

## **APPLICAZIONE DI UN ALGORITMO GENETICO PER LA CALIBRAZIONE OTTIMALE DI MODELLI ISTERETICI**

### **APPLICATION OF A GENETIC ALGORITHM FOR THE OPTIMAL CALIBRATION OF HYSTERETIC MODELS**

Sabatino Di Benedetto, Massimo Latour, Gianvittorio Rizzano  
University of Salerno  
Department of Civil Engineering  
Via Giovanni Paolo II, Fisciano, Italy  
sdbenedetto@unisa.it, mlatour@unisa.it, g.rizzano@unisa.it

#### **ABSTRACT**

The computing improvements that have taken place in the last decades have allowed ever greater use of numerical models to predict phenomena concerning the most various topics. In particular, Civil Engineering represents a field deeply affected by this occurrence; in fact, the possibility of having greater computational capabilities has guaranteed to investigate of themes inherent both to the global and local behaviour of buildings with greater attention and precision so that currently many software programs allow modelling with more accuracy the different structural components. One of the aspects of interest, with particular reference to the field of steel structures, concerns the calibration of mathematical models helpful in modelling the mechanical behaviour of particular devices or connections. With this in mind, many efforts have recently been aimed at solving this problem by implementing particular computational codes called "Genetic Algorithms". Genetic Algorithms are instructions through which it is possible to reach the optimal desired calibration according to a procedure that conceptually refers to the evolutionary process of living species.

This work focuses on implementing a Genetic Algorithm that, starting from the knowledge of assigned hysteretic curves, allows their modelling through the "hysteretic" uniaxial material element of the OpenSees software through an appropriate calibration of the parameters. In particular, the code is developed in the Matlab environment and is validated by calibrating 44 force-displacement hysteretic curves obtained from finite element simulations relating to the cyclic behaviour of connections between circular hollow section profiles and through-plates subjected to displacement histories in the axial direction. Finally, to highlight the effectiveness of the proposed algorithm, the same hysteretic curves have also been calibrated using the MultiCal tool, a calibration software already available and exploited in the research.

#### **SOMMARIO**

Gli sviluppi informatici occorsi negli ultimi decenni hanno permesso un sempre maggiore utilizzo di modelli numerici per la previsione di fenomeni riguardanti le più diverse tematiche. In particolare, l'Ingegneria Civile rappresenta un campo fortemente investito da tale avvenimento;

infatti, la possibilità di disporre di maggiori capacità computazionali ha permesso di investigare con maggiore attenzione e precisione aspetti inerenti sia al comportamento globale che locale degli edifici tanto che, attualmente, sono disponibili molti software di calcolo che permettono di modellare con maggiore accuratezza le diverse componenti strutturali.

Uno degli aspetti di interesse, con particolare riferimento al campo delle strutture in acciaio, riguarda la calibrazione di modelli matematici utili per la modellazione del comportamento meccanico di particolari tipologie di dispositivi o connessioni. In tale ottica, molti sforzi si stanno recentemente rivolgendo alla risoluzione di tale problematica tramite l'implementazione di particolari codici di calcolo definiti "Algoritmi Genetici".

Gli Algoritmi Genetici sono delle istruzioni fornite al calcolatore tramite cui è possibile pervenire alla desiderata calibrazione tramite una procedura che concettualmente si richiama al processo evolutivo delle specie viventi.

Tale lavoro è incentrato sull'implementazione di un algoritmo genetico che permetta, a partire dalla conoscenza di assegnate curve isteretiche, la loro modellazione tramite l'elemento di tipo hysteretic del software di calcolo OpenSees mediante un'opportuna calibrazione dei parametri. In particolare, il codice è stato sviluppato in ambiente Matlab ed è stato validato calibrando 44 curve isteretiche forza-spostamento ottenute da simulazioni agli elementi finiti relative al comportamento ciclico di connessioni tra profili circolari cavi e piatti caricati assialmente. Infine, per evidenziare l'efficacia dell'algoritmo proposto, si è proceduto anche alla calibrazione delle stesse curve isteretiche tramite MultiCal, un software di calibrazione già disponibile ed utilizzato nell'ambito della ricerca.

## 1 INTRODUZIONE

Gli sviluppi tecnologici a cui si è assistito negli ultimi decenni hanno reso la società contemporanea sempre più high-tech. I riflessi di tale fenomeno sono stati osservati anche nell'ambito dell'Ingegneria Civile poiché le maggiori capacità computazionali a disposizione hanno ampliato le possibili tematiche da investigare. Al giorno d'oggi, infatti, sono stati implementati software agli elementi finiti (FE) più avanzati per eseguire simulazioni numeriche tenendo conto di complesse non linearità a livello di materiali, sezioni, membrature e dispositivi.

