

## NTC e software tecnico: la passione da sola può bastare?

### Premessa

Se consideriamo la teoria del ciclo di vita applicata alla normativa tecnica per le strutture si può dire che le NTC segnano la fine del ciclo iniziato negli anni 70, rappresentato dai vari D.M. di cui l'ultimo pubblicato nel 96, caratterizzato dal calcolo alle tensioni ammissibili e ne aprono uno nuovo iniziato con la OPCM 3274 del 2003 e consolidato con il D.M. del 2008 con la logica degli stati limite.

Allo stesso tempo, da fine anni 70, venivano immessi sul mercato i primi personal computer e da allora si è verificata la diffusione e l'applicazione sempre più capillare di strumenti di calcolo automatico, del loro progressivo minor costo e sempre maggiore capacità di calcolo.

Altrettanto imponente è stata la crescita del software: nel tempo di una generazione il software è diventato una delle principali fonti di ricchezza nel mondo, lo sviluppo e l'utilizzo del software è cresciuto più velocemente di qualsiasi prodotto realizzato nella storia del mondo e probabilmente nulla vanta il suo tasso di crescita.

Dunque le nuove NTC sono nate in un mondo in cui le capacità di calcolo automatico sono enormemente maggiori rispetto al passato, facilmente disponibili in ogni ufficio tecnico, ma sembra che il normatore non ne voglia tener conto, pur avendo dedicato un capitolo apposito al software (Capitolo 10 - Redazione dei progetti strutturali esecutivi e delle relazioni di calcolo) peraltro ripreso da una vecchia norma, la CNR 10024/86 - Analisi di strutture mediante elaboratore: impostazione e redazione delle relazioni di calcolo.

Cerchiamo di capire meglio il perché.

### La natura del software

Per capire bene cosa si intende in premessa è necessario puntualizzare alcuni aspetti che caratterizzano la natura del software.

Esaminando la comunicazione dei produttori si ha l'impressione che il software sia un prodotto; è distribuito in scatole, ha un prezzo, segue le leggi del marketing come tutti i prodotti commerciali.

Ma il software non è un prodotto nel senso classico del termine, la vera natura del software è di essere un medium, cioè uno strumento per la diffusione e l'applicazione concreta della conoscenza; rispetto ad altri media, come per esempio i libri, non solo trasmette conoscenza ma ne consente direttamente l'applicazione attraverso la capacità dei computer di elaborare i dati ed interagire con l'operatore.

Il prodotto vero è la conoscenza contenuta e trasmessa nel software, le informazioni e gli algoritmi di calcolo implementati e resi disponibili.

Il software è conoscenza resa attiva.

Se il software non è un prodotto, ed il prodotto vero è la conoscenza contenuta nel software, lo sviluppo di software può essere solo un'attività di acquisizione di conoscenza da parte di chi sviluppa il software e di codifica secondo un linguaggio formale.

Per questo sono necessarie due competenze: la conoscenza del problema che si intende trattare e la capacità di tradurlo in modo ottimale in un linguaggio comprensibile al computer.

La scrittura in un linguaggio di programmazione (Basic, Fortran, Pascal, C++, ecc.) è attività comune per tutti gli argomenti trattati dall'informatica, mentre per la conoscenza del problema, nel nostro caso il calcolo strutturale, è necessaria una competenza specifica ed approfondita della materia.

Ed è proprio su questo punto che le NTC manifestano la loro criticità.

## Il contenuto del cap. 10

È davvero stupefacente l'apertura del § 10.2 che recita: “**Qualora** l'analisi strutturale e le relative verifiche siano condotte con l'ausilio di codici di calcolo automatico, il progettista dovrà...”: ma davvero esiste ancora qualcuno che pensa che si possano applicare le NTC in modo manuale da valutare come poco probabile l'uso del software?

