

Prototipazione virtuale di sistemi mecatronici

Con Modelica

Gianni Ferretti, Gianantonio Magnani, Paolo Rocco

Miglioramenti significativi nelle prestazioni dinamiche di sistemi mecatronici si possono conseguire solo comprendendo a fondo le mutue interazioni fra meccanica, elettronica e controllo. Un possibile approccio è costituito dalla prototipazione virtuale, che comporta l'integrazione della simulazione dinamica multi-dominio nel processo di progettazione, al fine di riprodurre e analizzare gli effetti delle scelte di progetto sulle prestazioni complessive. Questo articolo definisce i requisiti del linguaggio di modellistica e descrive lo strumento software Dymola utilizzato per la simulazione di un centro di lavorazione.

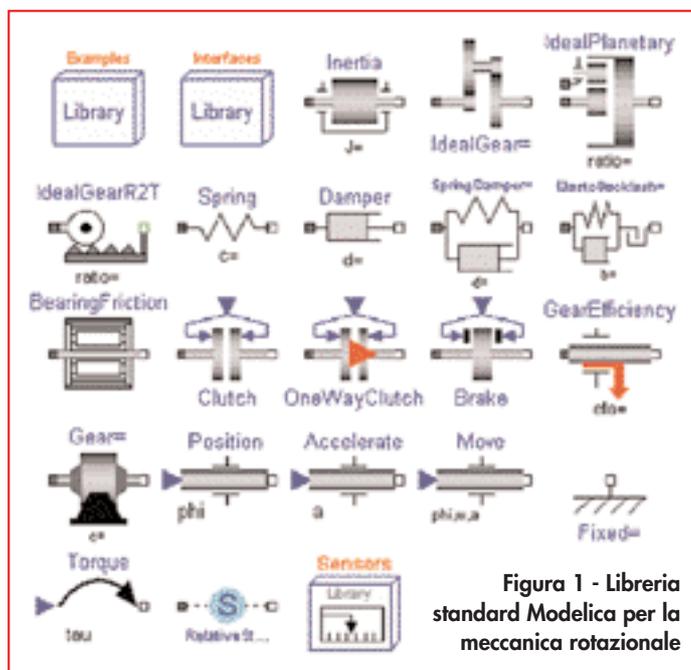


Figura 1 - Libreria standard Modelica per la meccanica rotazionale

Ogni sistema tecnologico in cui la meccanica, l'elettronica e il controllo concorrano a determinarne le prestazioni complessive appartiene alla famiglia dei sistemi mecatronici. Esempi di tali sistemi sono i robot, le macchine utensili e i centri di lavoro. Mentre il processo di progettazione tradizionale dei sistemi mecatronici non comporta interazioni significative fra i progettisti meccanici e gli esperti di controllo ed elettronica di potenza, la richiesta del mercato di precisioni sempre maggiori impone oggi un progetto sinergico, in cui le mutue interazioni fra le scelte fatte nel progetto meccanico e nel progetto dell'elettronica di potenza siano comprese appieno. Le prestazioni dinamiche, piuttosto che il dimensionamento statico, costituiscono oggi il problema principale, e le prestazioni dinamiche non possono essere migliorate se la macchina non viene analizzata chiudendo l'anello fra elettronica, controllo e meccanica. Lo studio sperimentale di prototipi fisici è però generalmente oneroso in termini economici e temporali. La *prototipazione virtuale*, un nuovo modo di definire

la simulazione dinamica [1, 2] della macchina controllata, si adatta meglio ai limiti di tempo e budget dei centri di ricerca e sviluppo industriali. Se le scelte di progetto, dal punto di vista sia meccanico sia elettronico, vengono verificate prima di realizzare un prototipo fisico, si approfondisce maggiormente la conoscenza del sistema. Ciò è vero, ovviamente, soltanto se il modello simulato è una replica affidabile della realtà e se lo strumento di simulazione è di facile utilizzo per progettisti di aree diverse.

È piuttosto diffusa la sensazione che parecchie imprese, in cui la competizione e i miglioramenti delle prestazioni dinamiche delle macchine sono avvertiti come obiettivi fondamentali, siano in qualche modo pronte ad introdurre questo tipo di strumenti nel processo di progettazione, ma che non ci sia una piena consapevolezza sulle reali potenzialità consentite dal nuovo approccio. Le incertezze nascono spesso anche sui requisiti del linguaggio di modellistica e dello strumento software da adottare. Ambienti di simulazione come Simulink sembrano essere estremamente po-

tenti nel replicare il sistema di controllo, ma si evidenziano difficoltà non appena si tenta una modellistica *fisica* del sistema, in quanto il flusso unidirezionale del segnale, implicato nella descrizione procedurale a blocchi, risulta inadeguato. Viceversa, alcuni pacchetti commerciali (per esempio Adams, Altair, Dads) consentono un'accurata e relativamente semplice simulazione della meccanica, dal dimensionamento statico allo studio delle vibrazioni, ma generalmente l'utente incontra qualche difficoltà nel simulare gli anelli di regolazione, comprensivi di elettronica e controllo. Nel presente articolo si discutono i requisiti che devono caratterizzare uno strumento software per la simulazione di sistema di dispositivi mecatronici. Varietà dei domini fisici, approccio orientato agli oggetti, affidabilità ed efficienza della simulazione numerica, integrazione con Cad meccanici, sono le caratteristiche principali evidenziate.

Successivamente si descrive l'utilizzo di Dymola, basato sul linguaggio Modelica [3, 4], che sembra soddisfare la maggior parte dei requisiti menzionati. Dapprima si descrivono le librerie standard di Modelica per la simulazione di sistemi mecatronici. Quindi si descrivono alcuni modelli sviluppati appositamente, relativi ai motori brushless e ai driver di corrente, e utilizzati nella prototipazione virtuale di un centro di lavoro.

