Using efference copy for external and self-generated sensory noise cancellation

Poramate Manoonpong and Florentin Wörgötter

Bernstein Center for Computational Neuroscience, University of Göttingen, Göttingen, Germany

In the early 1950s, von Holst [1] proposed that motor commands copied (efference copy) within the central nerve system (CNS) help to distinguish 'reafference' (afference activity due to self-generated motion) from 'exafference' (afference activity due to external stimulus). An efference copy can be also used to compare it with the actual sensory feedback in order to subtract the self-generated sensation.

Based on these biological findings, we apply such principles to eliminate unwanted disturbances and cancel self-generated noise of a biped robot [2]. RunBot, as it is called, is a planar dynamic walking robot driven by its proprioceptors, like ground contact sensors (G), angle sensors (S), and stretch sensors (A), through reflexive neural control (see Fig. 1A). In addition, it has an infrared eye (IR) and a vestibular-like sensor (AS) allowing it to detect a slope and to learn to adapt its gait and upper body component (UBC) posture in order to walk up the slope (see [2] for more details of learning process). However, because of simulated disturbances on its walking path (see Fig. 1B, white spots) the IR sensor gives unwanted noise (see Fig. 2A, yellow areas). In addition, RunBot's egomotion causes the AS sensor to produce self-generated sensory events (see Fig. 2B, green areas). These perturbations will destabilize the activation parameters for the gait.

To filter the unwanted noise of the IR and AS signals, we copy motor signals (N, efference copy, see Figs. 1A and C) of the leg joints and transform them into a noise expectation through neural forward models (see Fig. 1A). These forward models are manually constructed by utilizing the discrete-time dynamical properties of recurrent neural networks (e.g., hysteresis effect) [3]. The neural forward model of the IR signal consists of a series of 15 recurrent neurons while the neural forward model of the AS signal has five recurrent neurons. The outputs of these forward models (called expected sensory noise signals) are fed into compensator units to subtract the unwanted noise of the actual IR and AS sensory feedback. Each compensator unit is modeled as a standard additive neuron [3]. As a result, the compensated signals of the IR and AS sensors (see Figs. 2C and D) can be applied to a learning circuit which lets RunBot learn to stably adapt its locomotion to different terrain, e.g. level floor versus up a ramp.

References

[1] Das Reafferenzprinzip (Wechselwirkung zwischen Zentralnervensystem und Peripherie). E.v. Holst and H. Mittelstaedt, *Naturwissenschaften* 37:464–476, 1950.

[2] Adaptive, Fast Walking in a Biped Robot under Neuronal Control and Learning. P. Manoonpong, T. Geng, T. Kulvicius, P. Bernd, F. Wörgötter. *PLoS Computational Biology* 3(7):e134, 2007.

[3] Neural Preprocessing and Control of Reactive Walking Machines: Towards Versatile Artificial Perception-Action Systems. P. Manoonpong, *Cognitive Technologies*, Springer-Verlag, 2007.

