2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

# Learnable Tegotae-based Feedback in CPGs with Sparse Observation Produces Efficient and Adaptive Locomotion

Christopher Herneth[1], Mitsuhiro Hayashibe[1], and Dai Owaki[1]

***Abstract*—Central Pattern generators (CPG) are a biologically inspired, decentralized control architecture that enables model-free, but yet adaptively stable and computational lightweight locomotion capabilities on complex robots. Nevertheless, no unified design guidelines for closed-loop CPG controllers are available in the literature. Therefore, we propose a task-distributed, end-to-end trainable, closed-loop CPG control policy by generalizing and extending Tegotae control. The Tegotae approach modulates CPG activity by quantifying the discrepancy between internal belief states and environmental reactions. Spontaneous and adaptive gait formation towards situationally efficient locomotion patterns are intrinsic properties of Tegotae control. The Tegotae control policy is trained and benchmarked in simulation on a 1D hopping robot. We found that our approach can learn efficient and adaptive locomotion on minimal feedback information, while outperforming unstructured, classic reinforcement learning policies of equal complexity. To the best of our knowledge, this is the first study to fully generalize the Tegotae approach and construct unimpeded, end-to-end trainable Tegotae control policies.**

## I. Introduction

Contemplating the effortless grace with which animals and man walk the earth, it appears natural to take inspiration from the mechanisms that drive these astounding bodies. Modern approaches for controlling highly complex robots, separate motion planning and execution in distinct processes [6], [16]. The former plans trajectories over limited time horizons, which are subsequently tracked by a model-based controller relying on known, full-body dynamics. Such methods are robust to external disturbances but remain sensitive to unmodelled environmental facets [25]. At the same time, the full-body dynamics of the fast-growing field of soft robotics are often unknown or intractable.

Nevertheless, even the simplest creatures that lack sophisticated brain structures, but yet possess multifaceted bodies, exhibit complex, and adaptive locomotion behavior. Such abilities have been linked to self-organized neural circuits called central pattern generators (CPGs) [18] that can generate rhythmic patterns in the absence of tonic stimulation. An important phenomenon for independent CPG activity is the synchronization of individual oscillators via mechanical entrainment through the physical body [10]. This enables CPG circuits to autonomously react to sensory stimuli and even induce gait transitions [3] which reduces the strain on superior control entities. Absorbing the CPG-based control architecture into robot control loops offers, among others, the following key advantages: limit cycle behavior, low computational footprint, the possibility of distributed and/or parallel implementation, dimensionality reduction of high-level control signals, and the possibility to integrate sensory information [10].

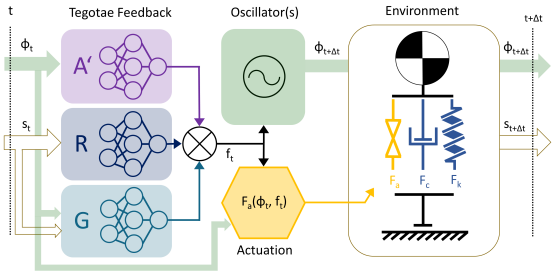


Fig. 1. Proposed generalized Tegotae Control architecture. Action (A), Reaction (R), and Gain (G) terms are replaced by Multi-Layer Perceptrons (MLPs). The actuation function is optionally learnable. We deploy and tune the parametric Tegotae policy on a 1D-Monoped hopper in a Reinforcement learning (RL) setup.

Owing to these traits CPGs have been widely adopted in the robotics community and demonstrated the ability to replicate the characteristics of animal locomotion. While sensory information is essential to ensure adaptivity to unexpected environmental disturbances, its expedient integration into the control architecture is nontrivial. As a solution, the authors of [21], [24] proposed the *Tegotae* framework, a systematic design principle for a local sensory feedback mechanism for CPG controllers. By quantifying the discrepancy between the intention of a controller and perceived environmental reactions, the Tegotae control law modulates CPG activity to achieve synchronization of oscillators and body dynamics with a minimal feedback law.

In this work, we generalize Tegotae control to an end-to-end learnable, parametric control policy, that retains the inherent structure of Tegotae control. We then demonstrate the viability of this approach compared to handcrafted Tegotae solutions and benchmark our CPG-based control architecture against a standard, non-distributed RL policy of equal complexity.

Our contributions are as follows: (1) extension of Tegotae control from constant to active gain modulation, (2) a parametric formulation and implementation of the Tegotae control loop in a unified policy, (3) extensive study on the behavior characteristics of different degrees of generalization and the choice of feedback variables, as well as a comparison to a standard single network policy of equal complexity.

