Simultaneous control of multiple 0D parameters by using deep reinforcement learning in KSTAR tokamak

Jaemin Seo[1], Y.-S. Na[1], B. Kim[1], C.Y. Lee[1], M.S. Park[1] and Y.H. Lee[2]

[1]*Department of Nuclear Engineering, Seoul National University, Seoul, South Korea*

[2]*Korea Institute of Fusion Energy, Daejeon, South Korea*

It needs to operate tokamak plasmas within the window of interest for detailed physics studies. The operation scenarios are frequently characterized by the specific window of several 0D global parameters such as β_N (or β_p), q_{95}, q_0 or l_i [1,2]. Those 0D parameters represent the plasma pressure, the magnetic field structure and the current profile which can provide the information of the confinement and stability of the fusion plasma. Attempts to control the global or local plasma parameters have been widely conducted by using PID algorithm in the experiment [3-5] and in the real-time predictive modelling [6]. The performance control by stabilizing the MHD instabilities such as NTMs have been also developed [7,8]. However, controlling multiple parameters simultaneously with machine learning has not been carried out much [9,10]. In this work, we introduce an algorithm to control multiple 0D parameters (β_N, q_{95}, and l_i) simultaneously into the arbitrary target regime using deep reinforcement learning (RL) technique [11]. First, we construct a data-driven simulator as a virtual KSTAR tokamak environment. Then, we trained the RL-based controller that manipulates the external actuator variables to match the 0D plasma parameters with the given target regime.

In tokamak plasmas, global plasma responses are determined by combination of internal physical phenomena such as turbulent transport, heating & current drive, and MHD stability. These physical properties can be predicted by self-consistent simulations with a integrated modelling suite [12]. It provides multi-dimensional kinetic profiles which are crucial for detailed physics studies, but it is computationally heavy to be used for tons of virtual experiments required for RL agent training. Therefore, in this work, an LSTM-based [13] data-driven simulator shown in Figure 1 that imitates the KSTAR plasma responses is used for the virtual experimental environment instead, which is introduced and validated in the precedent research [10].

Figure 1. LSTM-based data-driven simulator as a virtual tokamak environment introduced in [10].

In the predict-first tokamak scenario development [14], the predictive simulation is conducted with manually seeking the optimal operation control to achieve the target performance with lots of trials and errors. However, it is hard to find the multi-dimensional control solution to match multiple 0D parameters with arbitrary targets simultaneously. Therefore, we introduced an RL artificial decision-making agent, the TD3 algorithm [15], which is suitable for multi-input and multi-output control. The RL agent observes the current plasma state (β_N, q_{95}, and l_i) and sets the target state (β_{N_target}, q_{95_target}, and l_{i_target}), then it determines the action composed of actuator controls to match the consequent plasma state into the target regime. During the RL agent's training in the virtual tokamak shown in Figure 1, it receives the reward according to how close the consequent plasma state is to the given target each episode, so it is gradually trained to provide optimal decision-making that yields the highest reward at given states.

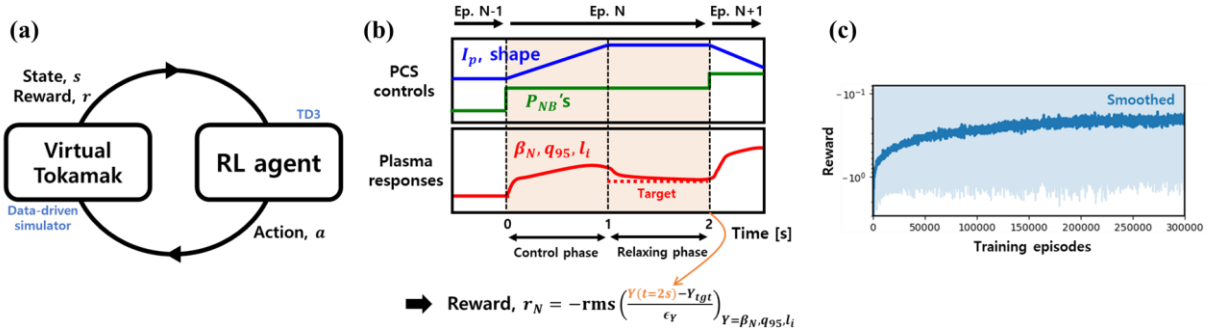

Figure 2. (a) Schematic view of the RL training in the virtual tokamak. (b) Modelling of the actuator controls and reward evaluation in each training episode. (c) The received reward versus training episode.