Requisiti del linguaggio di modellistica e dell'ambiente di simulazione

La scelta di un linguaggio di modellistica e di uno strumento software nell'ambito di un progetto industriale richiede l'attenta considerazione di alcuni aspetti, di seguito discussi brevemente.

Simulazione multi dominio

L'ambiente di simulazione deve consentire di integrare facilmente modelli appartenenti ai domini meccanico, elettrico e dei sistemi di controllo. Come già menzionato nell'introduzione, uno studio sistematico delle prestazioni dinamiche di un sistema mecatronico non può ignorare le mutue interazioni fra i tre domini. Non è consigliabile, ad esempio, l'utilizzo di strumenti software che consentono una simulazione dettagliata della meccanica ma che richiedono aggiunte *ad-hoc* da parte dell'utente relativamente ai componenti elet-

tronici e di controllo, magari sviluppati in un linguaggio di modellistica diverso. Una tale soluzione determina generalmente lunghi tempi di sviluppo e una ridotta accuratezza.

Paradigma dichiarativo

Per la modellistica dei componenti fisici del sistema, siano essi meccanici o elettrici, si deve utilizzare un linguaggio *dichiarativo*. I linguaggi procedurali, in cui si postula un flusso unidirezionale di informazione da un ingresso a un'uscita, risultano inadeguati a descrivere gli scambi istantanei di potenza nei sistemi fisici. I modelli *non causali*, descritti da sistemi Dae (Differential Algebraic Equations) [5], consentono di rappresentare i componenti del sistema mecatronico nel modo più naturale e consistente dal punto di vista fisico. Anziché all'utente, il compito di definire la causalità computazionale del modello assemblato deve essere affidato all'ambiente di simulazione.

Paradigma object-oriented

Il linguaggio di modellistica deve conformarsi ai moderni paradigmi object-oriented dell'ingegneria del software. Gli aspetti del paradigma object-oriented maggiormente attinenti alla modellistica dei sistemi fisici sono la *modularità* (il sistema è composto dall'aggregazione di moduli), l'*astrazione* (la descrizione interna del modulo è separata dalla sua interfaccia), l'*incapsulamento* delle informazioni (solo le variabili di interfaccia sono accessibili agli altri moduli), il *riutilizzo* dei moduli mediante la parametrizzazione. Inoltre, soprattutto nel caso di progetti complessi, l'utilizzo dell'ambiente di simulazione è notevolmente facilitato se i moduli vengono strutturati in librerie.

Modificabilità

Ogni ambiente di simulazione è dotato di librerie di componenti di base, che generalmente coprono le necessità relative alla simulazione di semplici sistemi mecatronici. Nascono comunque facilmente situazioni in cui è necessario progettare moduli appositi. Un esempio è il modello del motore brushless realizzato in Dymola, descritto nel paragrafo *Adattamento dei modelli*. Da questo punto di vista, ovviamente, la possibilità di espandere moduli di libreria, piuttosto che la necessità di uno sviluppo *ex-novo*, facilita notevolmente il compito dell'utente.

Trattamento efficiente di vincoli ed equazioni

L'assemblaggio di moduli descritti da equazioni Dae genera facilmente sistemi di *indice* elevato [5] e *loop* algebrici. L'insieme di equazioni deve quindi essere rielaborato, ad esempio mediante manipolazione simbolica, per consentire l'integrazione numerica. Alcuni strumenti di simulazione utilizzano l'algoritmo di Pantelides [6] (o simili) per ridurre l'indice del sistema.

Altri utilizzano tecniche di rilassamento dei vincoli per generare un sistema Ode (Ordinary Differential Equations) in cui, però, i vincoli non sono identicamente soddisfatti durante i transitori. L'efficienza di questa fase preliminare di elaborazione risulta fondamentale per l'efficienza dell'ambiente di simulazione complessivo.

Efficienza e affidabilità della simulazione numerica

Il solutore deve essere in grado di gestire *eventi*, come discontinuità e variazioni improvvise delle variabili esterne.

È necessario quindi individuare l'istante di attivazione dell'evento con precisione, adottando un passo variabile nell'integrazione numerica. La simulazione dei sistemi in anello chiuso deve essere affrontata nel modo più affidabile: soluzioni in cui la parte fisica del sistema è simulata con un sistema Dae, mentre la parte di controllo procedurale è simulata con un solutore Ode, pur mantenendo la sincronizzazione fra i due solutori, non sembrano garantire la stessa affidabilità di un singolo solutore applicato all'intero sistema.

Integrazione con Cad meccanici

I progettisti meccanici sono propensi a utilizzare strumenti software che possono essere facilmente integrati con i Cad meccanici di uso corrente (ad esempio SolidWorks).

Integrazione significa, in particolare, un utilizzo diretto dei parametri di massa e di inerzia generati dal Cad nell'ambiente di simulazione dinamica, senza alcuna trascrizione manuale (che generalmente allunga i tempi di sviluppo e si presta alla generazione di errori). Inoltre, per problemi in cui la flessibilità dei materiali è rilevante, risulta assai utile l'integrazione con strumenti Fea (Finite Element Analysis).

Interfacce di pre e post-processing

I moduli relativi ai componenti devono essere assemblati mediante una Gui (Graphical User Interface). Sebbene le interfacce bidimensionali siano più comuni e di più facile utilizzo si dovrebbero poter utilizzare anche interfacce di tipo tridimensionale. Ogni modulo deve essere associato a un simbolo grafico, che sia il più evocativo possibile della natura fisica del sistema.

