Reinforcement Learning Approach for Sub-Critical Current SOT-MRAM Switching

Johannes Ender[1,2], Roberto L. de Orio[2], Simone Fiorentini[1],
Siegfried Selberherr[2], Wolfgang Goes[3], and Viktor Sverdlov[1,2]
[1]Christian Doppler Laboratory for Nonvolatile Magnetoresistive Memory and Logic
at the [2]Institute for Microelectronics, TU Wien, Gußhausstraße 27-29, A-1040 Wien, Austria
[3] Silvaco Europe Ltd., Cambridge, United Kingdom
Email: ender@iue.tuwien.ac.at

Abstract – We present the use of reinforcement learning for the discovery of pulse sequences for optimal switching of spin-orbit torque magnetoresistive memory devices. A neural network trained on fixed material parameters is able to switch a memory cell for a wide range of material parameter variations as well as for sub-critical current values. Micromagnetic simulations are used to prove the reliability of the trained neural network.

Keywords – ***Reinforcement learning, SOT-MRAM, field-free switching***

I. Introduction

The increasing power consumption of semiconductor memory devices due to the progressive down-scaling and the entailing higher leakages is a pressing issue which could be solved with magnetoresistive random access memories (MRAM). Spin-orbit torque (SOT) MRAM is a highly promising contender for replacing existing charge-based random access memories due to its nonvolatility, high operation speed and large endurance. The need for an external magnetic field for deterministic switching [1] is eliminated in a recently introduced purely electrically controllable SOT-MRAM cell [2].

For the development of MRAM devices, accurate simulation tools are of paramount importance. They benefit from the increasing computational power of simulation hardware, but nevertheless the analysis of the equally increasing amounts of simulation results is challenging and employing advanced machine learning algorithms to manage and optimize the data becomes attractive. Machine learning assisted scientific research has become more widespread and has achieved impressive results [3]. We show how the machine learning sub-field of reinforcement learning (RL) [4] can be used in order to discover optimal pulse sequences for deterministically switching the SOT-MRAM cell proposed in [2].

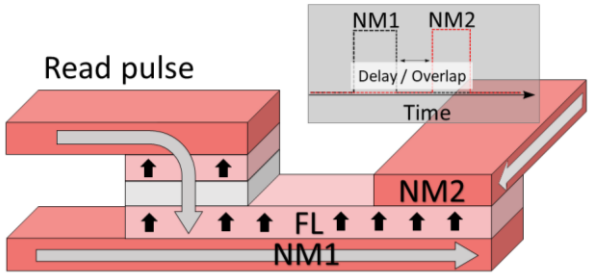

Figure 1: SOT-MRAM cell for switching based on two orthogonal current pulses. The pulses are sent through the structure via two nonmagnetic heavy metal wires, one of which is fully overlapping the FL (NM1) and one only partially (NM2).

II. Spin-Orbit Torque Memory

Besides spin-transfer torque MRAM, SOT-MRAM is the most promising type of magnetoresistive memory. The writing operation is performed by sending a charge current through a metal wire with a large spin Hall angle, which is attached to the free layer. This creates a transverse spin current, exerting a torque on the free layer magnetization and initiating a precessional motion, eventually leading to switching. Perpendicularly magnetized SOT-MRAM devices, however, require in addition a magnetic field to deterministically reverse the magnetization [1]. Among several proposed field-free schemes, e.g. [5], a memory cell introduced in [2] solves the problem by adding a second metal wire, orthogonal to the first one (cf. Fig. 1). It was shown that by sending current pulses through the two metal wires NM1 and NM2, the memory cell can be switched deterministically and purely electrically. Although the write path in SOT-MRAM is separated from the read path and thus oxide reliability issues are not a problem, a reduction of the write current is still desirable to reduce the stress on the surrounding circuitry [6].