Questo approccio si basa sul presupposto che le attuali normative richiedono la progettazione di strutture in grado di dissipare l'energia sismica che investe gli edifici in elementi ben definiti e noti a priori. Di conseguenza, le non linearità possono essere modellate attraverso elementi a plasticità concentrata o diffusa, assegnando un comportamento elastico alle altre componenti strutturali ([1], [2]).

In quest'ottica, le maggiori capacità computazionali hanno consentito di sfruttare diverse tipologie di modelli non lineari che differiscono per il grado di complessità; per chiarezza, vale la pena citare le più note implementazioni matematiche attualmente incorporate nei comuni software strutturali: Ramberg e Osgood [3], Bouc e Wen ([4], [5]), Richard e Abbott [6], Sivaselvan e Reinhorn [7]. Tuttavia, sono disponibili anche altri modelli più semplici, come la tradizionale legge bilineare elastico-plastica (nota come "Steel01" in OpenSees [8] o "bl\_sym" in SeismoStruct [9]).

I parametri dei modelli non lineari sopra citati hanno principalmente un significato fenomenologico piuttosto che meccanico. Tuttavia, il loro pregio consiste nella possibilità di caratterizzare risposte sperimentali tensione-deformazione, forza-spostamento o momento-rotazione, ignorando anche le connotazioni fisiche su cui si basano i fenomeni osservati.

Pertanto, la calibrazione di questi modelli fenomenologici rispetto a risposte note rappresenta un passo fondamentale per il loro effettivo utilizzo nei comuni software strutturali. In particolare, questo scopo può essere perseguito implementando una routine di calcolo basata su Algoritmi Genetici (AG) che identifichi e ottimizzi i parametri che meglio replicano la risposta non lineare nota.

Gli Algoritmi Genetici rappresentano una trasposizione della teoria darwiniana dell'evoluzione delle specie in ambito informatico, con l'intento di ottimizzare la calibrazione di leggi matematiche appartenenti alle più diverse tematiche. La prima applicazione di questa tecnica riguardò il campo della biologia e avvenne nel 1975 da parte di Holland [10]. Invece, l'introduzione all'Ingegneria Civile è avvenuta nel 1986 da parte di Goldberg e Samtani [11]. Nel 2000 Pezeshk et al. [12] hanno proposto la prima routine basata sugli AG per la progettazione ottimale di telai in acciaio, mentre sviluppi significativi si sono verificati nel 2005 e nel 2010 quando Del Savio et al. [13] e Cheng [14] hanno promosso codici genetici per progettare, rispettivamente, strutture in acciaio con giunti semirigidi e ponti ad arco con tiranti e torri in acciaio. Inoltre, nel 2019 Falcone et al. [15] hanno proposto una procedura basata sugli AG per l'adeguamento sismico di strutture esistenti in cemento armato mediante l'introduzione di controventi concentrici in acciaio o tramite il rinforzo delle colonne tramite FRP.

La strategia basata sugli AG è stata applicata anche per la calibrazione di modelli strutturali isteretici attraverso lo sviluppo dell'applicazione MultiCal (Multi-Objective Calibration) [16], concepita per l'ottimizzazione multi-obiettivo di diversi modelli fenomenologici implementati in OpenSees [8] e SeismoStruct [9]. La procedura permette l'ottimizzazione con riferimento: alla storia delle tensioni generalizzate (o forze o momenti); alla dissipazione di energia; alla curva di involuppo. MultiCal è stato ampiamente testato [16], dimostrando la sua affidabilità per la calibrazione di curve monotone, cicliche e di risposte isteretiche caratterizzate da storie di carico non predefinite (questo è il caso, ad esempio, dei risultati di prove pseudo-dinamiche [16]). Tuttavia, poiché MultiCal è sviluppato in linguaggio C++ [17], è richiesta una conoscenza tecnica specifica per leggere o modificare il codice sorgente. Inoltre, il processo di ottimizzazione viene eseguito utilizzando un singolo processore della CPU, rendendo onerosi i tempi di calibrazione.