Deve essere possibile visualizzare i risultati della simulazione sia mediante grafici bidimensionali, sia mediante animazioni tridimensionali. I grafici bidimensionali sono più utili ai fini di un'analisi dettagliata delle proprietà dinamiche, mentre le animazioni tridimensionali aiutano a comprendere il moto complessivo del sistema.

Librerie standard di Modelica per la mecatronica

Nessuno degli ambienti software commerciali disponibili soddisfa completamente tutti i requisiti descritti. È necessario quindi raggiungere un compromesso fra diverse esigenze. In questo articolo si illustra un'esperienza di utilizzo di Dymola (basato sul linguaggio Modelica) [3, 4], relativa alla simulazione di centri di lavorazione.

Dymola soddisfa pienamente i requisiti di approccio multi-dominio, object-orientation, adozione di paradigmi di modellistica non causale e dichiarativa e modificabilità. L'efficienza dell'integrazione numerica è garantita da un sofisticato strato software, in cui il modello assemblato viene inizialmente elaborato (*tradotto*) mediante manipolazione simbolica delle equazioni, al fine di ridurre l'indice e produrre un codice di simulazione efficiente. La successiva integrazione numerica è basata su un solutore Dae di concezione avanzata. Dymola è inoltre dotato di un'interfaccia bidimensionale per l'assemblaggio del modello e di un semplice modulo di animazione 3D per il post-processing dei risultati di simulazione. Non sono attualmente commercializzate interfacce con ambienti Cad, si sta però studiando l'interfacciamento del linguaggio Modelica con SolidWorks [7]. Tutti i moduli in Dymola sono scritti nel linguaggio Modelica [4], prodotto dai lavori di un'associazione, fondata nel 1996, avente come obiettivo la specifica di un linguaggio di modellistica orientato agli oggetti adatto alla

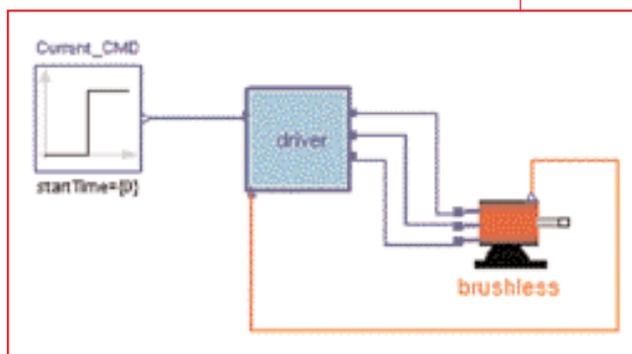
modellistica dei sistemi fisici. L'associazione aggiorna le specifiche dei linguaggi di modellistica e sviluppa le librerie standard di Modelica, che sono liberamente accessibili. Dymola è uno degli ambienti software commerciali che supportano completamente il linguaggio Modelica.

Esistono parecchie librerie Modelica utilizzabili per un progetto mecatronico. Ad esempio, la libreria Modelica.Mechanics.Rotational (Figura 1) fornisce componenti per la simulazione di sistemi meccanici rotatori monodimensionali, utili a riprodurre catene di trasmissione, riduttori, inerzie, elementi molla/smorzatore, giochi e attriti.

È significativo sottolineare come la connessione rigida di due inerzie, operazione estremamente utile nella simulazione di catene di trasmissione, che genera un sistema Dae di indice 3, possa essere eseguita senza alcun particolare accorgimento: l'ambiente di simulazione è in grado di risolvere i vincoli sulle posizioni e sulle velocità angolari. È disponibile una libreria simile (Modelica.Mechanics.Translational) per la simulazione di sistemi meccanici traslazionali monodimensionali. I modelli dei sistemi meccanici tridimensionali, come i robot, le macchine utensili o i veicoli, possono essere costruiti mediante la libreria Modelica.Additions.Multi-body. I componenti di questa libreria possono essere combinati con le librerie meccaniche monodimensionali, mediante flange, nei modelli Joint (i giunti introducono i gradi di libertà nel sistema meccanico). Gli elementi Body richiedono invece la specifica dei parametri geometrici e inerziali (massa, centro di massa e tensore di inerzia), che potrebbero essere assegnati in maniera più efficiente se fosse disponibile l'interfaccia con un modellatore solido. A differenza di altri ambienti di simulazione, il dominio elettrico ed i sistemi di controllo possono essere simulati con la stessa facilità del dominio meccanico.

La libreria Modelica.Electrical.Analog consente la simulazione di semplici circuiti elettrici, sempre rispettando il paradigma dichiarativo. Tutti i componenti della libreria sono dotati di *terminali* elettrici, in cui avviene lo scambio di potenza (tramite le variabili di tensione e corrente). Di conseguenza, ogni elemento fisico (sia esso un resistore o un condensatore, ad esempio) ha una ed una sola controparte nella libreria, indipendentemente dal fatto che l'elemento sia comandato in tensione o in corrente in un particolare assemblaggio. In questo modo si garantisce una

completa riusabilità del modello. La libreria Modelica.Blocks contiene blocchi procedurali, del tipo ingresso/uscita, utilizzabili per costruire semplici schemi a blocchi nella simulazione dei regolatori (digitali e analogici). Sulla base della nostra esperienza la libreria, sebbene alquanto limitata, consente di simulare la maggior parte delle funzioni dei regolatori industriali, ovviamente mediante aggregazione di elementi di libreria. Nel prossimo paragrafo si illustra un esempio di simulazione di un driver di corrente per un motore brushless.



Adattamento dei modelli

I più comuni sistemi di attuazione utilizzati nella robotica, nell'industria delle macchine utensili e nei centri di lavorazione sono di gran lunga i servomeccanismi con motori brushless, connessi al carico mediante catene di trasmissione (o riduttori).

