegrati*, può portare a matrici singolari quando si ha uno stato di deformazione costante in tutti i punti di integrazione. Mentre per la matrice di rigidità è sufficiente un solo punto Gauss, per la matrice delle masse sono necessari minimo tre punti Gauss.

## 4.7 Stato piano di deformazione

Uno stato di tensione, o di deformazione, si può dire piano se sono esprimibili, nell'ordine, mediante tre sole componenti di tensione, oppure mediante tre sole componenti di deformazione.

I problemi piani riguardano solidi di forma cilindrica riferiti ad un sistema di assi cartesiani  $x, y$  e  $z$  avente gli assi  $x$  ed  $y$  nel piano della sezione trasversale retta e l'asse  $z$  parallelo all'asse del cilindro.

Un corpo è in uno stato piano di deformazione quando le componenti del tensore delle deformazioni con l'indice  $z$  sono tutte nulle e contemporaneamente le restanti sono indipendenti da  $z$ . Per ragioni di simmetria, gli spostamenti dei punti del solido appartengono a piani perpendicolari all'asse  $z$  e si riproducono identicamente in tutte le sezioni rette del cilindro.

La matrice elastica di questo problema è:

$$\mathbf{D} = \frac{1}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \quad (4.71)$$

## 4.8 Matrice di rigidezza

La matrice di rigidezza del metodo, che lega spostamenti nodali alle forze nodali può essere derivata tramite il Principio dei Lavori Virtuali.

Si consideri l'equazione che esprime l'uguaglianza tra il lavoro virtuale esterno fatto da un sistema di forze ed il lavoro virtuale interno:

$$\mathbf{u}\mathbf{F} = \int_V \boldsymbol{\varepsilon}^T \boldsymbol{\sigma} dV \quad (4.72)$$

Il vettore della deformazione virtuale è legato al vettore degli spostamenti virtuali tramite la matrice  $\mathbf{B}$ :

$$\boldsymbol{\varepsilon} = \mathbf{B}\mathbf{u} \quad (4.73)$$

Sostituendo questa ultima equazione nella precedente e notando che lo spostamento virtuale è arbitrario si può scrivere:

$$\mathbf{F} = \int_V \mathbf{B}^T \boldsymbol{\sigma} dV \quad (4.74)$$

Infine, dato che  $\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon} = \mathbf{D}\mathbf{B}\mathbf{u}$  si ottiene:

$$\left( \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} dV \right) \mathbf{u} = \mathbf{F} \quad (4.75)$$

o:

$$\mathbf{K}\mathbf{u} = \mathbf{F} \quad (4.76)$$

In questo lavoro, la matrice di rigidezza è computata utilizzando tecniche di quadratura numerica per la valutazione dell'integrale.

## 4.9 Matrice di massa

La matrice di massa dell'elemento è fondamentale nell'analisi dinamica. La forma della matrice di massa è definita dalla:

$$\mathbf{M} = \int_{\Omega_e} \mathbf{N}^T \mathbf{c} \mathbf{N} d\Omega_e \quad (4.77)$$

dove  $\mathbf{c}$  è una matrice di coefficienti o uno scalare, nel caso in esame è lo scalare che indica la densità  $\rho$  del solido analizzato. Quando la matrice  $\mathbf{M}$  ha rango completo allora essa viene calcolata *esattamente* e la matrice delle masse così definita viene detta *consistent* perché utilizza le stesse funzioni di forma utilizzate per formare la matrice di rigidezza. Utilizzando le funzioni di forma 4.6 si ottiene:

$$\mathbf{M} = \int_{\Omega_e} \begin{bmatrix} \mathbf{N}_1 \mathbf{N}_1 & 0 & \mathbf{N}_1 \mathbf{N}_2 & 0 & \mathbf{N}_1 \mathbf{N}_3 & 0 \\ 0 & \mathbf{N}_1 \mathbf{N}_1 & 0 & \mathbf{N}_1 \mathbf{N}_2 & 0 & \mathbf{N}_1 \mathbf{N}_3 \\ \mathbf{N}_2 \mathbf{N}_1 & 0 & \mathbf{N}_2 \mathbf{N}_2 & 0 & \mathbf{N}_2 \mathbf{N}_3 & 0 \\ 0 & \mathbf{N}_2 \mathbf{N}_1 & 0 & \mathbf{N}_2 \mathbf{N}_2 & 0 & \mathbf{N}_2 \mathbf{N}_3 \\ \mathbf{N}_3 \mathbf{N}_1 & 0 & \mathbf{N}_3 \mathbf{N}_2 & 0 & \mathbf{N}_3 \mathbf{N}_3 & 0 \\ 0 & \mathbf{N}_3 \mathbf{N}_1 & 0 & \mathbf{N}_3 \mathbf{N}_2 & 0 & \mathbf{N}_3 \mathbf{N}_3 \end{bmatrix} d\Omega_e \quad (4.78)$$

$$\mathbf{M} = \frac{\rho A}{12} \begin{bmatrix} 2 & 0 & 1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 0 & 1 \\ 1 & 0 & 2 & 0 & 1 & 0 \\ 0 & 1 & 0 & 2 & 0 & 1 \\ 1 & 0 & 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 1 & 0 & 2 \end{bmatrix} \quad (4.79)$$

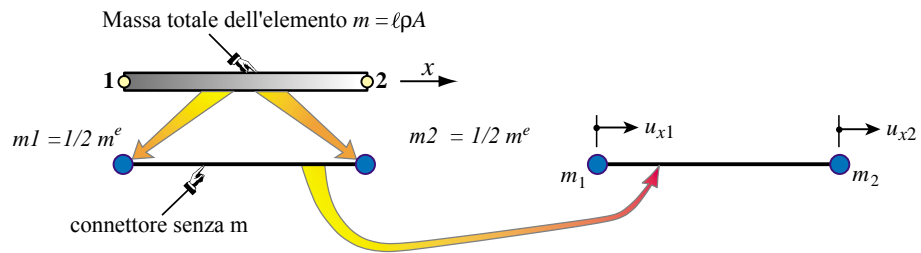
dove:

$$\int_{\Omega_e} \mathbf{N}_i \mathbf{N}_i d\Omega_e = \frac{A}{3} \quad (4.80)$$

$$\int_{\Omega_e} \mathbf{N}_i \mathbf{N}_j d\Omega_e = \frac{A}{6} \quad (4.81)$$

Questa stessa formulazione si ottiene con almeno tre punti Gauss in modo da ottenere una matrice di rango completo.

Esiste un'altra forma di questa matrice che rende gli algoritmi dinamici più efficienti senza sensibile perdita di accuratezza. Questa forma consiste nel *diagonalizzare* la matrice delle masse o più spesso indicata come *lumped* in ambito ingegneristico.



**Figura 4.7:** Matrice di massa: elemento asta.

Esistono diversi metodi per formare una matrice lumped. Uno dei metodi più semplici è quello di considerare la condizione di conservazione della massa nell'elemento. Questo ragionamento è supportato dal fatto che ogni forma di lumping che conserva la massa dell'elemento risulta convergente. Una delle forme nate da questo approccio è quella di sommare le entrate della riga  $j$ -esima della matrice delle masse e porre il risultato sull'elemento diagonale  $jj$  di quella riga mentre tutte le altre entrate sono nulle. Si ottiene allora:

$$\mathbf{M} = \frac{\rho A}{3} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.82)$$

## Capitolo 5

# Soluzione Numerica

### 5.1 Problemi di equilibrio

L'equazione differenziale che rappresenta il problema in esame è definito nella forma:

$$\mathbf{A}\ddot{\Phi} + \mathbf{B}\dot{\Phi} + \mathbf{L}(\Phi) = \mathbf{0} \quad (5.1)$$

dove:

- Il termine  $\Phi$  rappresenta un vettore di variabili dipendenti che è rappresentato dalle incognite spostamenti  $\mathbf{u}$ ;
- Il termine  $\mathbf{L}$  è un'operatore differenziale che contiene le derivate spaziali come  $\partial/\partial x$  e  $\partial/\partial y$  che possono essere, come spesso accade, nonlineari;
- I termini  $\mathbf{A}$  e  $\mathbf{B}$  rappresentano matrici costanti. In particolare si nota che rispetto al capitolo 3 è presente anche un termine addizionale che rappresenta il termine inerziale che diventa non trascurabile in campo dinamico;
- È presente una dipendenza dal tempo delle variabili che richiede anche la discretizzazione temporale.

### 5.2 Discretizzazione nello spazio

Viene descritta in questa sezione la discretizzazione delle equazioni che governano il problema  $U$  e la procedura per la soluzione. Le equazioni del capitolo 3 vengono discretizzate nello spazio con il consueto metodo dei residui pesati di *Bubnov-Galerkin* incorporando anche le equazioni costitutive. Le incognite sono rappresentate dai valori degli spostamenti nodali  $\mathbf{u}$ .

Le funzioni di forma utilizzate per discretizzare le variabili incognite spostamenti  $\mathbf{u}$  del problema in esame sono:

$$\mathbf{u} \cong \mathbf{u}^h = \sum_{k=1}^n N_k \bar{\mathbf{u}}_k = \mathbf{N} \bar{\mathbf{u}} \quad (5.2)$$

L'equazione che entra in gioco è l'equazione di equilibrio elastico. Viene riportata nuovamente per maggior chiarezza:

$$\begin{aligned} \mathbf{L}^T \boldsymbol{\sigma} - \rho \ddot{\mathbf{u}} + \rho \mathbf{g} &= 0 \\ \mathbf{L}^T \boldsymbol{\sigma} - \rho \ddot{\mathbf{u}} + \mathbf{b} &= 0 \end{aligned} \quad (5.3)$$

Ovviamente, è necessario definire le condizioni iniziali e le condizioni al contorno. Le condizioni iniziali specificano i valori di partenza del campo di spostamento al tempo  $t = 0$ :

$$\mathbf{u} = \mathbf{u}_0 \quad \text{in } \Omega \text{ e su } \Gamma \quad (5.4)$$

in cui  $\Omega$  rappresenta il dominio di interesse e  $\Gamma$  il suo contorno. Per le condizioni al contorno si ha:

$$\begin{cases} \mathbf{t} = \bar{\mathbf{t}} & \text{su } \Gamma_t \\ \mathbf{u} = \bar{\mathbf{u}} & \text{su } \Gamma_u \\ \Gamma = \Gamma_t \cup \Gamma_u \end{cases} \quad (5.5)$$

Partendo dall'equazione (5.3), e con rispetto delle condizioni al contorno, si applica il metodo dei residui pesati descritto precedentemente con i pesi  $\mathbf{w}$  e  $\bar{\mathbf{w}}$ :

$$\int_{\Omega} \mathbf{w}^T (\mathbf{L}^T \boldsymbol{\sigma} - \rho \ddot{\mathbf{u}} + \mathbf{b}) d\Omega + \int_{\Gamma} \bar{\mathbf{w}}^T (\mathbf{I}^T \boldsymbol{\sigma} - \bar{\mathbf{t}}) d\Gamma = 0 \quad (5.6)$$

Le funzioni peso sono scelte in modo tale da:

$$\begin{aligned} \mathbf{w} &= 0 & \text{su } \Gamma^u \\ \bar{\mathbf{w}} &= -\mathbf{w} & \text{su } \Gamma^u \end{aligned} \quad (5.7)$$

La precedente equazione richiede, nell'integrazione, il calcolo della derivata seconda degli spostamenti. Si può allora *rilassare* questa condizione applicando il teorema di Green alla precedente. Si ottiene:

$$\int_{\Omega} \phi \frac{\partial \Psi}{\partial x} d\Omega = - \int_{\Omega} \frac{\partial \phi}{\partial x} \Psi d\Omega + \int_{\Gamma} \phi \Psi n_x d\Gamma \quad (5.8)$$

quindi:

$$- \int_{\Omega} (\mathbf{L}\mathbf{w})^T \boldsymbol{\sigma} d\Omega + \int_{\Gamma} \mathbf{w}^T \mathbf{I} \boldsymbol{\sigma} d\Gamma + \int_{\Omega} \mathbf{w}^T \rho \ddot{\mathbf{u}} d\Omega + \int_{\Omega} \mathbf{w}^T \mathbf{b} d\Omega$$



$$+ \int_{\Gamma_t} \bar{\mathbf{w}}^T (\mathbf{I}^T \boldsymbol{\sigma} - \bar{\mathbf{t}}) d\Gamma + \int_{\Gamma_c} \bar{\mathbf{w}}^T \mathbf{c} d\Gamma = 0 \quad (5.9)$$

che grazie alla (5.7) diventa:

$$\begin{aligned} & \int_{\Omega} (\mathbf{L}\mathbf{w})^T \boldsymbol{\sigma} d\Omega + \int_{\Omega} \mathbf{w}^T \rho \ddot{\mathbf{u}} d\Omega \\ &= \int_{\Omega} \mathbf{w}^T \mathbf{b} d\Omega + \int_{\Gamma_t} \mathbf{w}^T \bar{\mathbf{t}} d\Gamma + \int_{\Gamma_c} (\bar{\mathbf{w}})^T \mathbf{c} d\Gamma \end{aligned} \quad (5.10)$$

Per ottenere la discretizzazione nello spazio, secondo Bubnov-Galerkin, inserendo le funzioni peso pari  $\mathbf{w} = \bar{\mathbf{w}} = (\mathbf{N})^T$  nella precedente equazione:

$$\begin{aligned} & \int_{\Omega} (\mathbf{L}\mathbf{N})^T \boldsymbol{\sigma} d\Omega + \int_{\Omega} (\mathbf{N})^T \rho \ddot{\mathbf{u}} d\Omega \\ &= \int_{\Omega} \mathbf{N}^T \mathbf{b} d\Omega + \int_{\Gamma_t} (\mathbf{N})^T \bar{\mathbf{t}} d\Gamma + \int_{\Gamma_c} (\mathbf{N})^T \mathbf{c} d\Gamma \end{aligned} \quad (5.11)$$

Si ottiene infine (si veda il [48] per maggiori dettagli):

$$\int_{\Omega} \mathbf{B}^T \boldsymbol{\sigma} d\Omega + \left[ \int_{\Omega} (\mathbf{N})^T \rho (\mathbf{N}) d\Omega \right] \ddot{\mathbf{u}} = \mathbf{f}^{(u)} + \int_{\Gamma_c} (\mathbf{N})^T \mathbf{c} d\Gamma \quad (5.12)$$

dove la matrice  $\mathbf{B}$  è data da:

$$\mathbf{B} \equiv \mathbf{L}\mathbf{N} \quad (5.13)$$

con  $\mathbf{L}$  operatore differenziale definito come:

$$\mathbf{L} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & 0 \\ 0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial x} & 0 & \frac{\partial}{\partial z} \end{bmatrix} \quad (5.14)$$

Il vettore delle forze esterne  $\mathbf{f}^u$  contiene le forze di volume e le trazioni sul contorno. Si ha:

$$\mathbf{f}^u = \int_{\Omega} (\mathbf{N})^T (\rho \mathbf{g}) d\Omega + \int_{\Gamma_t} (\mathbf{N})^T \bar{\mathbf{t}} d\Gamma \quad (5.15)$$

in cui  $\bar{\mathbf{t}}$  rappresenta il vettore delle trazioni sul contorno  $\Gamma_t$  e  $\mathbf{c}$  rappresenta il vettore delle trazioni sul contorno  $\Gamma_c$ . L'equazione differenziale discretizzata diventa infine:

$$\boxed{\mathbf{M} \ddot{\mathbf{u}} + \int_{\Omega} \mathbf{B}^T \boldsymbol{\sigma} d\Omega - \mathbf{f}^{(u)} - \int_{\Gamma_c} (\mathbf{N})^T \mathbf{c} d\Gamma = 0} \quad (5.16)$$

Considerando un mezzo elastico lineare si definisce:

$$\boldsymbol{\sigma} = \mathbf{D}\boldsymbol{\varepsilon} = \mathbf{D}\mathbf{B}\bar{\mathbf{u}} \quad (5.17)$$

- La matrice della masse  $\mathbf{M}$  è data da:

$$\mathbf{M} = \int_{\Omega} (\mathbf{N})^T \rho (\mathbf{N}) d\Omega \quad (5.18)$$

- La matrice di rigidità  $\mathbf{K}$  è data da:

$$\mathbf{K} = \int_{\Omega} \mathbf{B}^T \mathbf{D} \mathbf{B} d\Omega \quad (5.19)$$

### 5.3 Discretizzazione nel tempo

Dato che il dominio del tempo è infinito, esso dovrà necessariamente essere limitato ad un incremento finito  $\Delta t$  e relazionato alla condizione iniziale al tempo  $t_n$  e ai successivi  $t_{n+1} + \Delta t$ , ottenendo le *relazioni di ricorrenza*. Il punto di partenza è chiaramente dato dall'equazione (5.16):

$$\mathbf{M}\ddot{\bar{\mathbf{u}}} + \mathbf{K}\bar{\mathbf{u}} - \mathbf{f}^{(u)} - \int_{\Gamma_c} (\mathbf{N})^T \mathbf{c} d\Gamma = 0 \quad (5.20)$$

Sarebbe possibile considerare un approccio che vede la discretizzazione temporale simultanea a quella spaziale. Questo, eccetto alcuni casi (come i problemi di fluidodinamica), non porta a vantaggi sensibili, soprattutto per i problemi analizzati nell'ambito di questa tesi.

La soluzione dell'equazione differenziale che regge i problemi dinamici può essere ottenuta sia tramite schemi impliciti, sia tramite schemi espliciti. Se la risposta all'iterazione corrente ( $t + \Delta t$ ) dipende interamente dalla soluzione al passo precedente ( $t$ ), si avrà una soluzione esplicita, mentre se la risposta all'iterazione corrente dipende dalla velocità e dall'accelerazione correnti si avrà uno schema implicito.

Tenendo conto della nonlinearietà, il problema di equilibrio dinamico deve essere soddisfatto ad ogni passo temporale  $i + 1$ :

$$\mathbf{M}\ddot{\bar{\mathbf{u}}}_{i+1} + \mathbf{C}\dot{\bar{\mathbf{u}}}_{i+1} + \mathbf{K}\bar{\mathbf{u}}_{i+1} = \mathbf{F}_{i+1} \quad (5.21)$$

Il sistema può essere risolto con i consueti metodi utilizzati nei problemi dinamici strutturali: differenze finite, Newmark, Discontinuous Galerkin.

La differenza con i sistemi lineari sta nel fatto che la matrice tangente deve essere ricalcolata ad ogni passo temporale con il relativo contributo

incrementale di soluzione. Durante ogni passo temporale viene quindi applicato il metodo di Newton-Raphson, calcolato l'incremento di soluzione che viene aggiunto a quella nota al passo precedente. Una volta noto vengono calcolate da questo le derivate prime e seconde della soluzione che permette di determinare i contributi, associati a questi gradi di libertà, nel termine delle forze interne da cui si ricalcola il residuo. Una volta calcolato il residuo la procedura viene ripetuta.

### Differenze Finite

Gli schemi di integrazione numerica in tempo alle differenze finite sono classificati, generalmente, come espliciti o impliciti. Negli schemi espliciti l'equazione differenziale che regola il problema viene scritta al tempo  $t$  per cui tutte le variabili note della soluzione e l'equazione alle differenze viene risolto per trovare la soluzione al passo successivo  $t + \Delta t$ . Le approssimazioni alle differenze finite, a seconda del loro ordine e precisione, possono utilizzare la soluzione nota in tempi  $t - \Delta t$ ,  $t - 2\Delta t$  etc. Gli schemi impliciti possono essere sia condizionatamente che incondizionatamente stabili. La stabilità incondizionata (un limite massimo  $\Delta t$ ) è solitamente garantita solo per problemi lineari, e può essere persa in casi non lineari. Tuttavia è generalmente preferibile utilizzare anche nelle analisi non lineari quei metodi che sono stabili in analisi lineari. I metodi impliciti sono più adatti per problemi in cui la variazione della soluzione nel tempo è meno rapida, e sono ammessi passi temporali maggiori.

I metodi espliciti sono solo condizionatamente stabili, cioè l'ampiezza del passo temporale  $\Delta t$  dovrebbe essere inferiore a una ampiezza critica per ottenere una soluzione stabile. La soluzione non converge se viene utilizzato un passo temporale  $\Delta t$  maggiore del limite critico. I metodi espliciti sono generalmente preferiti per la soluzione di problemi in cui la parte interessante della soluzione cambia rapidamente nel tempo, come in problemi di propagazione delle onde. Negli schemi impliciti, l'equazione che governa il problema è scritta al tempo  $t + \Delta t$ , mentre la soluzione è nota al tempo  $t$ . L'equazione risultante dall'uso di un appropriato differenza finita è poi risolta per la variabile soluzione a  $t + \Delta t$ .

Si può pensare di approssimare la derivata di una variabile  $\mathbf{x}$  rispetto al tempo:

$$\frac{d\mathbf{x}}{dt} = \begin{cases} \frac{1}{\Delta t}(\mathbf{x}_{t+1} - \mathbf{x}_t) & \text{Differenze in Avanti;} \\ \frac{1}{\Delta t}(\mathbf{x}_{t+1} - \mathbf{x}_t) & \text{Differenze Indietro;} \\ \frac{1}{\Delta t}(\mathbf{x}_{t+1} - \mathbf{x}_{t-1}) & \text{Differenze Centrali.} \end{cases} \quad (5.22)$$

Mentre l'approssimazione della variabile  $\mathbf{x}$  stessa:

$$\mathbf{x} = \begin{cases} \mathbf{x}_t \\ \mathbf{x}_{t+1} \\ \frac{1}{2}(\mathbf{x}_{t+1} + \mathbf{x}_t) \end{cases} \quad (5.23)$$

con  $n$  un intero positivo:

$$\begin{cases} t = n\Delta t \\ t + 1 = (n + 1)\Delta t \end{cases}$$

Uno schema di validità generale rispetto alle approssimazioni introdotte consiste nello scrivere la (5.21):

$$\begin{aligned} & (\mathbf{Z}\Delta\ddot{\bar{\mathbf{x}}}_{t+1} + \mathbf{V}\Delta\dot{\bar{\mathbf{x}}}_{t+1} + \mathbf{A}\Delta\bar{\mathbf{x}}_{t+1})(\alpha) \\ & (\mathbf{M}\Delta\ddot{\bar{\mathbf{u}}}_t + \mathbf{C}\Delta\dot{\bar{\mathbf{u}}}_t + \mathbf{K}\Delta\bar{\mathbf{u}}_t)(1 - \alpha) \\ & = (\alpha)(\mathbf{F}_{t+1}) + (1 - \alpha)(\mathbf{F}_t) \end{aligned} \quad (5.24)$$

Dove  $\alpha$  è un parametro temporale per l'integrazione definito come:

- $\alpha = 0$  differenze in avanti;
- $\alpha = 1/2$  differenze centrali;
- $\alpha = 1$  differenze all'indietro;
- $\alpha = 1/3$  o  $\alpha = 2/3$  metodo di Galerkin.

Gli schemi con  $\alpha = 0$  sono detti *espliciti* gli altri *impliciti*.

### 5.3.1 Metodi di Newmark

Con *Metodi di Newmark* si intendono un classe di problemi introdotti originariamente per la soluzione di equazioni differenziali del secondo ordine nella dinamica strutturale.

#### Introduzione

Il tempo, variabile indipendente, viene discretizzato in passi di dimensioni finite  $\Delta t$ . Questo si traduce in una espressione che coinvolge valori noti ad un determinato passo temporale e valori ignoti al passo successivo. La soluzione al passo successivo viene quindi calcolata da una semplice manipolazione algebrica, o in caso di sistemi non lineari, quando tale riarrangiamento algebrico non è possibile, con il metodo di Newton-Raphson.

### Metodi di Newmark

Per quanto riguarda la discretizzazione in tempo e la soluzione dell'equazione (5.20) è possibile impiegare il metodo di Newmark, introdotto nel 1959 da N.M. Newmark.

In quanto segue si assume che all'inizio dell'intervallo, cioè all tempo  $t = t_{n+1}$ , siano noti i valori della funzione incognita  $\mathbf{u}$  e delle sue derivate, cioè  $\mathbf{u}_n$ ,  $\dot{\mathbf{u}}_n$  e  $\ddot{\mathbf{u}}_n$  fino a  $\mathbf{u}_n^{p-1}$  dove  $p$  è l'ordine della serie approssimante.

L'algoritmo di Newmark per la risoluzione dei problemi in esame si ottiene utilizzando un'espansione in serie troncata al secondo termine. Si ha:

$$\begin{aligned}\mathbf{u}_{n+1} &= \mathbf{u}_n + \Delta t \dot{\mathbf{u}}_n + \frac{1}{2}(1 - 2\beta)\Delta t^2 \ddot{\mathbf{u}}_n + \beta \Delta t^2 \ddot{\mathbf{u}}_{n+1} \\ &= \tilde{\mathbf{u}}_{n+1} + \beta \Delta t^2 \ddot{\mathbf{u}}_{n+1} \\ \dot{\mathbf{u}}_{n+1} &= \dot{\mathbf{u}}_n + (1 - \gamma)\Delta t \ddot{\mathbf{u}}_n + \gamma \Delta t \ddot{\mathbf{u}}_{n+1} \\ &= \dot{\tilde{\mathbf{u}}}_{n+1} + \gamma \Delta t \ddot{\mathbf{u}}_{n+1}\end{aligned}\tag{5.25}$$

Queste due equazioni, assieme all'equazione di equilibrio (5.20), forniscono le basi per il calcolo delle incognite al tempo  $n + 1$ . Inoltre la matrice delle masse è di tipo *lumped* (come generalmente avviene), le accelerazioni sono disaccoppiate e non si deve fare la fattorizzazione della matrice stessa.

I parametri  $\beta$  e  $\gamma$  definiscono la variazione dell'accelerazione all'interno del time step e determinano la stabilità e l'accuratezza del metodo. Nella classe dei metodi di Newmark rientra anche il *Metodo dei Trapezi*, corrispondente al metodo di Newmark originariamente definito, che adottava come costanti:

$$\gamma = \frac{1}{2}\tag{5.26}$$

$$\beta = \frac{1}{4}\tag{5.27}$$

Il metodo così definito viene chiamato anche *schema ad accelerazione media*.

Inoltre si ha che:

- $\beta \neq 0$  formulazione *Implicita*;
- $\beta = 0$  formulazione *Esplicita*.

Viene scelto uno step di integrazione molto basso (dell'ordine dei centesimi di secondo) e se il metodo non converge allora si abbassa ancora. La convergenza viene misurata dallo squilibrio delle forze non lineari non bilanciate rispetto ad una certa tolleranza. I valori dei coefficienti  $\gamma$  e  $\beta$ , che rispettano le condizioni di convergenza e stabilità, sono definiti come:

$$\gamma = \frac{1}{2}\tag{5.28}$$

$$\frac{1}{6} < \beta < \frac{1}{4} \quad (5.29)$$

Il metodo con  $\beta = \frac{1}{6}$  viene chiamato *schema ad accelerazione lineare*. Con  $\beta = 0$  il metodo di Newmark diventa il più semplice *Metodo di Newmark alle Differenze Centrali* che è un'altro membro della classe di problemi alla Newmark.

Ad ogni passo la soluzione dipende dal valore della soluzione nei passi precedenti. Nasce quindi un problema all'istante iniziale perché in tal caso non esiste un passo precedente; si dice che l'algoritmo di Newmark non è *self-starting*. Le condizioni iniziali all'istante  $t = 0$  riguardano velocità e spostamento, ma non si conosce l'accelerazione  $\ddot{\mathbf{x}}$ . Per determinarla si impone l'equazione di equilibrio dinamico all'istante iniziale:

$$\ddot{\mathbf{u}} = (\mathbf{F}_0 - \mathbf{C}\dot{\mathbf{u}} - \mathbf{K}\mathbf{u})\mathbf{M}^{-1} \quad (5.30)$$

Inglobando il contributo sul contorno della frattura nel vettore dei termini noti  $\mathbf{f}^{(u)}$ , nel caso in cui la risposta del sistema sia non lineare, è possibile scrivere (ricordando che, per una generica quantità  $\phi$  è possibile scrivere  $\Delta\phi_n = \phi_{n+1} - \phi_n$ ):

$$\Psi(\mathbf{u}_{n+1}) = \mathbf{f}_{n+1} - \frac{1}{\beta\Delta t^2}\mathbf{M}[\mathbf{u}_{n+1} - \tilde{\mathbf{u}}_{n+1}] - \mathbf{K}\mathbf{u}_{n+1} = \mathbf{0} \quad (5.31)$$

La soluzione di questo set di equazioni non lineari richiede una procedura iterativa. In particolare, è utilizzato lo schema di Newton-Raphson derivato nel precedente Capitolo. Si può scrivere:

$$\Psi_{n+1}^{k+1} \approx \Psi_{n+1}^k + d\Psi_{n+1}^{k+1} = \mathbf{0} \quad (5.32)$$

assumendo  $\mathbf{f}_{n+1}$  indipendente dalla deformazione. L'incremento da calcolare nell'equazione (5.32) è dato da:

$$d\Psi_{n+1}^{k+1} = -\left[\frac{1}{\beta\Delta t^2}\mathbf{M} - \frac{\partial\mathbf{K}}{\partial\mathbf{u}_{n+1}}\right]d\mathbf{u}_{n+1}^k = -\mathbf{A}_{n+1}^k d\mathbf{u}_{n+1}^k \quad (5.33)$$

quindi, il calcolo dell'incremento della soluzione è dato da:

$$d\mathbf{u}_{n+1}^k = (\mathbf{A}_{n+1}^k)^{-1}\Psi_{n+1}^{k+1} \quad (5.34)$$

La soluzione è aggiornata utilizzando le seguenti formule:

$$\begin{aligned} \mathbf{u}_{n+1}^{k+1} &= \mathbf{u}_{n+1}^k + d\mathbf{u}_{n+1}^k \\ \ddot{\mathbf{u}}_{n+1}^{k+1} &= \frac{1}{\beta\Delta t^2}[\mathbf{u}_{n+1}^{k+1} - \tilde{\mathbf{u}}_{n+1}] \\ \dot{\mathbf{u}}_{n+1}^{k+1} &= \dot{\mathbf{u}}_{n+1}^k + \gamma\Delta t\ddot{\mathbf{u}}_{n+1}^{k+1} \end{aligned} \quad (5.35)$$

Alla prima iterazione si pone:

$$\begin{aligned}\mathbf{u}_{n+1}^1 &= \mathbf{u}_n \\ \ddot{\mathbf{u}}_{n+1}^1 &= \frac{1}{\beta \Delta t^2} [\mathbf{u}_n - \tilde{\mathbf{u}}_{n+1}] \\ \dot{\mathbf{u}}_{n+1}^1 &= \dot{\tilde{\mathbf{u}}}_{n+1} + \gamma \Delta t \ddot{\mathbf{u}}_{n+1}^1\end{aligned}\tag{5.36}$$

in cui le quantità senza apice  $k$  sono i valori giunti a convergenza del passo precedente. Il procedimento iterativo continua fino a che non si raggiunge la convergenza desiderata:

$$\|\Psi_{n+1}^k\| \leq \varepsilon \|\Psi_{n+1}^1\|\tag{5.37}$$

Un valore appropriato di  $\varepsilon$  può essere scelto in base alla precisione dell'elaboratore; ad esempio, se l'elaboratore può calcolare fino a 16 cifre decimali, un valore di  $\varepsilon$  appropriato può essere assunto pari a  $10^{-8}$ .