The dynamics of the magnetization in the free layer of SOT-MRAM cells can be simulated by solving the following extended Landau-Lifshitz-Gilbert equation.

$$\begin{aligned}\frac{\partial \mathbf{m}}{\partial t} = &-\gamma\mu_0 \mathbf{m}\times\mathbf{H}_{\mathbf{eff}} + \alpha\mathbf{m}\times\frac{\partial \mathbf{m}}{\partial t} \\ &-\gamma\frac{\hbar}{2e}\frac{\theta_{SH} j_1}{M_{Sd}}[\mathbf{m}\times(\mathbf{m}\times\mathbf{y})]f_1(t) \\ &+\gamma\frac{\hbar}{2e}\frac{\theta_{SH} j_2}{M_{Sd}}[\mathbf{m}\times(\mathbf{m}\times\mathbf{x})]f_2(t)\end{aligned} \quad (1)$$

978-1-6654-0685-7/21/$31.00 ©2021 IEEE

m is the normalized magnetization, γ is the gyromagnetic ratio, μ_0 is the vacuum permeability, α is the Gilbert damping factor, and M_S is the saturation magnetization. The effective field $\mathbf{H_{eff}}$ consists of several contributions, namely the exchange field, the uniaxial perpendicular anisotropy field, the demagnetizing field, the current-induced field, and a stochastic thermal field at 300 K. f_1 and f_2 are functions defining when the NM1 pulse and the NM2 pulse are active. The decision of when and how long to apply current pulses is primarily based on intuition so far and an automatic approach for the discovery of optimal current pulse sequences would be important.

III. Reinforcement Learning

Reinforcement learning algorithms can be used to solve sequential decision-making problems for achieving a certain goal or objective. The two main entities in such scenarios in the context of RL are the agent and the environment. Over the length of one learning episode which is defined by the environment, the agent interacts with the environment by performing actions and gathering observations. Each performed action changes the state of the environment and the observations made by the agent are returned as signals from the environment consisting of information about the current state of the environment and a reward. The reward signal tells the agent how good (or bad) the previously performed action was for achieving the objective. Usually, many such episodes have to be performed, in which the agent refines its policy π, a mapping from states to actions, in such a way, that the received reward over the course of an episode is maximized. The basis for decision-making in value-based reinforcement learning algorithms is the action-value function [2].

$$Q_\pi(s,a) = \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t R_t \mid S_t = s, A_t = a\right] \qquad (2)$$

It assigns a value to state-action pairs, expressed as the expectation value of the cumulated and discounted reward R_t, given that the current state is s and the performed action is a, following the current policy π. The discount factor γ is a parameter controlling the influence of rewards received later in an episode on the estimate of the action-value function at the current time step. Having an estimate for the action-value function, the policy tells the agent, which action is best to take in a certain state. During training, greedily always taking the action promising the highest reward, i.e. the action with the highest value of the action-value function, can lead to an agent getting stuck in a local minimum, and a trade-off between exploitation of existing knowledge and exploration of new strategies has to be made. In ε-greedy policies, this trade-off is controlled by the exploration probability ε. Here, the greedy action is taken with a probability of ε, and a random action is taken with probability 1-ε.

TABLE II. DQN Parameters

Parameter	Value
Neural network size	11 × 150 × 100 × 4
Discount factor, γ	0.9997
Learning rate	7.5×10^{-4}
Exploration fraction	0.2
Final exploration probability, ε	0.01
Replay buffer size	3×10^5
Batch size	512

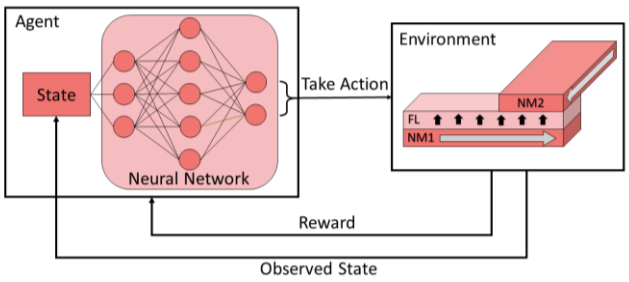

Figure 2: General setup of the reinforcement learning procedure: The simulation of the SOT-MRAM cell acts as environment which an agent interacts with, to build up a policy based on a neural network.

IV. RL Applied to SOT Switching

For applying RL in order to learn how to best apply pulses to switch the pulsed SOT memory cell as fast as possible, we employed the deep Q-network (DQN) algorithm [7]. This algorithm uses a neural network to approximate the action-value function. Based on an RL Python library [8], which provides an