In un motore brushless la commutazione elettromeccanica tipica dei motori a corrente continua è sostituita dalla commutazione elettronica delle correnti nelle tre fasi degli avvolgimenti di statore. Ciò in teoria dovrebbe garantire che la coppia elettromagnetica sviluppata all'albero motore sia indipendente dalla posizione del rotore. In realtà, alcune imperfezioni costruttive nel motore o nel drive, ove si realizza la commutazione elettronica, producono un'ondulazione (*ripple*) sulla coppia effettiva [8, 9]. Dal punto di vista del dimensionamento statico del sistema di attuazione questo potrebbe essere considerato un problema minore, mentre dal punto di vista delle prestazioni dinamiche è di notevole rilevanza. Il ripple di coppia potrebbe, infatti, eccitare le risonanze del sistema meccanico, generalmente associate all'accoppiamento elastico fra motori e carichi [10].

La prototipazione virtuale è un utile strumento per studiare questi fenomeni e, in particolare, per separare gli effetti delle singole fonti di disturbi sulle prestazioni del sistema. I do-

Figura 2 - Modello completo del sistema

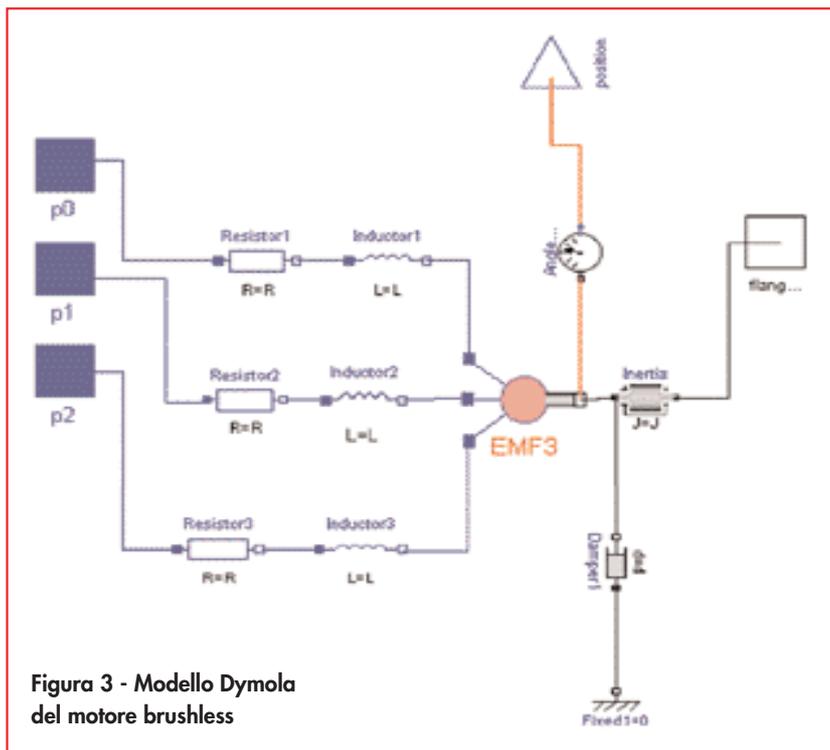


Figura 3 - Modello Dymola del motore brushless

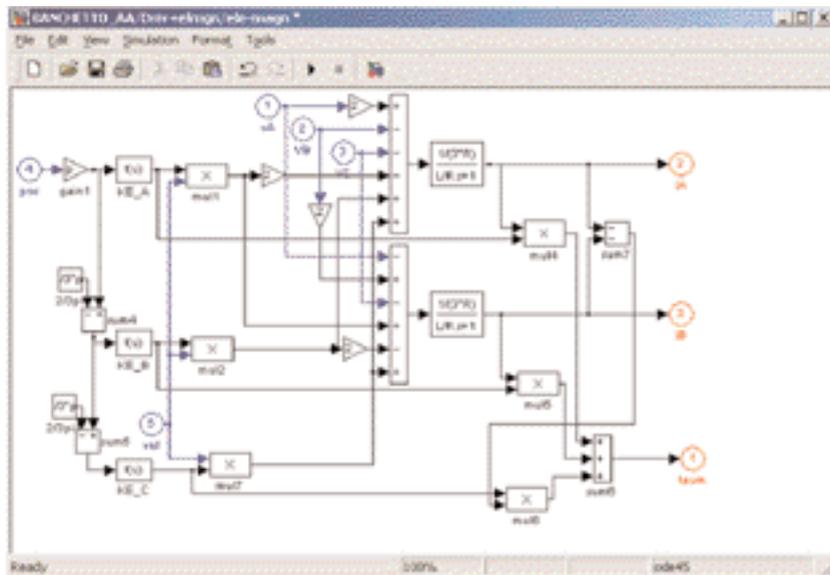


Figura 4 - Modello Simulink della parte elettrica del motore brushless

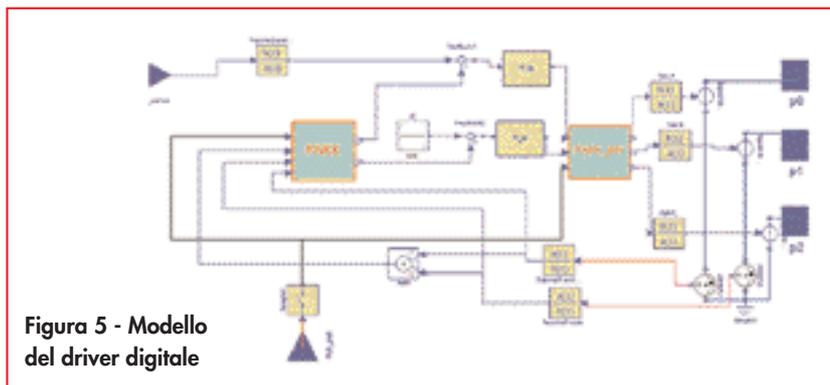


Figura 5 - Modello del driver digitale

mini meccanico, elettronico e del controllo sono, infatti, coinvolti in modo paritetico in questo tipico problema mecatronico.