Tale articolo propone una procedura numerica integrata tra Matlab [18] ed OpenSees [8] per calibrare i parametri dell'elemento "hysteretic" appartenente alla libreria di OpenSees attraverso un semplice Algoritmo Genetico (nel seguito indicato come pGA, acronimo di "proposed Genetic Algorithm"). In particolare, l'efficacia della soluzione proposta è dimostrata applicando la routine implementata a 44 curve forza-spostamento cicliche che rappresentano i risultati delle simulazioni Abaqus [19] riguardanti connessioni tra profili circolari cavi e piatti passanti caricati assialmente. Tuttavia, a differenza di MultiCal, l'algoritmo proposto è stato implementato per ottimizzare solo le risposte forza-spostamento generalizzate, trascurando l'ottimizzazione in termini di energia ed involuppo. Di conseguenza, la funzione obiettivo da minimizzare è costituita dalla somma degli scarti quadratici normalizzati tra le forze delle curve di riferimento e le forze derivanti dai modelli fenomenologici.

Infine, le stesse 44 curve sono state calibrate tramite MultiCal e sono state confrontate con i risultati forniti dal pGA, dimostrando la soddisfacente accuratezza della routine implementata.

## 2 METODOLOGIA

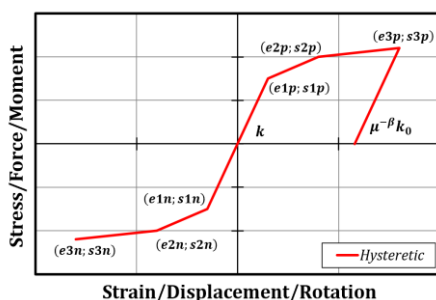
Il presente lavoro ha lo scopo di proporre l'implementazione di un algoritmo genetico per calibrare i parametri (Tabella 1) del modello "hysteretic uniaxialmaterial" di OpenSees (Fig. 1) a partire dalla conoscenza di storie di carico cicliche che possano essere rappresentative di relazioni tensione-deformazione, forza-spostamento o momento-rotazione.

In particolare, si intende testare l'algoritmo proposto rispetto a 44 curve forza-spostamento cicliche ottenute da simulazioni agli elementi finiti (FE) eseguite su connessioni tra profili circolari cavi e piatti passanti caricati assialmente (Fig. 2). Prima di eseguire queste analisi, il modello agli elementi finiti, sviluppato in Abaqus [19], è stato validato sulla base di tre prove cicliche [20]. Questi risultati fanno parte di un più ampio programma di ricerca attualmente in corso presso l'Università di Salerno riguardante la caratterizzazione delle componenti dei nodi tra colonne circolari cave e profili IPE passanti. Per brevità le prove sperimentali e le validazioni dei modelli

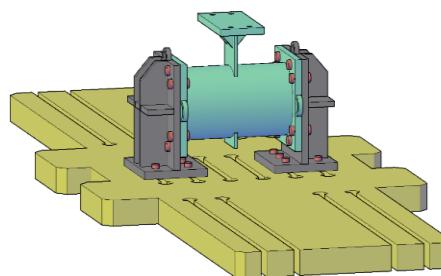
numerici non sono riportate nel presente lavoro, ma tutte le informazioni necessarie possono essere consultate in [20].

**Tabella 1.** Parametri dell'elemento hysteretic

Parametri	Significato meccanico
$s1p, e1p$	Forza e spostamento positivi del primo punto dell'involuppo
$s2p, e2p$	Forza e spostamento positivi del secondo punto dell'involuppo
$s3p, e3p$	Forza e spostamento positivi del terzo punto dell'involuppo
$s1n, e1n$	Forza e spostamento negativi del primo punto dell'involuppo
$s2n, e2n$	Forza e spostamento negativi del secondo punto dell'involuppo
$s3n, e3n$	Forza e spostamento negativi del terzo punto dell'involuppo
$pinchx$	Fattore di pinching legato alla deformazione in fase di ricarica
$pinchy$	Fattore di pinching legato alla forza in fase di ricarica
$damage1$	Danno dovuto alla duttilità
$damage2$	Danno dovuto all'energia
$beta$	Potenza usata per determinare il degrado di rigidità basato sulla duttilità



**Fig. 1.** Legge del modello hysteretic



**Fig. 2.** Connessione tra profilo circolare cavo e piatto passante

### 3 IMPLEMENTAZIONE DELL'ALGORITMO GENETICO

Gli Algoritmi Genetici (AG) rappresentano un'interessante soluzione da adottare nel caso di problemi di ottimizzazione poiché sono concepiti per calibrare i parametri di leggi matematiche al fine di ottenere il miglior fitting di dati sperimentali o numerici. L'implementazione degli AG evoca la teoria dell'evoluzione di Darwin: gli esseri viventi sono il risultato di un processo evolutivo governato principalmente da variabilità casuali che permettono la sopravvivenza dei soli individui caratterizzati dalle migliori capacità di adattamento all'ambiente in cui vivono. Gli algoritmi genetici intendono trasporre questo approccio in ambito computazionale, ottimizzando molteplici parametri che possono influenzare un modello fenomenologico. La ricerca della configurazione ottimale dei parametri inizia con la preliminare generazione casuale di individui le cui "capacità di sopravvivenza" (rappresentative della corrispondenza con la risposta sperimentale) vengono valutate attraverso un'apposita Fitness-Function. Questa rappresenta l'operatore selettivo per l'identificazione dei migliori individui. Nella fase successiva, tali individui vengono impiegati per creare nuove generazioni. Questo approccio viene replicato generazione dopo generazione, incorporando mutazioni casuali ed incroci (o crossover) tra individui.

