

Approfondimento seminario
Frontiere dei sistemi autonomi

Prof. Andrea Omicini

Collegio superiore - A.A. 2017/2018
Alma Mater Studiorum

Zonzini Federica

Indice

1	Sistemi Autonomi nell'era dell'IoE	9
1.0.1	Caso di studio: Big Data & Cloud Computing	11
1.1	Real Time Damage Detection in applicazioni SHM	13
	Bibliografia	15

Abstract

Lista degli acronimi

AC Autonomic Computing

IoT Internet of Things

IoE Internet of Everything

NoSQL Not only SQL

SHM Structural Health Monitoring

Capitolo 1

Sistemi Autonomi nell'era dell'IoE

I sistemi autonomi nell'ambito dell'ingegneria dell'informazione nascono con l'intento di creare architetture e processi di calcolo in grado di auto-regolarsi e gestire in modo automatico non soltanto l'elaborazione dei dati, capacità da sempre acquisita, quanto piuttosto di controllare lo stato del proprio funzionamento senza richiedere intervento umano.

Questa classe di sistemi viene più propriamente chiamata con il termine inglese *Autonomic Computing (AC)*, cioè "*computing systems that can manage themselves given high-level objectives from administrators.*" [1]. Coniata per la prima volta nel 2001 da Paul Horn, vicepresidente del settore ricerca IBM, l'idea di queste infrastrutture di calcolo cerca di emulare a livello informatico ed elettronico i principi del sistema nervoso autonomo dell'uomo, in grado di interfacciarsi, controllare funzioni di vitale importanza e reagire agli stimoli esterni indipendentemente dalla consapevolezza individuale. L'aggettivo *autonomic* si riferisce infatti alla capacità insita nelle moderne risorse di calcolo distribuito di auto-controllarsi, adattandosi agli imprevedibili cambiamenti in maniera del tutto trasparente all'utilizzatore, a cui rimane nascosta la complessità algebrica e computazionale sottesa a questi meccanismi.

L'importanza dell'utente, nel contesto così delineato, consiste nell'andare a definire un insieme di regole o politiche di alto livello che consentono al sistema di compiere scelte adattative: controllando periodicamente le proprie variabili interne di stato, il sistema computazionale diviene una entità autonoma capace di compiere scelte sulla base di regole di ottimizzazione, tali da indurre la transizione ad una nuova condizione più idonea all'attuale

configurazione degli ingressi.

Gli elementi base di una struttura AC sono detti "componenti autonomi", ciascuno dei quali è costituito da due schemi di controllo, a cui si aggiungono sensori per l'auto-monitoraggio e attuatori per l'auto-correzione di stato. In altri termini, quello che ricercatori e industria hanno cercato di aggiungere ai precedenti strumenti di calcolo è stata la proprietà di consapevolezza delle architetture sviluppate, derivante dalla sinergia tra efficienti processi di controllo e intelligenti politiche decisionali.

IBM ha definito quattro principi fondamentali per l'effettiva costruzione di un sistema autonomo:

- Auto-configurazione: coincide con la configurazione automatica dei componenti interni in modo da adattarsi dinamicamente ai cambiamenti provenienti dall'ambiente
- Auto-guarigione: capacità di rilevare malfunzionamenti e agire in modo correttivo su una o più componenti al fine di ripristinare la condizione nominale.
- Auto-ottimizzazione: autonomia del sistema nelle fasi di controllo e monitoraggio automatico delle risorse per garantire la configurazione operativa ottima. Riallocazione dei compiti e controllo della priorità con cui le azioni vengono eseguite rientrano in questa caratteristica.
- Auto-protezione: abilità nel proteggersi da possibili attacchi.

