Summary of replies to the editor and reviewer comments

Tao Sun, Zhendong Dai, Poramate Manoonpong

RC: ***Reviewer Comment***, **AR:** ***Author Response***

We would like to thank the editor and reviewers for their thorough review and comments, as well as for their substantial praise of the work. We provide our replies to editor's and all reviewers' comments. The editor's and reviewers' comments are shown in blue. Our answers are shown in black. The modified and added content in the revised manuscript is shown in red.

1. Reviewer #3

RC: I have read the manuscript. Authors have revised the manuscript as per suggestion. So, this manuscript is suitable for publication in this journal.

AR: We greatly appreciate your careful review and your agreement to the publication of this manuscript.

2. Reviewer #4

RC: The revised version gives much information on developing the control system of the robot. However, information about how to modify, or not to modify, the initial values of the controller parameters during the training or testing process is not given. That is, how to initialize all the values of the network parameters every time before starting to run the robot?

AR: We have now summarized all the control parameters of the controller as shown in the following table (Table S.2). The parameters belong to the four basic modules (the CPGs, MNs, DFFB, and DIL) of the controller. Their initial values and short descriptions are listed in the table. Most of these parameters are constant and remain unchanged after initialization. They are initialized every time before starting to run the robot. Only three parameters (e.g., $\beta(n)$, $w_1(n)$, and $w_2(n)$) are changed overtime. They are adapted based on the DFFB and DIL modules when the controller is running.

To make it easy to understand these parameters, we have now added the table in the supplementary material and mentioned this in line 336 in the revised manuscript as:

"The control parameters of the adaptive quadruped motor controller are listed in Table S.2 of the Supplementary material."

Table S.2. Parameters of the adaptive quadruped motor controller.

Modules	Symbols	Initial values	Description	Adaptive /constant
CPGs	$\mathbf{w}$	$\begin{pmatrix} 1.4 & 2.6 \\ -2.6 & 1.4 \end{pmatrix}$	Synaptic weights of the SO(2) CPG neurons	Constant
	$\mathbf{b}$	$\begin{pmatrix} 0.01 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.01 \end{pmatrix}$	Biases of the SO(2) CPG neurons	Constant
	$\mathbf{\Phi}$	$\begin{pmatrix} 0.0 & -\pi & -\pi & 0.0 \\ \pi & 0.0 & 0.0 & \pi \\ \pi & 0.0 & 0.0 & \pi \\ 0.0 & -\pi & -\pi & 0.0 \end{pmatrix}$	Desired relative phases among the four CPGs	Constant
	ϵ	0.01	CPG communication gain	Constant
MNs	α	0.16 for Lilibot and 0.12 for Laikago	Synaptic weight projection from the CPGs to MNs	Constant
	$\beta(n)$	0	Online-modulated by the DFFB	**Adaptive**
DFRL	$w_1(n)$	0.003	Online-modulated by the DIL	**Adaptive**
	$w_2(n)$	0.0032	Online-modulated by the DIL	**Adaptive**
	w_3	2	DFFB output gain of the knee joint	Constant
	w_4	1	DFFB output gain of the hip joint	Constant
DIL	A_f	0.01	Retention rate of the fast learner	Constant
	B_f	0.05	Learning rate of the fast learner	Constant
	C_f	0.001	Integral learning rate of the fast learner	Constant
	A_s	0.1	Retention rate of the slow learner	Constant
	B_s	0.01	Learning rate of the slow learner	Constant
	C_s	0.0001	Integral learning rate of the slow learner	Constant

3. The third round comments

RC: There are so many initial values of the parameters to be given before starting the simulation work. Were all the initial values determined by experience? For instance, the initials of the CPGs can be formed by a matrix. How do the engineers give the elements of the matrix zero initially or not? or all elements are given by the same values? Is there any systematic algorithm for giving these initials?