Lo scopo del presente lavoro consiste nell'implementazione in Matlab [18] di un algoritmo genetico per la calibrazione dei parametri relativi al modello "hysteretic uniaxialmaterial" appartenente alla libreria OpenSees [8]. Questo paragrafo è interamente incentrato sulla descrizione dell'algoritmo

genetico proposto, che, per chiarezza, sarà indicato come pGA. La procedura adottata per una generica generazione  $i$  –esima del pGA è riportata in Fig. 3.

I parametri di input sono la popolazione ( $P$ ), i semi ( $S$ ) ed il crossover ( $C$ ).  $P$  rappresenta il numero di generazioni da creare,  $S$  è il numero di individui prodotti attraverso l'operatore di mutazione all'interno di ciascuna generazione e  $C$  rappresenta il numero di individui creati applicando l'operatore di crossover. I precedenti parametri sono stati impostati rispettivamente pari a 100, 70 e 30. In particolare, si precisa che la mutazione modifica casualmente i valori assunti dai parametri dell'ultimo migliore individuo, mentre il crossover è un operatore che crea nuovi individui combinando casualmente i valori dei parametri appartenenti a diverse soluzioni ottimali definite in precedenza. Altra assunzione alla base del presente algoritmo consiste nella simmetria del modello hysteretic; ciò implica che  $s1n=-s1p$ ,  $e1n=-e1p$ ,  $s2n=-s2p$ ,  $e2n=-e2p$ ,  $s3n=-s3p$ ,  $e3n=-e3p$ .

A partire dalla conoscenza delle 44 curve forza-spostamento, per definire i valori iniziali di  $s1p$ ,  $e1p$ ,  $s2p$ ,  $e2p$ ,  $s3p$ ,  $e3p$ , si è ipotizzato che il primo punto dell'involuppo coincida con il punto della curva di riferimento dove per la prima volta è attinto l'80% della resistenza massima. Il secondo punto dell'involuppo corrisponde invece al picco di resistenza. Infine, il terzo punto dell'involuppo è stato assunto come punto in cui la curva di riferimento attinge lo spostamento massimo. Queste formulazioni non forniscono i valori corretti dei parametri, ma inizializzano il vettore del primo individuo. Invece i parametri relativi a  $pinchx$ ,  $pinchy$ ,  $damage1$ ,  $damage2$ ,  $beta$  sono stati impostati rispettivamente pari a 0.2, 0.8, 0.05, 0.1 e 0.1.

Lo step 1 del pGA prevede la generazione della popolazione di individui a partire dal migliore individuo della precedente generazione ( $B_{i-1}$ ) se  $i > 1$ , oppure dall'iniziale configurazione di parametri precedentemente assegnata se  $i = 1$ . Tale generazione è eseguita tramite gli operatori mutazione e crossover. Come mostrato in Fig. 3, è stato assunto un coefficiente di mutazione del 10% per i parametri  $s1p$ ,  $e1p$ ,  $s2p$ ,  $e2p$ ,  $s3p$ ,  $e3p$  e un coefficiente del 50% per  $pinchx$ ,  $pinchy$ ,  $damage1$ ,  $damage2$ ,  $beta$ . Questa ipotesi implica che il codice generi una matrice di mutazione ( $M_m$ ) di dimensioni  $11 \times S$ , ed una matrice di crossover ( $M_c$ ) di dimensioni  $11 \times C$ . In  $M_m$  gli elementi appartenenti alle prime sei righe variano casualmente tra 0,90 e 1,10, mentre la variazione per le righe rimanenti è compresa tra 0,50 e 1,50. Per le prime 10 generazioni non essendo disponibili abbastanza soluzioni, l'operatore di crossover non può essere applicato. In caso contrario, tale operatore seleziona nuovi individui mediante combinazione casuale delle precedenti 10 soluzioni. In generale, la nuova popolazione di individui ( $NP$ ) viene valutata come  $NP = [M_m B_{i-1} \quad M_c]$  (Step 3).