In aggiunta, affinché un sistema rientri nella categoria AC è necessario possedere le otto seguenti importanti proprietà, da concretizzare in maniera assolutamente trasparente all'utilizzatore:

- Auto-consapevolezza: il sistema deve conoscere l'obiettivo della computazione opportunamente gestendo le risorse disponibili a fronte delle richieste di calcolo attuali, partendo dalla conoscenza del contesto in cui è collocato e dalla connessione con altri sistemi.
- Auto-apprendimento: mediante tecniche di machine learning, tipicamente non supervisionate, il sistema apprende dall'esercizio nuove procedure di ottimizzazione, perfezionando di volta in volta la propria performance in virtù dell'esperienza di calcolo precedente. Facendo uso di tecniche unsupervised si elimina così completamente la necessità di controllo esterno.

- Auto-regolamentazione: il sistema deve soddisfare specifici livelli di qualità o parametri di elaborazione, per cui le metriche di valutazione delle prestazioni non possono scendere al di sotto di una data soglia anche nei casi più sfortunati.
- Auto-organizzazione: l'organizzazione del sistema deve dipendere da modelli fisici, sociali e ambientali senza alcuna forzatura derivante dall'esterno.
- Auto-creazione: i componenti del sistema devono essere in grado di auto-guidarsi e auto-motivarsi, originando complessità e ordine secondo una risposta creativa idonea al tipo di domanda corrente.
- Auto-gestione: il sistema gestisce se stesso senza interventi esterni, tipicamente riferito ad un insieme di processi piuttosto che ad una singola applicazione.
- Auto-descrizione: il sistema riesce a spiegarsi e mostrarsi in una forma ed un linguaggio facilmente comprensibile dagli umani senza bisogno di traduzioni

La computazione diventa in questo senso pervasiva, al punto che la scienza del *pervasive computing* introdotta in [1] è ormai concretamente realtà. Le architetture di calcolo, infatti, sulla base di procedure più o meno aleatorie riescono ad anticipare, prevedere e mantenere la variabilità delle complesse interazioni tra le diverse parti del sistema, prevenendo l'amministratore dal gravoso compito di dedicarsi a task di basso livello.

Pur vivendo nell'era dell'Internet of Things (IoT), la presenza di macchine di calcolo, eventualmente invisibili o di forma alternativa, connesse in uno spazio comune ha ormai dato avvio alla più corretta era dell'Internet of Everything (IoE).

1.0.1 Caso di studio: Big Data & Cloud Computing

Fornire strategie performanti in questo senso significa inoltre rispondere all'esigenza attuale di trattare l'estrema complessità dei dati costantemente prodotti in misura coerente, affidabile e sicura. Il trend attuale, condizionato certamente da fenomeni sociali e commerciali, è quello di immettere nella rete una enorme quantità di informazioni, per le quali sono stati definiti sia nuovi principi di memorizzazione che tecniche di signal processing alternative. La

dimensione, il volume, la rapidità nella produzione di dati hanno inaugurato a tutti gli effetti il tema dei Big Data.

Le due problematiche connesse alle caratteristiche dei dati appena citate sono state attualmente risolte in modo efficiente sposando i principi di ridondanza, replica e distribuzione delle risorse hardware e software. Per la dimensione dei database collezionati, infatti, le procedure di storing hanno condotto alla creazione di approcci Not only SQL (NoSQL), in cui la struttura statica relazionale è stata sostituita con meccanismi dinamici di aggiornamento e riscrittura delle informazioni. Questa prerogativa viene raggiunta mediante vettori di dati costituiti da terne di tempo-nome-valore, dove cioè al tipo di informazione prodotta e al suo valore si associa anche l'istante temporale in cui viene originato.

Cluster e server distribuiti e gerarchicamente strutturati consentono prima di tutto di far fronte a possibili failures, estremamente pericolose nel caso in cui un solo punto di accesso all'informazione sia disponibile. Viene così garantita la proprietà di auto-guarigione di fronte ad anomalie. L'auto-regolamentazione è poi garantita dalla gestione intelligente delle risorse computazionali, raggiunta con schemi di programmazione parallela in cui lo scheduling dei thread avviene sulla base degli acceleratori attualmente liberi. In questo senso si inserisce anche l'idea di auto-consapevolezza del sistema, realizzata dalla conoscenza capillare delle singole unità operative dell'architettura e dei loro rapporti funzionali. Ovviamente, vincoli di ottimizzazione sono congeniti al funzionamento, in quanto la riduzione dei tempi di accesso in memoria, lettura e riscrittura devono essere opportunamente minimizzati indipendentemente dalle dimensioni dei registri.