In each training episode, the RL agent delivers the actuator controls; plasma current (I_p), NB powers ($P_{NB\text{-}A,B,C}$), and plasma shape (R_{geo}, a_{min}, κ, δ_u and δ_l). Then, in the simulator, the actuator variables evolve to the RL-determined values for 1 s, while the NB powers change immediately as shown in Figure 2 (b). After 1 s of relaxation from the end of the control phase, the reward is evaluated at the final state. The final state is then loaded as the initial condition of the next episode and the target values are updated randomly. Here, other operation conditions such as toroidal field and Greenwald density fraction are fixed with those in shot #26719 in KSTAR. The detailed descriptions of the state, action, and reward for the RL training are presented below.

$$State = (\beta_N, q_{95}, l_i, \beta_{N\_target}, q_{95\_target}, l_{i\_target}) \quad (1)$$

$$Action = (I_p, P_{NB\text{-}A}, P_{NB\text{-}B}, P_{NB\text{-}C}, R_{geo}, a_{min}, \kappa, \delta_u, \delta_l) \quad (2)$$

$$Reward = -\mathrm{rms}[(y-y_{target})/\epsilon_y]_{\,y=\beta N,q95,li} \ \text{ with } \epsilon_y = 0.1, 0.4, 0.04 \text{ for } y = \beta_N, q_{95}, l_i \quad (3)$$

After a few 10^5 of training episodes, the average reward the RL agent receives becomes saturated as shown in Figure 2 (c) and it provides nearly optimal decision-making for each

given target regime. Then, we tried to apply it to find the optimal actuator control for several operation regimes in KSTAR. We set three target regimes; (i) normal H-mode ($\beta_N = 2.0$, $q_{95} = 5.0$, $l_i = 0.95$), (ii) hybrid regime (2.5, 4.0, 0.85) and (iii) high-l_i regime (2.5, 6.0, 1.05).

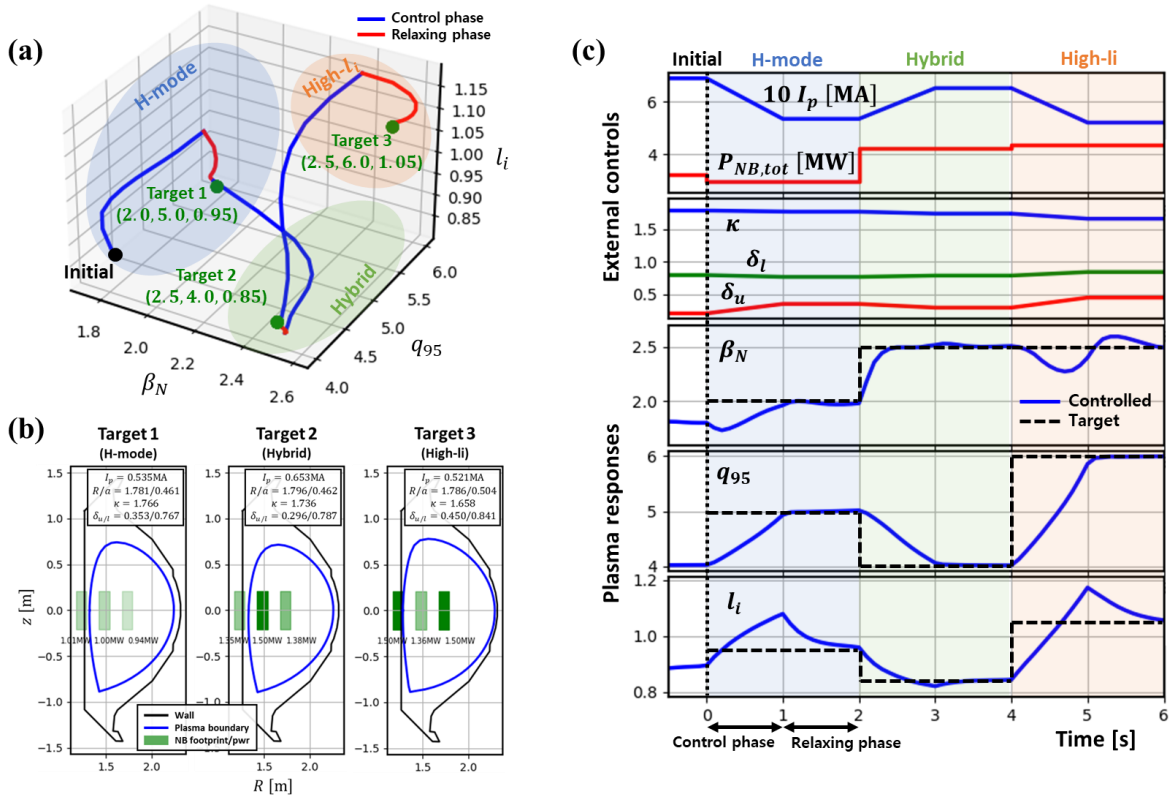

Figure 3. (a) 3D trajectory of plasma responses in parametric space (β_N, q_{95}, and l_i) by RL controls for three targets. (b) The RL control solutions (I_p, $P_{NB\text{-}A,B,C}$, and shape) for three different target regimes. (c) Time-evolution of the control variables and the consequent plasma responses.