## Capitolo 6

# Implementazione numerica

### 6.1 Introduzione

In questo capitolo vengono approfonditi alcuni aspetti relativi alla traduzione in codici di calcolo del metodo proposto per l'analisi agli elementi finiti della meccanica della frattura. Questa operazione di implementazione nasconde alcune insidie. La necessità di avere un'implementazione ad alta efficienza computazionale, oltre alle esigenze sintattiche del linguaggio di programmazione utilizzato, ha richiesto, infatti, una codifica che non è stata in genere l'immediata traduzione di quanto visto in sede di presentazione teorica.

Il codice impiegato è stato realizzato in un contesto di ricerca presso l'Università degli Studi di Padova dal Prof. Stefano Secchi. Esso è stato sfruttato e modificato, in parte, per le analisi svolte in questo lavoro. In particolare è stato esteso al problema dinamico con i relativi problemi. Il codice è scritto in C++, potente linguaggio orientato agli oggetti che permette di realizzare codici facilmente modificabili e riutilizzabili in perfetta armonia con i dettami dell'ingegneria del software, con gli obiettivi della ricerca e con lo scopo ingegneristico in esame.

Per descrivere la logica del programma saranno utilizzati diagrammi di flusso ed elenchi. Questa documentazione è necessaria per spiegare come il codice lavora e come è stato impiegato per risolvere il problema ingegneristico.

### 6.2 La programmazione orientata agli oggetti

La filosofia di programmazione orientata agli oggetti mostra caratteristiche di astrazione molto adatte per problemi matematico-numerici. L'astrazione insita nella trasversalità degli strumenti matematici sembra trovare, infatti,

un ottimo corrispettivo nell'astrazione propria della programmazione a oggetti, basata sulla progettazione di tipi di dato da parte dell'utente (più che su operazioni da svolgere, come avviene nella programmazione procedurale) e sul loro uso polimorfo, uno strumento estremamente potente. Accanto al Fortran (linguaggio procedurale tradizionalmente usato per il calcolo scientifico in virtù della sua efficienza computazionale), oggi sono sempre più diffusi anche in ambito scientifico linguaggi come il C++, nato come un miglioramento del linguaggio C orientato agli oggetti.

L'implementazione degli algoritmi in C++, a scapito di una lieve riduzione di efficienza rispetto al Fortran, consente al codice guadagni sia in termini di riusabilità sia in termini di maggior chiarezza per il programmatore.

I codici di calcolo normalmente impiegati dagli ingegneri rappresentano spesso delle cosiddette *scatole chiuse*, cioè non permettono di accedere al codice sorgente ma solo di utilizzarne le funzioni attraverso un'interfaccia (grafica). Questo avviene, oltre che per la proprietà intellettuale, anche per permettere un utilizzo veloce con strumenti di *pre-processing* e altre utilità, ma comporta il rischio di utilizzare un programma senza conoscerne bene il funzionamento. Detto questo dietro a un normale codice commerciale si nascondono essenzialmente tre componenti principali:

- Un libreria di elementi finiti. Attività tipiche di questi moduli sono di creare le matrici di rigidezza, di massa ecc. e di calcolare forze interne e deformazioni;
- Un libreria di modelli costitutivi;
- Un *kernel* che utilizza le librerie sopra descritte per risolvere i sistemi di equazioni discretizzate.

Si è utilizzato il C++ perché permette, rispetto ai classici linguaggi procedurali, di frammentare il codice in *oggetti* e utilizzare un approccio modulare che permette di gestire meglio codici molto estesi come quello in esame.

### 6.3 Modulazione del codice di calcolo

Nel seguito di questo paragrafo si vuole dare una breve spiegazione delle classi, degli oggetti e delle funzioni che gestiscono il fenomeno legato alla propagazione della frattura implementate nel codice C++. I passaggi più critici di queste righe di codice saranno esposti dopo questo paragrafo.

### 6.3.1 Main

Il file sorgente o il cosiddetto *main* si occupa tramite una funzione di creare un oggetto della classe TGeoModel a cui verrà delegata la funzione di risolvere il problema legato alla propagazione della frattura:

```

1
2 int main()
3 {
4     // Crea un oggetto di SolveGeo
5     SolveGeo();
6
7     return 0;
8 }

```

### 6.3.2 Soluzione del problema della frattura

Questa funzione della classe TGeoProblem si occupa di gestire il problema non lineare dinamico e di costruire il problema topologico e FEM:

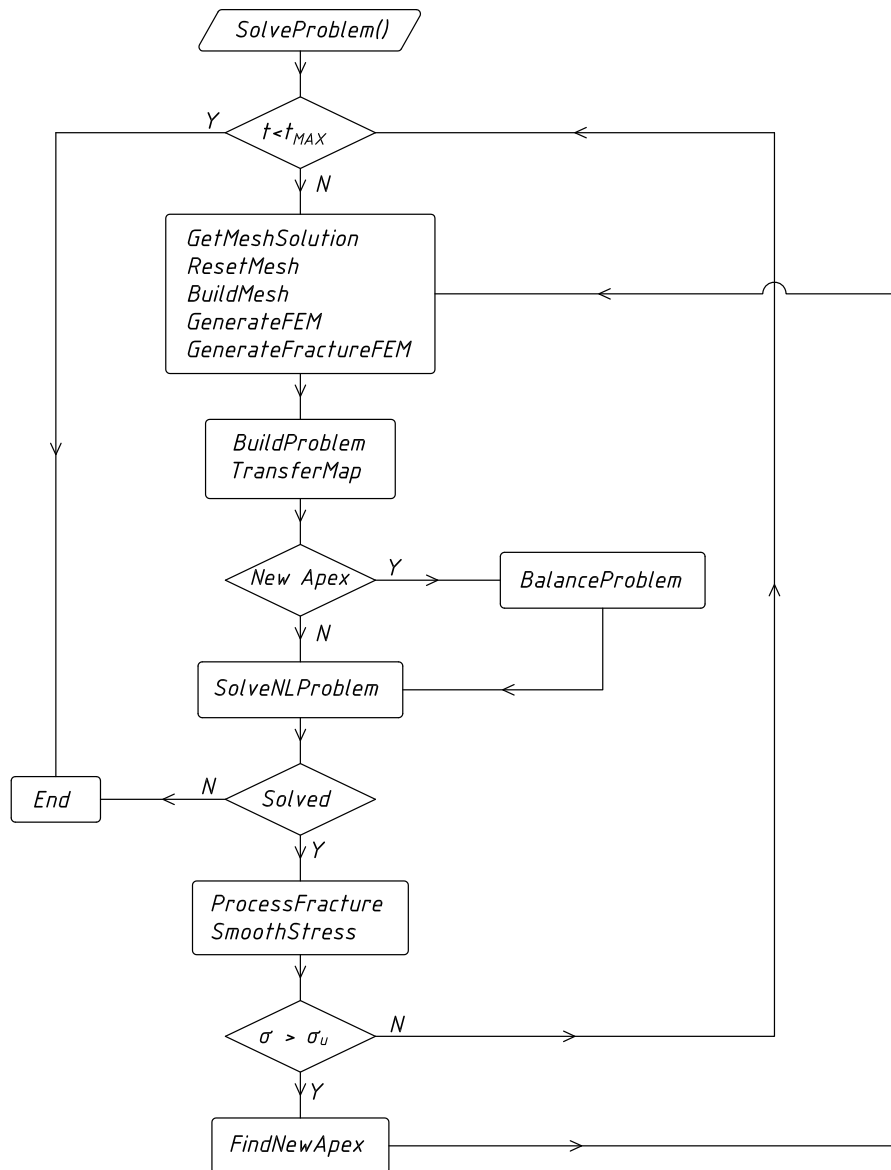
```

1 bool
2 TGeoProblem::SolveProblem()
3 {
4     // Altri passaggi
5
6     // Ciclo che si occupa di avanzare la frattura nel tempo
7     do
8     { //Si salvano tutte le variabili da proiettare
9       // ed anche la vecchia Mesh
10      GetMeshSolution(m_PrevSol);
11      // Si fa il reset della mesh attuale
12      m_pModel->ResetMesh();
13      // Si costruisce una nuova mesh
14      m_pModel->BuildMesh(pModelFem);
15      // Si genera il modello FEM
16      m_pModel->GenerateFEM();
17      // Si genera il modello FEM della frattura
18      m_pModel->GenerateFractureFEM();
19      // Si costruisce il problema non lineare
20      pProblem->BuildProblem();
21      // Si proiettano le variabili sulla nuova Mesh
22      TransferMap(m_PrevSol);
23      // Si esegue il ciclo di bilanciamento
24      pProblem->BalanceProblem();
25      //Solutore Non Lineare
26      pProblem->SolveProblem();
27      //Calcolo della massima tensione principale
28      SmoothStressValue(maxStressVal);
29      // Si salvano i valori delle soluzioni precedenti
30      pProblem->UpdateValue();
31      // Condizione di apertura di nuovi punti di frattura
32      if (!m_bRemeshing && maxStressVal.Value > pProperty->
          m_SigmaFracture)

```

```
33     {  
34         FindNewApex(maxStressVal);  
35     }  
36     // Altri passaggi  
37     } while (IsAdvancing());  
38 }  
39 return true;  
40 }
```

In Figura 6.1 a fronte viene riportato un diagramma di flusso che illustra i passi seguiti nella soluzione del problema.



**Figura 6.1:** Diagramma di flusso che illustra i passi seguiti nella soluzione del problema della frattura.

### 6.3.3 Soluzione del problema non lineare

La classe `TFemNonLinearProblem` si occupa di gestire il problema non lineare dinamico e statico della propagazione della frattura e si può riassumere nelle seguenti righe di codice. Qualche passaggio in più viene fornito in Appendice A:

```

1 //Funzione che si occupa di creare di creare il problem FEM
2 // Dichiaro e inizializza vettori e matrici
3 // gradi di liberta e gradi di vincolo
4 TNonLinearProblem::BuildProblem (...) { /*Altro*/ };
5
6 // Funzione che si occupa dello step di Bilanciamento
7 // del problema tempo-dipendente
8 TNonLinearProblem::BalanceProblem (...) { /*Altro*/ };
9
10 // Solutore del problema non lineare dinamico
11 TNonLinearProblem::SolveProblem (...) { /*Altro*/ };
12
13 // Calcolo della norma
14 TNonLinearProblem::ResidualNorm (...) { /*Altro*/ };
15
16 // Salvataggio delle soluzioni attuali
17 // per aggiungere la soluzione del prossimo step
18 TNonLinearProblem::UpdateValue (...) { /*Altro*/ };

```

Tutte le funzioni sono essenziali per il programma però si vuole evidenziare l'importanza della funzione `TFemNonLinearProblem::UpdateValue` che si occupa di aggiornare i valori tra gli step temporali e tra cicli Newton Raphson diversi nello stesso step temporale.

### 6.3.4 Soluzione del problema dinamico

Per affrontare il problema dinamico è stata creata la classe `TFemTimeSchema` che si occupa di selezionare lo schema di integrazione temporale in base al tipo di problema da affrontare. In particolare da questa viene ereditata una classe denominata `TFemTimeNewmark` che si occupa di gestire il problema dinamico con l'algoritmo di Newmark. Vengono proposti alcuni passaggi:

```

1 // Schema coerente con l'algoritmo GN22 di
2 // O.C. Zienkiewicz
3
4 // Funzione che si occupa di Calcolare i Parametri del
5 // Newmark
6 TFemTimeNewmark::ComputeNewmarkParameters() { /*Altro*/ };
7
8 // Impone il tipo di smorzamento
9 TFemTimeNewmark::SetDamping (...) { /*Altro*/ };
10
11 //Costruisce la matrice locale tangente del problema
12 TFemTimeNewmark::BuildTangent (...) { /*Altro*/ };
13
14 // Assembla la matrice globale tangente del problema
15 TFemTimeNewmark::AssemblyTangentMatrix (...) { /*Altro*/ };
16
17 // Impone le condizioni iniziali del sistema
18 TFemTimeNewmark::SetInitialConditions (...) { /*Altro*/ };

```

```
19 // Altro
20
21 // Aggiornamento degli Spostamenti
22 TFemTimeNewmark::UpdateDisplacement (...) { /* Altro */ };
23 // Aggiornamento delle Velocita
24 TFemTimeNewmark::UpdateVelocity (...) { /* Altro */ };
25 // Aggiornamento delle Accelerazioni
26 TFemTimeNewmark::UpdateAceleration (...) { /* Altro */ };
27
28 // Altro
```

## 6.4 Nucleazione e propagazione della frattura

L'evoluzione della frattura comporta un continuo cambiamento della topologia del dominio accompagnata da un aggiornamento delle condizioni al contorno. Mentre i cambiamenti topologici sono gestiti tramite l'algoritmo di remeshing descritto nel 6.6.1 a pagina 68, le condizioni al contorno sono trattate per mezzo della tecnica dei moltiplicatori di Lagrange [50].

Quando viene modellata l'apertura della frattura e la zona di processo, vengono utilizzati elementi finiti con forze coesive per rappresentare correttamente la fase solida e in grado di rappresentare in maniera completa la fase fluida. Gli elementi finiti posti lungo il percorso della frattura possono essere visti come speciali elementi finiti di contatto non-lineari, capaci di rappresentare il modo I e II di frattura, e di integrare l'equazione di continuità lungo il percorso della frattura. Questo non è noto a priori, ma è calcolato e aggiornato in maniera incrementale sulla base del campo di tensione, e rappresenta una delle incognite del problema.

In presenza di domini non omogenei, in un generico step delle soluzione e in presenza di condizioni piane di tensione o deformazione, vengono calcolati i campi di spostamento e pressione e derivati quelli di deformazione e tensione. Questi ultimi, dato che viene impiegata un'approssimazione lineare per rappresentare spostamenti e pressioni, saranno costanti all'interno di ogni elemento. I valori nodali di tensione sono calcolati applicando una procedura di *smoothing* solo su elementi dello stesso materiale, quindi possono essere presenti discontinuità solo su nodi giacenti sull'interfaccia che separa due regioni omogenee. Dato che le maglie della mesh non sono strutturate, ogni nodo può essere condiviso, in generale con  $n$  con  $n_m$  differenti proprietà di materiale. Se la frattura si muove all'interno di una sotto-regione omogenea, la nucleazione e la propagazione si sviluppano in accordo al modello coesivo proposto da Barenblatt e Dugdale [4][14]. La nucleazione della frattura si verifica quando il valore della tensione principale di trazione raggiunge il valore della resistenza a trazione del materiale,  $\sigma_t$ ; la direzione di propagazione è ortogonale a quella della tensione principale. Le labbra della frattura interagiscono e scambiano forze fintanto che non viene superato il valore critico

di apertura,  $w_{cr}$ . Una volta che questo valore è superato, le sole forze che i due lati della frattura possono scambiarsi sono di compressione, quando la frattura è richiusa e si ha contatto. La presenza di fluido viene gestita semplicemente utilizzando il concetto di tensioni efficaci. Quando il percorso frattura avvicina ad un'interfaccia tra due materiali diversi, la direzione di propagazione è diversa dal caso precedente e sarà dettata dal massimo rilascio di energia di frattura.

## 6.5 Sdoppiamento dei nodi

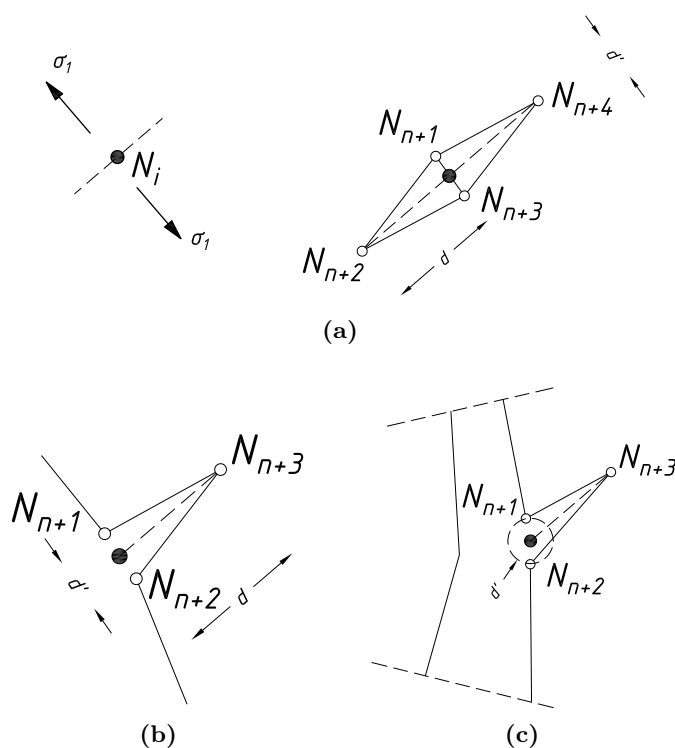
A seguito dell'avanzamento della frattura si rende necessario cambiare il dominio, e di conseguenza la mesh di elementi finiti. La procedura consiste nel trovare il nodo  $N_i$  con la tensione principale di trazione più alta tramite una procedura di smoothing nodale <sup>1</sup>. Se viene superato il valore limite del materiale una nuova frattura viene creata, o, nel caso in cui ci sia una frattura esistente, essa viene fatta avanzare. Se si verifica la prima situazione, sono possibili i seguenti casi:

- *Il nodo  $N_i$  è interno al dominio.* Il nodo esistente viene eliminato e quattro nuovi nodi sono inseriti al suo posto. Contemporaneamente vengono inseriti anche 4 nuovi bordi (Figura 6.2a). Il passo di avanzamento di prova è definito *a priori* dall'utente (parametro `IsAdvancing`). Se la nuova frattura avanza portando l'apice  $N_{n+2}$  o  $N_{n+4}$  in un intorno circolare di raggio  $d_{zip}$  di un nodo esistente, l'apice è riposizionato nel corrispondente nodo esistente. La distanza  $d'$  tra le due labbra della frattura è inizialmente assunta lievemente maggiore rispetto a  $d_{zip}$  per evitare l'eliminazione di uno dei nuovi nodi  $N_{n+1}$  o  $N_{n+3}$ . Le assunzioni introdotte per le dimensioni longitudinali e trasversali della frattura sono scelte meramente operative e non influenzano l'evoluzione del fenomeno, che è governato solamente dalle equazioni di equilibrio.
- *Il nodo  $N_i$  appartiene al contorno (esterno).* Una volta che il nodo esistente è eliminato, tre nuovi nodi e tre nuovi bordi vengono creati (Figura 6.2b). La direzione di propagazione è determinata in modo che sia garantita la formazione di un contorno convesso.
- *Il nodo  $N_i$  appartiene al contorno di una frattura esistente esterna alla zona di processo*(Figura 6.2c). Sono adottate le medesime scelte operative del caso precedente.
- *Il nodo  $N_i$  appartiene alla zona di processo*(Figura 6.2c). Sono adottate le medesime scelte operative dei casi precedenti, eccetto per la zona di

---

<sup>1</sup>estrapolazione dei valori di tensione dai punti Gauss ai nodi dell'elemento finito mediante funzioni di forma





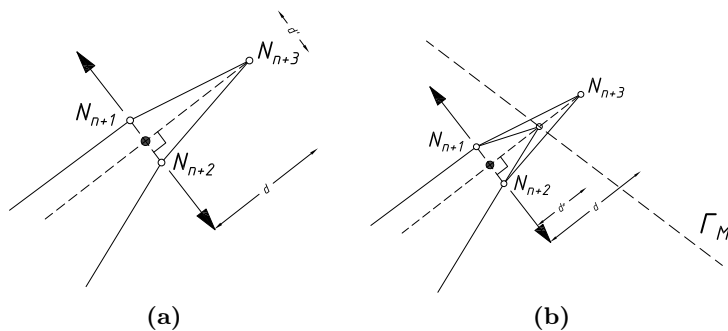
**Figura 6.2:** Formazione di una nuova frattura.

processo, in cui vengono applicate le forze coesive lungo la frattura esistente e lungo la nuova frattura appena formata;

Nel caso in cui una frattura esistente venga propagata, sono possibili i seguenti casi:

- *La frattura si propaga all'interno di una regione omogenea.* La direzione di avanzamento è assunta in modo tale da garantire la concavità del nuovo contorno. Se sono possibili alternative, la direzione di avanzamento è scelta in modo che sia garantita la massima dissipazione energetica sotto forma di energia di frattura. La frattura avanza ad un passo di lunghezza  $d$  definito dall'utente; dall'apice precedente il nuovo nodo  $N_{n+3}$  è inserito ad una distanza  $d$  (Figura 6.3a) mentre gli altri due nodi  $N_{n+1}$  e  $N_{n+2}$  sono posizionati di conseguenza.
- *La frattura avanza di una quantità predeterminata  $d$  ed il nuovo nodo attraversa un'interfaccia<sup>2</sup> o va a posizionarsi esternamente al dominio,  $\Gamma_M$  (Figura 6.3b).* La lunghezza di avanzamento è ridotta a  $d^*$ , distanza tra il nodo  $N_i$  e l'interfaccia o il contorno lungo la direzione di

<sup>2</sup>Nel senso generale del termine, un'interfaccia è il punto, l'area o la superficie sulla quale due entità qualitativamente differenti si incontrano.



**Figura 6.3:** Modi di avanzamento per fratture esistenti.

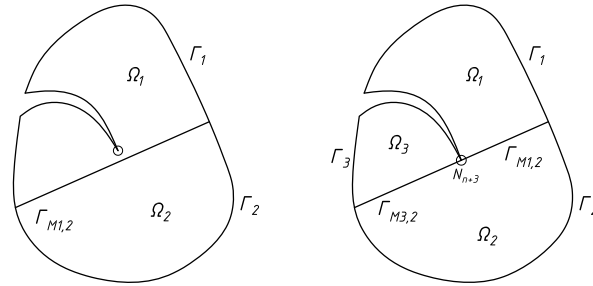
propagazione. Le entità geometriche che descrivono  $\Gamma_M$  sono quindi aggiornate, tenendo in considerazione l'inserimento di un nuovo nodo. Questo step può portare alla formazione di nuove regioni e contorni (Figura 6.4).

- *Il nodo  $N_i$  fa parte dell'interfaccia.* La frattura può propagarsi all'interno di una delle regioni adiacenti o lungo l'interfaccia. Anche in questo caso, la scelta è dettata dalla massima energia di frattura. Se si verifica il primo caso a causa della continuità del campo di spostamenti e isotropia assunto nei componenti, le direzioni principali di tensione sono coincidenti nei diversi materiali. Quindi, la direzione della propagazione della frattura può essere calcolata usando la medesima procedura vista nei casi precedenti. Una volta che il nuovo nodo apice è determinato, vengono calcolate le forze coesive per mezzo della relazione costitutiva del materiale della regione contenente il nuovo nodo. Quando l'angolo tra uno dei nuovi bordi e l'interfaccia è leggermente più piccolo di un valore prefissato dall'utente, il nuovo nodo viene proiettato sull'interfaccia,  $\Gamma_M$  e la procedura opera come nel caso di propagazione della frattura lungo un'interfaccia. In quest'ultimo caso, le forze tangenziali sono solitamente presenti tra le labbra della frattura.

## 6.6 Generazione della mesh e remeshing

La discretizzazione del dominio spaziale gioca un ruolo fondamentale nel metodo degli elementi finiti poiché la soluzione può dipendere da essa. Nel contesto degli algoritmi *h-adattivi*<sup>3</sup> basati su misure di errore a posteriori, un efficiente *mesh generator* gioca un ruolo molto importante per rappresentare in maniera fedele fenomeni in cui il dominio cambia con l'avanzamento

<sup>3</sup>infittimento/ingrossamento locale della mesh.

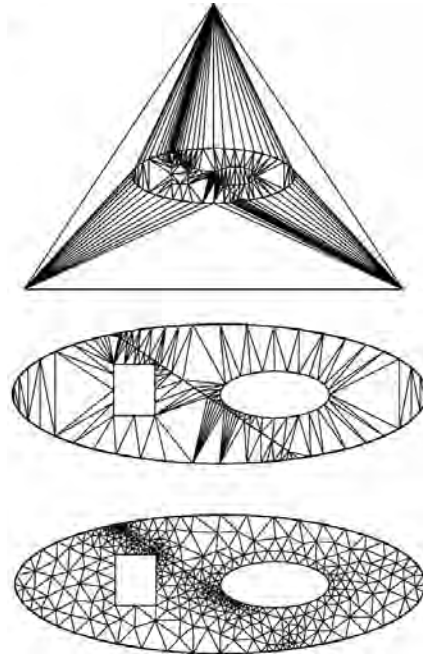


**Figura 6.4:** Propagazione della frattura con cambiamenti nella topologia del dominio.

del fenomeno. In questi casi sono spesso utilizzate mesh non strutturate per avere una soluzione indipendente dalla discretizzazione del dominio.

Nel codice di calcolo impiegato è implementato un generatore automatico di mesh 2D basato sulla triangolazione di Delaunay. Verranno di seguito esposti i passi principali seguiti dall'algoritmo implementato per la generazione della mesh. Per i dettagli si rimanda a [39].

- Il dominio da discretizzare viene definito dall'utente utilizzando una serie di nodi, curve e poligoni piani. Per ogni nodo e lungo ogni curva del contorno, una funzione detta *funzione spaziatura*  $\mathfrak{S}(x, y)$  deve essere assegnata in modo tale che si possa definire la massima lunghezza dei lati dei triangoli finali. La compatibilità della funzione spaziatura con le informazioni inserite dall'utente è automaticamente verificata e, se necessario, corretta. Possono essere inserite informazioni aggiuntive come regioni interne, punti di singolarità in cui è necessario infittire la mesh etc. In questo passaggio, il ruolo dell'utente è fondamentale.
- viene assunto un dominio ausiliario convesso di forma triangolare che contiene la regione da meshare.
- il dominio ausiliario viene triangolato inserendo dei punti nodali sul contorno secondo l'algoritmo di *Bowyer-Watson*. Questo risultato è ottenuto utilizzando il minimo numero di suddivisioni capaci di rappresentare le informazioni disponibili e di produrre una triangolazione del contorno. La procedura impiegata garantisce che tutti i triangoli soddisfino il criterio di Delaunay ma non, in generale, i requisiti della funzione spaziatura. Tutti i triangoli che non soddisfano la funzione spaziatura vengono inclusi ed ordinati (secondo la qualità, in maniera crescente) in una lista.
- in questa fase la rappresentazione completa non è garantita, poichè la triangolazione di Delaunay non è in grado di rappresentare le varietà di forme geometriche che fanno parte del contorno di un corpo. Una volta che la triangolazione del contorno è ottenuta, l'algoritmo controlla se è



**Figura 6.5:** Meshatura di una regione, esempio preso da [39].

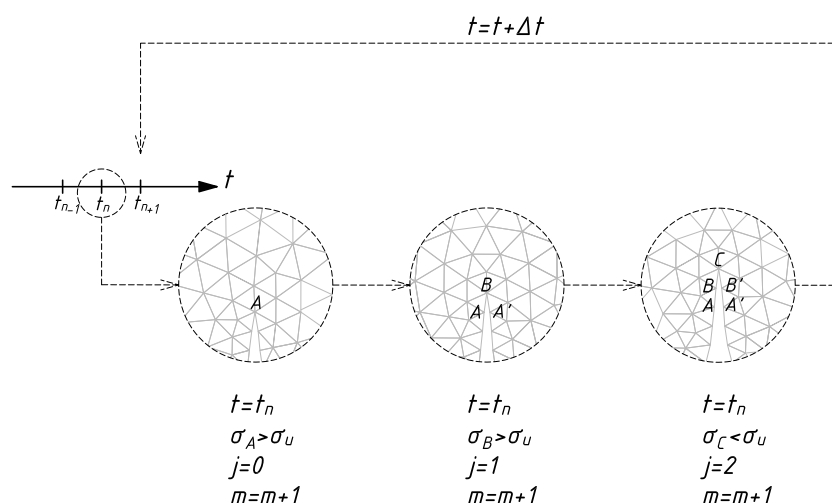
preservata l'integrità. In caso contrario, i vertici sono spostati lungo il contorno mancante e poi vengono triangolati. In questo caso, i valori della funzione spaziatrice sono cambiati fino ad ottenere un poligono chiuso che rappresenta il contorno.

- tutti i triangoli esterni al dominio vengono eliminati;
- qualora i requisiti imposti dalla funzione spaziatrice non siano soddisfatti, vengono inseriti nuovi vertici nei triangoli peggiori. La decisione del luogo in cui devono essere posti i punti dipende dalla lista sopra elencata. Quando la lista è vuota la procedura è interrotta.
- la qualità della mesh risultante è migliorata attraverso una serie di operazioni di pulizia topologica.

### 6.6.1 Remeshing

## 6.7 Proiezione delle variabili

La continua evoluzione del fenomeno nel tempo richiede un aggiornamento del dominio e della mesh; quest'ultimo, inteso sia come costruzione di una nuova mesh sul nuovo dominio, sia come infittimento dove si hanno forti



**Figura 6.6:** Avanzamento multiplo della frattura all'interno del medesimo passo temporale.

gradienti di tensione o deformazione. Poichè l'algoritmo di integrazione temporale ha bisogno, per il calcolo delle quantità al tempo corrente  $t_{n+1}$ , del vettore della soluzione al passo precedente  $t_n$  (chiamato a titolo esemplificativo  $\mathbf{V}_n$ ) è richiesta una *proiezione* delle variabili (spostamenti, forze ecc.) sulla nuova configurazione che può non solo avere una mesh diversa ma un dominio diverso qualora la frattura sia in avanzamento.