AR: Although there are many parameters in the proposed controller (i.e., the adaptive quadruped motor control (AQMC)), most of the parameters belonging to the conventional basic modules CPGs, motor neurons (MNs), and dual integral learner (DIL), which have been introduced in other previous work [1, 2, 3]. The novelty DFFB reflex was proposed in this work to integrate with the three modules to adapt the offsets of the expected joint movement positions. We have now summarized how to initialize the parameters of these modules in the improved supplementary material as:

The instructions of how to setup the parameters of the DFFB reflex, CPGs, MNs, and DIL are described as below (see table S.2):

1. DFFB reflex: the DFFB reflex has four parameters including: $w_1(n)$, $w_2(n)$, w_3 and w_4.

   i) The two adaptive synaptic weights $w_1(n)$ and $w_2(n)$ are positive and determines the sensory gain of the DFFB reflex. According to our experience, their proper initial values are 0.003 and 0.0032, respectively. However, when the DIL was induced to adjust the two parameter values online, their initial values can be set to zero without specific initialization. To sum it up, the initialization for the two values is only necessary for employing the DFFB reflex when without the DIL.

   ii) The two constant parameters w_3 and w_4 are the output gains of the DFFB reflex to the knee and hip joints, respectively. They are set to fixed values 2.0 and 1.0, respectively. This is because the foot displacement moved by the hip joint will be approximately twice that moved by the knee joint if the two joints receive the same command values. Therefore, we compensate for this by setting the knee-joints command value twice that of the hip joint, by setting the w_3 value to double the w_4 value.

2. CPGs: the CPGs here are four coupled special orthogonal (SO(2)) CPG proposed in [1]. They have four constant parameters including: $\mathbf{w}$, $\mathbf{b}$, $\mathbf{\Phi}$, and ϵ.

   i) $\mathbf{w}$ and $\mathbf{b}$ are synaptic weights and biases of the four CPGs. $\mathbf{w}$ value determines the shape and frequency of the CPG outputs. b is for activating the neurons of the CPGs, its value can be initialized in the range of (-0.085, 0.085). Based on the background of well understood neurodynamics of the recurrent neural network [1], the weight parameters were manually selected in accordance with the neurodynamics of the network staying near the Neimark-Sacker bifurcation set where quasi-periodic attractors occur (Pasemann et al. 2003). The attractors can drive the joints to perform rhythmic movements in experiments.

   ii) $\mathbf{\Phi}$ and ϵ represents relative phases and gain among the four CPGs, which determines the phase relationship among the CPG outputs. For instance, ϕ_{12} and ϕ_{13} represent the relative phases of the second CPG (controlling right hind leg) and the third CPG (controlling the left front leg) with respect to the first CPG (controlling the right front leg). Here we used a trot gait for the robot experiment, thus the diagonal legs move in phase but an-phase with other legs. Therefore, the CPGs of the right hind and left front legs are in phase but out phase with the CPG of the right front leg. Thus, $\phi_{23} = 0$, ϕ_{12} and ϕ_{13} equal to $-\pi$. ϵ determines the CPG coupling strength. The larger of this value, the stronger of the CPG relative phase locking. For this parameter setup, to increase its value from zero until the

CPGs can show expected phase relationship. This CPG phase locking technique also can be seen in [4] proposed by Shinya ei. al.

3. MNs: MNs is for summarizing signals from the CPGs and the DFFB reflex and linearly scaling the summed signals to expected joint movement positions. It has two parameters including: α and $\beta(n)$ [2, 5].

   i) α determines the amplitudes of the movement commands or robot walking step height and length. This parameter can properly match the commands to the robot joint movement work space. In the experiments, we set the parameter 0.16 for Lilibot and 0.12 for Laikago according to the robots' configurations, respectively.

   ii) $\beta(n)$ determines the joint movement offsets which influences the robot walking posture. This parameters were set to zero without specific initialization, because they were online modulated by the DFFB reflex.