Il modello del sistema in Dymola è ottenuto dalla connessione in retroazione di due sotto-modelli: uno rappresenta il motore brushless, l'altro il driver (Figura 2). I due modelli sono connessi mediante tre connettori elettrici (le tre fasi di un motore) e un connettore di controllo (la misura della posizione del rotore).

Il modello del motore brushless è mostrato in Figura 3. Le tre fasi sono connesse a stella nel blocco Emf3, in cui sono definiti i profili di forza controelettrica dovuti a ciascuna fase. A sua volta, il modello Emf3 generalizza il modello Emf della libreria Modelica.Electrical.Analog.Basic. I profili nominali sinusoidali possono quindi essere modificati per analizzare il ripple dovuto alle armoniche di ordine più elevato. La coppia applicata alla flangia del blocco Emf3 è determinata dalla somma dei prodotti delle correnti e dei profili di forza controelettrica delle singole fasi. È importante sottolineare come l'ambiente di modellistica non causale fornito da Dymola consenta di specificare nel modo più naturale possibile il vincolo algebrico sulle correnti (somma nulla delle correnti).

Tale vincolo è ovviamente descritto con maggiore difficoltà in ambienti di simulazione basati sulla specifica causale dei modelli, in forma di sistemi Ode. A titolo di esempio si consideri il modello Simulink della parte elettrica del motore brushless, mostrato in Figura 4. In particolare, si noti come il vincolo algebrico sia stato risolto (dall'utente), al fine di poter esprimere una corrente di fase in funzione delle altre due. Ovviamente l'elaborazione del modello ha richiesto tempo e il risultato è di difficile leggibilità, soprattutto in virtù del fatto che i blocchi introdotti non hanno un corrispondente fisico.

Si è inoltre realizzata una versione *full digital* del controllore di corrente (Figura 5), adottando uno schema di controllo vettoriale [11], in cui le correnti di fase sono dapprima trasformate in corrente *di quadratura* e corrente *diretta* mediante la trasformazione di Park. Tali correnti sono quindi regolate mediante due anelli digitali, assumendo come riferimento per la corrente di quadratura l'uscita del regolatore di posizione/velocità, ed un valore nullo per la corrente diretta. Le tensioni di comando all'uscita di due regolatori PI, dotati di dispositivo anti-windup, sono infine trasformate nelle tre tensioni di fase mediante la trasformazione di Park inversa.

Si è evitato di simulare l'amplificatore Pwm che, operando a una frequenza di 10÷20 kHz, con una modulazione dell'ampiezza degli impulsi di 1 μs, avrebbe determinato passi di integrazione inferiori a 1 μs. Passi di integrazione così piccoli sono assolutamente normali nella simulazione dell'elettronica di un driver, ma risultano inadeguati in una simulazione combinata del sistema meccatronico, le cui frequenze di interesse sono notevolmente inferiori alle frequenze indotte dal dispositivo Pwm. La simulazione dettagliata della dinamica elettronica comporterebbe quindi un eccessivo e non necessario allungamento dei tempi di simulazione, soprattutto nel caso di più istanze del modello del motore.

Simulazione di un centro di lavoro

Il modello del motore brushless è stato incluso in una libreria di elementi utilizzata per simulare un centro di lavorazione completo a tre assi Cartesiani (asse X, Y e Z). La Figura 6 mostra il livello più esterno del simulatore. Il modello è composto di tre parti: la simulazione del Cn (Controllore Numerico) e dei servodriver, realizzate interamente con blocchi Dymola, la simulazione della catena di trasmissione per ciascun asse, ove si è utilizzato il modello del motore brushless, e la simulazione della catena cinematica, realizzata con gli elementi della libreria Modelica.Additions.Multibody (Figura 7).

I modelli del riduttore e della trasmissione a cinghia sono mostrati, come esempio, in Figura 8 e in Figura 9 rispettivamente. Entrambi contengono un modulo realizzato appositamente: per facilitare la verifica si è introdotto nel riduttore un elemento che consente la simulazione di una trasmissione rigida o elastica (con gioco). Nel modulo della trasmissione a cinghia è presente un elemento che descrive un modello semplificato di una cinghia. Il primo elemento è stato realizzato componendo elementi della libreria di Modelica, il secondo per scrittura diretta delle equazioni di bilancio sulla cinghia. La Figura 10 mostra la struttura interna del modulo Cn, costituita da un interpolatore (responsabile della generazione di traiettoria) e da tanti controllori in anello chiuso quanti sono gli assi nella catena cinematica. A sua volta ciascun drive ha la struttura riportata in Figura 11, in cui un controllore di posizione/velocità digitale comanda un controllore digitale di corrente, descritto nel