Nascono quindi degli interrogativi a cui si deve rispondere prima di procedere [34]:

- Come trasferire le variabili in modo consistente in modo da rispettare le equazioni di equilibrio;
- Come rendere tra di loro le variabili consistenti;
- Minimizzare l'errore numerico.

È da notare che la proiezione dei valori da una configurazione ad un'altra è essenziale per ogni algoritmo tempo-dipendente che preveda l'evoluzione della mesh.

In letteratura i procedimenti adattivi sono ben noti, e hanno delle solide basi, sia per quanto riguarda l'infittimento della mesh, sia per la stima e il controllo degli errori grazie a contributi e molti autori tra cui si cita Zienkiewicz, [48]. Al contrario non esiste in letteratura una solida strategia di adattamento per quando riguarda i problemi tempo-dipendenti ma sono presenti delle proposte. Inoltre la strategia di proiezione deve tenere in conto che la mesh è non strutturata. Alcuni contributi in questo senso, molti importanti, sono contenuti in [29] ed in [32].

### 6.7.1 La mappa di proiezione consistente

Un operatore che proietti le variabili sulla nuova configurazione deve soddisfare i seguenti requisiti:

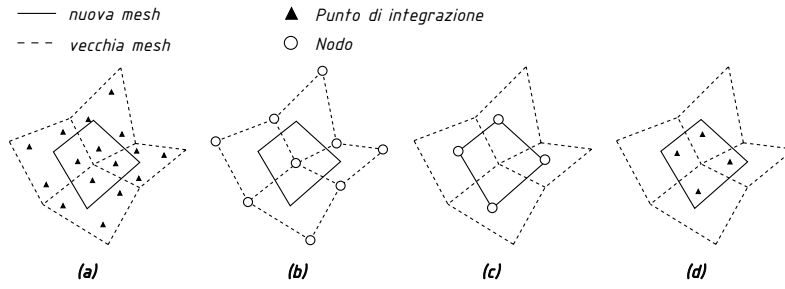
- Consistenza con le equazioni costitutive;
- Consistenza tra le variabili: per esempio tra forze e spostamenti;
- Soddisfare equilibrio statico o dinamico;
- Compatibilità con le condizioni al contorno e con l'evoluzione delle stesse.
- Minimizzazione dell'errore numerico;

Si consideri la soluzione sul dominio iniziale  $\Omega_0$  calcolata sulla mesh iniziale  $\mathcal{M}_0$  per mezzo di una delle equazioni risolutive introdotte nei capitoli precedenti. Ricordando che con  $m$  vengono indicati il numero di chiamate al solutore non lineare, all'iterazione  $m+1$ , la funzione  $\mathcal{F}(m)$  si occupa di restituire il tempo corrispondente ad ogni quantità di indice  $m$ , vale  $t = \mathcal{F}(m)$ . Se si raggiungono le condizioni per le quali una nuova frattura si crea o una frattura esistente si propaga, sono richiesti dei cambiamenti topologici del dominio  $\Omega_m \equiv \Omega_0$  e del suo contorno, accompagnati dall'introduzione di uno o più nodi.

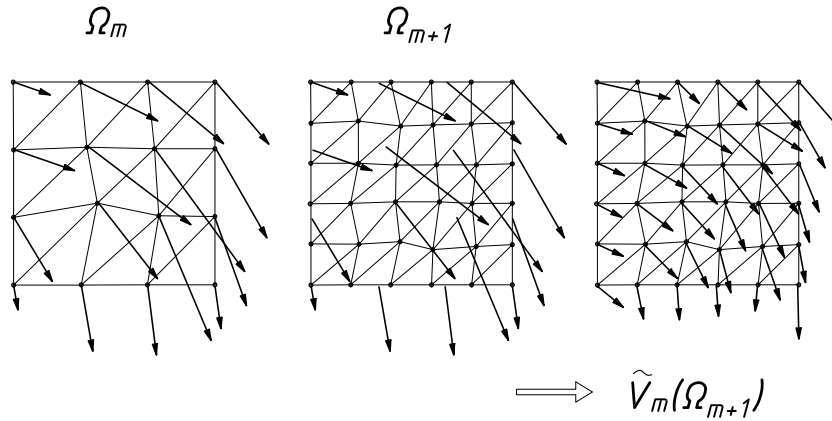
L'operatore di proiezione nel caso in esame affronta due situazioni diverse:

- La prima situazione è collegata al remeshing puro considerando come statico il dominio. In questo caso si ha solo un possibile errore dovuto alla proiezione delle variabili. Solitamente questa procedura non provoca errori pesanti, controllabili facilmente con un nuovo step di ricalcolo delle variabili interne;
- La seconda situazione è l'inserimento di un segmento di frattura, cioè lo split in due di un nodo dove è soddisfatto il criterio di rottura. L'avanzamento della frattura provoca la situazione di maggior criticità. In aggiunta ai possibili errori collegati alla proiezione delle variabili si ha anche una variazione del dominio e delle condizioni al contorno. Questa evoluzione può provocare un grande sbilanciamento tra le forze esterne ed interne e può portare alla stop della procedura iterativa.

Una discretizzazione del nuovo dominio  $\Omega_{m+1}$  viene quindi creata e una nuova mesh  $\mathcal{M}_{m+1}$  viene generata. La soluzione delle equazioni di equilibrio statico o dinamico sono quindi da ricercare sulla nuova mesh appena generata, tenendo in considerazione il fatto che il vettore  $\mathbf{V}_n$  è noto sulla mesh all'iterazione precedente  $\mathcal{M}_m$ . In generale, utilizzare operatori di trasferimento basati sulle sole funzioni di forma non garantisce il soddisfacimento



**Figura 6.7:** Trasporto: (a) punti Gauss dalla vecchia mesh, (b) nodi dalla vecchia mesh, (c) nodi sulla nuova mesh, (d) punti Gauss sulla nuova mesh.



**Figura 6.8:** Algoritmo di proiezione delle forze nodali: da sinistra verso destra si hanno le forze nodali al tempo  $n$  sulla mesh  $m$ ; le forze nodali al tempo  $n$  sulla mesh  $m + 1$  riferite ai nodi precedenti ed infine le forze nodali al tempo  $n$  sulla mesh  $m + 1$  riferite ai nodi di quest'ultima.

delle equazioni di bilancio. La mappa può essere scritta come:

$$\tilde{V}_m(\Omega_{m+1}) = \mathfrak{N}(\mathbf{V}_m(\Omega_m)) \quad (6.1)$$

in cui  $\mathfrak{N}$  è un operatore di trasferimento costruito utilizzando le funzioni di forma degli elementi costituenti la mesh (si veda la Figura 6.8).

Il valore delle variabili da proiettare è noto solo in punti discreti del dominio sulla vecchia mesh, il trasferimento su nuovi punti di un nuovo dominio porta sicuramente a degli errori che possono compromettere la soluzione. Un altro aspetto fondamentale è quello della consistenza delle variabili passate, esse devono soddisfare le equazioni di equilibrio anche se il dominio è cambiato a causa della propagazione della frattura. Inoltre la perdita di consistenza può essere dovuta al fatto che la proiezione delle variabili avviene in modo lineare mentre il legame tra i termini del problema è non lineare.

Se queste condizioni di proiezione non sono soddisfatte allora la convergenza dello step successivo alla proiezione non può essere soddisfatta.

Per minimizzare gli errori è bene passare il minor numero possibile di variabili ricalcolando le altre una volta che quelle proiettate sono consistenti. Quindi le variabili da proiettare possono essere suddivise in:

- *Variabili interpolate*: da proiettare direttamente tramite l'operatore di proiezione. Nel caso in esame sono tutte le forze: elastiche, inerzia, smorzamento ed esterne;
- *Variabili derivate*: ricalcolate da quelle proiettate in modo che tutto sia consistente. Nel caso in esame essi sono spostamenti, velocità ed accelerazioni.

La consistenza in questo caso deve essere intesa come una corretta mutua dipendenza tra le variabili in gioco con rispetto delle equazioni costitutive.

Nella proiezione utilizzata nel presente lavoro si assicura sempre l'equilibrio in forze (elastiche, inerzia, smorzamento, esterne) mentre quella delle variabili derivate non può essere assicurata, per come è costruito l'algoritmo di proiezione.

Il vettore di spostamenti al passo  $m$ ,  $(\mathbf{u}_m)$  è ricalcolato sul dominio  $\Omega_m$  usando la nuova mesh  $\mathcal{M}_{m+1}$ . È necessario ricordare che il dominio  $\Omega_{m+1}$  presenta condizioni al contorno diverse, cioè nucleazione/avanzamento della frattura e nodi che si sdoppiano lungo il percorso della frattura in avanzamento, in luogo di una mesh precedente continua in questa zona.

Il remeshing e quindi la proiezione dei valori è un'operazione computazionalmente dispendiosa. Nell'algoritmo in esame il remeshing viene fatto ogni qual volta si abbiano elevati gradienti di tensione o deformazione attraverso il criterio di Zhu-Zienkiewicz e ogni qual volta c'è un'evoluzione del dominio. In particolare il remeshing con la tecnica di Delaunay permettere un agevole infittimento locale.

### 6.7.2 Algoritmo di proiezione

L'operatore di proiezione può essere costituito in due modi:

- Trasferimento diretto dei valori dei punti Gauss della mesh precedente  $\mathcal{M}$  sulla nuova mesh  $\mathcal{M}_{+1}$ . Si tengono costanti i valori relativi all'area associata ad ogni punto Gauss;
- Costruire una soluzione che è continua. Essa consiste nel soddisfare la forma debole:

$$\int_{\Omega} \mathfrak{N}[\mathbf{V}_{n,N}^{\mathcal{M}_{+1}} - \mathbf{V}_{n,N}^{\mathcal{M}}] d\Omega = 0 \quad (6.2)$$

Nel caso in esame si è utilizzato il secondo approccio.

La scelta dell'operatore di trasferimento  $\mathfrak{N}$  invece non è banale. Il trasferimento avviene solitamente facendo coincidere l'operatore di proiezione con

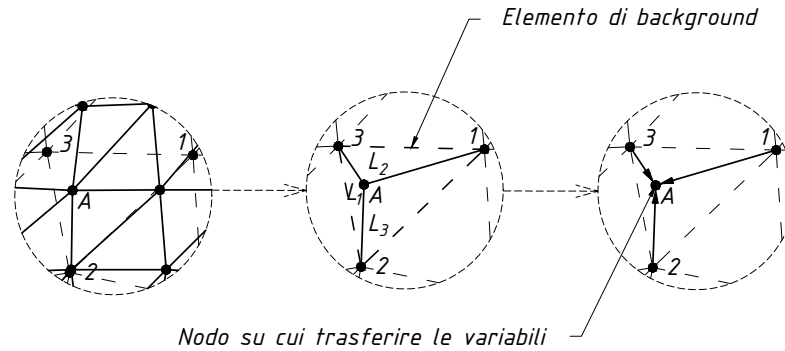


le funzioni di forma. Una volta noti i valori nodali della mesh precedente si devono portare sui nuovi nodi della mesh attuale utilizzando per questo scopo le funzioni di forma. Nel caso in esame si calcola l'area nodale di ogni nodo sulla nuova mesh  $\mathcal{M} + 1$  e viene trovato il cosiddetto *elemento di background* in cui esso è contenuto nella vecchia mesh  $\mathcal{M}$ . La mesh è infatti composta da triangoli CST e solitamente anche per domini discretizzati con elementi di ordine superiore il dominio per la proiezione viene suddiviso in triangoli perchè la meshatura con elementi triangolari può essere fatta agevolmente in modo automatico ed il calcolo dell'area nodale è banale. Successivamente l'area nodale viene utilizzata come funzione di forma.

La procedura può essere riassunta nel modo seguente:

- Nel primo step i valori calcolati nei punti Gauss si proiettano i valori delle variabili  $\mathbf{V}_{n,G}^{\mathcal{M}}$  da proiettare ai nodi  $\mathbf{V}_{n,N}^{\mathcal{M}}$ . Per i triangoli CST si ha banalmente  $\mathbf{V}_{n,G}^{\mathcal{M}} = \mathbf{V}_{n,N}^{\mathcal{M}}$ . Generalmente comunque si utilizzano funzioni di forma lineari per questa interpolazione.
- Nel secondo step si proiettano i valori nodali dalla vecchia mesh  $\mathbf{V}_{n,N}^{\mathcal{M}}$  sulla nuova mesh  $\mathbf{V}_{n,N}^{\mathcal{M}+1}$ . Questo passaggio è il più complesso e viene suddiviso ulteriormente nei seguenti substep:
  - Per ogni nodo  $A$  sulla nuova mesh  $\mathcal{M} + 1$  con coordinate note  $\mathbf{x}_{n,N}^{\mathcal{M}+1}$  viene trovato l'*elemento di background* della mesh precedente  $\mathcal{M}$ . Questo elemento  $\Omega_{(e)}^{\mathcal{M}}$  è tale per cui vale  $\mathbf{x}_{n,N}^{\mathcal{M}+1} \in \Omega_{(e)}^{\mathcal{M}}$ , cioè si deve trovare l'elemento a cui appartiene il nodo della nuova mesh nella vecchia mesh. Sarebbe possibile anche fare la procedura contraria, avendo però dei nodi in cui non è proiettata alcuna variabile;
  - Si trovano le coordinate  $(x_A^{\mathcal{M}+1}, y_A^{\mathcal{M}+1})$  del nodo  $A$  della nuova mesh;
  - Si individua in quale triangolo,  $\mathcal{M} + 1$  (*elemento di background*  $\Omega_{(e)}^{\mathcal{M}}$ ) sia contenuto il nodo  $A$  e note le coordinate di  $A$  e dei nodi del triangolo di background vengono calcolate le funzioni di forma  $N_i = L_i = A_i/A$ .
- Nel terzo step una volta noti i valori delle funzioni di forma vengono proiettati i valori necessari.

I valori così ricavati sono *meccanicamente equivalenti* poichè i valori nodali che al passo precedente erano in equilibrio sono in equilibrio anche sul nuovo dominio/mesh perchè la proiezione avviene scalarmente utilizzando un *coefficiente di mapping* che moltiplica ogni entrata del vettore della variabili da proiettare e non avviene un passaggio in senso vettoriale come per



**Figura 6.9:** Fasi del funzionamento dell’algoritmo di proiezione.

esempio in [34]. Il calcolo valore del *coefficiente di mapping* avviene tramite le funzioni di forma, in pratica coincide con  $L_i$  dove  $i$  è la  $i$ -esima funzione di forma.

### 6.7.3 Step di bilanciamento

Come spiegato sopra, le variabili derivate (spostamenti, velocità ed accelerazioni) non possono essere considerate consistenti in quanto l’algoritmo di proiezione garantisce solamente l’equivalenza meccanica delle forze.

Un ulteriore inconveniente che entra in gioco è l’avanzamento della frattura. Essa porta ad una variazione del dominio e laddove si aveva una continuità ora è presente una zona coesiva in cui agiscono delle forze di richiusura. Quindi, oltre al dominio sono variate anche le condizioni al contorno. Da un punto di vista meccanico si ha che la continuità viene ripristinata dalle forze coesive che vengono rilasciate, mediate il ciclo di bilanciamento.

Per ovviare a queste problematiche si è impostato un *ciclo di bilanciamento*. L’idea è quella di proiettare tutte le forze interne ed esterne, aggiornare quelle esterne con le nuove condizioni al contorno, ottenere un residuo (dato esclusivamente dalle nuove condizioni al contorno) e con un ciclo Newton Raphson annullarlo. L’entrata nel ciclo Newton Raphson è fatta utilizzando le forze interne precedenti e le forze esterne precedenti però con le condizioni al contorno aggiornate al passo attuale.

Inoltre, dato che le variabili cinematiche sono intrinsecamente connesse tra loro dall’algoritmo di Newmark allora si può vedere questo step come uno step di aggiornamento. Si proiettano allora anche le variabili cinematiche e si correggono con le seguenti formulazioni ricavate da semplici manipolazioni delle formulazioni del Newmark che si omettono per semplicità:

- Spostamenti:

$$\mathbf{x}_{t+1}^{i+1} = \Delta \mathbf{x}_{t+1}^{i+1} + \mathbf{x}_{t+1}^i \quad (6.3)$$

- Accelerazioni:

$$\ddot{\mathbf{x}}_{t+1}^{i+1} = \Delta \ddot{\mathbf{x}}_{t+1}^{i+1} + \ddot{\mathbf{x}}_{t+1}^i \quad (6.4)$$

- Velocità:

$$\dot{\mathbf{x}}_{t+1}^{i+1} = \Delta \dot{\mathbf{x}}_{t+1}^{i+1} + \dot{\mathbf{x}}_{t+1}^i \quad (6.5)$$

La correzioni via via annulleranno il residuo ottenuto ad ogni iterazione. Quindi questo step in pratica si può vedere come uno step di aggiornamento delle variabili cinematiche.

Da un punto di vista del codice la funzione che si occupa di gestire il ciclo di bilanciamento è la funzione `TNonLinearProblem::BalanceProblem`.

Alcuni passaggi salienti del codice:

```

1  bool
2  TNonLinearProblem::BalanceProblem()
3  {
4      //Impostazione dei parametri di Newmark
5
6      m_pNewmark->ComputeNewmarkParameters();
7      //Calcola la Matrice Tangente
8      m_pNewmark->AssemblyTangentMatrix(*m_pGlobalTangent);
9      // Aggiorna l'intensita delle forze coesive che dipendono
10     dagli spostamenti
11     UpdateFractureForce(m_PrevSol);
12     //Forma il Vettore dei Termini Noti
13     ArrangeContribution(m_NumIter, m_pNewmark);
14     // Si scompone mediante decomposizione LDL e
15     // e si inverte la matrice calcolando
16     m_pGlobalTangent->Solve(m_PrevDeltaSol, e_LDL)
17     //Aggiorna il valore degli spostamenti nel loop
18     Sum(m_PrevDeltaSol, &m_PrevSol);
19     // Aggiorna il valore delle accelerazioni nel loop
20     m_pNewmark->BalanceAcceleration(m_PrevDeltaSol,
21     m_PrevASol);
22     // Aggiorna il valore delle velocita nel loop
23     m_pNewmark->BalanceVelocity(m_PrevDeltaSol, m_PrevVSol);
24
25     // Altri passaggi intermedi
26
27     // Calcolo della norma del Residuo
28     res = ResidualNorm(m_Residual, m_PrevExtForce);
29
30     } while (res > m_ResToll);
31
32     return true;
33 }

```

## 6.8 Ciclo full Newton

La spiegazione matematica del ciclo Newton Raphson è stata già fornita in 4.4.1. Si vuole dare ora una breve spiegazione del suo funzionamento.

Il ciclo principale che si occupa di risolvere il problema non lineare è:

```

1  bool
2  TNonLinearProblem::SolveProblem()
3  {
4      do
5      {
6          if (m_NumIteration == 0)
7          {
8              if (m_CurrTime == m_IniTime)
9              {
10             }
11             else
12             {
13                 _____
14                 // Ciclo Iniziale Newton
15                 _____
16                 ResNormInit = Norm(...);
17             }
18         }
19         _____
20         // Il Solutore calcola la soluzione
21         m_DeltaSol.Copy(m_Residual);
22         if (!m_pGlobalTangent->Solve(m_DeltaSol, e_LDL))
23         {
24             return false;
25         }
26         // Aggiorna il Valore degli spostamenti nel loop
27         m_pNewmark->UpdateDisplacement(...);
28         // Aggiorna il valore delle accelerazioni nel loop
29         m_pNewmark->UpdateAceleration(...);
30         // Aggiorna il valore delle velocita nel loop
31         m_pNewmark->UpdateVelocity(...);
32
33         // Ricalcola le Forze di Elastiche: Calcolo nel locale ,
34         // Assemblaggio elementi( K x spostamenti)
35         m_pNewmark->ComputeElasticForce(...);
36         // Ricalcola le Forze di Damping: Calcolo nel locale ,
37         // Assemblaggio elementi( C x velocita)
38         m_pNewmark->ComputeDampingForce(...);
39         // Ricalcola le Forze d'Inerzia: Calcolo nel locale ,
40         // Assemblaggio elementi( M x accelerazione)
41         m_pNewmark->ComputeInertiaForce(...);
42         // Aggiorna l'intensita delle forze coesive
43         UpdateFractureForce(m_Sol);
44         // Altro
45         // RICALCOLO DEL RESIDUO
46         _____
47         //Altro
48         res = ResidualNorm(m_Residual, m_ExtForce);

```

```

49     m_NumIteration++;
50 } while (res > m_Tolerance || m_NumIteration >
        m_NumMaxIteration);
51 return true;
52 }

```

### 6.8.1 Criteri di uscita

I criteri di uscita nel caso attuale possono essere diversi. Sono state utilizzate due combinazioni:

- La prima combinazione riguarda la norma sugli spostamenti:

$$\frac{\|\Delta d\|}{\|d\|} < \varepsilon_d \quad (6.6)$$

Dove il valore  $\|\Delta d\|$  descrive la norma sull'incremento di spostamenti mentre  $\|d\|$  è la norma sugli spostamenti totali. Il termine  $\varepsilon_d$  rappresenta un'adeguata tolleranza. Nel caso in esame la tolleranza utilizzata è pari a  $1E - 4$ . e sul residuo:

$$\frac{\|R\|}{\|R_0\|} < \varepsilon_r \quad (6.7)$$

Dove il valore  $\|R\|$  descrive la norma sul residuo finale mentre  $\|R_0\|$  è la norma sul residuo iniziale. Il termine  $\varepsilon_r$  rappresenta un'adeguata tolleranza. Nel caso in esame la tolleranza utilizzata è pari a  $1E - 5$ .

- La seconda combinazione è la norma sul residuo:

$$\frac{\|f_{res}\|}{\|f_{ext}\|} < \varepsilon_f \quad (6.8)$$

Dove il valore  $\|f_{res}\|$  descrive la norma sul residuo finale mentre  $\|f_{ext}\|$  è la norma delle forze esterne. Il termine  $\varepsilon_f$  rappresenta un'adeguata tolleranza. Nel caso in esame la tolleranza utilizzata è pari a  $1E - 5$ .

- Il tutto con una limitazione sulle massime iterazioni possibili, pari a 50. Si vuole inoltre osservare come sia stata sufficiente appena qualche iterazione per raggiungere la desiderata convergenza.

Nel corso della analisi la norma utilizzata è stata quella della combinazione due.

## 6.9 Postprocessing

Mentre il *pre-processing* avviene con un tipico file di input testuale da cui il programma legge, in base a delle *keyword*, tutti i dati di input. Il modo con cui si inseriscono i dati di input è poco importante ai fini dell'analisi mentre il *post-processing* è al contrario molto importante e allo stesso tempo deve fornire il maggior numero di informazioni da ogni singola analisi.

Per questo e per altri motivi si è utilizzata una classe, TStreamFilter, che permette di esportare un modello compatibile con il file di input del programma commerciale agli elementi finiti Straus7 da cui, utilizzando il post-processore, si possono estrapolare tutte le informazioni utili all'analisi.

# Capitolo 7

## Applicazioni

### 7.1 Introduzione

In questo capitolo saranno trattati problemi semplici per introdurre al funzionamento del codice di calcolo e alla sua capacità di predire il percorso della frattura. In particolare, saranno effettuate alcune considerazioni sulla simulazione della prova di flessione in tre punti (*Three Point Bending test*) fatta su un provino in calcestruzzo. In tutte le analisi svolte in questo capitolo, sono assunte condizioni quasi-statiche.

### 7.2 Three point bending test

Il *three point bending test* è un classico problema della meccanica dei solidi. Consiste in una trave isostatica semplicemente appoggiata a cui è applicato uno spostamento impresso. In mezzzeria, dove si ha la massima sollecitazione flettente, è applicato un intaglio in modo tale da assicurare la formazione della frattura nel modo voluto.

Si vuole indagare l'effetto scala riportato anche in Carpinteri [11]. Con *effetto di scala* si intende la dipendenza di uno o più parametri caratterizzanti il comportamento meccanico di un materiale dalla dimensione strutturale del provino. Da un punto di vista ingegneristico, il problema è di estrema importanza.

Gli errori di progettazione attribuibili alla mancata considerazione dell'effetto di scala in natura e negli artefatti sono stati particolarmente frequenti nel corso della storia e continuano ad esserlo anche nella nostra epoca, caratterizzata da alta tecnologia e uso del computer. Sebbene fosse risaputo da oltre duemila anni che i pezzi di legno sono più deboli man mano che aumentano in lunghezza, i costruttori navali del Rinascimento non riuscivano a spiegarsi come mai le loro grandi navi di legno si spezzassero sotto il

proprio peso. Nell'introduzione al suo autorevole studio sulla resistenza dei materiali, Galileo cita casi di navi spezzate e altri guasti ricorrenti nell'ingegneria rinascimentale attribuibili alla dimensione. Un esempio moderno, che peraltro ha contribuito alla nascita della meccanica della frattura stessa, è quello delle navi militari della marina USA classe Liberty. Fra il 1939 ed il 1945 furono costruite 2708 pezzi di cui, al 1 aprile 1946, 1031 riportavano danni riconducibili al fattore di scala e alla frattura fragile [25].

Non è necessario comunque considerare casi estremi, per accorgersi come la duttilità non sia solamente una caratteristica del materiale, bensì una caratteristica dell'intera struttura.

Nel caso particolare del *three point bending test* l'effetto scala consiste nell'aver un comportamento di *softening*, cioè un comportamento via via più fragile (transizione *duttile-fragile*), mano a mano che la dimensione del provino aumenta. Questo effetto è stato dimostrato sperimentalmente in [11].

Le principali peculiarità del modello sono:

- Da un punto di vista dello stato di sollecitazione si tratta di uno stato piano di deformazione *Strain2D*;
- Si utilizzano elementi finiti triangolari;
- La forzante è data da uno spostamento impresso.

L'obiettivo di queste analisi è di studiare l'evoluzione dello stato di equilibrio con l'evoluzione del carico. Tutti questi stati di equilibrio definiscono la risposta, in termini di carico-spostamento, della struttura in cui il carico applicato varia proporzionalmente in funzione di un unico parametro. In tal caso, per un sistema ad  $n$  gradi di libertà (*DoF*),  $(n + 1)$  incognite definiscono completamente il problema.

Il metodo degli elementi finiti, sulla base della meccanica del continuo, offre  $n$  equazioni algebriche che descrivono gli stati di equilibrio della struttura:

$$\mathbf{R}_{t+1}(\mathbf{x}, \boldsymbol{\lambda}) = \boldsymbol{\lambda} \mathbf{F}_{t+1} - \mathbf{A} \mathbf{x}_{t+1} = 0$$

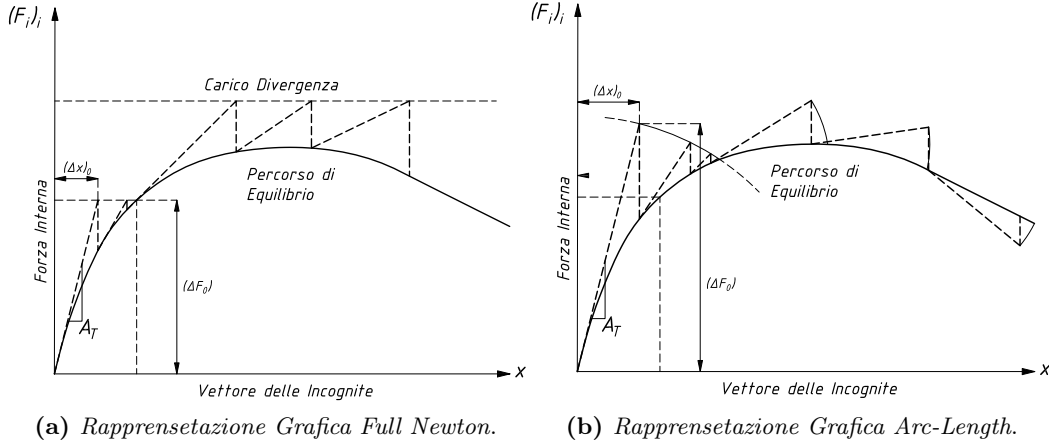
Per poter trovare la soluzione al problema si aggiunge un'ulteriore equazione per trovare la soluzione alle  $(n + 1)$  incognite avendo  $(n + 1)$  equazioni. L'imposizione di questa nuova equazione dipende dal metodo impiegato. Nel nostro caso si è impiegato il metodo dei moltiplicatori di Lagrange:

$$\mathbf{L}_m(\mathbf{u}, \boldsymbol{\lambda}) = 0$$

Le equazioni che governano la risposta della struttura sono nonlineari e richiedono una soluzione iterativa. All'iterazione corrente,  $i$ , i parametri  $\mathbf{x}$  e  $\boldsymbol{\lambda}$  vengono aggiornati:

$$\mathbf{x}_{t+1}^{i+1} = \mathbf{x}_{t+1}^i + \Delta \mathbf{x}_{t+1}^i$$





**Figura 7.1:** Concetto di modello coesivo

$$\boldsymbol{\lambda}_{t+1}^{i+1} = \boldsymbol{\lambda}_{t+1}^i + \Delta \boldsymbol{\lambda}_{t+1}^i$$

Si valutano ad ogni iterazione  $i$  gli incrementi  $\mathbf{x}_{t+1}^{i+1}$  e  $\boldsymbol{\lambda}_{t+1}^{i+1}$  tale per cui alla fine del processo iterativo si ottiene:

$$\mathbf{R}_{t+1}^{i+1}(\mathbf{u}_{t+1}^{i+1}, \boldsymbol{\lambda}_{t+1}^{i+1}) = 0$$

$$\mathbf{L}_{m_{t+1}}^{i+1}(\mathbf{u}_{t+1}^{i+1}, \boldsymbol{\lambda}_{t+1}^{i+1}) = 0$$

La valutazione degli incrementi dipende dal metodo impiegato. Inoltre queste procedure sono trattate in modo separate ad ogni iterazione  $i$ .