Esaminando in dettaglio il capitolo 10, il normatore ha esaminato due problematiche: come realizzare la relazione di calcolo e la validazione del progetto e per questo sono previsti tre momenti a carico di diverse figure.

Come noto la progettazione strutturale ruota attorno al concetto di modello, come rappresentazione sintetica e significativa della realtà in esame, e la modellazione di per sé è un'arte, quindi con un certo grado di soggettività, pur dovendo rispettare le regole della Scienza e della Tecnica delle Costruzioni.

Le norme a questo riguardo offrono diverse soluzioni, lasciando al progettista la scelta del tipo di analisi che si desidera effettuare (lineare, non lineare, ecc.).

Definita quindi la modellazione che si desidera, la prima validazione, a cura del progettista, riguarda il controllo che il software utilizzato sia effettivamente in grado di risolvere il problema in esame. Questa richiesta la si può assolvere conoscendo a fondo le caratteristiche del software, gli algoritmi utilizzati e l'implementazione che il produttore ha realizzato.

Si fa notare che ancora oggi non esiste una standardizzazione circa le caratteristiche dei software ed ogni produttore usa terminologie e convenzioni del tutto personali che rendono difficoltoso il confronto tra i vari software disponibili sul mercato, mentre una banale scatola di pelati deve denunciare in modo standardizzato i suoi ingredienti.

Per esempio, la teoria del calcolo agli elementi finiti propone diverse soluzioni, ognuna con un proprio ambito di applicazione, la cui estrapolazione può produrre sgradevoli sorprese: ci si chiede quanti progettisti oggi in Italia hanno consapevolezza di questo. È evidente che già per rispondere con precisione al primo punto sia necessaria una cultura non superficiale degli algoritmi e del calcolo automatico in generale.

Ma di questo la normativa non se ne cura; per questo AIST ha proposto una scheda normalizzata per la precisa identificazione delle caratteristiche del software.

La seconda validazione, a carico dei produttori di software, riguarda l'affidabilità dei software stessi cioè la loro capacità di risolvere correttamente i problemi trattati nel modo in cui sono stati implementati. Per questo sono necessari i cosiddetti "benchmark", cioè casi di confronto che, trattando un problema per volta offrono la ragionevole certezza circa la precisione ottenuta.

È questo un punto molto delicato in quanto se per alcuni aspetti (ad esempio per il calcolo elastico) si hanno diversi esempi e modelli consolidati, per quanto oggi è possibile realizzare con le NTC non sono disponibili modelli di valutazione condivisi (vedi analisi push-over).

Anche di questo la norma non dice nulla.

Inoltre qual è la precisione richiesta? Come qualche autore ha sottolineato sarebbe auspicabile un "criterio di fascia", cioè una tolleranza accettabile, come per esempio per le resistenze elettriche dove si specifica il valore con un range di tolleranza.

Ma chi stabilisce la tolleranza?

Il terzo ed ultimo punto riguarda la validazione del modello realizzato per il caso in esame, cioè che il modello sia coerente con il software utilizzato e con la realtà che vuole rappresentare. Si fa notare che questo punto, forse il più critico in quanto varia fortemente in base al problema trattato, è appena citato dalla normativa e comprende implicitamente anche la validazione del progettista, cioè di colui che si fa garante di tutte le validazione richieste.

Senza entrare nel merito della competenza dei progettisti, su cui dovrebbero vegliare gli Ordini Professionali, non si può non rilevare la confidenza, che in alcuni casi sconfinava in leggerezza, che molti tecnici ripongono nei programmi di calcolo accettando acriticamente le soluzioni proposte.

In molti casi il software svolge una funzione vicaria all'aggiornamento se non addirittura alla conoscenza dei problemi per l'applicazione delle NTC.

È evidente che ciò si verifica proprio per il contenuto di conoscenza insito nel software; resta però intatta la questione relativa al controllo dei dati introdotti e dei risultati ottenuti che, in mancanza di adeguata consapevolezza da parte del tecnico, potrebbero essere del tutto aleatori.