TABLE I. SOT Cell Simulation Parameters

Parameter	Value
Saturation magnetization, M_S	1.1×10^6 A/m
Perpendicular anisotropy, K	8.4×10^5 J/m^3
Exchange constant, A	1.0×10^{11} J/m
Gilbert damping factor, α	0.035
Spin Hall angle, θ_{SH}	0.3
Free layer dimensions	40 nm × 20 nm × 1.2 nm
NM1: w1 × l	20 nm × 3 nm
NM2: w2 × l	20nm × 3 nm

implementation of the DQN algorithm and allows to easily couple custom environments, the approach depicted in Fig. 2 was implemented. Apart from the parameters given in Table I, the DQN agent was used with the default settings. For the environment, an in-house developed finite difference simulator [9] was coupled to the RL library and was adapted to allow the exchange of action, reward, and state signals. The training of the agent was performed with the fixed material parameters given in Table II. The agent was allowed to perform 4 distinct actions, namely turning both currents on, turning both pulses off and turning one pulse on and the other one off and vice versa. To prevent the agent from turning the pulses on and off arbitrarily fast, a minimum pulse width of 100 ps was enforced. Results published in [10] determined the critical current for the given memory cell to be 120 µA. However, it was also shown that the current value for the NM2 wire could be reduced, while maintaining deterministic switching. Thus, the current values for NM1 and NM2 were chosen as 130 µA and 100 µA, respectively.

The state signal returned to the agent every time step consists of 11 variables: The average vector components of the magnetization (m_x, m_y, m_z), the difference of the average vector components of the magnetization to the previous iteration $(\Delta m_x, \Delta m_y, \Delta m_z)$, the average vector components of the effective magnetic field $(H_{eff,x}, H_{eff,y}, H_{eff,z})$, and two variables which indicate whether the currents on the NM1 and NM2 wire can currently be turned on.

The most important component besides the state vector is the rewarding scheme which gives the agent feedback about the goodness or badness of the actions it has taken and thus encodes

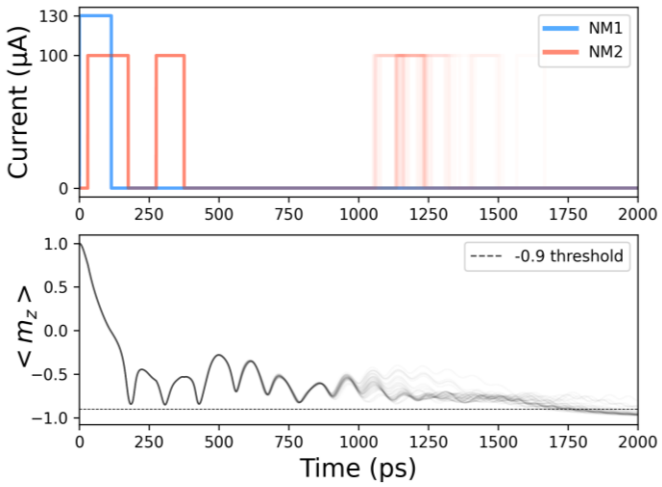

Figure 3: Results of 50 realizations for fixed material parameters and I_{NM1} = 130 µA and I_{NM2} = 100 µA using the trained neural network model. Results of the single runs are plotted slightly transparent, such that regions where multiple lines overlap appear more solid.

the objective. The reward function we employed is defined as follows.

$$r = m_{z,target} - m_{z,current} \tag{3}$$

With an $m_{z,target}$ of -1, the given function always produces a negative reward which is more negative, the farther away the current value of the average z-component of the magnetization is from the target value. This not only incentivizes the agent to reverse the z-component of the magnetization from +1 to -1, but also to do it fast, as the longer it takes, the more negative reward is accumulated.

V. Results

After training, the weights of the neural network are not adjusted any further and the agent-environment setup as shown in Fig. 2 can be used to perform switching simulations in which the agent dynamically decides when to apply pulses.

As there is a thermal field contribution to the effective magnetic field, we performed 50 realizations with the parameters given in Table II in which the agent tried to reverse the magnetization. The results are shown in Fig. 3. Due to the slight transparency of the single plot lines, one can see where multiple trajectories overlap, as those regions appear more solid. For the NM1 pulse, all the lines overlap exactly, which means that in all the realizations, the NM1 pulse was applied once in the beginning of the realization. The first two applied pulses on the NM2 wire were also applied in all realizations. Up until 1 ns, the magnetization trajectories match very closely, but then they start to drift apart. Between 1 ns and 1.5 ns, further NM2 pulses are applied in different realizations. The exact positions of these pulses vary, depending on the trajectory of the magnetization. The threshold of -0.9, at which we consider the memory cell to be switched, is reached deterministically in all realizations.