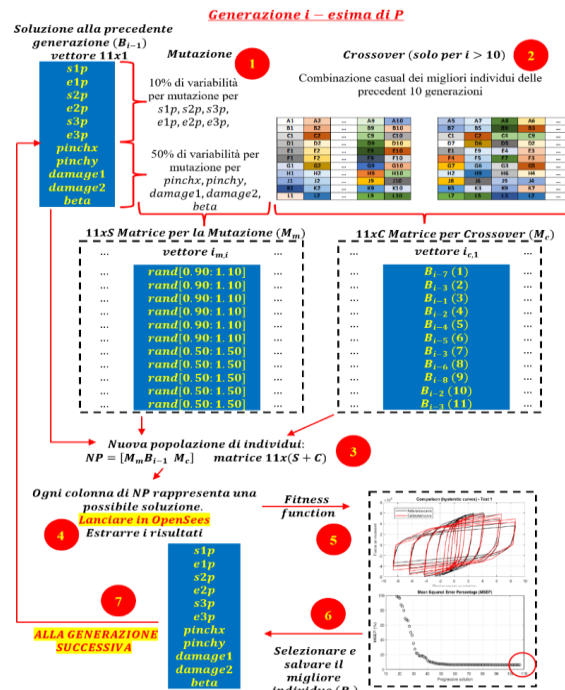


Fig. 3. Procedura riferita alla prima generazione della procedura proposta

Ciascuna colonna della matrice  $NP$  rappresenta una configurazione di parametri dell'elemento hysteretic da testare. Nello Step 4, vengono creati i modelli OpenSees, per ciascuna delle configurazioni individuate. In particolare, i parametri di input del modello hysteretic corrispondono a ciascuna colonna di  $NP$ . I modelli OpenSees sono stati concepiti secondo lo schema mostrato in Fig. 4: due nodi coincidenti sono collegati attraverso un elemento di lunghezza nulla caratterizzato dalle proprietà del modello hysteretic lungo l'unica direzione  $x$ . Tutti i gradi di libertà del nodo 1 sono fissi, mentre la storia ciclica di spostamenti è applicata al nodo 2. Alla fine delle analisi i risultati sono opportunamente salvati.



Fig. 4. Modellazione in OpenSees

Vale la pena evidenziare che il codice Matlab è stato implementato per essere compatibile con il calcolo parallelo in modo che più modelli OpenSees possano lavorare contemporaneamente su diversi processori (in questo caso, impostati pari a sei) della Central Processing Unit (CPU). Questa scelta è stata ispirata dalla necessità di velocizzare la procedura di ottimizzazione.

Quindi, per ciascuno dei 44 casi confrontando la curva di riferimento con quella numerica ottenuta, è possibile definire una funzione di fitness che, in questo caso, corrisponde alla percentuale degli scarti quadratici medi normalizzati (MSEP, acronimo di Mean Squared Error Percentage) nello Step 5. La formulazione matematica della Fitness-Function è riportata nell'Eq. 1.

$$MSEP (\%) = 100 \cdot \sum_{j=1}^T \left[ \left( \frac{F_{j,calcolato} - F_{j,riferimento}}{F_{j,riferimento}} \right)^2 W_j \right] \quad (1)$$

Per chiarezza,  $T$  rappresenta il numero degli step di applicazione degli spostamenti, mentre  $F_{j,calcolato}$  e  $F_{j,riferimento}$  sono i valori di forza ottenuti dal modello matematico OpenSees e dalle simulazioni in Abaqus allo stesso  $j$  -esimo istante.  $W_j$  è invece un fattore di ponderazione che è uguale a 0 se il valore assoluto di  $F_{j,riferimento}$  è inferiore al 10% della forza massima registrata sperimentalmente ( $F_{max,riferimento}$ ), altrimenti varia linearmente tra 5 e 1 per  $j$  compreso tra 1 e  $T$ . L'introduzione del fattore  $W_j$  è giustificata dalla necessità di trascurare gli errori eccessivi e privi di significato che si verificano per valori bassi del denominatore mostrato nell'Eq. 1 ( $F_{j,riferimento}$ ). Inoltre, poiché le storie di spostamento sono caratterizzate da molti cicli di ridotte ampiezze, il fattore  $W_j$  permette di amplificare gli errori nei cicli iniziali della curva isteretica, scongiurando che l'influenza degli ultimi cicli prevalga.

Infine, il miglior individuo della generazione attuale viene valutato nello Step 6 e, se il suo MSEP è inferiore alla soluzione iniziale, esso diventa il nuovo miglior individuo ( $B_i$ ) e viene utilizzato per la generazione successiva (Step 7). Ovviamente, se l'individuo ottimo ottenuto nell' $i$ -esima generazione è migliore di  $B_{i-1}$ , allora viene rinominato  $B_i$ , altrimenti  $B_i = B_{i-1}$ .

