

Il software, motore di innovazione nei sistemi industriali

Nell'inaugurare il nuovo anno di pubblicazione di *Automazione e Strumentazione*, mi sembra opportuno richiamare l'attenzione su uno degli oggetti più considerati e, in certa misura, temuti nell'ingegneria dell'automazione: il software.

La spinta crescente a fornire sistemi industriali più flessibili ed efficienti sta rivoluzionando la concezione di nuove macchine, apparati, dispositivi e processi. Non più ancorati all'applicazione di una tecnologia essenzialmente mono-disciplinare e tendenzialmente rigida, essi vengono progettati e costruiti mediante l'integrazione di molti componenti semplici e ripetitivi, la cui capacità di concorrere agli obiettivi complessivi viene determinata (ed eventualmente ridisegnata) dalla funzionalità di un complesso tessuto connettivo, "l'elettronica" o, più specificatamente, il software.

Si sviluppa un approccio alla progettazione basato sull'idea che la complessità delle funzioni richieste (e l'eccellenza delle prestazioni) si possa ottenere mettendo insieme componenti fisici relativamente standard (e quindi fortemente ripetibili) in numero eventualmente molto grande, garantendo la loro compatibilità di funzionamento attraverso l'azione di coordinamento del software di controllo e di gestione, destinato ad aumentare la propria complessità di pari passo con il crescere dell'articolazione del sistema complessivo in componenti. Che questa via possa consentire di sviluppare sistemi industriali innovativi e dotati di capacità impensabili fino a un decennio fa, appare evidente a tutti: basti pensare ai radicali cambiamenti in atto nel settore automobilistico o in quello delle macchine automatiche, dove sistemi essenzialmente meccanici arrivati ormai al loro massimo grado di sofisticazione stanno cedendo il passo a sistemi mecatronici, caratterizzati dal ruolo sempre più fondamentale del software. Si manifestano però di pari passo alcune questioni nuove, cui il progettista non sempre è preparato:

- in quale misura è possibile separare la progettazione dei componenti (e la loro ottimizzazione tecnico-economica) dalla verifica di funzionalità del sistema complessivo?
- in quale misura è possibile demandare alla progettazione del software i requisiti prestazionali, di sicurezza e di affidabilità richiesti al sistema ?
- come fare per prevedere (in fase di progettazione) e per verificare (in fase di collaudo) il comportamento di un sistema caratterizzato da complessità quasi illimitata e da scenari operativi soggetti a cambiamenti durante la sua vita (attraverso la parametrizzazione del software) ?
- attraverso quali passi può essere sviluppato il processo di integrazione ?
- come limitare i rischi di un funzionamento imprevedibile del software e le sue conseguenze ?

Appare talvolta attraente un punto di vista “ottimistico” secondo il quale sarebbe sufficiente dimensionare correttamente i componenti (dopo averne individuato il funzionamento “nominale”), fidando poi nel fatto che, come un collante miracoloso, il software sia in grado di coordinare l’azione di ogni parte sino al conseguimento degli obiettivi previsti, magari adattabili alle variabili esigenze di funzionamento. In realtà è necessario tener presente che esistono limiti nelle prestazioni di un sistema complesso che non dipendono singolarmente dai componenti (ma si determinano nella loro integrazione) e che non sono superabili attraverso pure modifiche del software. Dalle conseguenze negative di questo principio occorre proteggersi adottando un processo di progettazione integrata (che includa la sensoristica e il software) fin dalle prime fasi; da qui la necessità di dotarsi di metodologie di progetto del software di controllo che prevedano affinamenti successivi nelle specifiche funzionali per evitare inutili dettagli nelle fasi preliminari. Per mettere il progettista in grado di prevedere il comportamento del sistema in fieri, è necessario supportarlo con strumenti di modellistica e simulazione interdisciplinari (che integrino hardware e software), oltre a strumenti di progettazione assistita da calcolatore, in particolare per le funzioni software. La progettazione integrata, appoggiandosi naturalmente sull’affinamento contestuale delle varie scelte progettuali, porta a concepire ambienti di modellistica molto flessibili, cioè capaci di utilizzare modelli sempre più dettagliati, man mano che il processo di progettazione avanza, pur mantenendo la necessaria compatibilità tra le interfacce.

Fronteggiare l’insidia dei “buchi” nel software è esigenza vitale, soprattutto quando sono coinvolte funzioni rilevanti per la sicurezza o per la continuità della produzione. Esistono metodologie che consentono di individuare errori che compromettono la correttezza formale degli algoritmi specificati (o progettati); molti errori, invece, sono più direttamente legati allo scenario operativo prevedibile e al risultato “sostanziale” dell’azione di controllo e possono essere individuati solo “ricostruendo” tutti gli scenari rilevanti, ossia operando per simulazione. Su ambedue i fronti, i “buchi” formali e quelli “sostanziali”, gli strumenti disponibili sono di grande ausilio per limitare i rischi di malfunzionamento del software e di inadeguatezza del controllo in situazioni critiche; va riconosciuto che, data la crescente complessità dei sistemi industriali (e meccatronici in particolare), la verifica di correttezza in fase di progettazione e collaudo risulta ancora molto onerosa e spesso scoraggia una verifica esaustiva. Non avendo certo la pretesa di dare risposta conclusiva, qui, alle problematiche sollevate dalla riconversione tecnologica dei sistemi industriali a elevato grado di automazione, mi sembra che valga la pena, tuttavia, sottolineare che il crescente grado di integrazione tra tecnologie elettro-meccaniche e tecnologie dell’informazione nei prodotti e sistemi industriali dà rilievo al ruolo di una figura professionale nuova, attrezzata con un bagaglio metodologico e un insieme di conoscenze adeguate per gestire un processo di ideazione, progettazione e realizzazione fortemente interdisciplinare, solidamente fondato sull’impiego sistematico di modelli funzionali sia dei componenti fisici che del software di controllo (e, più in generale, di gestione). Su questo terreno si sta misurando anche la capacità delle nostre università tecniche di accompagnare efficacemente il processo di innovazione tecnologica.

