PROGETTAZIONE DI RETI NEURALI SPINTRONICHE

D. Rodrigues, A. Meo, E. Piccolo, A. Grimaldi, R. Tomasello, L. Carnimeo, S. Vergura, V. Puliafito, M. Carpentieri

Dipartimento di Ingegneria Elettrica e dell'Informazione, Politecnico di Bari, Bari

L'Internet of Things (IoT) sta promuovendo l'implementazione di reti neurali ai margini della rete (edge computing), riducendo la dipendenza dal cloud computing al fine di ottenere una latenza inferiore, una maggiore privacy e una reattività in tempo reale [1]. Questa transizione necessita di soluzioni hardware innovative che consentano un'implementazione compatta ed efficiente delle reti neurali su dispositivi edge. Un approccio promettente è l'utilizzo di tecnologie spintroniche, che offrono dispositivi scalabili ed efficienti dal punto di vista energetico, compatibili con i processi di produzione CMOS standard [2].

Tuttavia, persiste una sfida chiave: l'addestramento di tali sistemi è difficoltoso poiché non può essere eseguito direttamente a livello del dispositivo. Questa limitazione comporta elevati requisiti di memoria e impedisce una compensazione efficace delle variazioni da dispositivo a dispositivo.

Nel nostro lavoro [3], abbiamo sviluppato un dispositivo in grado di generare sia una funzione di attivazione non lineare standard sia il suo gradiente corrispondente - una caratteristica essenziale per gli algoritmi di apprendimento. Nello specifico, abbiamo utilizzato una giunzione magnetica ad effetto tunnel (MTJ), in cui l'orientamento relativo della magnetizzazione dello strato libero (FL) e dello strato di riferimento (RL) viene tradotto in un segnale di resistenza elettrica tramite la magnetoresistenza a effetto tunnel. Abbiamo codificato la funzione di attivazione tangente iperbolica (*tanh*) nella risposta del dispositivo. I nostri risultati mostrano che una componente della magnetizzazione dello strato libero produce la funzione *tanh*, mentre una componente perpendicolare fornisce simultaneamente il suo gradiente. Attraverso un'attenta progettazione del dispositivo, dimostriamo una soluzione a singolo dispositivo che fornisce in uscita sia la funzione di attivazione che il suo gradiente.

Questo metodo consente un addestramento efficiente e altamente parallelo delle reti neurali con ridotte richieste di memoria. La nostra implementazione algoritmica ha prodotto tre risultati chiave: (i) deviazioni minori tra le curve generate dalla MTJ e quelle esatte generate via software hanno un impatto trascurabile sulle prestazioni della retropropagazione (backpropagation); (ii) il dispositivo mostra una forte robustezza alle variazioni tra dispositivi e al rumore; (iii) il sistema proposto supporta efficacemente il trasferimento di apprendimento (transfer learning) e la distillazione della conoscenza (knowledge distillation).

Abbiamo emulato il comportamento del dispositivo utilizzando simulazioni micromagnetiche con il framework nativo Petaspin basato su CUDA. Sulla base di ciò, abbiamo implementato la risposta di attivazione e del gradiente in Python per simulazioni di reti neurali su larga scala utilizzando la libreria PyTorch. Per valutare le prestazioni, abbiamo applicato i pesi di un modello addestrato via software alla nostra rete di edge computing basata su MTJ. I risultati mostrano solo una perdita minima di accuratezza: 0.4% per il dataset Fashion-MNIST e 1.7% per CIFAR-100, rispetto all'implementazione software originale. Questi risultati evidenziano il potenziale del nostro design basato su MTJ per reti neurali compatte e integrate nell'hardware in applicazioni di edge computing, in particolare nel contesto del transfer learning.

[1] K. Cao, Y. Liu, G. Meng, and Q. Sun, An Overview on Edge Computing Research, IEEE Access **8**, 85714 (2020).
[2] G. Finocchio et al., Roadmap for unconventional computing with nanotechnology, Nano Futures **8**, 012001 (2024).
[3] D. Rodrigues, E. Raimondo, R. Tomasello, M. Carpentieri, and G. Finocchio, A design of magnetic tunnel junctions for the deployment of neuromorphic hardware for edge computing, Appl Phys Lett **126**, (2025).
[4] A. Giordano, G. Finocchio, L. Torres, M. Carpentieri, and B. Azzerboni, Semi-implicit integration scheme for Landau–Lifshitz–Gilbert-Slonczewski equation, J Appl Phys **111**, 07D112 (2012).