Tale procedura si ripete fino a quando  $i = P$ .

#### 4 VALIDAZIONE DELL'ALGORITMO GENETICO

L'algoritmo descritto al paragrafo 3 è stato applicato per calibrare le curve cicliche forza-spostamento di 44 diverse configurazioni geometriche di connessioni tra profili circolari cavi e travi passanti. A titolo di esempio, i risultati del Caso 2 sono mostrati in Fig. 5.

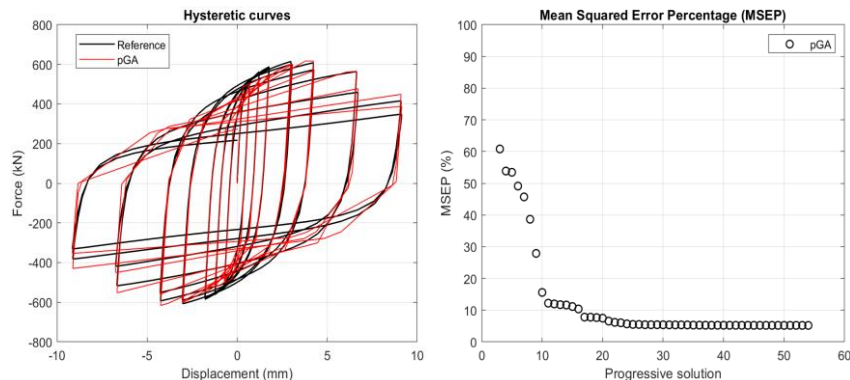
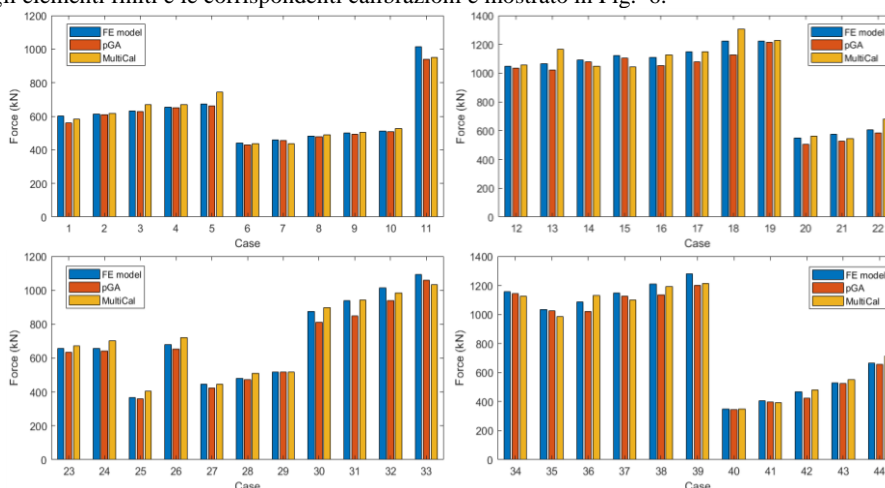


Fig. 5. Caso 2: curve forza-spostamento (sinistra); errore quadratico medio

In generale, l'algoritmo proposto è in grado di prevedere con buona precisione la rigidità e la resistenza delle risposte isteretiche. Tuttavia, anche il danno ed il degrado ciclico sono previsti con accettabile accuratezza. Per il caso 2 mostrato in Fig. 5 è stato riportato sia il confronto tra le curve forza-spostamento che il corrispondente grafico che rappresenta l'andamento della funzione MSEP con le soluzioni progressive che l'algoritmo definisce nel corso delle generazioni. Le configurazioni analizzate evidenziano che occorrono poche decine di casi per giungere ad una soluzione accettabile; infatti, per tutti i casi i grafici MSEP sono caratterizzati da una rapida riduzione dell'errore proprio all'inizio della procedura di esecuzione. Tuttavia, il processo di ottimizzazione

continua per trovare una soluzione migliore. Per validare il pGA, gli stessi casi sono stati calibrati utilizzando il software MultiCal applicando gli stessi input e disabilitando gli obiettivi che ottimizzano energia ed involuppo. Il confronto in termini di resistenza esibito dalle 44 simulazioni agli elementi finiti e le corrispondenti calibrazioni è mostrato in Fig. 6.



**Fig. 6.** Confronti in termini di resistenza

Infine, la funzione  $MSEP$  definita nell'Eq. 1 è stata applicata alle configurazioni ottimizzate dei parametri selezionati per i 44 casi sia con il pGA che MultiCal. I risultati di questa applicazione sono mostrati nella Fig. 7 per tutti i casi analizzati.