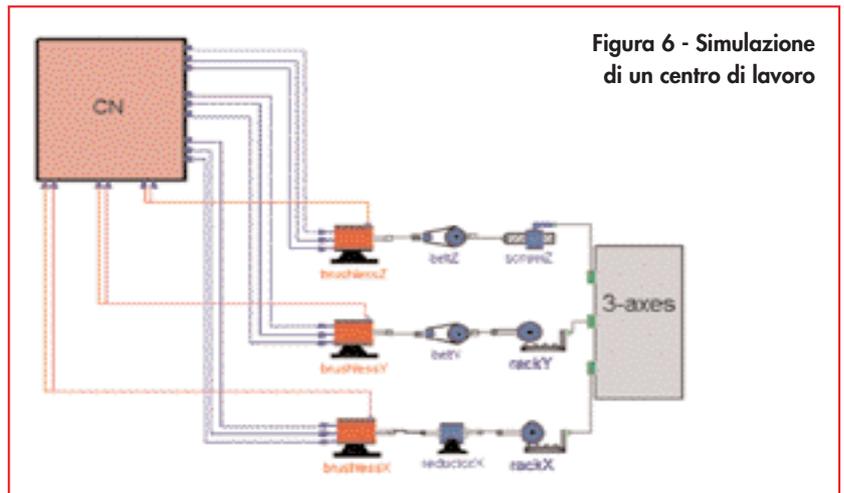


Figura 6 - Simulazione di un centro di lavoro

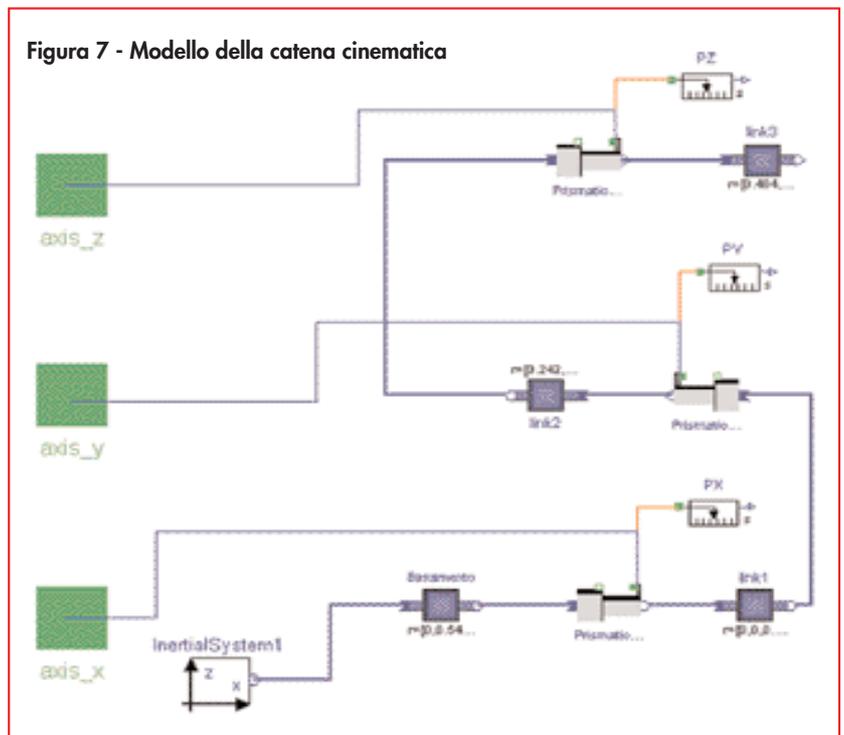


Figura 7 - Modello della catena cinematica

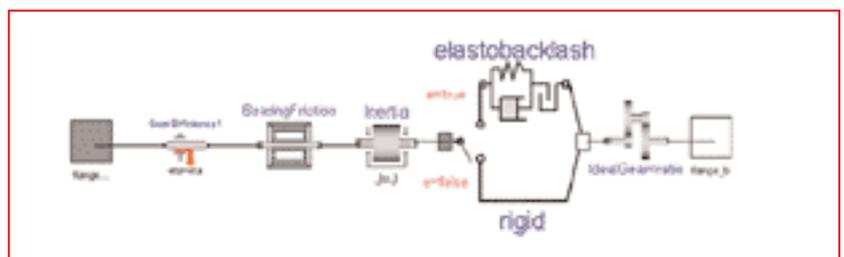


Figura 8 - Simulazione del riduttore

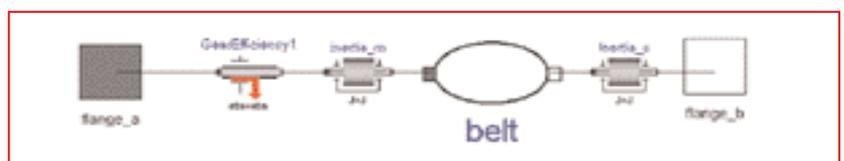


Figura 9 - Simulazione della cinghia di trasmissione

Figura 10
Simulazione del Cn

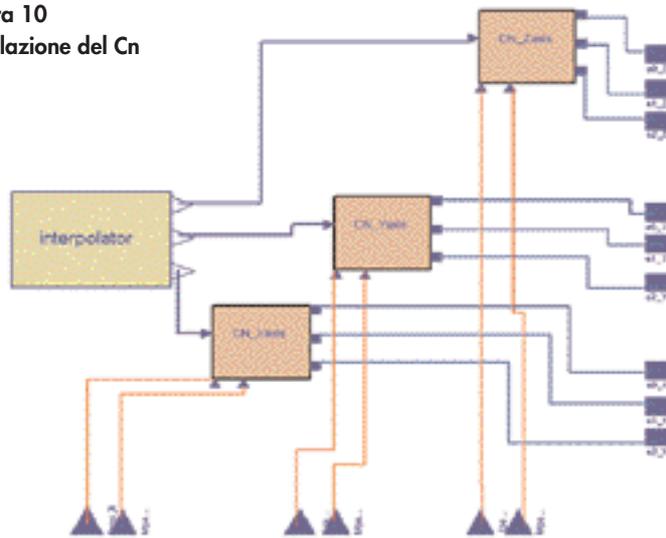
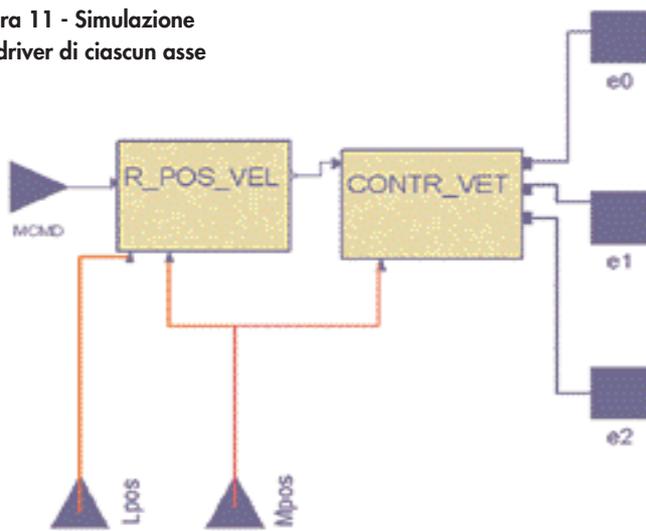
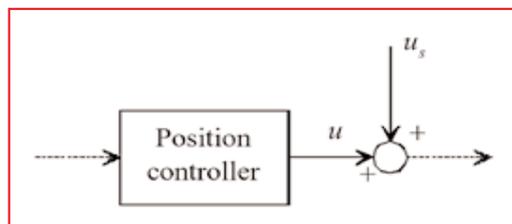


Figura 11 - Simulazione
del driver di ciascun asse



paragrafo precedente. La natura multidominio di Dymola e l'assemblaggio del modello basato sulle interconnessioni fisiche sono risultati elementi essenziali per realizzare un ambiente di simulazione affidabile e di facile utilizzo, anche senza una preparazione specifica nella modellistica dinamica. La prototipazione virtuale del centro di lavoro ha comportato però anche uno studio sperimentale, finalizzato a validare i moduli del modello di simula-

Figura 12
Esperimento



zione a fronte del comportamento della macchina reale. La validazione, che generalmente richiede di strumentare la macchina con sensori e dispositivi di memoria aggiuntivi, è un'attività piuttosto impegnativa che deve essere affrontata in modo sistematico. È stato necessario in particolare realizzare esperimenti ad-hoc per identificare parametri quali le costanti di rigidità delle trasmissioni, gli smorzamenti e i parametri di attrito.

Nell'effettuare questi esperimenti si è sfruttata una modalità operativa offerta dal controllore numerico utilizzato, che consente di determinare la risposta in frequenza dell'anello chiuso di posizione.

La modalità opera nel modo seguente: un segnale sinusoidale di test a frequenza crescente (*sweep*) $u_s(t)$ viene sovrapposto all'uscita del controllore di posizione (Figura 12). Si identifica quindi la risposta in frequenza fra tale ingresso e l'uscita $u(t)$. Le Figure 13 e 14 mostrano rispettivamente il segnale di test (in percentuale della corrente massima) e l'uscita del controllore, in un esperimento effettuato sull'asse X. Si evidenzia una coppia antirisonanza-risonanza principale.