4. DIL: The DIL has six parameters including: A_f, B_f, and C_f for fast learner, as well as A_s, B_s, and C_s for slow learner. The DIL does not require its parameters to be precisely adjusted to fit specific situations. Its further advantages can be seen in [3, 6]. The principle for setting the parameter values is that the fast learner has larger learning rates than the slow learner while has lower retention rate than the slow learner.

To be clear show the controller setup, we have now added those information and provided more detailed parameter table in the supplementary material as:

Table S.2. Parameters of the adaptive quadruped motor controller.

Modules	Symbols	Initial values	Description	Adaptive /constant
CPGs [1]	$\mathbf{w}$	$\begin{pmatrix} 1.4 & 2.6 \\ -2.6 & 1.4 \end{pmatrix}$	Synaptic weights of the SO(2) CPG neurons	Constant
	$\mathbf{b}$	$\begin{pmatrix} 0.01 & 0.01 & 0.01 & 0.01 \\ 0.01 & 0.01 & 0.01 & 0.01 \end{pmatrix}$	Biases of the SO(2) CPG neurons for triggering CPG activation	Constant
	$\mathbf{\Phi}$	$\begin{pmatrix} 0.0 & -\pi & -\pi & 0.0 \\ \pi & 0.0 & 0.0 & \pi \\ \pi & 0.0 & 0.0 & \pi \\ 0.0 & -\pi & -\pi & 0.0 \end{pmatrix}$	Desired relative phases among the four CPGs [4]	Constant
	ϵ	0.01	CPG communication gain	Constant
MNs [2, 5]	α	0.16 for Lilibot and 0.12 for Laikago	Synaptic weight projection from the CPGs to MNs	Constant
	$\beta(n)$	0	Online-modulated by the DFFB	**Adaptive**
DFFB reflex	$w_1(n)$	0.003	Online-modulated by the DIL	**Adaptive**
	$w_2(n)$	0.0032	Online-modulated by the DIL	**Adaptive**
	w_3	2	DFFB output gain of the knee joint	Constant
	w_4	1	DFFB output gain of the hip joint	Constant
DIL [3, 6]	A_f	0.01	Retention rate of the fast learner	Constant
	B_f	0.05	Learning rate of the fast learner	Constant
	C_f	0.001	Integral learning rate of the fast learner	Constant
	A_s	0.1	Retention rate of the slow learner	Constant
	B_s	0.01	Learning rate of the slow learner	Constant
	C_s	0.0001	Integral learning rate of the slow learner	Constant

References

[1] F. Pasemann, M. Hild, K. Zahedi, So (2)-networks as neural oscillators, in: International Work-Conference on Artificial Neural Networks, Springer, 2003, pp. 144–151.

[2] T. Sun, D. Shao, Z. Dai, P. Manoonpong, Adaptive neural control for self-organized locomotion and obstacle negotiation of quadruped robots, in: 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), IEEE, 2018, pp. 1081–1086.

[3] M. Thor, P. Manoonpong, Error-based learning mechanism for fast online adaptation in robot motor control, IEEE Transactions on Neural Networks and Learning Systems 31 (6) (2020) 2042–2051.

[4] S. Aoi, T. Yamashita, K. Tsuchiya, Hysteresis in the gait transition of a quadruped investigated using simple body mechanical and oscillator network models, Physical Review E 83 (6) (2011) 061909.

[5] T. Sun, X. Xiong, Z. Dai, D. Owaki, P. Manoonpong, A comparative study of adaptive interlimb coordination mechanisms for self-organized robot locomotion, Frontiers in Robotics and AI 8 (2021) 86. doi:10.3389/frobt.2021.638684.
URL https://www.frontiersin.org/article/10.3389/frobt.2021.638684

[6] M. A. Smith, A. Ghazizadeh, R. Shadmehr, Interacting adaptive processes with different timescales underlie short-term motor learning, PLoS Biol 4 (6) (2006) e179.