Il confronto tra questi risultati evidenzia che, in generale, l'algoritmo genetico proposto è in grado di raggiungere soluzioni migliori rispetto a MultiCal limitando la variabilità dell'intervallo del  $MSEP$  tra il 5 e il 20%, mentre MultiCal raggiunge errori più elevati, fino al 30%. Questo risultato rappresenta una caratteristica interessante per dimostrare l'affidabilità ed i vantaggi forniti dal codice proposto. Inoltre, l'implementazione del calcolo parallelo velocizza il processo di ottimizzazione poiché le 44 curve forza-spostamento sono state calibrate in 8 ore con il pGA, mentre la stessa calibrazione in MultiCal ha richiesto 40 ore.

Vale la pena sottolineare che il principale limite dell'algoritmo genetico proposto è che non può eseguire calibrazioni multi-obiettivo. Tuttavia, i risultati discussi in questo articolo possono rappresentare un punto di partenza significativo per l'aggiornamento di questo algoritmo utilizzando metodi avanzati per la calibrazione contemporanea di parametri aggiuntivi quali energia ed involuppo delle curve. Un altro limite è che il pGA è stato implementato per calibrare curve isteretiche tramite il solo modello hysteretic appartenente alla libreria OpenSees, mentre MultiCal può essere applicato a molti altri modelli fenomenologici. In conclusione, gli aspetti positivi e negativi relativi al pGA e MultiCal sono riportati nella Tabella 2.



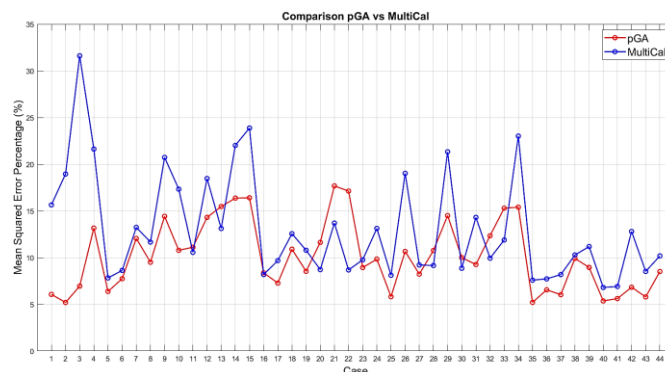


Fig. 7. Confronto pAG vs MultiCal

Tabella 2. pAG vs MultiCal: caratteristiche positive (verde) e negative (rosso)

Algoritmo Genetico proposto (pGA)	MultiCal
Ottimizzazione del processo attraverso l'analisi multi-core (processo più veloce)	Analisi single-core (processo più lento)
Nessuna calibrazione multi-obiettivo: ottimizzazione solo in termini di forza	Calibrazione multi-obiettivo: forza, energia, involuppo
Ottimizzazione solo del modello hysteretic di Opensees	Ottimizzazione di molti modelli attualmente disponibili nelle librerie Opensees e SeismoStruct
Soluzioni migliori rispetto a MultiCal in molti casi delle configurazioni analizzate	

## 5 CONCLUSIONI

Il presente articolo tratta le principali caratteristiche di un algoritmo genetico implementato per la calibrazione di curve isteretiche. Il codice è stato sviluppato in ambiente Matlab ed intende calibrare i parametri del modello hysteretic della libreria di OpenSees. Per dimostrare l'affidabilità dell'implementazione proposta, il pGA è stato applicato per la calibrazione di 44 curve isteretiche forza-spostamento cicliche ottenute da simulazioni numeriche su connessioni tra profili circolari cavi e piatti passanti caricati assialmente. Inoltre, la stessa calibrazione è stata eseguita con l'applicazione MultiCal per mostrare l'accuratezza del pGA. Le principali conclusioni sono di seguito riassunte.

1. Il pGA può calibrare i casi analizzati con una precisione maggiore rispetto a MultiCal poiché, nella maggior parte dei casi, le sue soluzioni forniscono valori della funzione *MSEP* inferiori rispetto a MultiCal. Infatti, le percentuali di *MSEP* relative alle configurazioni studiate variano tra il 5 e il 20% per il pGA e tra il 5 e il 30% per MultiCal.
2. Il codice proposto ha eseguito la calibrazione in un tempo inferiore (8 ore contro le 40 ore di MultiCal) perché sfrutta una procedura di calcolo parallelo tra i processori della CPU.
3. Il pGA ha il limite principale di non poter eseguire calibrazioni multi-obiettivo poiché può ottimizzare solo la forma delle curve isteretiche, trascurandone energia ed involuppo. Tale codice, inoltre, limita la calibrazione al solo modello hysteretic di OpenSees.