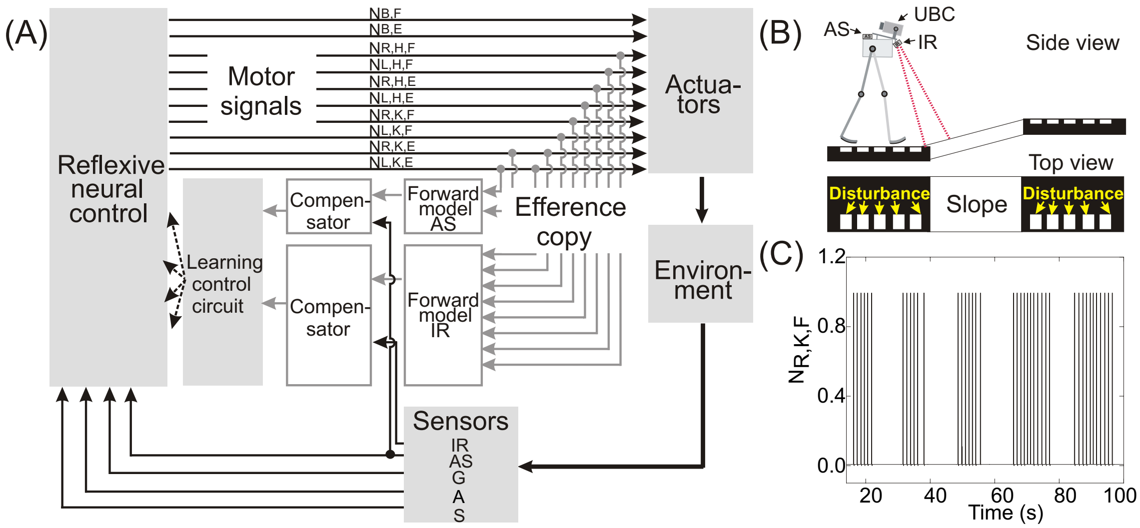

Figure 1: (A) Adaptive reflex neural control with forward models (see [2] for more details). Indexing of variables in this figure follows this structure: body-level (UBC = B, left-leg = L, right-leg = R); leg level (hip = H, knee = K); joint level (flexor = F, extensor = E). For example, $N_{B,F}$ applies to the flexor motor signal of the UBC while $N_{R,H,F}$ would apply to the flexor hip-motor signal of the right leg. (B) RunBot's walking path where white spots show the disturbance to the IR sensor. (C) The flexor knee-motor signal of the right ($N_{R,K,F}$); "efference copy". Note that other motor signals having similar patterns are not shown.

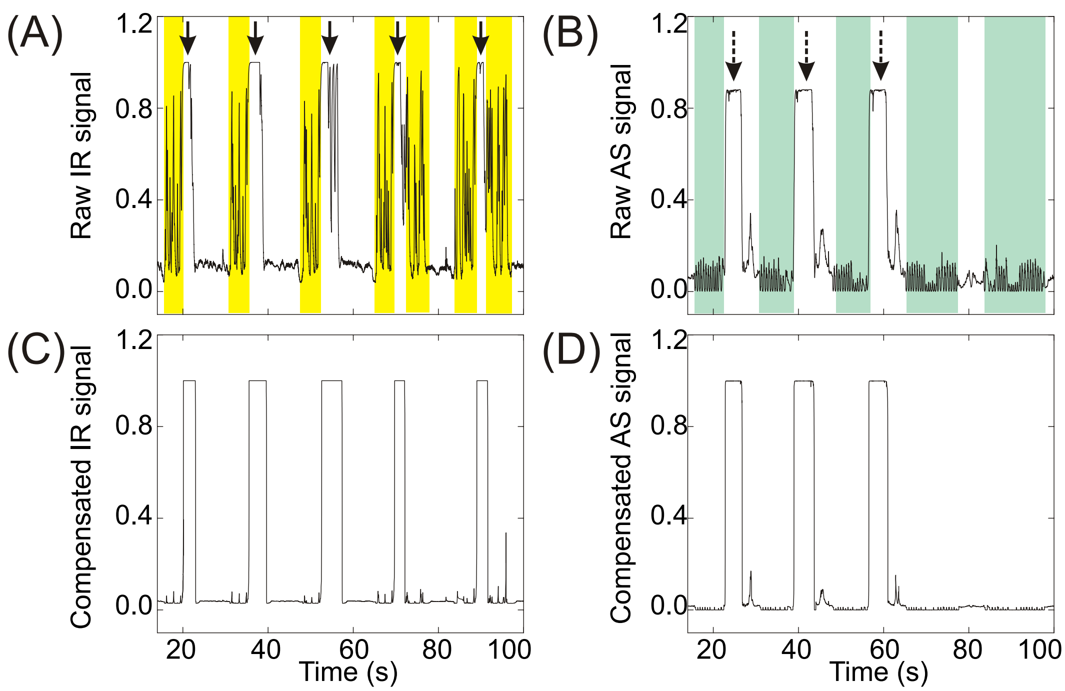

Figure 2: (A) and (B) Raw sensor signals. (C) and (D) Corrected sensor signals showing a clear improvement. Note that solid arrows depict the situation where RunBot detects a slope and dashed arrows where RunBot falls backward. It falls over backwards, as it has not yet learned to react to its IR input with a change in gait. RunBot learns from its failures, leading to a strengthening of the contact between the IR eye and the sites of movement control. As a result, after learning it can change its gait as soon as the IR input gives high activation (detecting the slope) and it will returns to its normal gait when it leaves the slope. In this experiment, RunBot can manage to walk on an eight-degree ramp after three falls.