Regole di auto-apprendimento, seppure in forma differente dalle istanze proprie del machine learning, sono indispensabili nell'ambito del controllo di flusso delle operazioni, poiché la presenza di istruzioni divergenti, nelle quali si hanno cioè possibili uscite sulla base di una condizione logicamente soddisfatta, portano ad esecuzioni differenti. L'obiettivo dei moderni sistemi autonomi in questo contesto è quello di definire delle politiche decisionali che, sulla base del set di istruzioni attualmente presenti, riesce ad eseguire una predizione non solo possibile ma anche corretta del subset di task proposto.

In maniera del tutto analoga, eventuali porzioni di codice errate vengono aggiornate correttamente grazie a funzioni di controllo nel trasferimento e nell'accesso organico ai dati.

L'autonomia dei moderni approcci ai dati propri dello scenario Big Data si riscontra quindi nelle nuove proposte di allocazione del calcolo e conse-

guente ripristino, gestione e prevenzione di potenziali danni predisponendo fondamentali meccanismi di backup.

1.1 Real Time Damage Detection in applicazioni SHM

Estendendo e rilassando in qualche misura il concetto di sistema autonomo sin qui delineato, che esula quindi dalla rigorosa definizione in termini informatici, si affronta qui l'idea di autonomia in applicazioni di monitoraggio strutturale. La scienza dello Structural Health Monitoring (SHM) ha come obiettivo quello di seguire lo stato di salute di una struttura sotto osservazione in maniera continuativa nel tempo, così da avvertire situazioni difformi dal normale regime operativo e diagnosticare l'origine del danno occorso.

L'esecuzione di questi step di calcolo si esplica non soltanto attraverso la stima di una serie di parametri caratteristici, ma deve essere svolta in maniera assolutamente autonoma proprio perché è spesso impossibile per l'operatore intervenire direttamente in loco, specialmente quando l'ambiente è ad alto rischio o limitato accesso.

La fase più delicata, che maggiormente deve soddisfare i requisiti in questo senso, è proprio quella di classificazione dell'irregolarità comparsa, raggiungibile attraverso approcci di machine learning o deep learning che sappiano introdurre una visione di ampio respiro sull'infinita possibilità di situazioni verificabili.

Nel contesto appena citato, infatti, dove la competenza della persona fisica e l'esperienza del settore non possono essere facilmente sostituiti da strumenti meccanici, la prospettiva delle ricerche sin qui condotte si è basata sulla volontà di fondere l'esperienza umana, emulata da una rete neurale di calcolo, con la vasta capacità di memorizzazione delle circuiterie, che superano quindi i limiti di visione e rappresentazione dell'operatore.

Le tecniche stesse che vengono definite devono evolvere verso scenari in cui la conoscenza aprioristica del sistema osservato deve tendere a zero, dal momento in cui questo implicherebbe la presenza necessaria del ricercatore per poter ottenere risultati coerenti e soddisfacenti. L'evoluzione a cui si sta assistendo, di migrare cioè l'intelligenza del sensing verso la dotazione strumentale, risponde a leggi simili a quelle delineate nell'ambito dei grandi numeri: alcuni approcci innovativi, basati semplicemente sulla definizione di un limitato set di regole e istruzioni da fornire come input agli algoritmi

implementati, ha consentito di raggiungere traguardi e sviluppi importanti nel settore, con guadagno in termini di costo, qualità, invasività e livello di informazioni prodotte. L'autonomia delle tecniche di signal processing si delinea come una delle prerogative fondamentali e di frontiera.

Bibliografia

- [1] Jeffrey O Kephart and David M Chess. The vision of autonomic computing. *Computer*, 33(1):41–50, 2003.