Per quanto riguarda il vettore che contiene anche gli spostamenti viene utilizzato il metodo *Newton-Raphson* mentre per quanto riguarda l'incremento di carico sono disponibili diversi metodi: carico controllato, spostamento controllato e metodo Arc-Length.

Nel metodo a carico controllato  $\boldsymbol{\lambda}_{t+1}^i$  è fissato (quindi  $\Delta \boldsymbol{\lambda}_{t+1}^i = 0$ ). Invece nel caso del controllo degli spostamenti si ha che  $\Delta \boldsymbol{\lambda}_{t+1}^i$  viene valutato per ogni incremento di spostamento. Nel caso in esame lo spostamento viene imposto tramite una *time-table* e  $\Delta \boldsymbol{\lambda}_{t+1}^i$  viene valutato dalla soluzione iterativa sempre con *Newton-Raphson*.

Viene invece scartato il metodo per carico imposto in quando non si riuscirebbe a cogliere il comportamento di softening. Si osserva, inoltre, dai vari grafici, che gli andamenti delle curve, nel tratto di softening, hanno un andamento al più verticale. Questo perché neanche il metodo agli spostamenti controllati riesce a cogliere in tutti i suoi aspetti il fenomeno. L'unico metodo che consente di farlo è il metodo *Arc-Lenght*.

Il metodo Arc-length è una strategia di soluzione in cui il percorso varia in base alle precedenti soluzioni convergenti ed, in qualsiasi punto, segue una

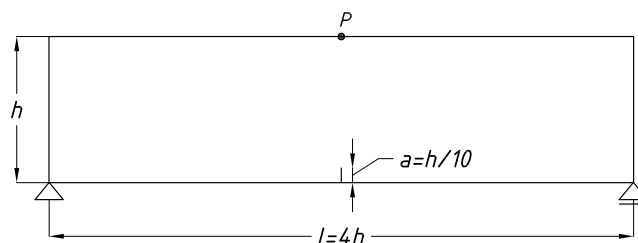
**Tabella 7.1:** Materiale e geometria del provino.

Quantità	Simbolo	Valore	Unità di misura
Altezza	$h$	100,0	$mm$
Lunghezza	$l$	400,0	$mm$
Dimensioni intaglio	$a$	10,0	$mm$
Modulo elastico	$E$	14700,0	$MPa$
Coefficiente di Poisson	$\nu$	0,2	/
Trazione coesiva massima	$\sigma_u$	2,0	$MPa$
Apertura massima ( $\sigma_u$ )	$w_{cr}$	0,04	$mm$

direzione ortogonale alla tangente della curva soluzione nota fino a quel momento. In questo metodo, sia il vettore dei carichi esterni applicati che il campo di spostamenti variano mentre nel metodo agli spostamenti controllati si è in grado di controllare solo il campo di spostamenti.

### 7.2.1 Geometria analizzata e svolgimento delle analisi

Tutti i provini analizzati rispettano i rapporti geometrici indicati in Figura 7.2. L'abbassamento viene applicato in maniera quasi-statica nel punto P, in mezzzeria, seguendo una time table prefissata. Per quanto riguarda il

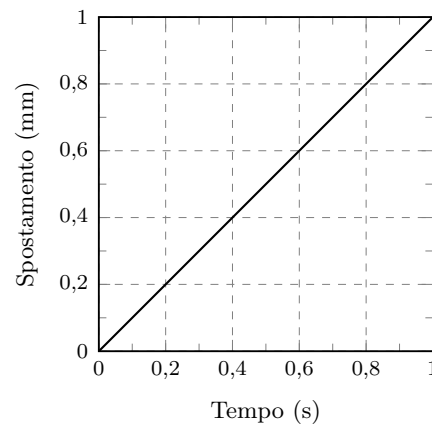
**Figura 7.2:** Rapporti geometrici dei provini analizzati.

materiale costituente il provino, e quindi i due legami costitutivi richiesti, è stato assunto calcestruzzo non armato, le cui caratteristiche sono riassunte in Tabella 7.1 assieme alla geometria del provino.

In merito alle modalità con le quali il programma segue la propagazione della frattura, si è già specificato nei precedenti capitoli, come venga effettuato un remeshing ogni qualvolta la frattura avanza.

### 7.2.2 Commenti alle analisi svolte

Dalle analisi svolte è emerso che mantenendo invariati materiale e forma geometrica, e aumentando la scala dimensionale, si riscontra una netta transizione verso il comportamento di tipo fragile, con una caduta improvvisa della capacità di carico e una propagazione rapida di fessura. Al contrario,



**Figura 7.3:** Table impiegata nelle analisi.

con provini di dimensioni relativamente modeste, si ha un comportamento duttile e una propagazione lenta della fessura. Nel caso studiato della flessione su tre punti, si può verificare la formazione di una cerniera plastica in mezzeria e l'impossibilità di separare in due parti distinte il provino con un semplice carico monotono, come si evince dalla Figura 7.4 in cui è rappresentata la forza, data dalla somma delle reazioni vincolari, all'avanzare dello spostamento del punto di controllo P per diverse scale del provino di partenza. In Figura 7.5 sono rappresentate ulteriori dimensioni del provino, per dare ulteriore risalto all'effetto scala. Si osserva dai vari grafici che gli andamenti delle curve nel tratto di softening hanno un andamento al più verticale. Questo perché neanche il metodo agli spostamenti controllati riesce a cogliere in tutti i suoi aspetti il fenomeno. L'unico metodo che consente di farlo è il metodo *Arc-Lenght*. Le tecniche di soluzione standard portano a instabilità in prossimità dei punti limite, nei problemi di snap-through e di snap-back, non riuscendo a predire la risposta al carico-spostamento completo.

Nelle Figure 7.6 e 7.7 a pagina 86 vengono riportate alcune immagini relative alla mesh e al contour plot delle tensioni sul provino in diversi istanti temporali.

### 7.3 Ulteriori considerazioni

Un altro aspetto che viene analizzato è la dipendenza dalle dimensioni della mesh. Nella Figura 7.8 a pagina 87 vengono riportati forza e abbassamento relative al provino in scala 1:1. Come si può osservare, mesh più fitte non portano a sensibili cambiamenti ma al più ad un incremento di tempo computazionale.

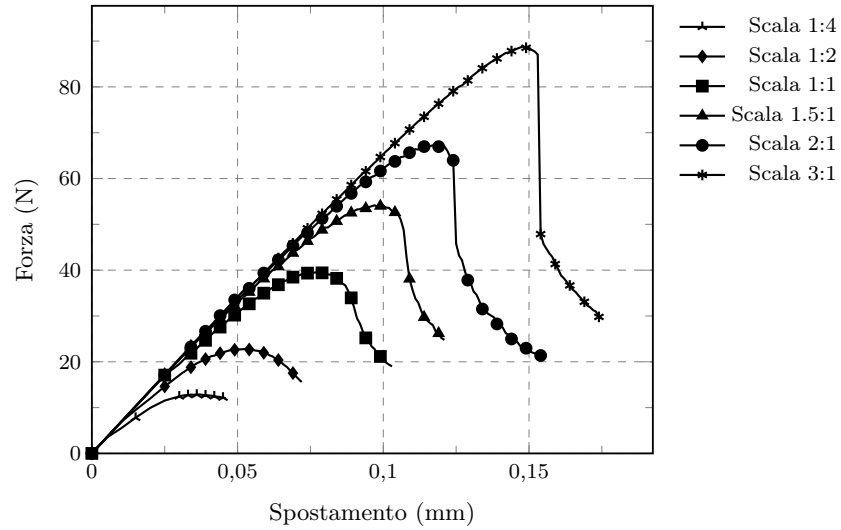


Figura 7.4: Transizione duttile-fragile all'aumentare della dimensione del provino.

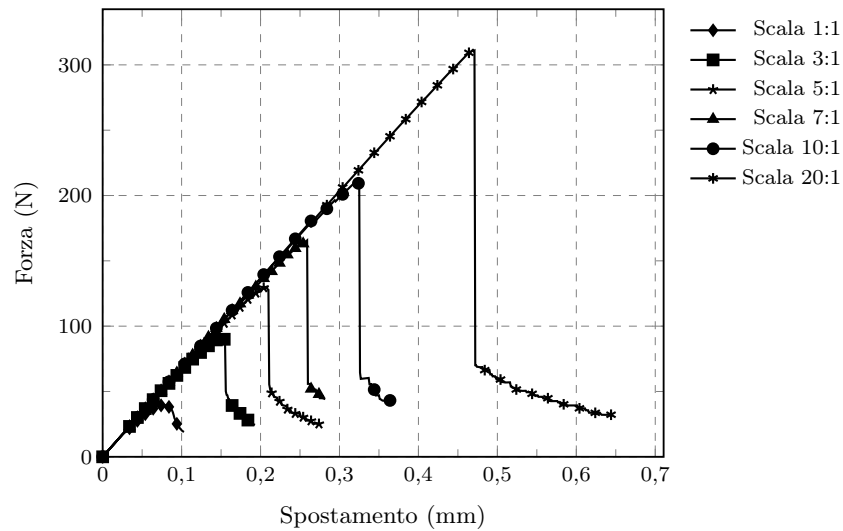
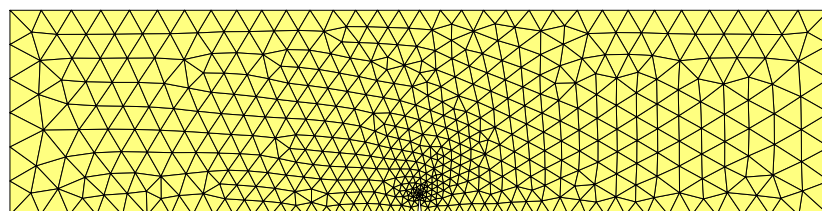
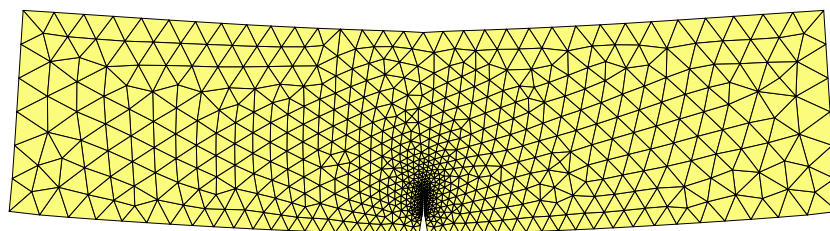


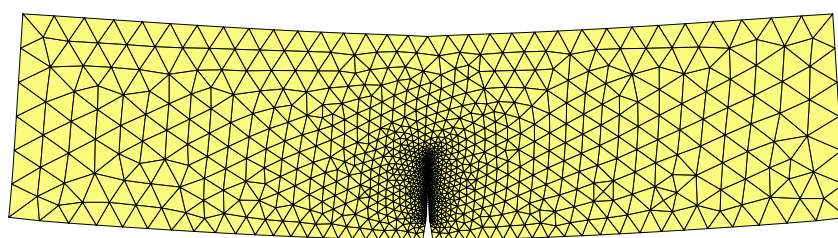
Figura 7.5: Transizione duttile-fragile per diverse scale del provino.



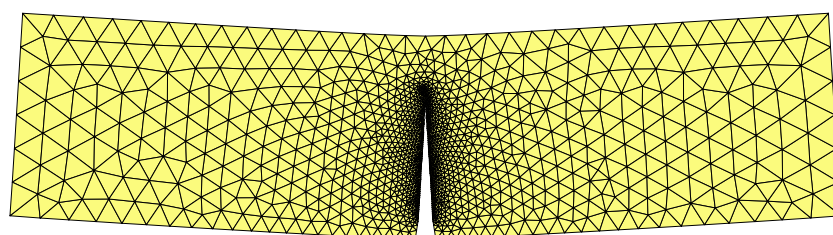
(a) Avanzamento della frattura al tempo  $t = 0$  s.



(b) Avanzamento della frattura al tempo  $t = 30$  ms.



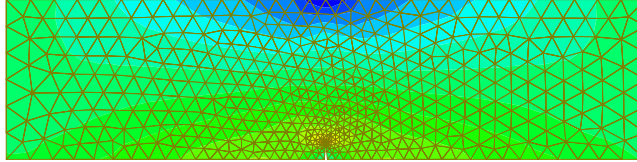
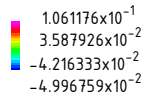
(c) Avanzamento della frattura al tempo  $t = 60$  ms.



(d) Avanzamento della frattura al tempo  $t = 105$  ms prima del collasso

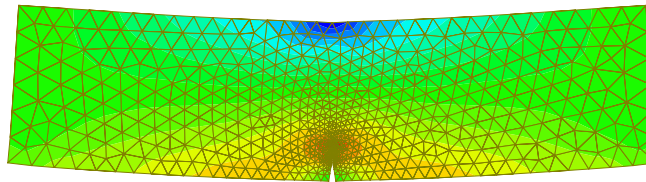
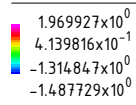
**Figura 7.6:** Avanzamento della frattura: deformata (non in scala) di alcuni istanti temporali.

Plate Stress:XX (MPa)



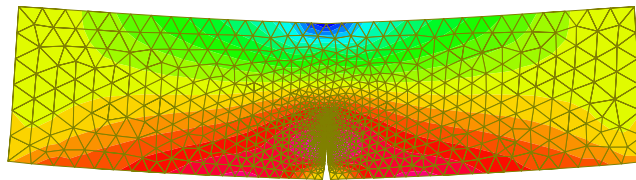
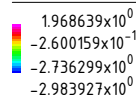
(a) Mappa di colore relativa alla tensione  $XX$  al tempo  $t = 0$  s.

Plate Stress:XX (MPa)



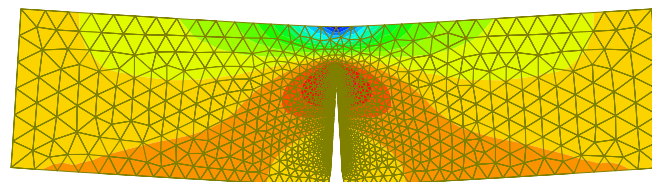
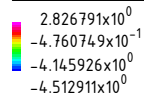
(b) Mappa di colore relativa alla tensione  $XX$  al tempo  $t = 30$  ms.

Plate Stress:XX (MPa)



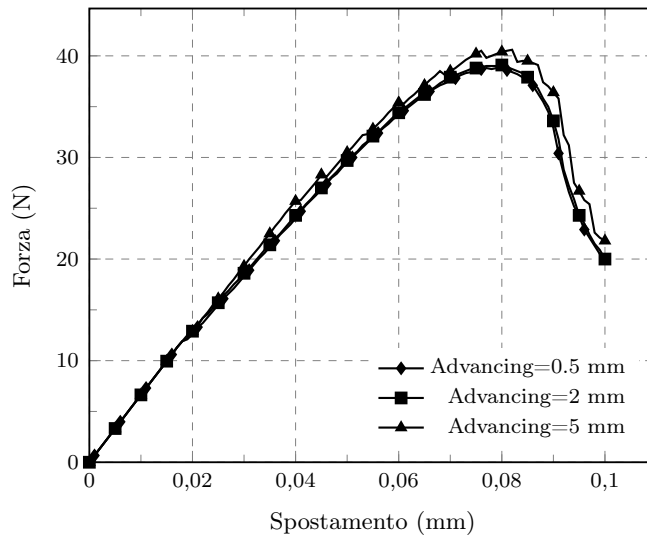
(c) Mappa di colore relativa alla tensione  $XX$  al tempo  $t = 60$  ms.

Plate Stress:XX (MPa)



(d) Mappa di colore relativa alla tensione  $XX$  al tempo  $t = 105$  ms prima del collasso

**Figura 7.7:** Avanzamento della frattura: mappe di colore relative alla tensione  $XX$ . (I valori sono relativi ai punti Gauss)

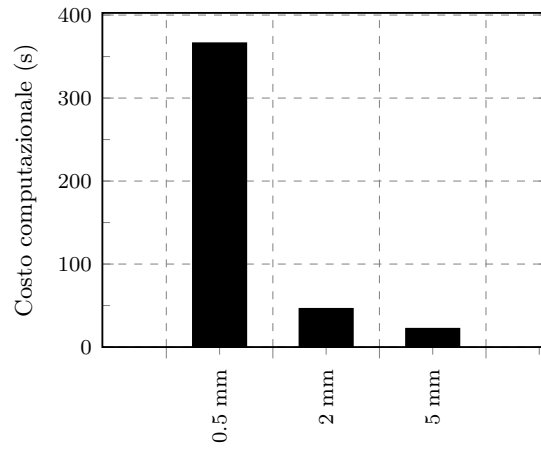


**Figura 7.8:** Influenza della dimensione della mesh.

I modelli coesivi possono portare ad un'intabilità numerica quando il numero di elementi finiti della zona di processo è insufficiente per rappresentarla. Esiste quindi un numero minimo di elementi finiti, *element threshold number*, da rispettare, affinché l'analisi sia accurata. Una pratica comune è quello di utilizzare un'analisi statica per determinare la dimensione massima degli elementi da utilizzare ma questo approccio non è conservativo perché non permette di tenere conto che il dominio è in continua evoluzione. In merito alla lunghezza della zona coesiva  $l$ , è possibile fare una stima utilizzando la teoria di Barenblatt:

$$l = \frac{\pi E G_c}{8(1 - \nu^2)\sigma_0^2} \cong 59 \text{ mm} \quad (7.1)$$

in cui  $E$  è il modulo elastico del materiale,  $G_c$  l'energia di frattura,  $\nu$  il coefficiente di Poisson del materiale e  $\sigma_0$  la tensione di trazione massima. I risultati numerici ottenuti sono in accordo con la soluzione proposta da Barenblatt in quanto la lunghezza della zona coesiva risultante dalle simulazioni è di circa 60 mm. Alcuni studi ([38]) hanno dimostrato che per ottenere risultati mesh indipendenti è necessario avere almeno 5 elementi finiti all'interno della zona di processo. Il criterio, nelle analisi eseguite risulta soddisfatto in quanto il passo di avanzamento è stato impostato a 2 mm. Nel grafico 7.8 si ha la conferma di quanto detto; come si può osservare, passi di avanzamento minori di 5 mm non portano a sensibili cambiamenti ma al più ad un incremento di tempo computazionale, come si evince in 7.9.



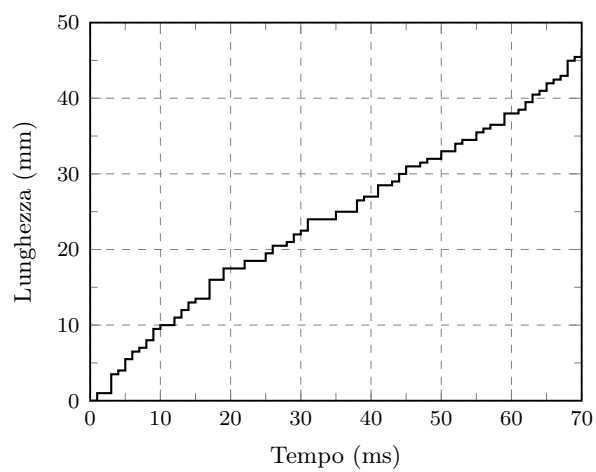
**Figura 7.9:** Costo computazionale (espresso in secondi) per le tre prove.

## 7.4 Andamento a scatti della frattura

Si vuole ora indagare l'andamento della frattura nel tempo. Esiste infatti una similitudine tra l'andamento della frattura e la registrazione dei terremoti o delle emissioni acustiche registrate da test di laboratorio. Tramite questi test è stato appunto dimostrato che l'evoluzione della frattura all'interno di un solido avviene *a scatti* e non in maniera regolare. Fisicamente il fenomeno non ha bisogno di giustificazioni profonde: si ha un avanzamento della frattura fino a che il criterio di rottura del materiale è rispettato. La frattura si arresta e cessa di propagarsi fino a che il carico esterno continua a crescere. Da questo ne deriva un comportamento a scatti.

Conducendo un'analisi quasi statica e plottando il grafico tempo - lunghezza di frattura si ottiene il grafico riportato in Figura B.4. Dal grafico si nota come l'evolversi della frattura non avviene in maniera continua e regolare ma, al contrario, in maniera discontinua.





**Figura 7.10:** Andamento a scatti della frattura.



## Capitolo 8

# Analisi dinamiche

### 8.1 Introduzione

Nelle applicazioni del capitolo precedente sono state effettuate analisi in ambito puramente statico, senza considerare la presenza della matrice delle masse.

In questo capitolo sarà impiegato il codice di calcolo precedentemente descritto per svolgere analisi dinamiche nel transitorio tramite l'algoritmo GN22 (Generalized Newmark) implicito.

Nella prima parte del capitolo sarà esposto il problema già anticipato nel Capitolo 1. Verranno poi fatte alcune applicazioni per dimostrare l'efficienza di quanto implementato, ovvero il ciclo di bilanciamento e l'algoritmo di proiezione delle variabili.

Nella seconda parte del capitolo, attraverso un *caso test* verrà motivata l'introduzione dell'algoritmo di integrazione temporale, e verrà mostrato come, nello studio della propagazione della frattura, sia necessario svolgere analisi nel transitorio per una corretta simulazione del fenomeno. Il caso statico-quasi statico rappresenta infatti un caso particolare del caso dinamico.

### 8.2 Ciclo di bilanciamento

L'obiettivo di questo paragrafo è l'applicazione ad un semplice caso test, di quanto presentato e sviluppato in precedenza. Si è scelto di riprendere l'esempio della prova di flessione in tre punti presentato alla fine del capitolo precedente, con la medesima geometria. Si è scelto di applicare una forza concentrata nel punto centrale del provino. I dati sono riassunti in Figura 8.1.

Questo primo caso test è stato condotto principalmente per verificare l'efficienza del ciclo di bilanciamento e osservare le correzioni apportate ai valori di spostamenti, velocità ed accelerazioni. L'analisi è condotta consi-

**Tabella 8.1:** Materiale e geometria del provino.

Quantità	Simbolo	Valore	Unità di misura
Altezza	$h$	100,0	$mm$
Lunghezza	$l$	400,0	$mm$
Dimensioni intaglio	$a$	10,0	$mm$
Modulo elastico	$E$	14700,0	$MPa$
Coefficiente di Poisson	$\nu$	0,2	/
Forza applicata	$F$	100	$N$
Trazione coesiva massima	$\sigma_u$	2,0	$MPa$
Apertura massima ( $\sigma_u$ )	$w_{cr}$	0,04	$mm$

derando una table avente pendenza pari a 1:1. Il tempo di integrazione  $\Delta t$  è assunto pari a  $10^{-3}$  secondi.

### 8.2.1 Balance step: andamento della convergenza

Negli step precedenti all'apertura della frattura, il dominio non varia e non è necessario ricorrere al ciclo di bilanciamento poiché le variabili vengono proiettate in maniera consistente.

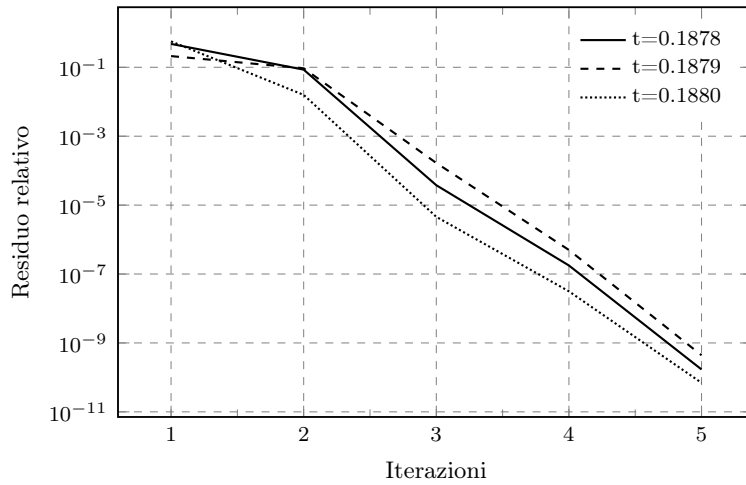
Quando la frattura si propaga, ed il dominio cambia, i valori vengono proiettati dal vecchio dominio a quello nuovo. Per assicurare il rispetto dell'equilibrio della vecchia soluzione sul nuovo dominio è necessario utilizzare il ciclo di bilanciamento. Nella Figura 8.1 si riportano alcuni i profili di convergenza dei vari time step che seguono all'apertura della frattura. Si nota facilmente la convergenza quadratica, tipica del metodo iterativo di Newton Raphson.

Nella Figura 8.2 si riportano alcuni i profili di convergenza ai alcuni time step dell'analisi. In particolare viene riportato uno step intermedio, in cui si nota che la convergenza è raggiunta in 6 iterazioni. Sono inoltre riportati due step finali, precedenti all'arresto dell'analisi in cui si nota la maggiore difficoltà a trovare una situazione equilibrata.

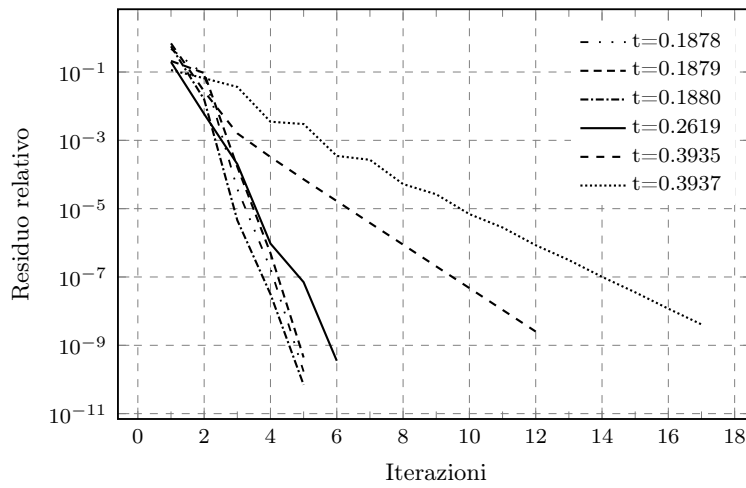
### 8.2.2 Balance step: correzioni apportate

Nei grafici 8.3, 8.4 e 8.5 vengono riportati spostamenti, velocità ed accelerazioni della prova eseguita con e senza ciclo di bilanciamento.

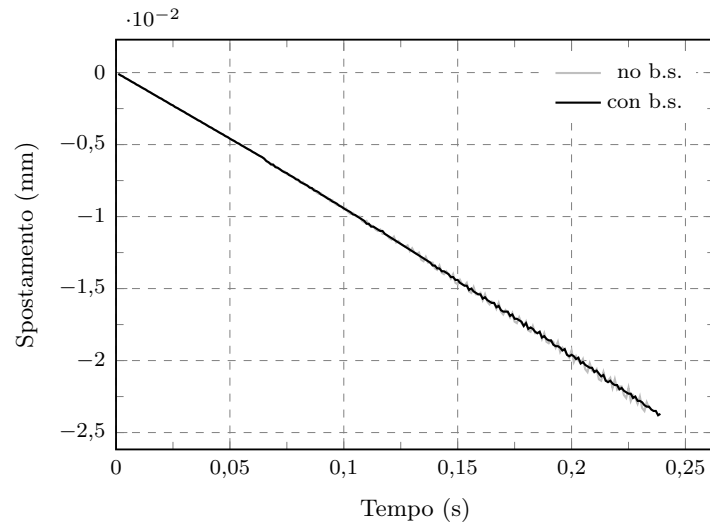
Si nota come gli spostamenti non siano apparentemente influenzati in maniera sensibile dalla correzione che si apporta nel ciclo di bilanciamento. In realtà, una piccola variazione di spostamento porta a grandi variazioni di velocità ed accelerazioni, qualora i tempi in gioco siano relativamente bassi. Nel caso carichi applicati rapidamente, le correzioni agli spostamenti sono più pesanti, a dimostrazione della necessità di utilizzare un ciclo di bilanciamento nelle analisi.



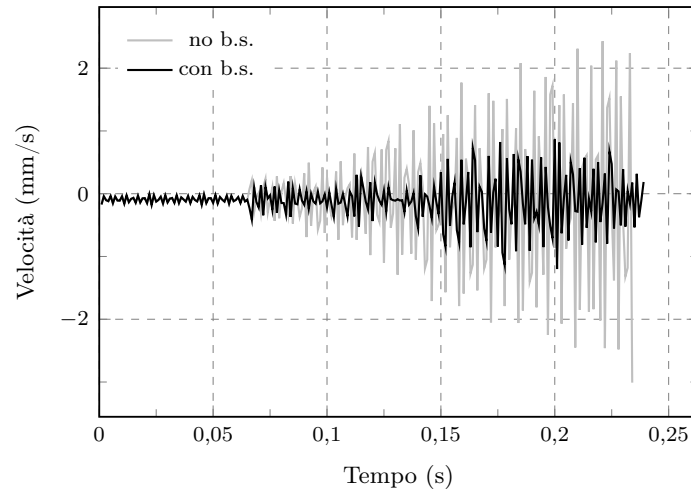
**Figura 8.1:** Profili di convergenza per gli step immediatamente successivi all'apertura della frattura.



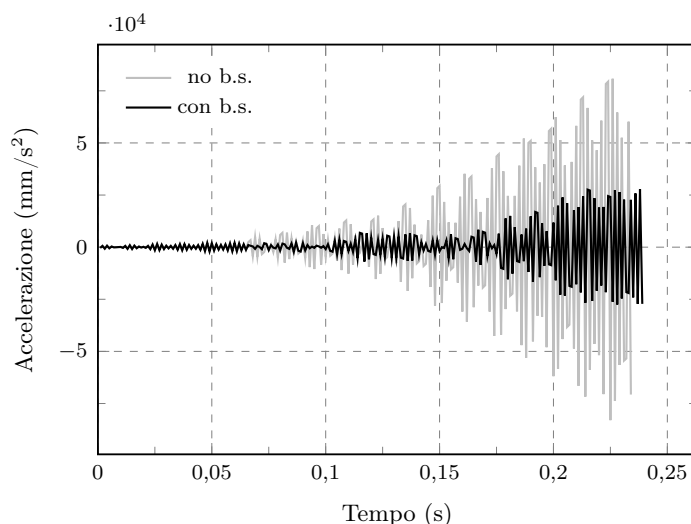
**Figura 8.2:** Profili di convergenza per alcuni step dell'analisi.