Figure 3 shows the results of the RL controls for achievement of three different target regimes and consequent plasma responses in the virtual KSTAR environment. The RL agent determines different I_p, combination of NBs, and plasma boundary shape to achieve each target regime as shown in Figure 3 (b). It is noteworthy that since the heating & current drive profiles differ according to the combination of NBs due to their different geometries, and the plasma current and boundary shape also strongly affect the plasma, it is hard to manually figure out the optimal controls from the nonlinearly entangled physical interactions that determine the plasma states. However, we can see that the RL agent could control the multiple plasma state parameters into the given target values simultaneously by manipulating nine control knobs in Equation (2).

Interestingly, when we set hybrid regime targets (Target 2 in Figure 3), the RL agent determines relatively higher off-axis tangential beam power which yields higher NBCD efficiency. It is related to the broadening of the current density profile that induces hybrid-like q profile at the favourable plasma current and shape condition. Even though the target β_N's are

identical in Target 2 and 3, the plasma current, NBs' combination, and boundary shape are different. In the high-l_i target case (Target 3), it determines a lower off-axis tangential beam power contrary to the hybrid regime so to have higher l_i with a more peaked current profile, while the total NB power in both cases is almost the same each other. Furthermore, it tries "fat" shape of the plasma boundary with a large minor radius, low elongation, and high triangularity for high-l_i as shown in Figure 3 (b). The reason is under analysis but the similar l_i dependencies on a_{min} and κ were reported in [16,17].

It is notable that the plasma responses by RL control are not converging directly into the target, rather showing delayed response in the relaxing phase in Figure 3 (a) and (c). Especially l_i evolves slowly in the resistive time-scale of ~1 s and β_N is also affected by l_i. The RL control solution reflects the time-dependent delayed response and it makes the plasma states detour to reach the target point in the 3D trajectory, which the simple PID control can hardly perform. The RL-based control algorithm introduced in this work can be the first step for developing the autonomous operation of future tokamak reactor.

Acknowledgement

This research was supported by National R&D Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science & ICT (NRF-2019M1A7A1A03089798).

References

[1] Na Yong-Su *Exploration of advanced tokamak operation in KSTAR*. No. NIFS-PROC-100. 2016.

[2] Na Yong-Su, et al. *Nuclear Fusion* 60.8 (2020): 086006.

[3] Allen S. L., and DIII-D Team. *Nuclear fusion* 41.10 (2001): 1341.

[4] Joffrin E., et al. *Plasma physics and controlled fusion* 45.12A (2003): A367.

[5] Han Hyunsun, et al. *Fusion Engineering and Design* 95 (2015): 44.

[6] Kim H-S., et al. *Fusion Engineering and Design* 135 (2018): 1.

[7] Kim M., et al. *Nuclear Fusion* 55.2 (2015): 023006.

[8] Park Minho, et al. *Nuclear Fusion* 58.1 (2017): 016042.

[9] Felici Federico, and Tom Oomen. *54th IEEE Conference on Decision and Control (CDC)*. (2015): 5370.

[10] Seo Jaemin, et al. *Nuclear Fusion* (submitted)

[11] Mnih Volodymyr, et al. *Nature* 518.7540 (2015): 529.

[12] Lee Chanyoung, et al. *28th IAEA Fusion Energy Conf.* (2020): TH/P8

[13] Hochreiter Sepp, and Jürgen Schmidhuber. *Neural computation* 9.8 (1997): 1735.

[14] Staebler Gary, et al. *28th IAEA Fusion Energy Conf.* (2020): TH/OV2

[15] Fujimoto Scott, Herke Hoof, and David Meger. *International Conference on Machine Learning.* (2018): 1587.

[16] Elahi Ahmad Salar, and Mahmood Ghoranneviss. *Journal of Nuclear and Particle Physics* 2.4 (2012): 91.

[17] Xue Erbing, et al. *Journal of Fusion Energy* 38.2 (2019): 244.