A partire dai data-sheet dei componenti si sono stimati i parametri di elasticità delle cinghie e del sistema pignone-cremagliera. Successivamente, a partire da tali dati, si è proceduto ad una taratura per tentativi dei parametri viscoelastici delle trasmissioni, per confronto fra simulazioni e dati sperimentali. Un primo risultato è mostrato in Figura 15. Il picco principale a circa 9 s è dovuto ad una differente taratura dei parametri del regolatore rispetto all'esperimento (si è verificata una certa ambiguità nella definizione dei valori effettivi dei parametri utilizzati dal controllore numerico), che hanno determinato un minor margine di fase ed un picco di risonanza più elevato. Comunque, la coppia antirisonanza-risonanza a più bassa frequenza sembra essere stata replicata con una certa accuratezza. D'altro canto, la frequenza di risonanza più elevata visibile in Figura 14 verso la fine dell'esperimento non appare in Figura 15, probabilmente a causa di una sovrastima del coefficiente di attrito. Gli stessi esperimenti realizzati sull'asse Y hanno posto in evidenza due risonanze vicine a bassa frequenza. Dopo un'accurata analisi si è stabilito che questo comportamento non può essere attribuito alle trasmissioni e si è quindi concentrata l'attenzione anche sulla cedevolezza strutturale. Si sono ottenuti alcuni incoraggianti risultati ma la ricerca in questa direzione è tutt'ora in corso.

Conclusioni

La scelta del linguaggio di modellistica e dell'ambiente di simulazione è un passo fondamentale per il buon esito di un'attività di prototipazione virtuale di una macchina. In questo articolo si sono fornite le specifiche per una scelta razionale. L'ambiente Dymola, basato sul linguaggio Modelica, è risultato soddisfare la maggior parte dei requisiti, ed è stato applicato alla modellistica di un centro di lavorazione.