**Figura 8.3:** Grafico tempo - spostamento: confronto tra soluzione ottenuta con e senza ciclo di bilanciamento.



**Figura 8.4:** Grafico tempo - velocità: confronto tra soluzione ottenuta con e senza ciclo di bilanciamento.



**Figura 8.5:** Grafico tempo - accelerazione: confronto tra soluzione ottenuta con e senza ciclo di bilanciamento.

### 8.3 Influenza dello smorzamento

Tutte le strutture manifestano la capacità di dissipare energia. Lo smorzamento è definito come la perdita di energia dissipata in un ciclo di oscillazione rispetto alla massima energia accumulata nella struttura. Ci sono principalmente due meccanismi che contribuiscono in modo significativo allo smorzamento delle strutture:

- *Smorzamento materiale:* dovuto a un comportamento isteretico. Esso deriva principalmente da una complessa interazione molecolare nel materiale. Questo meccanismo è dipendente dal tipo di materiale e dal modo di produzione della struttura. Inoltre, ogni struttura presenta delle differenze microscopiche rilevanti portando la quantificazione di questo meccanismo ad un'estrema complicazione.
- *Smorzamento di interfaccia:* non entra in gioco in questo lavoro. Deriva dall'attrito tra i diversi componenti strutturali.

Senza entrare nel dettaglio, lo smorzamento, in questo lavoro, è trattato introducendo delle forze dissipative viscosive equivalenti attraverso il rapporto di smorzamento  $\xi$ . Questo consente di snellire di molto la trattazione analitica.

Il modello implementato nel seguente lavoro è il cosiddetto *smorzamento alla Rayleigh* (proposto per primo da Lord Rayleigh) che viene descritto nel seguito.

### 8.3.1 Smorzamento alla Rayleigh

Solitamente la matrice di dissipazione viscosa viene espressa in funzione di quella modale  $\Phi$ , una volta assegnato il rapporto di smorzamento dei vari modi. La matrice modale permette di diagonalizzare la matrice delle masse  $\mathbf{M}$  e la matrice di rigidità  $\mathbf{K}$  ma in generale non diagonalizza quella di smorzamento  $\mathbf{C}$ . Tuttavia è possibile definire la matrice di smorzamento senza dapprima aver definito la matrice modale infatti per definizione tale matrice deve essere ortogonale alla base di autovettori che diagonalizzano le matrici di massa e di rigidità. Esiste quindi un caso speciale in cui la matrice modale diagonalizza la matrice di smorzamento e questo accade quando la matrice di smorzamento può essere espressa come combinazione lineare della matrice di massa e di rigidità dove la matrice di smorzamento  $\mathbf{C}$  viene costruita come:

$$\mathbf{C} = \alpha\mathbf{M} + \beta\mathbf{K} \quad (8.1)$$

Dove i coefficienti  $\alpha$  e  $\beta$  sono detti coefficienti scalari:

- $\alpha$  rappresenta la resistenza espressa dal fluido in cui la massa modellata si muove;
- $\beta$  rappresenta la dissipazione viscosa espressa dalla struttura stessa internamente;

Senza entrare nel dettaglio, una volta definito il rapporto di smorzamento  $\xi$  è possibile definire la seguente relazione:

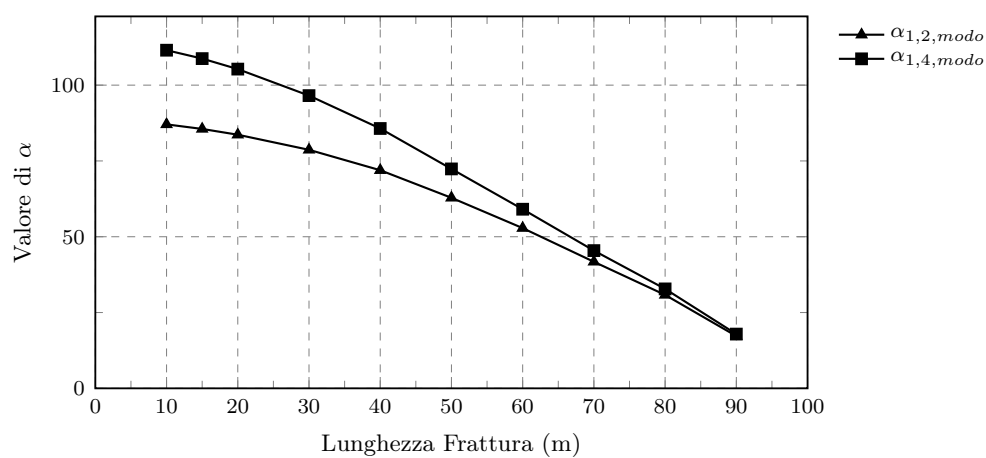
$$\xi = \frac{\alpha}{2} \frac{1}{\omega_k} + \frac{\beta}{2} \omega_k \quad (8.2)$$

Attraverso questa espressione è possibile ricavare un rapporto di smorzamento che presenta delle modeste variazioni di intensità nel campo delle frequenze di interesse progettuale. Ciò si ottiene assegnando i rapporti di smorzamento in corrispondenza dei due modi che definiscono il campo di interesse. Sul piano pratico si fissano i valori delle frequenze  $\omega_k$  e quello del rapporto di smorzamento  $\xi$  si ricavano parametri  $\alpha$  e  $\beta$ . Si osserva che la scelta delle frequenze a cui si assegna il rapporto di smorzamento influisce il comportamento dei modi più alti. Per esempio scegliendo il primo ed il secondo gli altri modi si esauriranno rapidamente, mentre scegliendo il primo ed il quarto si avrà che il secondo ed il terzo si esauriranno meno velocemente del quarto. Quindi la scelta dei modi influisce sullo smorzamento alla Rayleigh, da cui è necessaria un'appropriata scelta.

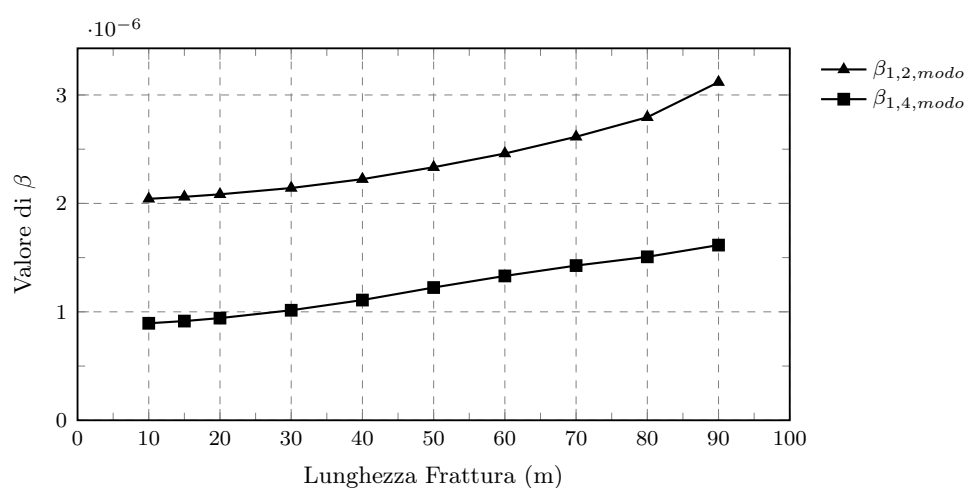
### 8.3.2 Smorzamento e frattura

L'introduzione dello smorzamento nel codice di calcolo fa notare come sia presente una grande quantità di energia dissipata nell'area della frattura. Le caratteristiche del provino utilizzato sono riassunte in Figura 8.2.





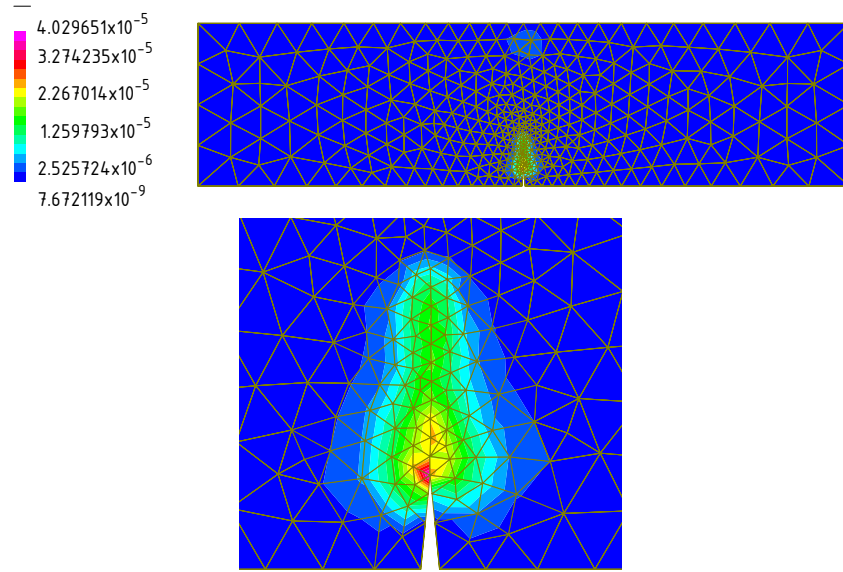
**Figura 8.6:** Andamento di  $\alpha$  in funzione della lunghezza della frattura e della scelta dei modi.



**Figura 8.7:** Andamento di  $\beta$  in funzione della lunghezza della frattura e della scelta dei modi.

**Tabella 8.2:** Caratteristiche provino.

Lunghezza (mm)	400
Altezza (mm)	100
Altezza Intaglio (mm)	10
$\Delta l$ (mm)	2
$\sigma_u$ (MPa)	2
$w_{cr}$ (mm)	0.04
$F$ (N)	1000



**Figura 8.8:** Densità di energia di smorzamento ( $\text{J}/\text{mm}^2$ )

**Tabella 8.3:** Parametri di smorzamento.

$\alpha$ (1/s)	60
$\beta$ (s)	$1E - 6$

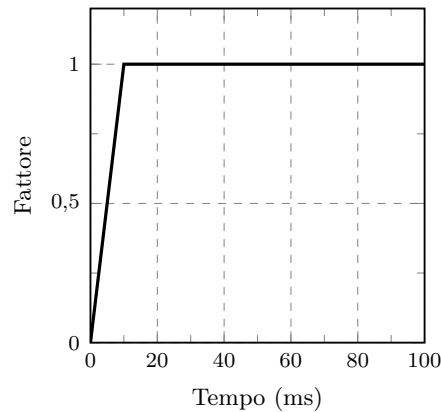
Lo smorzamento calcolato in questo caso è dato dalla storia temporale dall'inizio dell'analisi (non è quello istantaneo). Dalla figura 8.8 si deduce che c'è una notevole variazione delle quantità materiali nella zona della frattura rispetto a tutte le altre zone del sistema meccanico. Nelle analisi in esame si è sempre utilizzato uno smorzamento come indicato in Figura 8.3.

## 8.4 Analisi energetica

Nel seguente paragrafo verrà esaminato l'aspetto energetico della dinamica del sistema. Dato che l'ambiente esterno non interagisce in alcun modo con il sistema, in assenza di frattura deve risultare:

$$\int_0^{\mathbf{u}(t)} \mathbf{M}\ddot{\mathbf{u}} d\mathbf{u} + \int_0^{\mathbf{u}(t)} \mathbf{C}\dot{\mathbf{u}} d\mathbf{u} + \int_0^{\mathbf{u}(t)} \mathbf{K}\mathbf{u} d\mathbf{u} = \int_0^{\mathbf{u}(t)} \mathbf{F}\mathbf{u} d\mathbf{u} \quad (8.3)$$

Nell'equazione appena scritta sono facilmente riconoscibili le varie forme del lavoro che competono ad un sistema in movimento. Il primo termine è legato al lavoro della forza d'inerzia  $W_I$ , il secondo è legato al lavoro della forza di smorzamento  $W_D$ , il terzo è legato al lavoro della forza di richiamo elastica



**Figura 8.9:** Table impiegata nelle analisi - caso test 3.

$W_E$  ed infine il termine a destra dell'equazione rappresenta il lavoro compiuto dalle forze esterne  $W_{Ext}$ . E' immediato notare perciò che, dovendo le forze agenti sulla massa rispettare il concetto di equilibrio dinamico, i lavori dovranno anch'essi equilibrarsi tra loro. Si può dunque scrivere la seguente equazione di equilibrio dei lavori:

$$W_I + W_D + W_E - W_{Ext} = 0 \quad (8.4)$$

Il fatto che la somma dei lavori compiuti da tutte le forze agenti sul sistema, attive e passive, debba annullarsi è una conclusione del tutto generale valida per qualsiasi sistema equilibrato. Accertato che il lavoro risultante di tutte le forze è sempre nullo, è di ovvia dimostrazione qualsiasi altra scrittura che, componendo in varie categorie i lavori, e confrontando queste tra loro, si concluda con l'uguaglianza dei termini o equivalentemente con l'annullamento del lavoro complessivo.

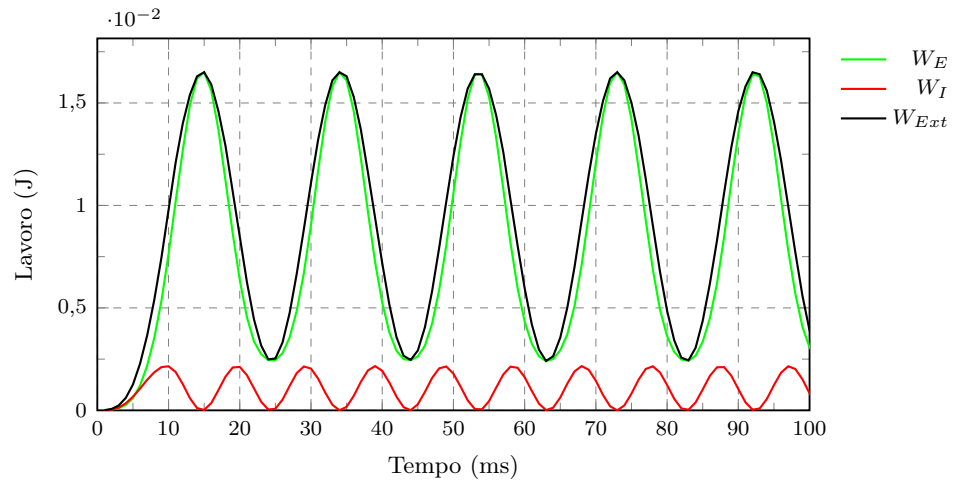
Applicando al provino relativo alla prova di flessione in tre punti una rampa di carico del tipo:

e applicando un carico tale per cui non si vada a raggiungere la massima tensione di trazione in nessun punto del solido, in assenza di smorzamento si ottiene il grafico riportato in Figura 8.10.

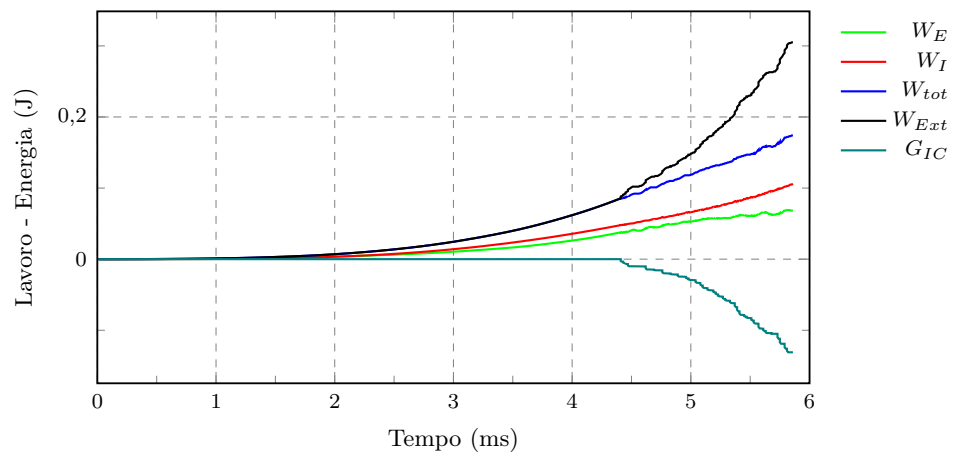
Nel caso in cui la massima tensione di trazione nel provino venga superata, si apre una nuova frattura. In questo caso, trascurando la dissipazione di natura termica, non considerata nelle equazioni che reggono il fenomeno, si può scrivere:

$$W_I + W_D + W_E - W_{Ext} = \int_0^{w_c} \sigma(w) dw A \quad (8.5)$$

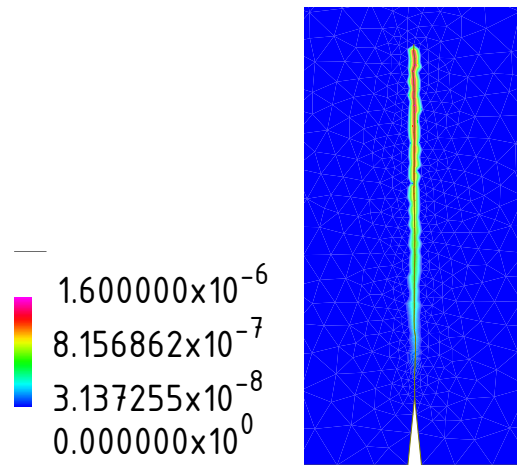
in cui il termine a destra dell'equazione (8.5) rappresenta l'energia dissipata sulla superficie di frattura. Ripetendo il test precedente, in assenza di



**Figura 8.10:** Andamento dei lavori: sistema conservativo.



**Figura 8.11:** Andamento dei lavori: sistema non conservativo.



**Figura 8.12:** Densità di energia di frattura ( $\text{J}/\text{mm}^2$ )

smorzamento, e considerando un carico tale per cui la resistenza a trazione del materiale venga superata, si ottiene il grafico in Figura 8.11.

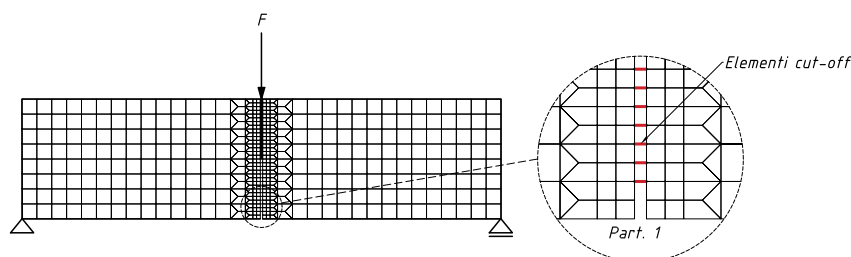
Come si può notare dal grafico,  $W_{tot}$  dato dalla somma del lavoro delle forze di richiamo e di quelle inerziali (in assenza di smorzamento) non coincide con il lavoro delle forze esterne  $W_{Ext}$ . La differenza è dovuta al fatto che, per aprire la frattura, il sistema dissipa energia. La parte di lavoro persa a seguito dell'avanzamento della frattura è rappresentata da  $G_{IC}$ . In Figura 8.12 è riportato il contour plot relativo alla densità di energia di frattura negli istanti finali della prova.

## 8.5 Rilascio di energia elastica

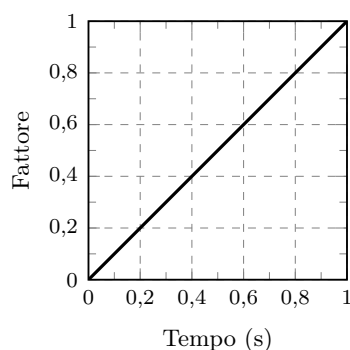
Per cercare di capire cosa avviene quando l'energia elastica, accumulata nel corpo, viene rilasciata quasi istantaneamente dando luogo ad una frattura è possibile costruire un semplice modello ad elementi finiti tramite un software commerciale. Il modello è costruito con elementi bidimensionali quadrangolari in condizioni piane di deformazione. Viene introdotta una discontinuità riproducendo un intaglio nel centro del provino. La fessura così ottenuta viene "cucita" con elementi *cut-off* in grado di disattivarsi quando raggiungono una predeterminata tensione di trazione. In questo modo, quando viene superata la tensione di trazione massima all'interno di un elemento, esso si disattiverà traferendo le tensioni alla rimanente parte del modello. Il percorso della frattura risulta, in questo caso semplicistico, predeterminato.

Il modello è esposto in Figura 8.13. Le caratteristiche del materiale sono le stesse delle analisi svolte in precedenza, reperibili in Figura 7.1.

Viene applicata una forza concentrata in mezzzeria crescente linearmente secondo quanto riportato in Figura 8.14 e condotta un'analisi non lineare



**Figura 8.13:** Mesh e BC del modello numerico: il percorso della frattura risulta predeterminato la frattura è "cucita" con barre cut-off.



**Figura 8.14:** Table impiegata nelle analisi del provino cucito con elementi cut-off.

dinamica.

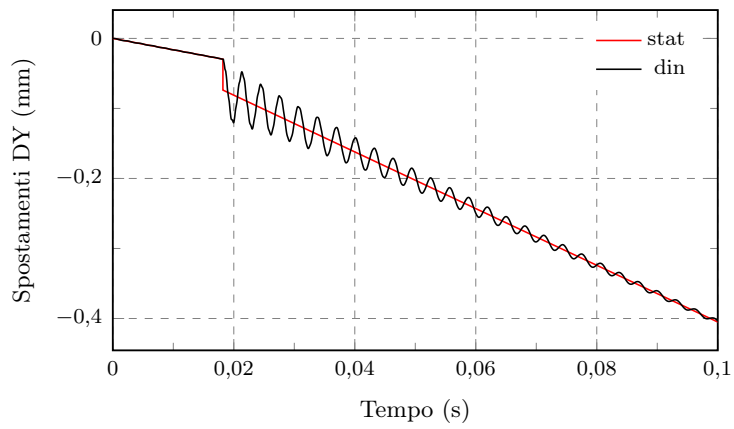
Dal grafico riportato in Figura 8.15 si nota come, nell'intorno degli 0.02 s venga superata la resistenza a trazione degli elementi cut off, aprendo quindi la frattura. In seguito al rilascio delle barre cut-off, il sistema inizia ad oscillare. Come si evince dal grafico, si ha un incremento di spostamento rispetto al caso statico dovuto alle oscillazioni causate dalla movimentazione delle masse nell'intorno dell'apice della frattura. L'incremento da luogo a tensioni più elevate e ad un incremento delle reazioni vincolari, come mostrato in Figura 8.18.

Di seguito si riportano spostamenti, velocità, accelerazioni relative al nodo centrale e le reazioni vincolari relative all'appoggio destro.

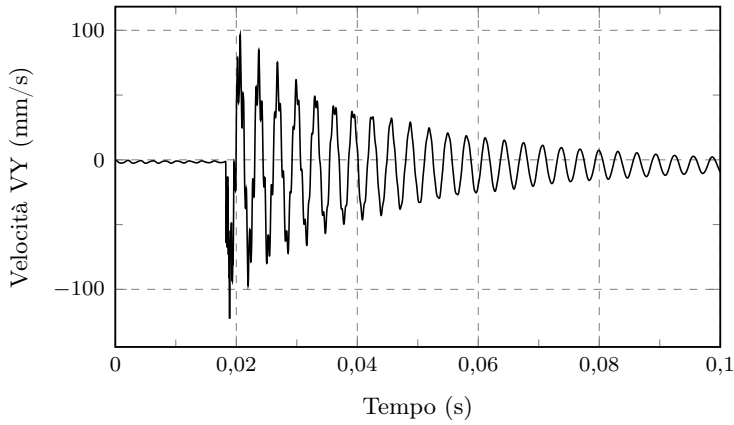
In Figura 8.19, infine, vengono riportate le mappe di colore relative alle accelerazioni degli elementi. Si nota come, immediatamente dopo l'apertura della frattura, le accelerazioni, e quindi le forze inerziali, crescano di diversi ordini di grandezza nell'intorno del percorso di frattura.

### 8.5.1 Analisi critica della propagazione statica

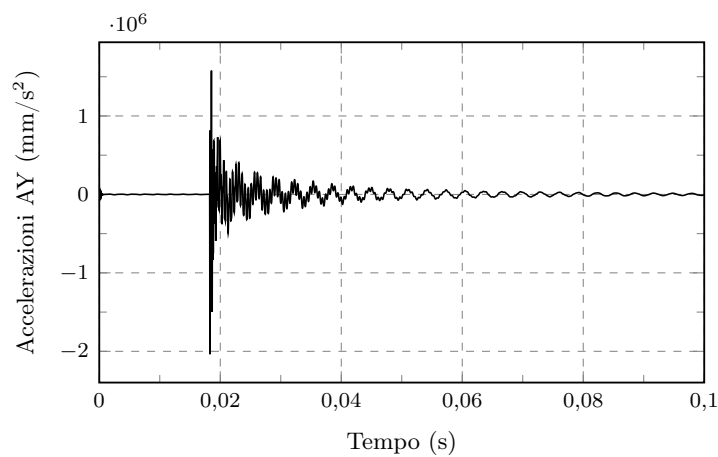
Si ritiene che lo studio del fenomeno senza il contributo delle forze inerziali possa portare a risultati fuorvianti. La propagazione della frattura è infatti



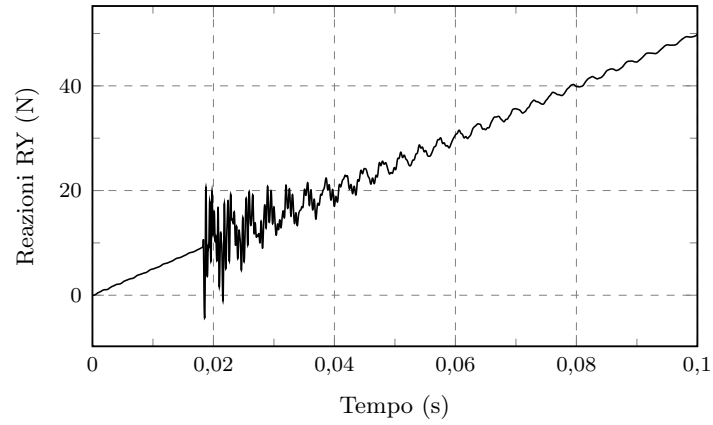
**Figura 8.15:** Grafico tempo - spostamenti relativo al nodo centrale.



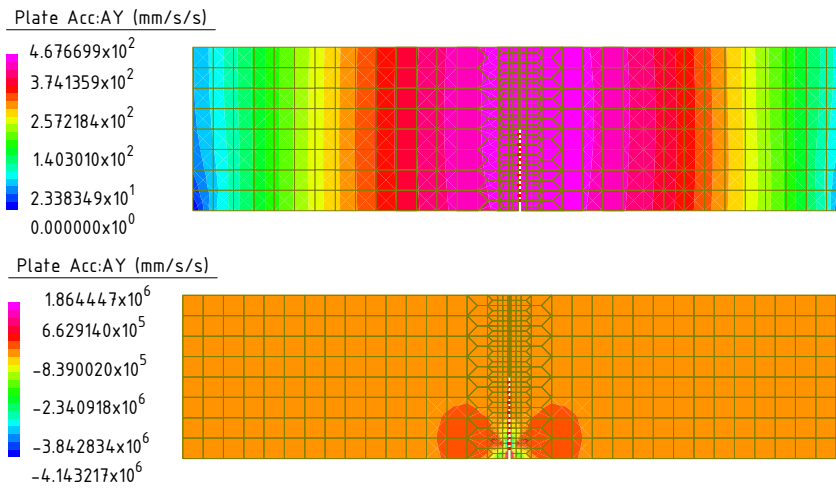
**Figura 8.16:** Grafico tempo - velocità relativo al nodo centrale.



**Figura 8.17:** Grafico tempo - accelerazioni relativo al nodo centrale.



**Figura 8.18:** Grafico tempo - reazioni vincolari.



**Figura 8.19:** Accelerazione AY negli istanti prima e dopo il rilascio degli elementi cut-off.



un fenomeno estremamente rapido ( $1 \sim km/s$ ) e la sua propagazione avviene in tempistiche il cui ordine di grandezza è il milionesimo di secondo.

Si può pensare che l'evoluzione quasistatica della frattura sia da considerare come un caso limite di processi dinamici. Se la velocità con cui variano i carichi imposti è molto piccola è lecito trascurare gli effetti inerziali, a patto che anche la corrispondente evoluzione del sistema meccanico sia lenta. In questo caso non c'è dubbio che si debba usare un modello quasistatico. Però, anche se si suppone che l'evoluzione sia quasistatica, ci sono degli istanti in cui il modello quasistatico stesso impone un salto della frattura, che si potrebbe considerare come un brevissimo intervallo di crescita velocissima. È chiaro che in questi intervalli viene meno l'ipotesi che l'evoluzione dello stato del materiale sia lenta. Quindi in questi intervalli non si può trascurare l'inerzia. In un modello quasistatico questi brevissimi intervalli vanno considerati come infinitesimi, e va considerato il limite dell'evoluzione dinamica, che porterebbe istantaneamente da uno stato a uno stato diverso.

Nell'intorno dell'apice quindi, la mobilitazione delle masse non è trascurabile e dovrebbe essere tenuta in considerazione.

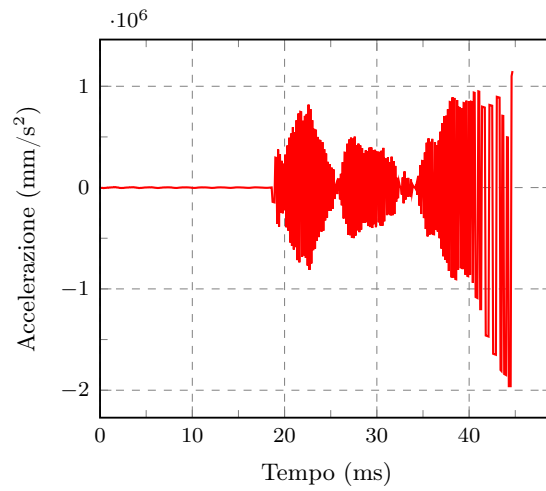
### 8.5.2 Effetti inerziali dovuti all'apertura della frattura

Si è detto nel precedente paragrafo 8.5.1 che l'apertura di una frattura comporta una variazione veloce del sistema facendo venire meno l'ipotesi che l'evoluzione dello stato del materiale sia lenta. Quindi in questi brevissimi intervalli non si può trascurare l'inerzia. Dalle indagini fin'ora svolte non si sono trovati significativi effetti inerziali ma non è ancora possibile escludere che questi non esistano, ancora una volta si vuole evidenziare l'analogia con la sismica. Sono necessarie anche in questo caso ulteriori indagini per capire se vi è un limite nel modello numerico impiegato (elementi finiti utilizzati, algoritmo di frattura, etc.).