Una proposta, in linea con la logica probabilistica delle NTC, è valutare con un diverso livello di affidabilità e con un conseguente coefficiente di sicurezza che si riflette nella progettazione, così come è previsto per la determinazione dei tassi di lavoro del materiale per le strutture esistenti attraverso il Fattore di Confidenza, la competenza dei tecnici in funzione dei corsi frequentati, degli esami sostenuti, delle esperienze reali dimostrabili.

Questo produrrebbe un circolo virtuoso che indurrebbe all'approfondimento ed allo studio sistematico della materia.

È importante segnalare che ai produttori di software giungono quotidianamente richieste di chiarimenti e di consulenza che superano lo stretto utilizzo del software, per compensare il "gap" di conoscenza e l'interpretazione della normativa, diventando collettore di tutti i dubbi e delle perplessità che si manifestano nell'attività corrente di progettazione.

## **L'esame dei progetti da parte degli enti di controllo**

Ancora più critica e preoccupante è la situazione che si verifica dall'esame dei progetti da parte degli uffici pubblici di controllo che spesso determinano richieste assurde e immotivate causate da un'applicazione rigida e burocratica della normativa; anche in questo caso i riflessi di questo fenomeno coinvolgono i produttori di software costringendoli a soluzioni irrazionali pur di conseguire il benessere del progetto.

Questo fenomeno è ancora più accentuato dalla mancanza di coordinamento tra i vari uffici e le interpretazioni spesso contrastanti da ufficio a ufficio.

Anche in questo caso è necessaria un'attività di normalizzazione, stabilendo criteri comuni e condivisi, a partire dal contenuto e dalla modalità di stesura della Relazione di Calcolo, citata dal Capitolo 10.

Anche in questo AIST si è fatta parte diligente proponendo una bozza, presentata in occasione del Convegno tenuto a Bologna in occasione del SAIE ed esposta al Presidente del Consiglio Superiore dei Lavori Pubblici, professor Karrer che ne ha apprezzato i contenuti.

Con tale documento si intende proporre un semplice schema, con la flessibilità necessaria, in grado di sintetizzare i dati salienti del progetto, separandoli dai voluminosi tabulati dei calcoli svolti, che di fatto risultano incomprensibili per natura.

AIST auspica inoltre l'adozione di uno standard per lo scambio dei file dati, così come indicato dalla norma stessa, in modo da rendere possibile il confronto tra diversi programmi.

Resterebbe un'ultima validazione che è la norma non cita e che costituisce invece la più forte criticità relativamente allo sviluppo del software, cioè la capacità della normativa stessa di essere tradotta in software.

Come già detto, se il software è un diffusore di conoscenza la mancanza di chiarezza nella normativa non può che riflettersi negativamente nel processo di sviluppo del software.

Si potrebbe osservare che difficilmente la normativa può trattare in modo esaustivo tutti i casi possibili e che è compito del progettista esaminare il caso specifico partendo da considerazioni di carattere del tutto generale fornite appunto dalla norma. Pur accettando questa considerazione, resta il fatto che le NTC in alcuni punti sono molto vaghe (valga l'esempio di come applicare la gerarchia delle resistenze per telai non ortogonali) e la bibliografia del tutto carente.

Resta addirittura il dubbio se esista davvero qualcuno in grado di fornire risposte concrete ai tanti dubbi che ancora le NTC manifestano.

## **Un dialogo necessario tra tutti i protagonisti**

Molti progettisti, soprattutto all'inizio delle carriere e freschi di studi, si sono cimentati con la programmazione, così come oggi gli attuali produttori di software tecnico hanno fatto ai loro inizi scegliendo invece di farne la propria attività principale, investendo energie e rischiando in attività imprenditoriali che sviluppano un volume d'affari non proporzionale al servizio fornito, che penalizza la ricerca e lo sviluppo di nuove soluzioni. La gran parte di chi ha iniziato ha però smesso presto, riconoscendo la grande difficoltà della materia, la necessità di impegnare tante energie per ottenere prodotti che poi devono essere commercializzati, con tutte le difficoltà del caso.