To determine how reliable the neural network can switch the memory cell under variations of material parameters, while having been trained on the fixed parameters given in Table II, further experiments were performed, in which the saturation magnetization M_S and the anisotropy constant K were varied individually by up to ±5%. Fig. 4 shows the results of these experiments, where the color-coding corresponds to how much overall reward the agent was able to accumulate. This total reward is higher, the closer the z-component of the magnetization is brought to the target value of -1. It is also higher if this transition is faster, as less negative reward is gathered. The blue-colored dots indicate a good overall performance. However, towards the top left corner, the performance of the switching realizations decreases. This can be

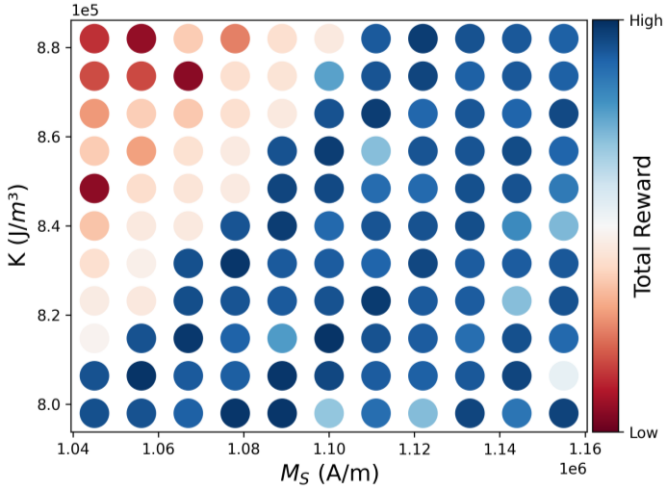

Figure 4: Accumulated reward achieved for an anisotropy constant K and a saturation magnetization M_S varied by ±5% with I_{NM1} = 130 µA and I_{NM2} = 100 µA. Results are shown for a total of 121 parameter combinations.

explained with the fact, that these combinations of values for the anisotropy constant and the saturation magnetization require higher current values to be switched, because the critical current is higher, as was shown in [10]. Out of the 121 performed switching simulations with varying parameters, ~75% were brought below the threshold of -0.9.

As presented in [10], the critical current for the used memory cell with the material parameters given in Table II is 120 µA. To evaluate the possibility of switching with a reduced current, the current value of the NM1 wire was lowered to a sub-critical value of 110 µA. Fig. 5 shows results of 50 switching realizations with the material parameters of the memory cell set to the ones used for training (Table II). Again, it can be observed

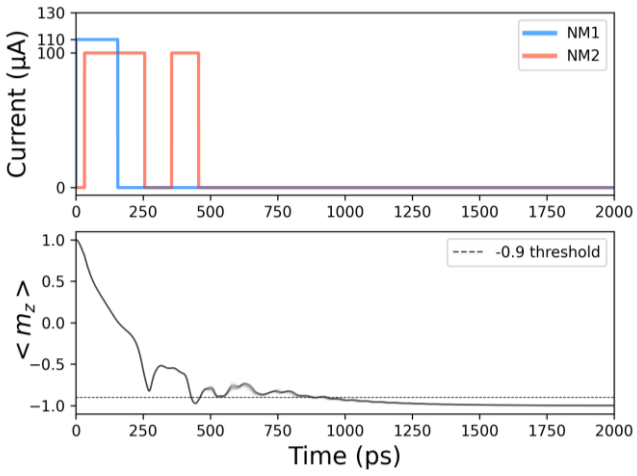

Figure 5: Results of 50 realizations for fixed material parameters and I_{NM1} = 110 µA and I_{NM2} = 100 µA using the trained neural network model. Results of the single runs are plotted slightly transparent, such that regions where multiple lines overlap appear more solid.

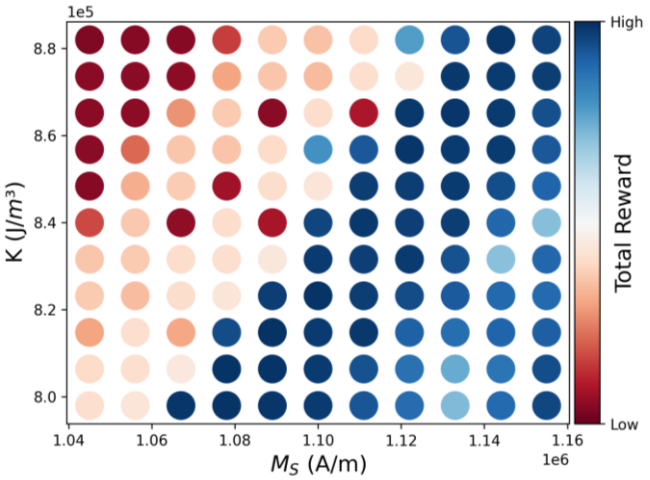

Figure 6: Accumulated reward achieved for an anisotropy constant K and a saturation magnetization M_S varied by ±5 % with I_{NM1} = 110 µA and I_{NM2} = 100 µA. Results are shown for a total of 121 parameter combinations.