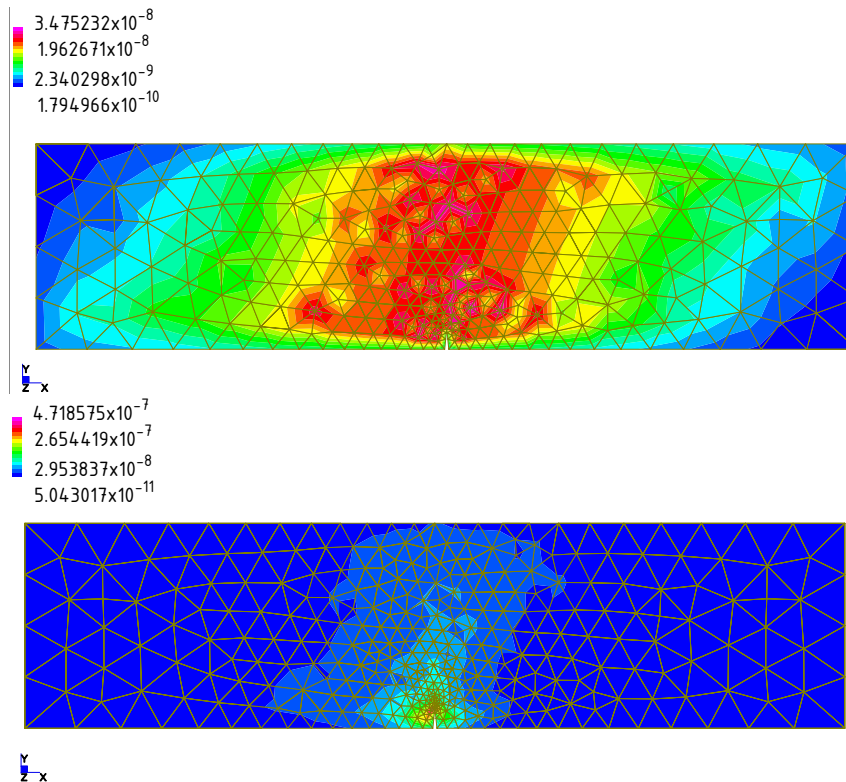
Impiegando il codice di calcolo sviluppato in questo lavoro, è possibile osservare come variano alcune quantità: accelerazione e densità di energia cinetica. In particolare nel grafico riportato in Figura 8.20 si è monitorato il nodo d'apice dell'intaglio applicato al provino.

Considerando l'istante temporale precedente e quello immediatamente successivo all'apertura della frattura, si nota come la densità di energia cinetica cresca (riportata in Figura 8.21) di circa un ordine di grandezza, a supporto di quanto osservato sopra.

Dalle indagini svolte si vuole evidenziare come nella zona della frattura si abbia un'istantanea variazione delle quantità materiali del sistema meccanico nell'istante in cui si apre la frattura. Da questo semplice test si può evincere come, per una simulazione più accurata del fenomeno, sia necessario svolgere un'analisi nel transitorio dinamico.



**Figura 8.20:** Grafico tempo - accelerazione AY del nodo d'apice dell'intaglio del provino durante la propagazione della frattura. Si nota come si ha un improvviso incremento dell'accelerazione.



**Figura 8.21:** Densità di energia cinetica ( $J/mm^2$ ): istanti prima e dopo l'apertura della frattura.

## 8.6 Algoritmo di integrazione temporale

La scelta di un algoritmo di integrazione temporale è un criterio essenziale per assicurare efficienza e robustezza alla simulazione numerica. Le difficoltà legate a questa scelta risiedono nell'abilità nel combinare robustezza, accuratezza e stabilità dell'algoritmo soprattutto con l'avanzare della soluzione. Data la natura di rapida espansione del fenomeno si potrebbe pensare che sia più adatto uno *schema esplicito* che non è influenzato dai modi delle alte frequenze e per ragioni di economia computazionale. In questo lavoro si è utilizzato invece uno schema implicito, ed in particolare l'algoritmo di Newmark implicito. A discapito dell'economia computazionale, si è scelto di utilizzare un metodo implicito poiché il suo impiego non maschera i problemi numerici legati alla frattura, non filtra la risposta alle alte frequenze e promuove valutazioni più oggettive sulla qualità della soluzione e della convergenza e rivela il dettaglio completo della risposta simulata.

Inoltre, l'onere computazionale dei metodi impliciti viene compensato dalla positiva proprietà di essere generalmente *incondizionatamente stabile*. Essi infatti consentono di ottenere soluzioni di precisione adeguata anche con passi di integrazione temporale ampi, consentendo un ridotto onere computazionale quando sia richiesto.

## 8.7 Fisica della propagazione della frattura

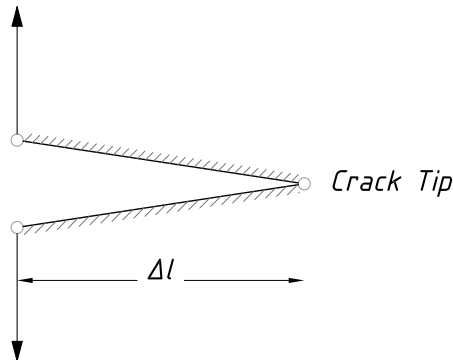
### 8.7.1 Introduzione

La propagazione della frattura all'interno di un solido non avviene in maniera continua. Fisicamente il fenomeno presenta un'analogia con la tettonica delle placche. Si ha infatti un accumulo di energia elastica nei corpi, indotto dai movimenti tettonici, fintanto che un determinato criterio di rottura non sarà soddisfatto, non si avranno movimenti. Una volta che esso è superato, si avrà un grande rilascio di energia causando l'improvviso e repentino spostamento della massa rocciosa coinvolta. Tale movimento improvviso, che in pochi secondi rilascia energia accumulata per decine o centinaia di anni, genera così le onde sismiche e il terremoto associato. Il passaggio tra energia elastica e onde sismiche è *inerzia*.

Da un punto di vista sperimentale è stato osservato (Pizzocollo, 2001 [33]) che l'avanzamento della frattura avviene in maniera discontinua, graduale, a scatti (*stepwise*).

### 8.7.2 Velocità infinita di propagazione

Come già descritto, nell'algoritmo numerico impiegato, si avrà un avanzamento istantaneo, seguito da uno split dei nodi ogni volta che si soddisfa il criterio di rottura.



**Figura 8.22:** Il  $\Delta l$  di cui la frattura aumenta ogni volta che viene soddisfatto il criterio di rottura.

Ad ogni passo temporale  $j$ , successivi avanzamenti  $\Delta l$  della frattura sono possibili finché il criterio di rottura rimane soddisfatto. Il loro numero in generale dipende dal passo temporale  $\Delta t$ , dal passo spaziale  $\Delta l$  e dall'incremento di carico applicato.

L'avanzamento a scatti della frattura è dunque un fenomeno noto, ma nel caso in esame si ritiene che esso sia viziato all'algoritmo di nucleazione della frattura implementato. Prendendo come riferimento la Figura 8.22 si osserva appunto che istantaneamente quando viene soddisfatto il criterio di rottura la frattura avanza della quantità  $\Delta l$ . Questo passo di avanzamento, che algoritmicamente avviene in un tempo infinito, nella realtà fisica impiega un determinato tempo, dipendente dalle caratteristiche del materiale.

Per una simulazione nel transitorio, che coinvolge il tempo sarà necessario controllare il passo di avanzamento della frattura, per non violare i principi fisici che reggono il fenomeno.

Facendo riferimento al grafico tempo - lunghezza della frattura, si avrà una linea verticale. Questo si nota utilizzando passi spaziali  $\Delta l$  troppo elevati. In quel caso si ha una propagazione infinitamente veloce della frattura. Se per esempio si immagina un carico talmente elevato da non uscire mai dallo step allora si ha che l'avanzamento avviene interamente all'interno di quello step, a discapito di tutte le leggi fisiche. Tradotto, la frattura avanza in un tempo infinitamente veloce.

Si può quindi affermare che manca una correlazione  $\Delta l - \Delta t$  per simulare il fenomeno in maniera fisicamente accettabile.

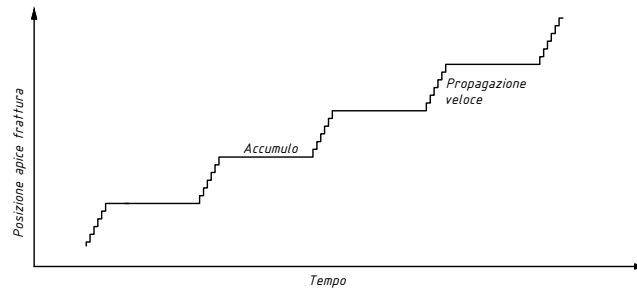
L'avanzamento graduale, o a scatti è infatti dovuto alla costruzione dell'algoritmo, e non ad una corretta simulazione fisica del fenomeno. Introducendo la dipendenza temporale nell'algoritmo, e tentando di eliminare le possibili dipendenze (dalla discretizzazione spaziale e temporale) si vuole capire se l'algoritmo è in grado di simulare il fenomeno in maniera coerente, o se il modello numerico necessita di essere arricchito di nuovi elementi per meglio descrivere comportamenti che non si riescono a cogliere.

### 8.7.3 Influenza delle forze inerziali

La *meccanica della frattura dinamica* è il sotto campo di meccanica della frattura che si occupa dei fenomeni per i quali il ruolo delle forze inerziali diventa significativo (Freund [17]). La questione se gli effetti inerziali siano significativi in ogni situazione in cui c'è la frattura dipende dalle condizioni di carico, dalle caratteristiche del materiale e dalla configurazione geometrica del corpo nonché dal comportamento della frattura in se come fenomeno. Circostanze in cui le forze inerziali sono davvero significative non possono essere specificate in modo inequivocabile ma sono intuibili alcuni aspetti.

Gli effetti inerziali possono derivare da:

- Da carico applicato rapidamente. In caso di carico rapido, l'influenza dei carichi viene trasferita alla fessura per mezzo di onde di tensione o *onde meccaniche* che si propagano attraverso il materiale. Per esempio, si supponga che esista una time table che scandisce l'applicazione del carico nel tempo. Se il tempo di applicazione del carico è elevato, cioè l'applicazione del carico è lenta, rispetto al tempo richiesto ad un'onda di tensione per attraversare, alla velocità dell'onda caratteristica del materiale, una lunghezza rappresentativa del corpo, come la lunghezza della fessura o la distanza che separa l'apice della fessura dal bordo caricato, è ragionevole pensare che gli effetti inerziali siano significativi. Inoltre in caso di avanzamento della fessura, gli effetti inerziali saranno probabilmente importanti se la velocità dell'apice della fessura o del suo bordo è una frazione significativa della velocità dell'onda caratteristica del materiale.
- Da una rapida propagazione della frattura. Le forze d'inerzia, quando si apre la frattura, non possono essere trascurate. Ipotizzando dei  $\Delta l$  e dei  $\Delta t$  piccolissimi si osserva un avanzamento che si avvicina al continuo. Si può pensare che l'apertura della fessura metta in moto le masse nell'intorno dell'apice. Una domanda a cui si vuole rispondere è quanto la dinamica incide sul fenomeno. Freund [17] afferma che nel caso di propagazione rapida della frattura, le particelle materiali sulle facce opposte della frattura si spostano l'una rispetto all'altra quando la frattura è passata, poichè non si ha più continuità del materiale. La resistenza inerziale a questo movimento può anche influenzare la risposta e deve essere presa in considerazione per una descrizione completa del processo. C'è anche un collegamento tra la rapida propagazione di una fessura e l'onda di tensione sviluppata da una frattura mobile (importante anche in sismologia).



**Figura 8.23:** Schematizzazione dell'andamento stepwise.

- Esiste inoltre anche una correlazione tra le onde meccaniche che si possono trasmettere nel solido e la massa. Esse hanno un limite fisico che dipende dalla massa.

#### 8.7.4 Comportamento stepwise

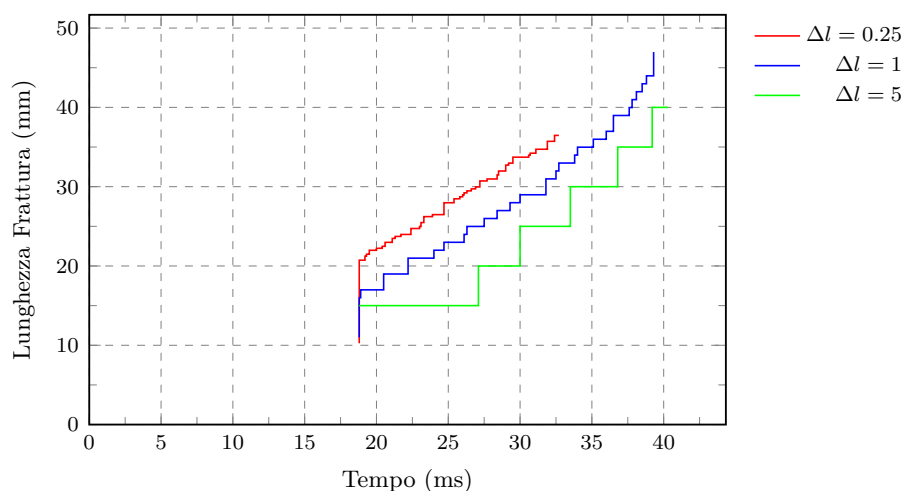
Come già introdotto al paragrafo 8.7 la frattura si propaga in modo graduale. Questo andamento deriva da un accumulo di energia elastica nel corpo seguito, quando il criterio di rottura è soddisfatto, da un repentino avanzamento della frattura.

In statica, nel momento in cui la frattura si apre si ha un avanzamento e affinché la frattura si propaghi ancora deve necessariamente aumentare la forzante esterna. Il comportamento statico inoltre è limitato dalla dimensione dell'avanzamento spaziale. Diminuendo via via tale avanzamento (al limite  $\Delta l \rightarrow 0$ ), l'andamento della frattura tende ad un comportamento continuo e non più a scatti come si evidenzia dal grafico riportato in Figura 8.24. La differenza tra un'analisi statica e dinamica è data dall'influenza delle masse dal momento in cui si apre la frattura. L'effetto inerziale che si crea fornisce un incremento tensionale che fa avanzare più rapidamente la frattura. In Figura 8.23 si ha una rappresentazione schematica del fenomeno

### 8.8 Caso test

In questo paragrafo verrà preso in considerazione l'esempio relativo alla prova di flessione in tre punti. In particolare, saranno svolte analisi dinamiche, per capire quanto il contributo inerziale incida nell'analisi. Sia un criterio per scelta del passo di integrazione temporale sia un criterio per la scelta del passo di discretizzazione spaziale sono necessari per ben rappresentare il principio fisico. La difficoltà principale sta nel fatto che non possono essere considerate separatamente ma devono essere necessariamente collegate tra loro.

Tutte le prove eseguite in questo contesto vengono svolte applicando solo



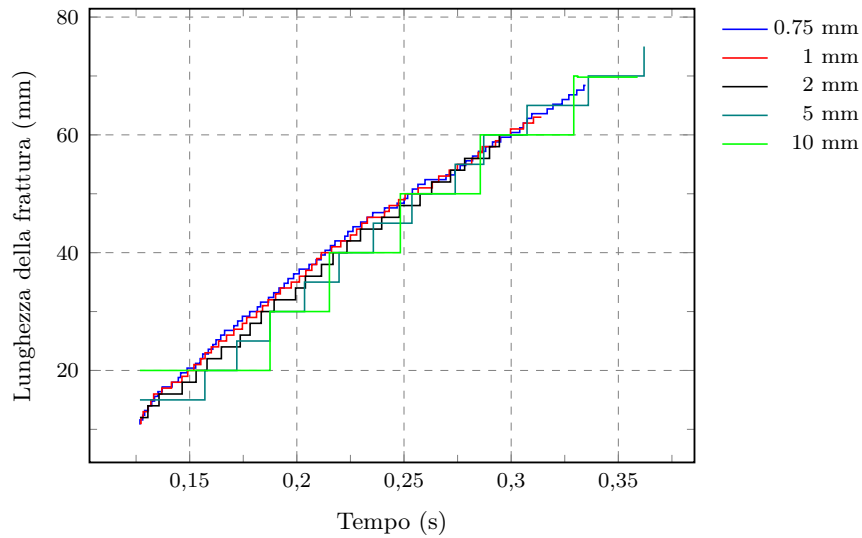
**Figura 8.24:** Andamento della frattura derivante da un'analisi statica. Si nota come l'andamento tenda al continuo mano a mano che diminuisce la lunghezza di avanzamento.

forze. La ragione di questo sta nel non imporre a priori (tramite spostamenti impressi) la velocità di propagazione della frattura ma bensì che essa sia il risultato naturale di un simulazione numerica completa.

### 8.8.1 Dipendenza dalla discretizzazione spaziale

Per discretizzazione spaziale si intende il passo  $\Delta l$  che viene scelto come avanzamento della frattura. Questo passo spaziale può influenzare in vario modo l'andamento della frattura. Lo scopo di questo paragrafo è tentare di capire in che misura esso influenzi la soluzione cercata. La discretizzazione spaziale dipende da vari fattori. I due principali sono una corretta discretizzazione della zona di processo (nel caso in esame l'avanzamento della frattura è anche il fattore di discretizzazione della zona di processo) e anche della stabilità della soluzione. Avanzamenti più piccoli sono sinonimi di maggiore stabilità perché le forze della zona coesiva variano meno.

Svolgendo diverse analisi, imponendo diversi passi di avanzamento e tenendo fissati i rimanenti parametri, dalla Figura 8.25 si nota come passi di avanzamento  $\leq 2$  mm non influenzino la soluzione e forniscano una buona rappresentazione della zona di processo. Utilizzando un passo  $\Delta l = 1$  mm, quando la zona di processo è pienamente sviluppata, il numero di elementi finiti presenti è pari circa a 40. In [44] viene evidenziato come non ci sia un criterio per stabilire, a priori, quanti elementi siano necessari nella zona di processo tuttavia viene suggerito di utilizzare almeno 10 elementi.



**Figura 8.25:** Grafico tempo - posizione della frattura per diverse dimensioni della mesh.

**Tabella 8.4:** Caratteristiche provino.

Lunghezza (mm)	400
Altezza (mm)	100
Altezza Intaglio (mm)	10
$\Delta l$ (mm)	10
$\sigma_u$ (MPa)	2
$w_{cr}$ (mm)	0.04
$F$ (N)	1000

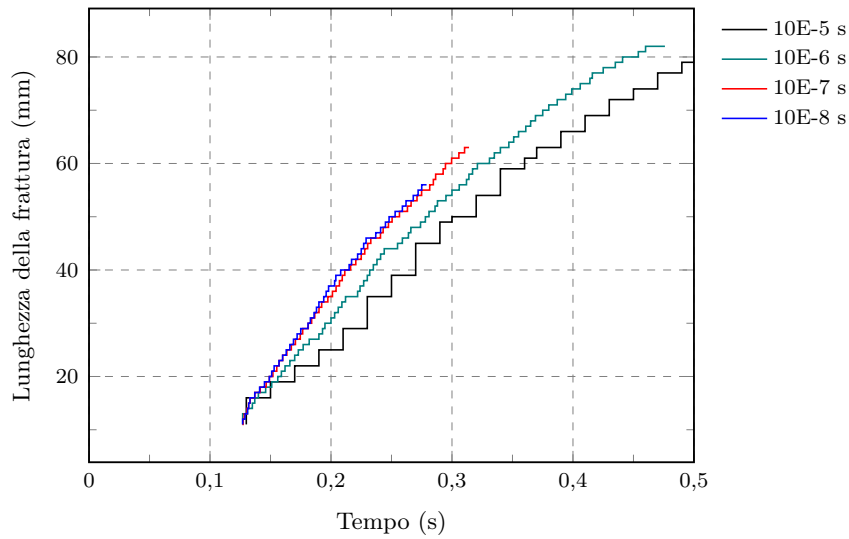
### 8.8.2 Dipendenza dalla discretizzazione temporale

Al pari della discretizzazione spaziale, anche il passo di integrazione  $\Delta t$  può influire sull'analisi. Si vuole indagare in queste analisi quanto incide il passo d'integrazione.

Il provino analizzato presenta le caratteristiche della Figura 8.4.

Una prima differenza rispetto ad un'analisi statica, è che impiegando un tempo pari, o al limite più piccolo di quelli elencati in Figura 8.4 si otterrà una soluzione indipendente dal tempo. Una conferma è data dal grafico riportato in Figura 8.26, in cui si vede che per tempi minori di  $10^{-7}$  (utilizzando un  $\Delta l$  di 1-2 mm) non si ottengono variazioni sensibili nella soluzione, in termini di percorso e velocità di propagazione della frattura neanche variando il carico applicato.





**Figura 8.26:** Grafico tempo - posizione della frattura per diversi tempi di integrazione.

### 8.8.3 Dipendenza dal legame coesivo

La scelta dei parametri della zona di processo non è banale. La lunghezza della zona di processo dipende da:

- Caratteristiche del materiale in base alla sua resistenza a trazione e alla sua fragilità;
- Velocità di propagazione della frattura. La lunghezza della zona di processo diminuisce con la velocità di propagazione della frattura così come evidenziato anche da Abedi [1].

Esiste anche una dipendenza dal legame coesivo impiegato. Mentre la massima tensione  $\sigma_u$  del legame coesivo è la massima tensione di trazione del materiale la massima apertura  $w_{cr}$  dipende dal materiale. Una apertura  $w_{cr}$  grande è sinonimo di propagazione lenta mentre una apertura  $w_{cr}$  piccola è sinonima di una propagazione veloce. Si sono analizzati varie leggi coesive con differenti aperture  $w_{cr}$ . Il provino analizzato presenta le caratteristiche della Figura 8.5. Si osserva come al variare della legge coesiva la frattura avanzi più velocemente.

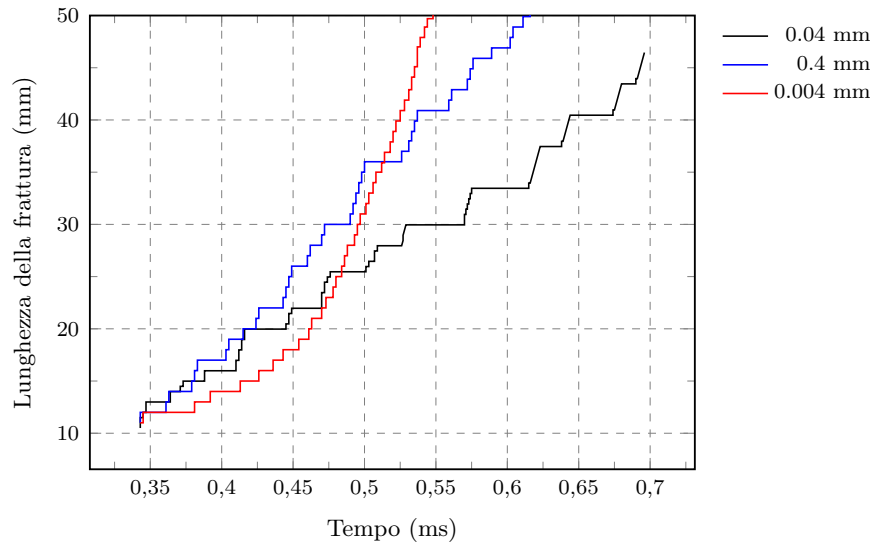
### 8.8.4 Andamento stepwise

L'andamento stepwise che si configura dalle analisi e in cui l'inerzia diventa fondamentale si osserva dal grafico della Figura 8.28. I dati del provino sono riportati nella tabella di Figura 8.6.

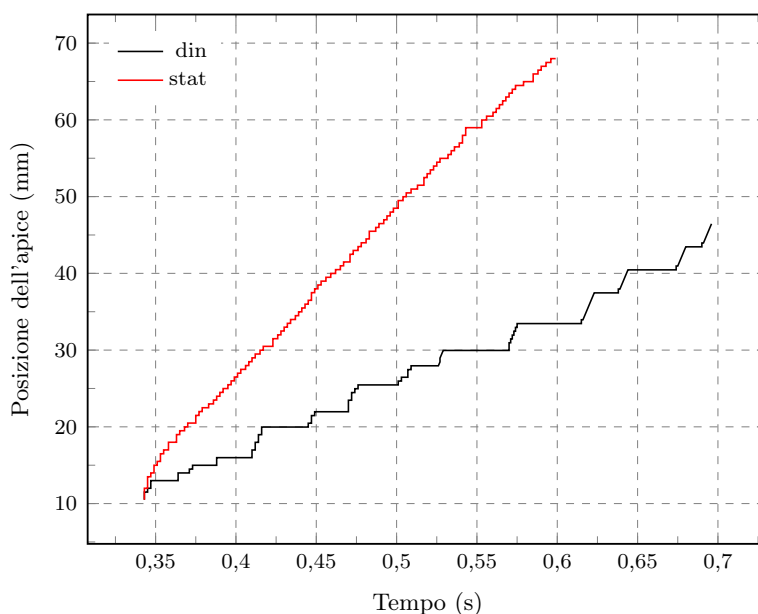
Si osserva come, scegliendo un passo di avanzamento sufficientemente piccolo, ed un passo di integrazione temporale da non violare il limite di

**Tabella 8.5:** Caratteristiche provino.

Lunghezza (mm)	400
Altezza (mm)	100
Altezza Intaglio (mm)	10
$\Delta l$ (mm)	2
$\Delta t$ (s)	$1E - 4$
$w_{cr}$ (mm)	0.04
$F$ (N)	1000

**Figura 8.27:** Grafico tempo - posizione della frattura per vari legami coesivi.**Tabella 8.6:** Materiale e geometria del provino

Quantità	Simbolo	Valore	Unità di misura
Altezza	$h$	100,0	$mm$
Lunghezza	$l$	400,0	$mm$
Dimensioni intaglio	$a$	10,0	$mm$
Modulo elastico	$E$	14700,0	$MPa$
Coefficiente di Poisson	$\nu$	0,2	/
Forza applicata	$F$	100	$N$
Trazione coesiva massima	$\sigma_u$	2,0	$MPa$
Apertura massima ( $\sigma_u$ )	$w_{cr}$	0,04	$mm$



**Figura 8.28:** Grafico tempo - posizione della frattura in cui si evince l'andamento a scatti, o stepwise della frattura.

velocità, l'andamento rispetto al caso statico sia chiaramente diverso. Si nota quanto descritto nei paragrafi precedenti, ovvero un repentino avanzamento della frattura, seguito da un accumulo di energia fino al momento in cui il criterio di rottura viene violato nuovamente.

Si vuole ancora una volta evidenziare come incidono i fattori di discretizzazione spaziale sul provino:

Si osserva come a parità di passo spaziale la soluzione, sotto un certo passo temporale, non varia sensibilmente.

## 8.9 Legame $\Delta t - \Delta l$ con la frattura a scatti

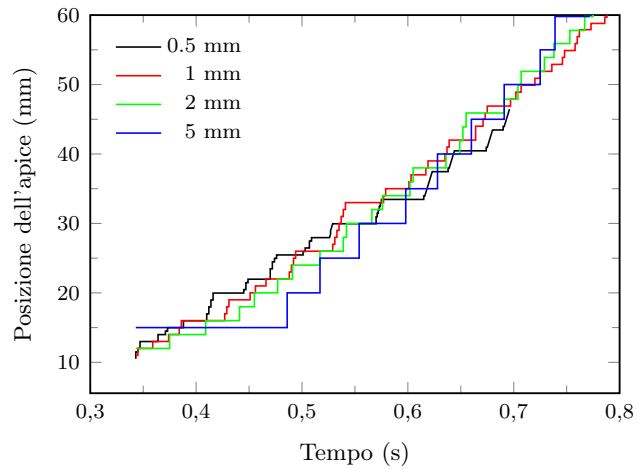
L'algoritmo di frattura utilizzato confronta, in ogni punto del continuo discretizzato, lo sforzo locale con la resistenza del materiale.

In un problema statico, lo sforzo è la soluzione di un'equazione differenziale del tipo:

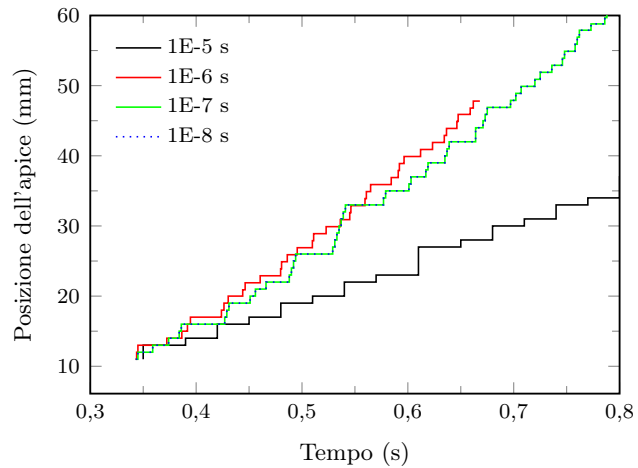
$$\nabla \sigma + \mathbf{b} = \mathbf{0} \quad (8.6)$$

Tradotto in termini discreti, si deve risolvere un sistema di equazioni algebriche in cui le incognite sono gli spostamenti. In un problema dinamico, l'equazione si modifica con le forze di inerzia:

$$\nabla \sigma + \mathbf{b} = \rho \mathbf{a} \quad (8.7)$$



**Figura 8.29:** Grafico tempo - posizione della frattura al variare del passo spaziale di avanzamento. Si nota come l'andamento stepwise diventi meno evidente all'aumentare del passo spaziale di avanzamento.



**Figura 8.30:** Grafico tempo - posizione della frattura al variare del passo temporale di avanzamento. Si nota come l'andamento stepwise diventi meno evidente all'aumentare del passo temporale di integrazione.

In dinamica gli sforzi locali cambiano per tenere conto delle accelerazioni. In termini discreti, si risolve un sistema di equazioni differenziali ordinarie di secondo grado, usando l'algoritmo di Newmark.