Sulla base di tale lavoro, gli sviluppi futuri includeranno: i) l'aggiornamento del codice per eseguire calibrazioni multi-obiettivo; ii) l'aggiunta di ulteriori modelli fenomenologici da calibrare.

**BIBLIOGRAFIA**

- [1] Latour, M., Piluso, V., Rizzano, G., Rotational behaviour of column base plate connections: Experimental analysis and modelling, *Engineering Structures*, vol. 68, pp. 14-23, 2014.
- [2] Latour, M., Rizzano, G., A theoretical model for predicting the rotational capacity of steel base joints, *Journal of Constructional Steel Research*, vol. 91, pp. 89-99, 2013.
- [3] Ramberg, W., Osgood, W., Description of stress-strain curves by three parameters, National Advisory Committee on Aeronautics, Technical Note 902, 1943.
- [4] Bouc, R., Forced vibration of mechanical systems with hysteresis, in *Proceedings of the fourth conference on non-linear oscillation*, Prague, Czechoslovakia, 1967.
- [5] Wen, Y. K., Method for random vibration of hysteretic systems, *ASCE J Eng Mech Div*, vol. 102, no. 2, pp. 249-263, 1976.
- [6] Richard, R. M., Abbott, B. J., Versatile Elasto-Plastic Stress-Strain Formula, *Journal of the Engineering Mechanics Division*, vol. 101, no. 4, pp. 511-515, 1975.
- [7] Sivaselvan, M. V., Reinhorn, A. M., Hysteretic models for deteriorating inelastic structures, *J. Eng. Mech.*, vol. 126, no. 6, pp. 633-640, 2000.
- [8] McKenna, F., OpenSees: a framework for earthquake engineering simulation, *Comput Sci Eng*, 13:58–66, 2011.
- [9] SeismoStruct 2016, A computer program for static and dynamic analysis of framed structures, SeismoSoft, 2016.
- [10] Holland, J. H., *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications To Biology, Control, and Artificial Intelligence*, The University of Michigan, 1975.
- [11] Goldberg, D. E., Samtani, M. P., *Engineering optimization via genetic algorithm*, Engineering optimization via genetic algorithm, United States, Birmingham, Alabama, 1986.
- [12] Pezeshk, S., Camp, C. V., Chen, D., Design of Nonlinear Framed Structures Using Genetic Optimization, *J Struct Eng*, New York, 126, 387–388, [https://doi.org/10.1061/\(asce\)0733-9445\(2000\)126:3\(382\)](https://doi.org/10.1061/(asce)0733-9445(2000)126:3(382)), 2000.
- [13] Del Savio, A. A., de Andrade, S., Vellasco, P., Martha, L., Genetic algorithm optimization of semi-rigid steel structures, in *8th International Conference on the Application of Artificial Intelligence to Civil, Structural and Environmental Engineering*, Kippen, Stirling Civil-Comp Press, c2005., Roma, p. 16, 2005.
- [14] Cheng, J., Optimum design of steel truss arch bridges using a hybrid genetic algorithm, *J. Constr. Steel Res.* 66, 1011–1017. <https://doi.org/10.1016/j.jcsr.2010.03.007>, 2010.
- [15] Falcone, R., Carrabs, F., Cerulli, R., Lima C., Martinelli, E., Seismic retrofitting of existing RC buildings: a rational selection procedure based on Genetic Algorithms, *Structures*, Volume 22, Pages 310-326, ISSN 2352-0124, <https://doi.org/10.1016/j.istruc.2019.08.006>, 2019.
- [16] Chisari, C., Francavilla, A. B., Latour, M., Piluso, V., Rizzano G., Amadio, C., Critical issues in parameter calibration of cyclic models for steel members, *Engineering Structures*, 132:123-138. DOI: 10.1016/j.engstruct.2016.11.030, 2017.
- [17] Stroustrup, B., *The C++ Programming Language*, Addison-Wesley, Mass. 1995.
- [18] MATLAB, R2018b, Natick, Massachusetts: The MathWorks Inc.; 2018.
- [19] Abaqus, *Analysis User's Manual Version 6.17*, Abaqus Inc., 2017.
- [20] Sica, R., Di Benedetto, S., Latour, M., Rizzano, G., Preliminary study on the cyclic response of welded joints with CHS beams and through-all plates, *STESSA Conference 2022*.

**KEYWORDS**

Algoritmo Genetico; mutazione; crossover; selezione; ottimizzazione, dissipazione di energia.