È indubbio che solo una grande passione ed un grande amore possono fornire la spinta necessaria per svolgere questo mestiere, ma da sole possono bastare?

Si ha invece l'impressione che il software tecnico sia considerata scienza di serie B, lontana e molto più in basso rispetto alla torre eburnea in cui si isolano i ricercatori puri e che vedono nel mercato una contaminazione avvilente.

Ma da chi altri, se non dal mercato, può provenire il software professionale?

Se il normatore ha a cuore la sicurezza delle strutture, e non ci sono dubbi su questo, la prima cura dovrebbe essere il completo sostegno ai produttori di software, in quanto snodo fondamentale per l'applicazione vera delle NTC, coinvolgendoli nel processo di sviluppo della normativa stessa.

Per fortuna si registra qualche segnale positivo.

AIST si è fatta parte diligente organizzando un Gruppo di lavoro costituito da funzionari sensibili all'argomento, come l'arch. Di Mambro del Ministero delle Infrastrutture, che ha fornito supporto e incoraggiato un'iniziativa che dovrebbe essere appena normale, e che invece risulta straordinaria.

In questa attività il MIT, pur non partecipando alle varie Commissioni di revisione delle NTC, ha dimostrato lungimiranza e sensibilità per l'argomento. Lungimiranza in quanto il tema sarà sempre più importante per l'applicazione di questa e delle prossime normative, sensibilità in quanto ha dato spazio alla ns. associazione, credendo nell'iniziativa e rendendosi conto di quanto ritardo culturale ci sia ancora nel settore.

Con l'apporto di Reluis, coordinato dal prof. Gaetano Manfredi, si sono aperte prospettive molto interessanti con obiettivi di sviluppo che però richiedono risorse che nessuna società di software singolarmente potrà sostenere.

Argomenti che, si ripete, dovrebbero prima di tutto interessare chi per missione è tenuto ad incentivare lo sviluppo e la crescita della nazione.

In questi giorni si è aperto il processo di revisione delle NTC, già con ritardo rispetto a quanto prevede la legge. Saranno colti questi spunti?

L'Italia con le NTC ha colmato un ritardo immotivato rispetto ad altre nazioni, avvicinandosi all'Europa, ma la reale applicazione dipende anche dagli strumenti software che i progettisti disporranno e dalla loro capacità di utilizzarli al meglio.

Indipendentemente da tutto ciò, chi sviluppa software lo continuerà a fare, ma la passione da sola può bastare?

## **Serve cultura (e formazione)**

Se esiste una approfondita cultura dell'ingegneria, e su questo non c'è dubbio che gli Atenei ed i Progettisti italiani non sono secondi a nessuno, è tempo che si sviluppi anche una cultura del software per l'ingegneria e forse questo ci metterebbe all'avanguardia.



Per cultura si intende la conoscenza teorica ma soprattutto pratica dei concetti che stanno alla base dei programmi di calcolo strutturale, i limiti applicativi, i criteri di scelta, insomma tutto quanto la norma (ed il buon senso) richiede.

Sempre di più questa richiesta arriva dal mondo dei progettisti, i quali consapevoli di queste problematiche, in molti casi sono letteralmente spiazzanti, denunciano da tempo la situazione ma una risposta vera non è ancora arrivata.

È anche tempo che si veda i produttori di software per l'ingegneria come una risorsa indispensabile al ciclo di progettazione e realizzazione di opere edili.

Per questo AIST può offrire un valido contributo: chi meglio di chi il software lo crea può illustrarlo, soprattutto se come associazione si risolve il sempre presente equivoco del torna conto commerciale che ogni azienda singolarmente presenta.