In dinamica lo sforzo indotto da un carico si trasmette attraverso onde meccaniche. Le onde meccaniche hanno bisogno di materia per propagarsi. Dato che la materia entra in gioco (tramite  $\rho$ ) entra in gioco anche la massima velocità *teorica* delle onde e quindi anche la massima trasmissione delle tensioni. Quindi le tensioni hanno un limite di propagazione all'aumentare del carico.

Detto questo entra in gioco anche l'avanzamento della frattura che, per come l'algoritmo è costruito, è infinitamente veloce. Si è detto però che esiste un limite alla propagazione della tensione e un prefissato  $\Delta l$  di cui la frattura avanza istantaneamente. Dato che esiste un limite alla propagazione si può considerare come questa distanza  $\Delta l$  venga percorsa in un determinato tempo. In un'analisi dinamica quindi, ci sarà sempre un valore di  $\Delta t$  limite da assumere, un *tempo critico*.

Da quello che si è detto nel paragrafo del caso test si è osservato come sotto un determinato passo temporale  $\Delta t$  (con  $\Delta l$  prefissato) la soluzione non varia sensibilmente. Il passo successivo è quindi quello di poter pensare che esista un legame diretto tra  $\Delta t$  e  $\Delta l$  da cui il tempo critico possa essere un risultato.

Dato che esiste un limite di propagazione nel materiale si può pensare il passo temporale come funzione del parametro di avanzamento  $\Delta l$  fissato e del materiale tramite una relazione del tipo:

$$\Delta t = \Delta t_{cr} = \frac{\Delta l}{c} \quad (8.8)$$

dove  $c$  rappresenta una generica velocità dipendente dal materiale. In questo caso si è considerata la massima velocità che si può propagare nel mezzo, cioè quella delle onde meccaniche (o del suono) espressa dalla (8.9).

$$c_{m,l} = \sqrt{\frac{E}{\rho}} \quad (8.9a)$$

$$c_{m,s} = \sqrt{\frac{G}{\rho}} \quad (8.9b)$$

La dinamica, rispetto alla statica, oltre ad un controllo sui carichi, scandisce il tempo e introduce la massa che algoritmicamente si trasforma nella matrice delle masse. La massa, quindi la densità, introduce gli effetti inerziali che si trasformano, localmente, in un aumento di tensioni e anche il

**Tabella 8.7:** Valori del passo di integrazione in funzione del parametro di avanzamento.

$\Delta l$ (mm)	$\Delta t$ (s)
10	$4.12E - 6$
5	$2.06E - 6$
2	$8.25E - 7$
1	$4.12E - 7$
0.5	$2.06E - 7$

controllo della propagazione delle onde meccaniche. Le propagazione delle onde meccaniche dipende, come si osserva, solo dalle caratteristiche elastiche  $E$  e  $G$  e dalla densità del materiale  $\rho$ .

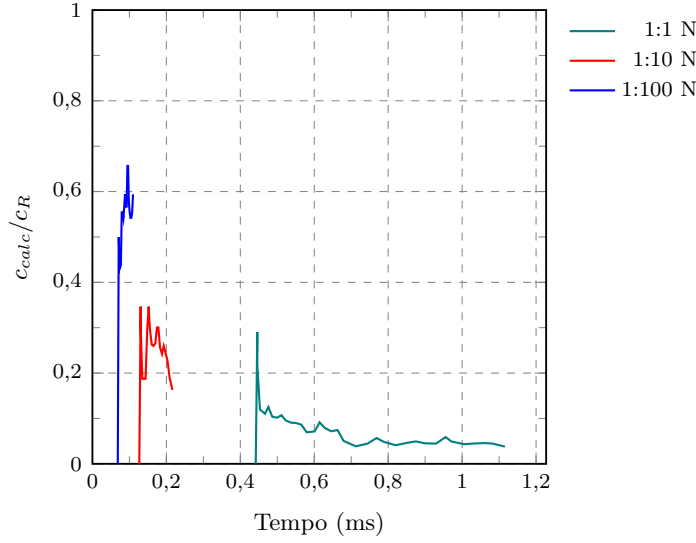
Considerando i passi di avanzamento impiegati nello studio della prova di flessione in tre punti si ottengono i tempi riportati in Figura 8.7. Utilizzando diverse time table si è visto generalmente che un tempo di  $1E-7$ , utilizzando un passo spaziale  $\Delta l$  di 0.50-2 mm, non porti a variazioni sensibili della soluzione.

Questo limite fisico impone che anche abbassando il tempo il comportamento non varia perché la tensione ha necessariamente, per quanto detto, bisogno di un tempo di propagazione. Questo limite potrebbe essere dato dalla massa. Quindi la presenza delle forze inerziali dovrebbe essere sempre presente non solo per rappresentazione gli effetti inerziali che possono essere tuttavia importanti come in una qualsiasi analisi dinamica, dove si presenta una rapida variazione delle quantità materiali del sistema meccanico, ma anche per imporre il limite di propagazione delle onde meccaniche.

Una volta scelto il  $\Delta l$  (tale da soddisfare i criteri della zona di processo) la scelta del passo temporale  $\Delta t$  diventerebbe una funzione  $\Delta t = f(\Delta l)$  tale da soddisfare il criterio fisico introdotto cioè quello di rispettare la massima propagazione delle onde nel materiale. Sotto un tale limite le analisi non sono più influenzate.

Alla luce di quanto visto la velocità di propagazione della frattura può essere suddivisa in due categorie:

- Velocità globale: si tratta essenzialmente della velocità calcolata tenendo in conto la posizione della frattura rispetto al tempo totale. Viene calcolata con le differenze finite in tre punti. Questa velocità viene a dipendere essenzialmente dalla velocità di applicazione dei carichi. Infatti il criterio di rottura si raggiunge in tempi minori e la frattura avanza più velocemente.
- Velocità locale: si tratta della velocità definita a livello locale, cioè quando la frattura avanza velocissima della quantità  $\Delta l$ . Questa velocità non è nota ed è oggetto di studio in questo scritto. Si presuppone vada a dipendere essenzialmente dal materiale.



**Figura 8.31:** Grafico tempo - velocità al variare dell'inclinazione della table normalizzata con la velocità delle onde di Rayleigh.

### 8.9.1 Velocità di propagazione

La velocità dell'apice della frattura è differente rispetto alla velocità *materiale* calcolata ad ogni passo discreto dall'algoritmo di integrazione temporale (Newmark per le analisi in esame). Quando si utilizzano schemi alle differenze finite, la pratica comune per calcolare la velocità consiste nel memorizzare la posizione dell'apice ad ogni istante  $t_n$  e di applicare formule di differenziazione numerica. Ad esempio è possibile calcolare la velocità come:

$$f'(t_i) = \frac{f(t_{i+1}) - f(t_{i-1}))}{2h} \quad (8.10)$$

in cui  $2h = x_{i+1} - x_i$ . Una volta calcolata la velocità è possibile interpolare i dati per ottenere una curva sufficientemente regolare.

Freund ([17]) dimostrò analiticamente che la velocità di propagazione della frattura in un corpo è limitata dalla massima velocità delle onde di Rayleigh che si propagano in quel corpo, calcolata come:

$$c_d = \sqrt{\frac{E(1-\nu)}{\rho(1+\nu)(1-2\nu)}}, \quad c_s = \sqrt{\frac{E}{2\rho(1+\nu)}} \quad (8.11)$$

$$c_R \approx c_s \frac{0.862 + 1.14\nu}{1 + \nu} \quad (8.12)$$

in cui  $c_d$ ,  $c_s$  e  $c_R$  rappresentano rispettivamente le velocità delle onde di dilatazione, di taglio e di Rayleigh. Si riportano di nella Tabella 8.31 alcune velocità di propagazione *globali* calcolate. Come si evince dalle analisi, la

velocità dell'apice della frattura risulta sempre inferiore alla velocità di propagazione delle onde di Rayleigh. Non si può affermare con certezza che la velocità delle onde di Rayleigh  $c_R$  rappresenti un limite superiore alla velocità di propagazione della frattura in un corpo ma sicuramente vari autori, oltre a questo lavoro, oltre a indagini sperimentali tra cui Larcher [26] sono arrivati a simili conclusioni. Sono quindi necessarie ulteriori indagini per stabilirlo con certezza.

### 8.9.2 Tempo di integrazione

Il passo di integrazione entra in gioco fondamentalmente per due ragioni:

- Avere una soluzione accurata;
- Cogliere il fenomeno fisico.

Si è visto che la velocità di applicazione del carico influisce sul fenomeno fisico, quindi in tal caso c'è bisogno di un tempo piccolo per avere una soluzione accurata. Nel caso invece del fenomeno fisico (la propagazione della frattura) un passo temporale piccolo è necessario per poter cogliere l'avanzamento stesso che si sviluppa dell'ordine di grandezza dei milionesimi di secondo. Generalmente il passo temporale dipende esclusivamente dalla necessità di cogliere il fenomeno fisico.

La capacità dell'algoritmo di fornire informazioni dipende dal passo temporale  $\Delta t$ . Questa capacità può essere suddivisa in quantità e qualità. Abbassando il passo temporale  $\Delta t$  l'algoritmo di integrazione nel tempo è capace a fornire soluzioni più dettagliate, quindi da un punto di vista algoritmico per poter cogliere il fenomeno della dinamica nella frattura si devono utilizzare passi di integrazione molto bassi dell'ordine, in genere, dei decimilionesimi di secondo.



## Capitolo 9

# Conclusioni

In questo lavoro di tesi è stata trattata la simulazione numerica della propagazione della frattura in solidi bidimensionali. Tramite gli elementi finiti tradizionali non è possibile modellare un campo di spostamenti discontinuo, quindi, per riprodurre una frattura in avanzamento è necessario un continuo aggiornamento del dominio e, di conseguenza, della mesh. Si effettua quindi un *remeshing adattivo* ogni qualvolta la frattura avanza. Sebbene siano presenti metodi alternativi, che permettono di trattare il problema in maniera più agevole (tra i quali si citano, ad esempio, il Metodo XFEM o l'impiego di elementi di interfaccia) si ritiene che la procedura impiegata porti a risultati più realistici, in quanto non si hanno mesh dipendenza o percorso di frattura stabilito a priori.

Per una simulazione accurata del fenomeno, è stata introdotta la tempo dipendenza, implementando l'algoritmo di integrazione temporale di Newmark. Si ritiene infatti, come si è visto da una semplice simulazione tramite software commerciale, che gli effetti inerziali indotti dalla propagazione della frattura non siano trascurabili.

Se la velocità con cui variano i carichi imposti è molto piccola è lecito trascurare gli effetti inerziali, a patto che anche la corrispondente evoluzione del sistema meccanico sia lenta, o meglio, la velocità con cui varia lo stato del materiale (deformazione e frattura) siano entrambe molto piccole.

Anche supponendo che l'evoluzione sia quasistatica, ci sono degli istanti in cui il modello quasistatico impone un salto della frattura, che si potrebbe considerare come un brevissimo intervallo di crescita velocissima. È chiaro che in questi intervalli viene meno l'ipotesi che l'evoluzione dello stato del materiale sia lenta. Quindi, in questi brevissimi intervalli non si può trascurare l'inerzia. L'evoluzione quasistatica della frattura è quindi da considerare come un caso limite di processi dinamici.

L'introduzione di tale algoritmo ha richiesto un sistema di proiezione delle variabili tra un istante temporale ed un altro, eventualmente su di un

nuovo dominio e su di una nuova mesh. Inoltre per avere delle variabili consistenti è stato implementato anche un ciclo di bilanciamento, che si occupa di ristabilire l'equilibrio dinamico delle variabili proiettate sulla nuova mesh/dominio, sotto l'effetto delle forzanti al passo precedente.

L'algoritmo che gestisce la frattura, per costruzione, prevede un avanzamento istantaneo (fissato a priori dall'utente) all'interno dello stesso step temporale, ogni volta che è rispettato un predeterminato criterio di rottura. Tale avanzamento, che numericamente avviene istantaneamente, fisicamente avviene in un tempo finito. Per poter rappresentare in maniera fisicamente coerente il fenomeno quindi, risulta necessario porre un limite alla propagazione istantanea della frattura.

Una volta trovato il passo di avanzamento che rende la soluzione indipendente dalla mesh, è necessario imporre il passo di integrazione temporale in modo tale da non violare il limite alla propagazione della frattura, ovvero la velocità delle onde di Rayleigh che si propagano nel solido.

Fissati passo di avanzamento e tempo coerenti con la fisica del problema, è stata fatta un'analisi dinamica su di un provino relativo alla prova di flessione in tre punti. Quello che si è osservato è che la frattura si propaga all'interno del corpo in maniera graduale e non in maniera continua. Il fenomeno, fisicamente, presenta un'analogia con i terremoti in cui si ha un accumulo di energia seguito da un repentino movimento delle faglie. Diversamente, analizzando il fenomeno senza la partecipazione delle forze inerziali, questo comportamento non emerge.

# Appendice A

## Implementazione

In questa appendice vengono riportate alcuni passaggi del codice.

### A.1 Implementazione matrici

Si vogliono ora presentare alcuni passaggi di codice in cui si mostra la formazione delle matrici di rigidezza, di smorzamento e delle masse.

#### A.1.1 Matrice di rigidezza

```
1  bool
2  TFemElasticIntegrator::BuildK()
3  {
4      TFemIntegrPoint* pIntgrPoint = m_pIntgrPoint;
5
6      int kDim = m_pElement->GetElementDoF()->GetTotalDoF();
7      m_K.ChangeDimension(kDim, kDim);
8
9      static TMatrix scalD;
10     static TMatrix gaussK;
11
12     for (int i = 0; i < pIntgrPoint->GetNumPoint(); i++)
13     {
14         double ar = pIntgrPoint->GetWeight(i) * m_DetJ[i];
15
16         Product(ar, m_D, &scalD);
17         Product_At_B_A(m_B[i], scalD, &gaussK);
18         Sum(gaussK, &m_K);
19     }
20     return true;
21 }
```

#### A.1.2 Matrice di smorzamento

```

1  bool
2  TFemElasticIntegrator::BuildDamping()
3  {
4      static TMatrix alphaM;
5      static TMatrix betaK;
6      BuildK();
7      Product(m_AlphaRayleigh, m_K, &alphaM);
8      BuildMass();
9      Product(m_BetaRayleigh, m_M, &betaK);
10     Sum(alphaM, &m_C);
11     Sum(betaK, &m_C);
12     return true;
13 }

```

### A.1.3 Matrice delle masse

```

1  bool
2  TFemElasticIntegrator::BuildMass()
3  {
4      TElasticProperty* pProp = (TElasticProperty*)m_pElement->
5          GetProperty();
6      TFemIntegrPoint* pIntgrPoint = m_pIntgrPoint;
7
8      int kDim = m_pElement->GetElementDoF()->GetTotalDoF();
9      m_M.ChangeDimension(kDim, kDim);
10
11     static TMatrix scalM;
12     static TMatrix gaussM;
13
14     for (int i = 0; i < pIntgrPoint->GetNumPoint(); i++)
15     {
16         double ar = pProp->m_Density * pIntgrPoint->GetWeight(i)
17             * m_DetJ[i];
18         Product(ar, m_N[i], &scalM);
19         TProduct(scalM, m_N[i], &gaussM);
20         Sum(gaussM, &m_M);
21     }
22     return true;
23 }

```

### A.1.4 Matrice tangente

```

1  bool
2  TFemTimeNewmark::BuildTangent(TFemElementIntegrator*
3      pIntegrator)
4  {
5      // Matrice di Massa fattorizzata
6      // (2/(m_Beta2*dt*dt))*M

```

```

7 // Costruzione della Matrice nel locale
8 pIntegrator->BuildMass();
9 pIntegrator->GetMass(m_pM);
10
11 static TMatrix tmpTMatrixM;
12 Product(m_factor4, *m_pM, &tmpTMatrixM);
13 m_T.Copy(tmpTMatrixM);
14 // Matrice di rigidezza fattorizzata
15 // Costruzione della Matrice nel locale
16 pIntegrator->BuildElastic();
17 pIntegrator->GetElastic(m_pK);
18 Sum(*m_pK, &m_T);
19
20 // Matrice di Damping fattorizzata
21 // (2*m_Beta1)/(m_Beta2*dt)*M
22 // Costruzione della Matrice nel locale
23 switch (m_Damping)
24 {
25
26 case DAMPING_VISCOUS:
27     pIntegrator->SetViscousDamping(m_Viscosity);
28     pIntegrator->BuildDamping(1);
29     break;
30 case DAMPING_RAYLEIGH:
31     pIntegrator->SetRayleighDamping(m_AlphaRayleigh,
32     m_BetaRayleigh);
33     pIntegrator->BuildDamping(2);
34     break;
35 case DAMPING_NULL:
36     pIntegrator->BuildDamping(3);
37     break;
38 }
39 pIntegrator->GetDamping(m_pC);
40
41 static TMatrix tmpTMatrixC;
42 Product(m_factor5, *m_pC, &tmpTMatrixC);
43 Sum(tmpTMatrixC, &m_T);
44
45 return true;
46 }

```

## A.2 Cicli Newton-Raphson

Si riportano di seguito, per esteso, i due cicli descritti nel capitolo 6.

### A.2.1 Ciclo di bilanciamento

```

1 bool
2 TNonLinearProblem::BalanceProblem()
3 {

```

```

4
5     printf("\r\n\t      Start of balance step          ");
6     double res = Norm(m_PrevExtForce, 0, m_PrevExtForce.NumRows
7     ());
8     printf("\r\n\t      PROJECTION-extforce:      \t%.10e", res);
9
10    if (!ArrayLoadCase().size())
11        return false; // Nessuna condizione di carico presente
12
13    m_NumIter = 0;
14    res = 1;
15
16    do
17    {
18        //Imposta I parametri di Newmark
19        m_pNewmark->SetBeta(0.5, 0.5);
20        //Imposta il tipo di smorzamento
21        m_pNewmark->SetDamping(TFemTimeNewmark::DAMPING_NULL);
22
23        m_pNewmark->ComputeNewmarkParameters();
24        //Calcola la Matrice Tangente
25        m_pNewmark->AssemblyTangentMatrix(*m_pGlobalTangent);
26
27        // Aggiorna l'intensita delle forze coesive che dipendono
28        // dagli spostamenti
29        UpdateFractureForce(m_PrevSol);
30        //Forma il Vettore dei Termini Noti
31        ArrangeContribution(m_NumIter, m_pNewmark);
32        // Parametri temporali
33        double dt = TFemModel::m_Time.GetTimeStep();
34        m_CurrTime = TFemModel::m_Time.GetCurrTime();
35        m_IniTime = TFemModel::m_Time.GetInitialTime();
36
37        if (m_NumIter == 0)
38        {
39
40            m_Residual.Copy(m_PrevExtForce);
41
42            for (int k = 0; k < m_PrevExtForce.NumRows(); k++)
43            {
44                m_Residual(k, 0) -= m_PrevTotalForce(k, 0);
45            }
46        }
47        else
48        {
49            m_Residual.Copy(m_PrevExtForce);
50
51            for (int k = 0; k < m_PrevExtForce.NumRows(); k++)
52            {
53                m_Residual(k, 0) -= m_tmpTotalForce(k, 0);
54            }
55        }
56    }

```

```

56
57
58 // Correzione spostamenti
59 m_PrevDeltaSol.Copy(m_Residual);
60 if (!m_pGlobalTangent->Solve(m_PrevDeltaSol, e_LDL))
61 {
62     PrintOut("\r\n CICLO 1: Matrice di rigidezza singolare.
63         ");
64     ASSERT(false);
65     return false;
66 }
67 // Aggiorna il valore degli spostamenti nel loop
68 m_pNewmark->BalanceDisplacement(m_PrevDeltaSol, m_PrevSol
69     );
70 // Aggiorna il valore delle accelerazioni nel loop
71 m_pNewmark->BalanceAcceleration(m_PrevDeltaSol,
72     m_PrevASol);
73 // Aggiorna il valore delle velocita nel loop
74 m_pNewmark->BalanceVelocity(m_PrevDeltaSol, m_PrevVSol);
75 m_tmpElasticForce.Reset();
76 m_tmpDampingForce.Reset();
77 m_tmpInertiaForce.Reset();
78 m_tmpTotalForce.Reset();
79
80 // Ricalcola le Forze Elastiche
81 m_pNewmark->ComputeElasticForce(m_PrevSol,
82     m_tmpElasticForce, m_nActiveLC);
83 // Ricalcola le Forze di Damping
84 m_pNewmark->ComputeDampingForce(m_PrevVSol,
85     m_tmpDampingForce, m_nActiveLC);
86 // Ricalcola le Forze d'Inerzia
87 m_pNewmark->ComputeInertiaForce(m_PrevASol,
88     m_tmpInertiaForce, m_nActiveLC);
89
90 Sum(m_tmpElasticForce, &m_tmpTotalForce);
91 Sum(m_tmpDampingForce, &m_tmpTotalForce);
92 Sum(m_tmpInertiaForce, &m_tmpTotalForce);
93
94 // Ricalcolo del residuo
95 m_Residual.Copy(m_PrevExtForce);
96 for (int k = 0; k < m_PrevExtForce.NumRows(); k++)
97 {
98     m_Residual(k, 0) -= m_tmpTotalForce(k, 0);
99 }
100
101 res = ResidualNorm(m_Residual, m_PrevExtForce);
102
103 double resIter = Norm(m_Residual, 0, m_Residual.NumRows
104     ());
105 printf("\r\n\t RESIDUAL N ITER: \t%.10e", res);

```

```

103     printf("\r\n\t  NUM ITER:          %100d", m_NumIter + 1);
104
105     m_NumIter++;
106
107 } while (res > m_ResToll);
108
109
110
111 printf("\r\n\t      End of balance step      ");
112
113
114 return true;
115 }

```

### A.2.2 Ciclo Newton-Raphson

```

1  bool
2  TNonLinearProblem::SolveProblem()
3  {
4      if (!ArrayLoadCase().size())
5          return false; // Nessuna condizione di carico presente
6
7      m_NumIteration = 0;
8      double res = 1;
9
10     TMatrix tempForce;
11     do
12     {
13         printf("\r\n\tIter. %3d  ", m_NumIteration + 1);
14         m_ExtForce.Reset();
15
16         m_pNewmark->SetBeta(0.50, 0.5);
17         //Imposta il tipo di smorzamento
18         m_pNewmark->SetDamping(TFemTimeNewmark::DAMPING_NULL);
19
20         m_pNewmark->ComputeNewmarkParameters();
21         //Calcola la Matrice Tangente
22         m_pNewmark->AssemblyTangentMatrix(*m_pGlobalTangent);
23         //Calcola la Matrice Elastica
24         m_pNewmark->AssemblyElasticMatrix(*m_pGlobalElastic);
25         //Calcola la Matrice di Smorzamento
26         m_pNewmark->AssemblyDampingMatrix(*m_pGlobalDamping);
27         //Calcola la Matrice delle Masse
28         m_pNewmark->AssemblyMassMatrix(*m_pGlobalMass);
29         //Forma il Vettore dei Termini Noti
30         m_pNewmark->BuildKnownValue(m_ExtForce, m_TotalForce);
31         // Aggiunge nel Termine Noto e nella Matrice Tangente
32         // i contributi delle forze coesive
33         ArrangeFractureForces(m_NumIteration);
34         // Aggiunge nel Termine Noto e nella Matrice Tangente
35         // i contributi dei constraints
36         ArrangeConstraints(m_NumIteration);

```



```

37     m_tmpExtForce = m_ExtForce;
38
39     // Parametri Temporal
40     double dt = TFemModel::m_Time.GetTimeStep();
41     m_CurrTime = TFemModel::m_Time.GetCurrTime();
42     m_IniTime = TFemModel::m_Time.GetInitialTime();
43
44     if (m_NumIteration == 0)
45     {
46         m_Sol.Reset();
47         m_DeltaSol.Reset();
48
49         if (m_CurrTime == m_IniTime)
50         {
51             m_Residual.Copy(m_ExtForce);
52             m_PrevSol.Reset();
53             m_PrevVSol.Reset();
54             m_PrevASol.Reset();
55             m_Sol.Reset();
56
57         }
58         else
59         {
60
61             m_TotalForce.Reset();
62             m_ElasticForce.Reset();
63             m_DampingForce.Reset();
64             m_InertiaForce.Reset();
65
66             Sum(m_PrevCycleElasticForce, &m_TotalForce);
67             Sum(m_PrevCycleDampingForce, &m_TotalForce);
68             Sum(m_PrevCycleInertiaForce, &m_TotalForce);
69
70             m_Residual.Copy(m_ExtForce);
71             for (int k = 0; k < m_ExtForce.NumRows(); k++)
72             {
73                 m_Residual(k, 0) -= m_PrevCycleTotalForce(k, 0);
74             }
75
76
77         }
78     }
79
80 -
81     else
82     {
83         m_Residual.Copy(m_ExtForce);
84
85         for (int k = 0; k < m_ExtForce.NumRows(); k++)
86         {
87             if (k < m_nNormalEquat)
88                 m_Residual(k, 0) -= m_TotalForce(k, 0);
89             else
90                 m_Residual(k, 0) = 0;

```

```

91     }
92
93 }
94
95
96 res = Norm(m_Residual, 0, m_ExtForce.NumRows());
97 printf("\r\n\t      R-Norm:      \t%.10e", res);
98
99
100 // Il Solutore calcola la soluzione all'iterazione
      attuale
101 m_DeltaSol.Copy(m_Residual);
102 if (!m_pGlobalTangent->Solve(m_DeltaSol, e_LDL))
103 {
104     PrintOut("\r\n CICLO2: Matrice di rigidezza singolare."
              );
105     return false;
106 }
107
108 // Aggiorna il Valore degli spostamenti nel loop
109 m_pNewmark->UpdateDisplacement(m_DeltaSol, m_Sol,
      m_NumIteration);
110 // Aggiorna il valore delle accelerazioni nel loop
111 m_pNewmark->UpdateAceleration(m_PrevSol, m_Sol,
      m_PrevVSol, m_PrevASol, m_ASol, m_NumIteration);
112 // Aggiorna il valore delle velocita nel loop
113 m_pNewmark->UpdateVelocity(m_Sol, m_ASol, m_PrevVSol,
      m_PrevASol, m_VSol, m_NumIteration);
114
115
116 m_ElasticForce.Reset();
117 m_DampingForce.Reset();
118 m_InertiaForce.Reset();
119 m_TotalForce.Reset();
120
121 // Ricalcola le Forze Elastiche
122 m_pNewmark->ComputeElasticForce(m_Sol, m_ElasticForce,
      m_nActiveLC);
123 // Ricalcola le Forze di Damping
124 m_pNewmark->ComputeDampingForce(m_VSol, m_DampingForce,
      m_nActiveLC);
125 // Ricalcola le Forze d'Inerzia
126 m_pNewmark->ComputeInertiaForce(m_ASol, m_InertiaForce,
      m_nActiveLC);
127 // Aggiorna l'intensita delle forze coesive che dipendono
      dagli spostamenti
128 UpdateFractureForce(m_Sol);
129 // Somma le forze elastiche, di smorzamento e d'inerzia
      nel vettore delle forze totali
130 Sum(m_ElasticForce, &m_TotalForce);
131 Sum(m_DampingForce, &m_TotalForce);
132 Sum(m_InertiaForce, &m_TotalForce);
133

```

```
134 // Aggiunge le reazioni dei link nel vettore delle forze
      interne m_ElasticForce.
135 // La reazione del link e' infatti da considerare come
      una forza interna.
136 for (int k = 0; k < (int)ArrayLink().size(); k++)
137 {
138     ArrayLink()[k]->AddLinkReactions(m_Sol, m_TotalForce,
      m_nActiveLC);
139 }
140
141 for (int k = 0; k < m_ExtForce.NumRows(); k++)
142 {
143     if (k < m_nNormalEquat)
144         m_Residual(k, 0) = m_ExtForce(k, 0) - m_TotalForce(k,
      0);
145     else
146         m_Residual(k, 0) = 0;
147 }
148
149 // Ricalcolo del residuo
150 res = ResidualNorm(m_Residual, m_ExtForce);
151 printf("\r\n\t F-Norm: \t%.10e", res);
152
153 m_NumIteration++;
154 if (m_NumIteration > m_NumMaxIteration)
155 {
156     printf("\r\n\t *** Solution is not converged in %d
      iteration\r\n", m_NumIteration);
157
158     return false;
159 }
160 } while (res > m_Tolerance);
161
162 return true;
163 }
```



## Appendice B

# Validazione del codice di calcolo

### B.1 Introduzione

In questa appendice saranno presentati alcuni test per verificare la bontà di quanto implementato nel codice di calcolo. Si ritiene che sia utile pratica ingegneristica fare sempre riferimento a modelli più semplici possibili. Secondo questo criterio si è andati a implementare attraverso l'uso di Visual Basic ed Excel delle utility che permettono di studiare il problema in modo semplice.

In particolare si farà riferimento all'implementazione dello schema di Newmark implicito.