Bibliografia

- [1] F. Cellier, (1991), *Continuous system modelling*, Springer Verlag.
- [2] G. Ferretti, S. Filippi, C. Maffezzoni, G. Magnani, P. Rocco, (1999), *A modular approach to dynamic virtual reality modeling of robotic systems*, Ieee Robotics and Automation Magazine, Vol. 6, pp.13–23.
- [3] M. Tiller, (2001), *Introduction to physical modelling with Modelica*, Kluwer.
- [4] S.E. Mattsson, H.Elmqvist, M. Otter, (1998), *Physical system modeling with modelica*, Control Engineering Practice, Vol. 6, pp. 501–510.
- [5] K.E. Brennan, S.L. Campbell, L. R. Petzold, (1989), *Numerical solution of initial-value problems in differential algebraic equations*, North-Holland.
- [6] C.C. Pantelides, (1988), *The consistent initialisation of differential-algebraic systems*, Siam Journal of Scientific and Statistical Computing, Vol. 9, pp. 213–231.
- [7] V. Engelson, H. Larsson, P. Fritzson, (1999), *A design, simulation and visualization environment for object-oriented mechanical and multi-domain models in modelica*, In: Proc. Of Ieee International Conference on Information Visualization, London, UK, pp. 188-93.
- [8] T.M. Jahns, W.L. Soong, (1996), *Pulsating torque minimization techniques for permanent magnet ac motor drives - a review*, Ieee Transactions on Industrial Electronics, Vol. 43, No. 2, pp. 321-330.
- [9] G. Ferretti, G. Magnani, P. Rocco, (1998), *Modelling, identification and compensation of pulsating torque in permanent magnet ac motors*, Ieee Transactions on Industrial Electronics, Vol. 46, No. 6, pp. 912–920.
- [10] G. Ferretti, G. Magnani, P. Rocco (1999), *Force oscillations in contact motion of industrial robots: an experimental investigation*, Ieee/Asme Transactions on Mecha-

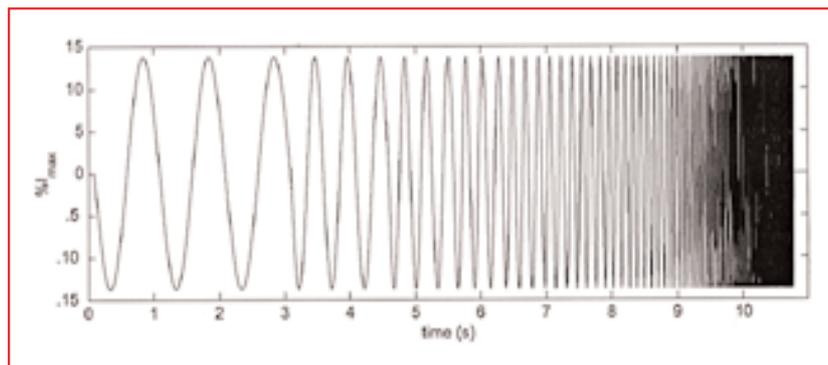


Figura 13 - Segnale di test

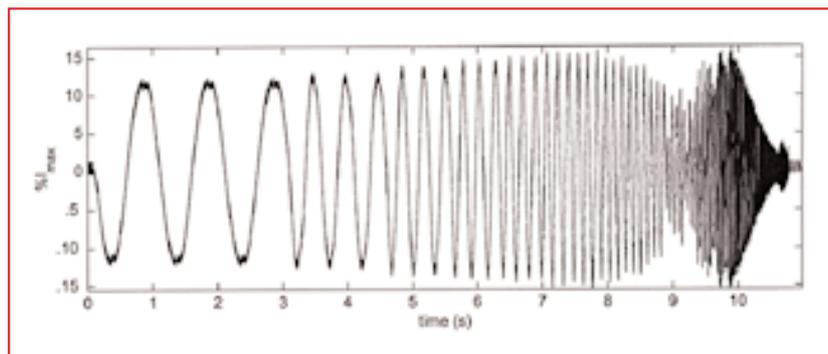


Figura 14 - Uscita del regolatore sperimentale

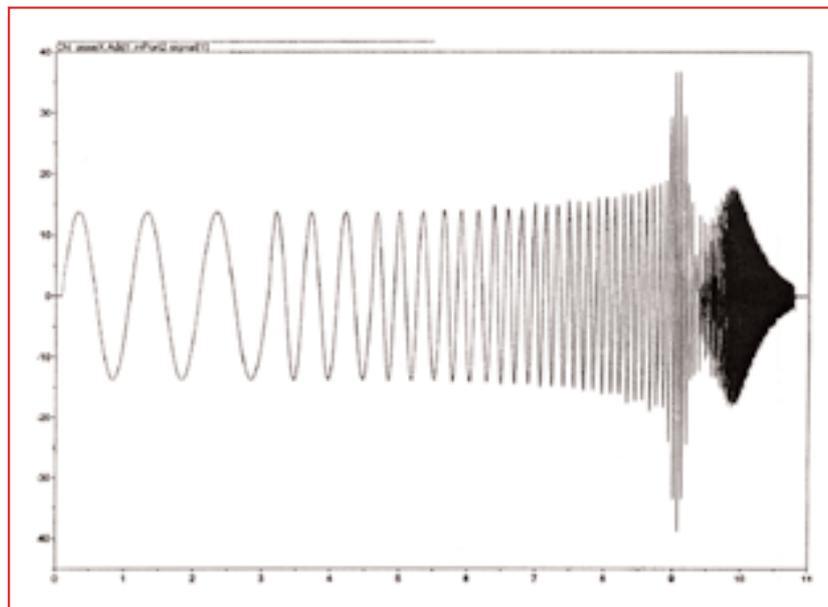


Figura 15 - Uscita del regolatore simulata

tronics, Vol. 4, No. 1, pp. 86-91.

- [11] Texas Instruments (1997), Digital signal processing solution for permanent magnet synchronous motor, Application Note, Literature Number: BprA044.

Prof. ing. G. Ferretti, Professore Associato, prof. ing. G. Magnani, Professore Ordinario, prof. ing. P. Rocco, Professore Associato, Dipartimento di Elettronica e Informazione, Politecnico di Milano.