that a single pulse on the NM1 wire and two pulses on the NM2 wire are applied in the beginning. These were applied in all realizations. The reduction of the NM1 current value, however, also reduces the variation in the trajectories of the z-component of the magnetization. The magnetization is oscillating less and settling faster towards the target value of -1. The crossing of the -0.9 threshold happens approximately 800 ps earlier than with the higher NM1 current. Again, performing simulations with varying the saturation magnetization as well as the anisotropy constant by up to ±5%, the results presented in Fig. 6 were achieved. The line separating the better performing realizations from the worse performing ones has shifted towards the center of the plot and the performance of the realizations in the top left corner decreased. Nevertheless, in 59% of the performed realizations, good performance could still be achieved, and the magnetization could successfully be reversed.

VI. Conclusion

We presented an RL approach, in which an RL agent autonomously learns how to reverse the magnetization in an SOT-MRAM cell. The agent dynamically applies pulses to achieve fast and deterministic switching. Even though being trained on a fixed material parameter set, the agent performs well under variation of these parameters. A current reduction to a sub-critical value shows even better switching performance for the fixed parameter case, but also achieves good results under variation of material parameters.

Acknowledgment

The financial support by the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development and the Christian Doppler Research Association is gratefully acknowledged.

References

[1] S. Fukami, T. Anekawa, C. Zhang, and H. Ohno, "A spin-orbit torque switching scheme with collinear magnetic easy axis and current configuration," *Nat. Nanotechnol.*, vol. 11, pp. 621–626, Mar. 2016. DOI: 10.1038/nnano.2016.29

[2] V. Sverdlov, A. Makarov, and S. Selberherr, "Two-pulse sub-ns switching scheme for advanced spin-orbit torque MRAM," *Solid-State Electron.*, vol. 155, pp. 49–56, Mar. 2019. DOI: 10.1016/j.sse.2019.03.010

[3] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N.Tishby, *et al.*, "Machine learning and the physical sciences," *Rev. Mod. Phys.*, vol. 91, no. 4, p. 045002, Dec. 2019. DOI: 10.1103/RevModPhys.91.045002

[4] R. S. Sutton, and A. G. Barto, "Reinforcement learning: An introduction," Cambridge, MA, USA: MIT press, 1998.

[5] S. Fukami, C. Zhang, S. DuttaGupta, A. Kurenkov, and H. Ohno, "Magnetization switching by spin-orbit torque in an antiferromagnet-ferromagnet bilayer system," *Nat. Mater.,* vol. 15, pp. 535–541, Feb. 2016. DOI: 10.1038/nmat4566

[6] K. Garello, F. Yasin, S. Couet, L. Souriau, J. Swerts, S. Rao, *et al.*, "SOT-MRAM 300mm integration for low power and ultrafast embedded memories," in *Proc. IEEE Symp. VLSI Circuits*, pp. 81–82, Oct. 2018. DOI: 10.1109/VLSIC.2018.8502269

[7] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529-533, Feb. 2015. DOI: 10.1038/nature14236

[8] A. Raffin, A. Hill, M. Ernestus, A. Gleave, A. Kanervisto, N. Dormann. Stable Baselines 3. Available online: https://github.com/DLR-RM/stable-baselines3 (accessed on 20 April 2021).

[9] A. Makarov, "Modeling of emerging resistive switching based memory cells," Ph.D. Thesis, Institute for Microelectronics, TU Wien, Vienna, 2014. DOI: 10.13140/RG.2.2.11456.74242

[10] R. L. de Orio, J. Ender, S. Fiorentini, W. Goes, S. Selberherr, and V. Sverdlov, "Numerical analysis of deterministic switching of a perpendicularly magnetized spin-orbit torque memory cell," *IEEE J. Electron Devices Soc.*, vol. 9, pp. 61–67, Nov. 2020. DOI: 10.1109/JEDS.2020.3039544