### B.2 Codice sistema ad 1 gdl

**Listing B.1:** macro

```
1      '
2      ' DICHIARAZIONE VARIABILI
3      '
4      Private K As Double
5      Private KT As Double
6      Private C As Double
7      Private M As Double
8      Private DT As Double
9      Private DispToll As Double
10     Private ResToll As Double
11     Private DispNorm As Double
12     Private ResNorm As Double
13     Private Res1 As Double
14     Private Res2 As Double
15     Private ERR As Double
```

```

16 Private ITER As Integer
17 Private Gamma As Double
18 Private Beta As Double
19 Private u0 As Double
20 Private v0 As Double
21 Private a0 As Double
22 Private i As Integer
23 Private j As Integer
24 Private uk As Double
25 Private vk As Double
26 Private ak As Double
27 Private ukp1 As Double
28 Private vkp1 As Double
29 Private akp1 As Double
30 Private vp As Double
31 Private ap As Double
32 Private Duk As Double
33 Private DDuk As Double
34 Private deltalambda As Double
35 Private lambdak As Double
36 Private lambdakp1 As Double
37 Private FORZANTE As Double
38 Private NORM As Double
39 Private PrevElasticForce As Double
40
41 '
42 ' LETTURA DATI
43 '
44 Sub READINPUT()
45 '
46 ' DATI SISTEMA SDOF
47 '
48 K = Worksheets(NEWMARK_LOOP).Cells(4, 5).Value
49 C = Worksheets(NEWMARK_LOOP).Cells(5, 5).Value
50 M = Worksheets(NEWMARK_LOOP).Cells(6, 5).Value
51 DT = Worksheets(NEWMARK_LOOP).Cells(7, 5).Value
52 DispToll = Worksheets(NEWMARK_LOOP).Cells(8, 5).Value
53 ResToll = Worksheets(NEWMARK_LOOP).Cells(9, 5).Value
54 ITER = Worksheets(NEWMARK_LOOP).Cells(10, 5).Value
55 '
56 ' PARAMETRI NEWMARK
57 '
58 Gamma = Worksheets(NEWMARK_LOOP).Cells(12, 5).Value
59 Beta = Worksheets(NEWMARK_LOOP).Cells(13, 5).Value
60 '

```

```

61  ' CONDIZIONI AL CONTORNO
62  '
63  u0 = Worksheets(NEWMARK_LOOP).Cells(15, 5).Value
64  v0 = Worksheets(NEWMARK_LOOP).Cells(16, 5).Value
65  a0 = Worksheets(NEWMARK_LOOP).Cells(17, 5).Value
66  End Sub
67
68  '
69  ' SETUP NEWMARK
70  '
71  Sub INITIALIZENEWMARK()
72  uk = u0
73  vk = v0
74  ak = a0
75  i = 1
76  j = 54
77  PrevElasticForce = K uk
78  ResNorm = 1
79  DispNorm = 1
80  Res1 = 1
81  Res2 = 1
82  End Sub
83
84  '
85  ' CALCOLO RIGIDEZZA TANGENTE
86  '
87  Sub KTANGENT()
88  KT = (1 - (Beta DT DT)) M + K
89  End Sub
90
91  '
92  ' CALCOLO PREDICTOR
93  '
94  Sub PREDICTOR()
95  ap = -(1 - (Beta DT)) vk - (1 - (2 Beta) - 1) ak
96  vp = (1 - (Gamma Beta)) vk + (1 - Gamma (2 Beta)) ak
97  End Sub
98  '
99  ' AGGIORNAMENTO VALORI
100  '
101  Sub UPDATEVALUE()
102  DDuk = DDuk + Duk
103  uk = uk + Duk
104  ukp1 = uk
105  vkp1 = vp + Gamma (Beta DT) DDuk
106  akp1 = ap + 1 - (Beta DT DT) DDuk
107  uk = ukp1
108  vk = vkp1
109  ak = akp1
110  'lambdak = lambdakp1
111  End Sub

```

```

112     '
113     ' AGGIORNAMENTO FORZA
114     '
115     Sub UPDATEFORCE()
116         PrevElasticForce = K uk
117     End Sub
118     '
119     ' CALCOLO NORMA
120     '
121     Sub NORMA()
122         DispNorm = Abs(Duk uk)
123         ResNorm = Abs(Res2 Res1)
124     End Sub
125     '
126     ' FULL NEWTON
127     '
128     '
129     Sub FULLNEWTON()
130         Duk = 0
131         DDuk = 0
132         FORZANTE = Worksheets(NEWMARK_LOOP).Cells(j, 5).Value
133         Do
134             If i = 1 Then
135                 Duk = (1 KT) (FORZANTE - M ap - PrevElasticForce)
136                 Res1 = (FORZANTE - M ap - K uk)
137             Else
138                 Duk = (1 KT) (FORZANTE - M ak - K uk)
139             End If
140             UPDATEVALUE
141             Res2 = (FORZANTE - M ak - K uk)
142             NORMA
143             i = i + 1
144             'Loop Until (DispNorm DispToll) '(DispNorm DispToll And
145                 ResNorm ResToll)
146             If ResNorm ResToll And DispNorm DispToll Then Exit Do
147         Loop
148     End Sub
149     '
150     ' SCRITTURA DATI
151     '
152     Sub OUTPUT()
153         Worksheets(NEWMARK_LOOP).Cells(j, 6).Value = ukp1
154         Worksheets(NEWMARK_LOOP).Cells(j, 7).Value = vkp1
155         Worksheets(NEWMARK_LOOP).Cells(j, 8).Value = akp1
156         Worksheets(NEWMARK_LOOP).Cells(j, 9).Value = i
157         Worksheets(NEWMARK_LOOP).Cells(j, 10).Value = DispNorm
158         Worksheets(NEWMARK_LOOP).Cells(j, 11).Value = ResNorm
159         'Worksheets(NEWMARK_LOOP).Cells(j, 12).Value = FORZANTE
160     End Sub
161     '
162     ' MAIN DEL PROGRAMMA
163     '
164     Sub NEWMARK()
165         READINPUT

```



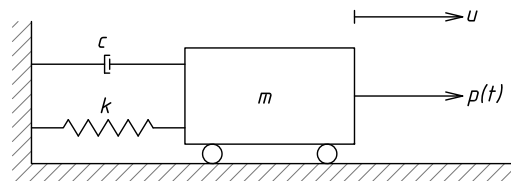
```

165     INITIALIZENEWMARK
166     KTANGENT           'costante in questo caso
167 Do
168     PREDICTOR
169     FULLNEWTON
170     UPDATEFORCE
171     OUTPUT
172     j = j + 1
173 Loop Until (j = 354)
174 End Sub
175
176 Sub RESET_CELL()
177 '
178 ' RESET_CELLE
179 '
180     Range(F54L1000).Select
181     Selection.ClearContents
182
183 End Sub

```

### B.3 Risultati sistema ad 1 gdl

Si considera il sistema ad 1 grado di libertà illustrato in Figura B.1.



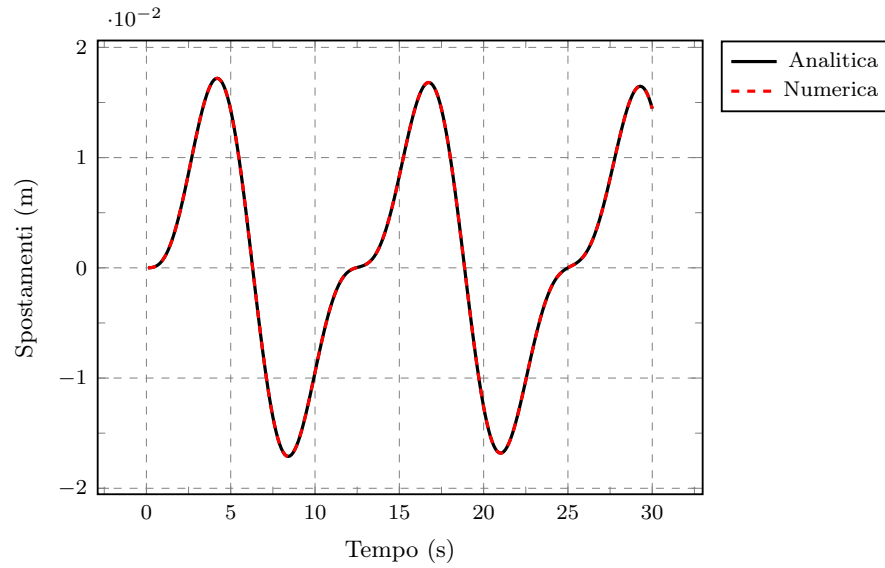
**Figura B.1:** Sistema ad 1 GDL.

L'equazione del moto per questo sistema ha la forma:

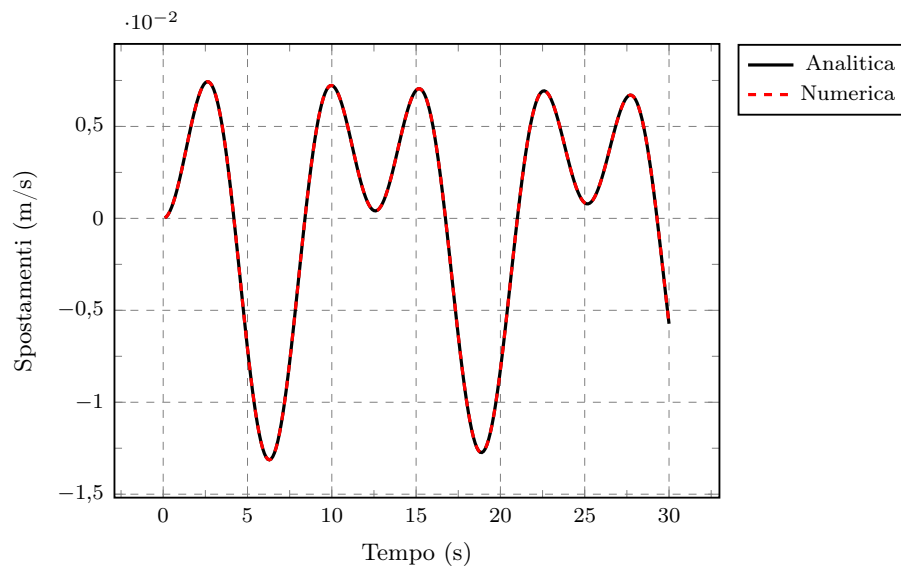
$$\begin{cases} \ddot{u} + 2\xi\omega\dot{u} + \omega^2u = p(t) \\ u(0) = u_0 \quad \dot{u}(0) = v_0 \end{cases} \quad (\text{B.1})$$

Si considera una vibrazione forzata con i parametri  $\omega = 1$ ,  $\xi = 0.05$ ,  $p(t) = \sin 0.5t$ ,  $u_0 = v_0 = 0$ . Il time step impiegato è pari a  $\Delta t = 0.1$  s. I parametri di Newmark (schema GN11)  $\beta_1$  e  $\beta_2$  risultano pari entrambi a 0.5. Si riportano di seguito i risultati.

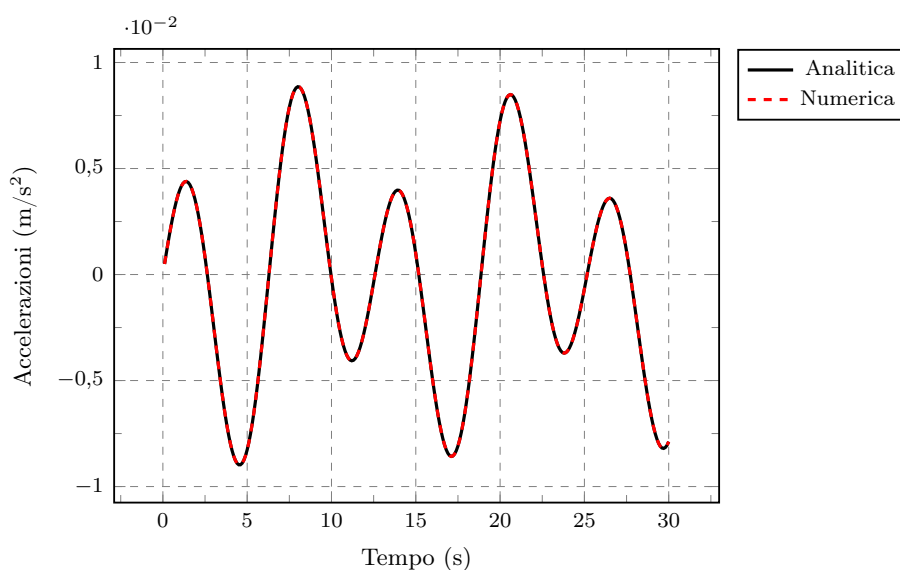
Come si evince dai grafici, non vi sono differenze sostanziali e si può affermare che lo schema implementato sia corretto.



**Figura B.2:** Risposta del sistema - spostamenti.



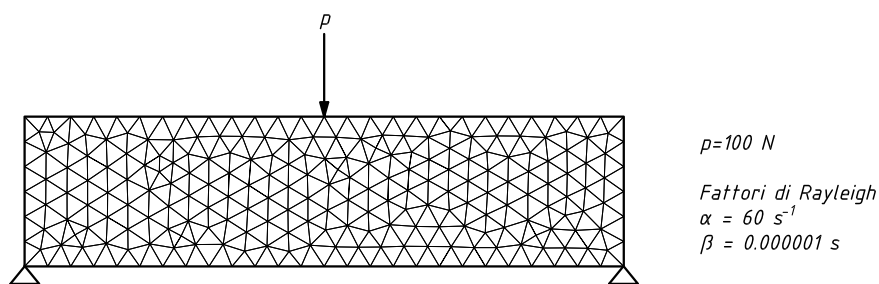
**Figura B.3:** Risposta del sistema - velocità.



**Figura B.4:** Risposta del sistema - accelerazioni.

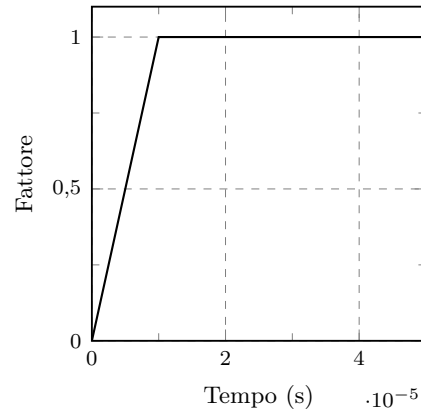
## B.4 Confronto con codice di calcolo commerciale

Si vuole fare anche un confronto con il software commerciale Straus7 della G+D Computing. In particolare, il confronto è fatto sulla geometria della prova di flessione in tre punti vista nei capitoli precedenti. In Figura B.5 è riportato il modello impiegato con i parametri dello smorzamento alla Rayleigh assunti.

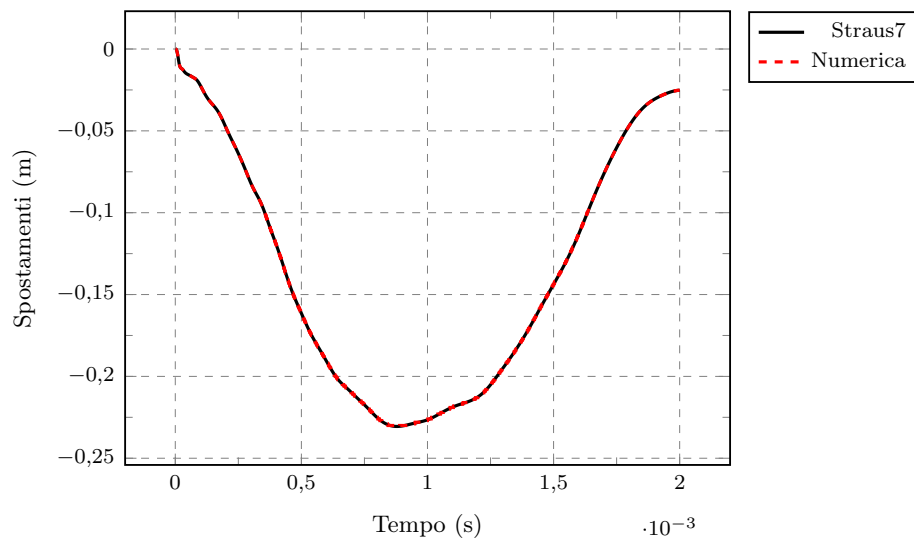


**Figura B.5:** Modello in condizioni piane di deformazione impiegato.

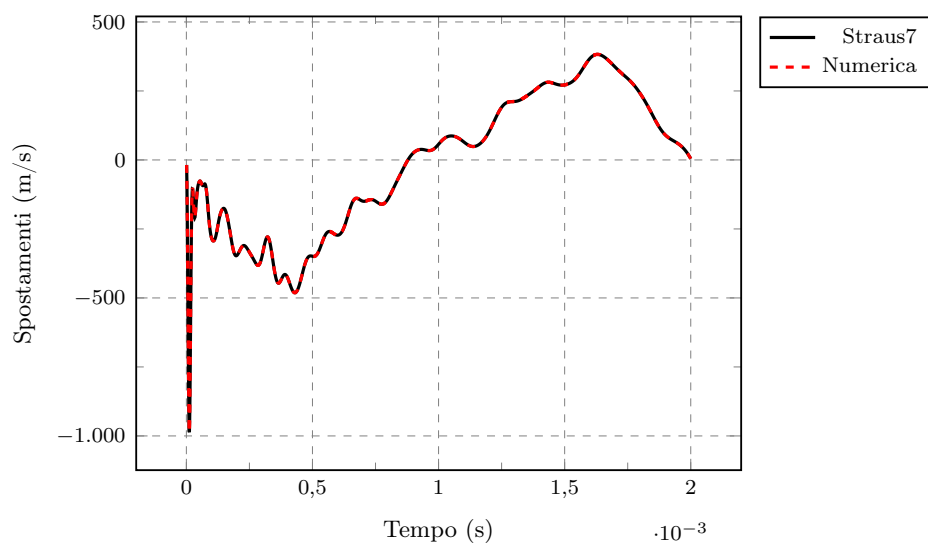
Il carico è applicato seguendo la rampa riportata in Figura B.6. Il time step impiegato è pari a  $\Delta t = 10^{-6}$  s. Vengono misurati spostamenti, velocità ed accelerazioni nel punto di applicazione del carico e le reazioni vincolari di uno dei due appoggi. Si riportano di seguito i risultati.



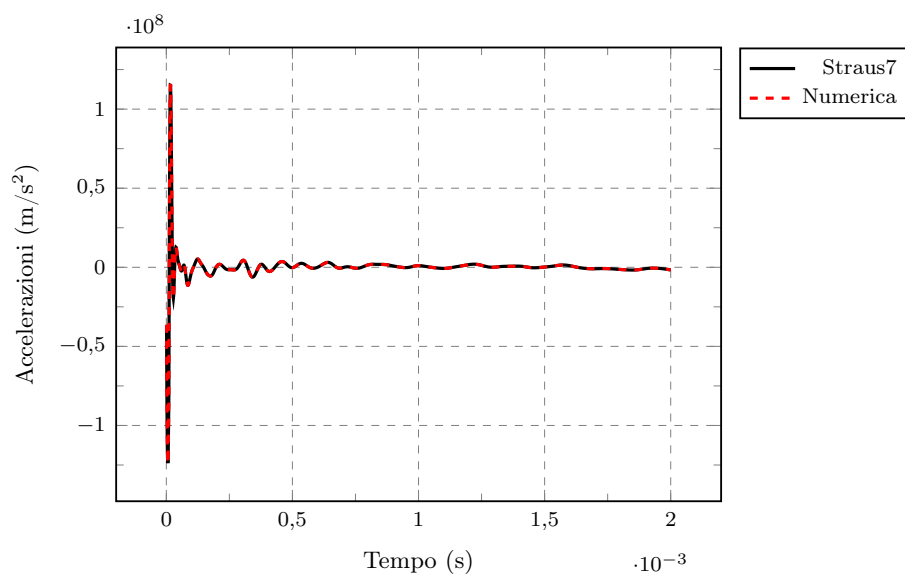
**Figura B.6:** Rampa impiegata per la validazione.



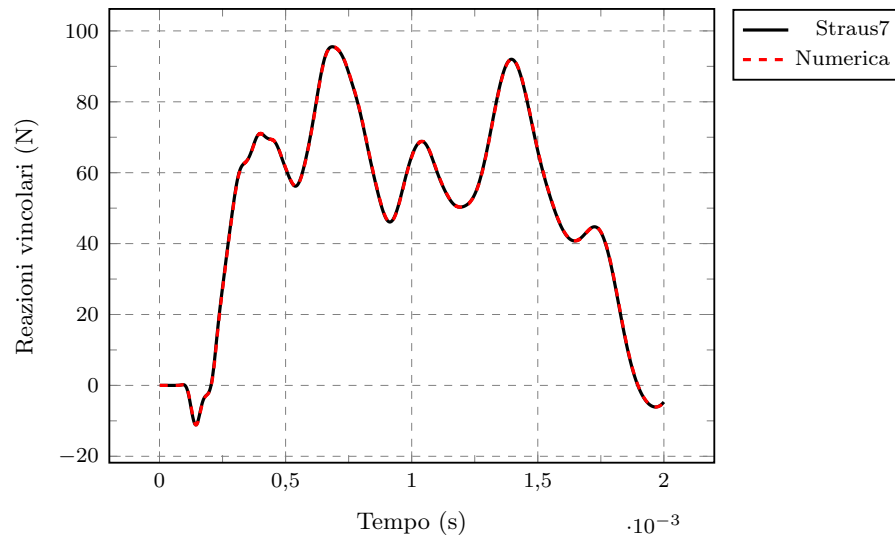
**Figura B.7:** Risposta del sistema - spostamenti.



**Figura B.8:** Risposta del sistema - velocità.



**Figura B.9:** Risposta del sistema - accelerazioni.



**Figura B.10:** Risposta del sistema - reazioni vincolari.

# Bibliografia

- [1] Reza Abedi, Boris Petracovici e Robert B Haber. «A space–time discontinuous Galerkin method for linearized elastodynamics with element-wise momentum balance». In: *Computer Methods in Applied Mechanics and Engineering* 195.25 (2006), pp. 3247–3273.
- [2] Reza Abedi et al. «Adaptive discontinuous galerkin method for elastodynamics on unstructured spacetime grids». In: *Proc. XXI International Congress of Theoretical and Applied Mechanics (ICTAM)*. 2004.
- [3] Reza Abedi et al. «Modeling evolving discontinuities with spacetime discontinuous Galerkin methods». In: *IUTAM Symposium on Discretization Methods for Evolving Discontinuities*. Springer Netherlands. 2007, pp. 59–87.
- [4] Grigory Isaakovich Barenblatt. «The mathematical theory of equilibrium cracks in brittle fracture». In: *Advances in applied mechanics* 7.1 (1962), pp. 55–129.
- [5] Klaus-Jürgen Bathe. *Finite element procedures*. Klaus-Jurgen Bathe, 2006.
- [6] Klaus-Jürgen Bathe e Gunwoo Noh. «Insight into an implicit time integration scheme for structural dynamics». In: *Computers & Structures* 98 (2012), pp. 1–6.
- [7] TN Bittencourt et al. «Quasi-automatic simulation of crack propagation for 2D LEFM problems». In: *Engineering Fracture Mechanics* 55.2 (1996), pp. 321–334.
- [8] Thomas J Boone e Anthony R Ingraffea. «A numerical procedure for simulation of hydraulically-driven fracture propagation in poroelastic media». In: *International Journal for Numerical and Analytical Methods in Geomechanics* 14.1 (1990), pp. 27–47.
- [9] R. Brighenti e S.E. Esculapio. *Analisi Numerica dei Solidi e delle Strutture: Fondamenti del Metodo degli Elementi Finiti*. 2014. ISBN: 9788874887989. URL: <http://books.google.it/books?id=9C6tBAAAQBAJ>.

- 
- [10] GT Camacho e M Ortiz. «Computational modelling of impact damage in brittle materials». In: *International Journal of solids and structures* 33.20 (1996), pp. 2899–2938.
- [11] Alberto Carpinteri e Giovanni Colombo. «Numerical analysis of catastrophic softening behaviour (snap-back instability)». In: *Computers & structures* 31.4 (1989), pp. 607–636.
- [12] Zhiyun Chen, Holger Steeb e Stefan Diebels. «A time-discontinuous Galerkin method for the dynamical analysis of porous media». In: *International journal for numerical and analytical methods in geomechanics* 30.11 (2006), pp. 1113–1134.
- [13] A.K. Chopra. *Dynamics of Structures*. Pearson Education, 2012.
- [14] DS Dugdale. «Yielding of steel sheets containing slits». In: *Journal of the Mechanics and Physics of Solids* 8.2 (1960), pp. 100–104.
- [15] Steven H Emerman, DL Turcotte e DA Spence. «Transport of magma and hydrothermal solutions by laminar and turbulent fluid fracture». In: *Physics of the earth and planetary interiors* 41.4 (1986), pp. 249–259.
- [16] Horacio D Espinosa, Pablo D Zavattieri e Gordon L Emore. «Adaptive FEM computation of geometric and material nonlinearities with application to brittle failure». In: *Mechanics of Materials* 29.3 (1998), pp. 275–305.
- [17] L.B. Freund. *Dynamic Fracture Mechanics*. Cambridge Monographs on Mechanics. Cambridge University Press, 1998.
- [18] Philippe H Geubelle e Jeffrey S Baylor. «Impact-induced delamination of composites: a 2D simulation». In: *Composites Part B: Engineering* 29.5 (1998), pp. 589–602.
- [19] Alan A Griffith. «The phenomena of rupture and flow in solids». In: *Philosophical transactions of the royal society of london. Series A, containing papers of a mathematical or physical character* (1921), pp. 163–198.
- [20] Arne Hillerborg, Mats Mod er e P-E Petersson. «Analysis of crack formation and crack growth in concrete by means of fracture mechanics and finite elements». In: *Cement and concrete research* 6.6 (1976), pp. 773–781.
- [21] Thomas JR Hughes e Gregory M Hulbert. «Space-time finite element methods for elastodynamics: formulations and error estimates». In: *Computer methods in applied mechanics and engineering* 66.3 (1988), pp. 339–363.
- [22] ICOLD. *fifth international benchmarks workshop on numerical analysis of dam*. 1999.



- [23] George R Irwin. «Analysis of stresses and strains near the end of a crack traversing a plate». In: *J. appl. Mech.* (1957).
- [24] Breno Pinheiro Jacob e Nelson Francisco Favilla Ebecken. «An optimized implementation of the Newmark/Newton-Raphson algorithm for the time integration of non-linear problems». In: *Communications in Numerical Methods in Engineering* 10.12 (1994), pp. 983–992.
- [25] Hideo Kobayashi e Hisahiro Onoue. «Brittle Fracture of Liberty Ships». In: *Failure Knowledge Database* 100 (1943).
- [26] Martin Larcher. «Development of discrete cracks in concrete loaded by shock waves». In: *International Journal of Impact Engineering* 36.5 (2009), pp. 700–710.
- [27] R.W. Lewis e B.A. Schrefler. *The finite element method in the static and dynamic deformation and consolidation of porous media*. Numerical methods in engineering. John Wiley, 1998. ISBN: 9780471928096.
- [28] Xiang Dong Li e N-E WIBERG. «STRUCTURAL DYNAMIC ANALYSIS BY A TIME-DISCONTINUOUS GALERKIN FINITE ELEMENT METHOD». In: *International Journal for Numerical Methods in Engineering* 39.12 (1996), pp. 2131–2152.
- [29] J Mediavilla, RHJ Peerlings e MGD Geers. «A robust and consistent remeshing-transfer operator for ductile fracture simulations». In: *Computers & structures* 84.8 (2006), pp. 604–623.
- [30] A Needleman. «A continuum model for void nucleation by inclusion debonding». In: *Journal of applied mechanics* 54.3 (1987), pp. 525–531.
- [31] P. Pegon e A. Anthoine. «Numerical strategies for solving continuum damage problems with softening: Application to the homogenization of Masonry». In: *Computers and Structures* 64.14 (1997), pp. 623–642.
- [32] D Peri et al. «Transfer operators for evolving meshes in small strain elasto-plasticity». In: *Computer Methods in Applied Mechanics and Engineering* 137.3 (1996), pp. 331–344.
- [33] F Pizzocolo, JM Huyghe e K Ito. «Mode I crack propagation in hydrogels is step wise». In: *Engineering Fracture Mechanics* 97 (2013), pp. 72–79.
- [34] Raul Radovitzky e Michael Ortiz. «Error estimation and adaptive meshing in strongly nonlinear dynamic problems». In: *Computer Methods in Applied Mechanics and Engineering* 172.1 (1999), pp. 203–240.
- [35] J R Rice e Dennis Michael Tracey. «On the ductile enlargement of voids in triaxial stress fields». In: *Journal of the Mechanics and Physics of Solids* 17.3 (1969), pp. 201–217.

- [36] James R Rice e Jian-Sheng Wang. «Embrittlement of interfaces by solute segregation». In: *Materials Science and Engineering: A* 107 (1989), pp. 23–40.
- [37] BA Schrefler et al. «Mechanics of partially saturated porous media». In: *Numerical methods and constitutive modelling in geomechanics*. Springer, 1990, pp. 169–209.
- [38] Bernhard A Schrefler, Stefano Secchi e Luciano Simoni. «On adaptive refinement techniques in multi-field problems including cohesive fracture». In: *Computer Methods in Applied Mechanics and Engineering* 195.4 (2006), pp. 444–461.
- [39] S Secchi e L Simoni. «An improved procedure for 2D unstructured Delaunay mesh generation». In: *Advances in Engineering Software* 34.4 (2003), pp. 217–234.
- [40] Stefano Secchi, Luciano Simoni e Bernhard A Schrefler. «Mesh adaptation and transfer schemes for discrete fracture propagation in porous materials». In: *International journal for numerical and analytical methods in geomechanics* 31.2 (2007), pp. 331–345.
- [41] Angelo Simone. *Continuous-discontinuous modelling of failure*. TU Delft, Delft University of Technology, 2003.
- [42] Luciano Simoni e Stefano Secchi. «Cohesive fracture mechanics for a multi-phase porous medium». In: *Engineering Computations* 20.5/6 (2003), pp. 675–698.
- [43] DA Spence e DL Turcotte. «Magma-driven propagation of cracks». In: *Journal of Geophysical Research: Solid Earth (1978–2012)* 90.B1 (1985), pp. 575–580.
- [44] Albert Turon et al. «An engineering solution for mesh size effects in the simulation of delamination using cohesive zone models». In: *Engineering fracture mechanics* 74.10 (2007), pp. 1665–1682.
- [45] Viggo Tvergaard e John W Hutchinson. «The relation between crack growth resistance and fracture process parameters in elastic-plastic solids». In: *Journal of the Mechanics and Physics of Solids* 40.6 (1992), pp. 1377–1397.
- [46] MP Wnuk. «Quasi-static extension of a tensile crack contained in a viscoelastic-plastic solid». In: *Journal of Applied Mechanics* 41.1 (1974), pp. 234–242.
- [47] X-P Xu e Alan Needleman. «Numerical simulations of fast crack growth in brittle solids». In: *Journal of the Mechanics and Physics of Solids* 42.9 (1994), pp. 1397–1434.
- [48] O C Zienkiewicz, Robert L Taylor e J Zhu. *The finite element method: its basis and fundamentals*. Butterworth-heinemann, 2005.

- 
- [49] O.C. Zienkiewicz e T. Shiomi. «Dynamic behaviour of saturated porous media; the generalized Biot formulation and its numerical solution». In: *International journal for numerical and analytical methods in geomechanics* 8.1 (1984), pp. 71–96.
- [50] O.C. Zienkiewicz e Robert L Taylor. *The finite element method for solid and structural mechanics*. Butterworth-heinemann, 2005.