This work was supported by a JSPS KAKENHI Grant-in-Aid for Scientific Research on Innovative Areas through the Project "Hyper-Adaptability (JP20H05458)", "Science of Soft Robot (JP21H00317)", JP18H03167, JP19K22855, and JP20H04260.

All the authors are with the Neuro-Robotics Lab, Department of Robotics, Graduate School of Engineering, Tohoku University, Sendai 980-8579, Japan. christopher.herneth@tum.de/owaki@tohoku.ac.jp

## II. Related work

### A. Tegotae Control

Customizing sensory feedback and oscillator coupling mechanisms for CPG-based approaches is non-trivial and not well studied in literature. To correct this issue, the authors of [21], [23] proposed a systematic design framework for integrating sensory feedback in CPG based control loops. Their approach is inspired by the Japanese concept of Tegotae, which describes the extent to which an expectation matches an observation. In this framework, Tegotae is quantified by the Tegotae function in (1). The Action Term $C(\phi)$ holds a belief about the system/environment configuration based on the states of internal oscillators $\phi$. This expectation is compared against sensory reactions $s$, processed by a Reaction Term $S(s)$. The nature of the Tegotae equation is such that a match of expectation and reaction leads to "good" (positive) Tegotae and vice versa. A decay in Tegotae is caused by an increasing discrepancy between oscillator states and (localized) sensory information, which is the basis of the Tegotae feedback law (2). The feedback signal is designed to modulate the activity of a set of distributed oscillators, such that Tegotae is improved through the entrainment of oscillators and robot body dynamics.

$$T = C(\phi)S(s) \tag{1}$$

$$f = \sigma\frac{\delta T}{\delta \phi} = \sigma\frac{\delta C}{\delta \phi}S(s) \tag{2}$$

Tegotae control leads to unprompted emergency of locomotion patterns, that match with that of animals of compare-able body morphology and weight distribution and even exhibit spontaneous gait transitions in reaction to external circumstances [13], [14], [21]–[23]. An extension to the standard Tegotae framework was proposed in [30], where a completely model-free approach is promoted. The actuation signals are decoupled from the oscillator states, by reflexively reacting to changes in Tegotae itself (7). Previous studies relied on carefully hand-tuned, static Tegotae feedback strengths $\sigma$. However, constant gain settings are suboptimal outside their designated operating points (Fig. 4). Therefore, we propose an actively tuneable Tegotae gain for an effective and customize able reaction rule

### B. Generalizing Tegotae

While the Tegotae framework provides an intuitive technique to systematically design the oscillator feedback, the optimal design of the Action and Reaction terms remains unclear. In [20]–[23], [30] the Action terms act as simple gait phase discriminators (stance and flight phases), while the Reaction terms are limited to hand-tuned, localized sensory information. While animal locomotion circuits are far more complex, further sophistication of the system state and sensory information processing mechanisms are difficult to optimize manually, due to their high nonlinearity. Other authors have thus reiterated the Tegotae approach and other CPG-based approaches by taking advantage of artificial, evolution, and learning processes. In [13] higher Fourier dynamics of the Action Term were designed by learning the coefficients of a Fourier series expansion with a genetic algorithm and thus improved the locomotion efficiency of an earthworm-inspired robot. Similar results were achieved in [11], [12], [14] for a snake and a caterpillar robot, where the Reaction term was extended to global sensory information, using a learned Affine transformation. However, non of these works consider unrestricted and joint learning of the Action and Reaction terms. In this study, we generalize the Tegotae function in its entirety, by replacing each term with a dedicated, general function approximator. We use Multi-Layer Perceptrons (MLP) for that purpose and train the combined policy in a reinforcement learning (RL) environment. Since MLPs do not naturally exhibit oscillatory behavior [5], [31], we retained the internal Kuramoto oscillator model [17] of the original Tegotae control architecture without modification.

### C. CPGs and Reinforcement learning

Combining CPGs with RL leads to a powerful combination of intrinsically stable and computationally lightweight control, with automatic tuning to a variety of scenarios. Examples of CPG controllers, learned by Policy Gradient methods include [5], [7], [19]. In [5] the feedback mechanism and oscillator dynamics were coupled in a single, unified policy by directly integrating a series of nonlinear Hopf-oscillators into the Actor-network of an Actor-Critic algorithm, while [7], [19] make the oscillators part of the environment. Our approach differs from these works by separating the processing streams of internal and external information domains in a multi-headed feedback policy, structured after Tegotae control [24].

## III. METHODOLOGY

### A. Generalizing Tegotae

In this work, we introduce dedicated MLP networks for the Action (A) and Reaction (R) terms and align them according to the Tegotae formulation, resulting in the parameterized equation (3) and the network alignment depicted in Fig. 1 (Tegotae feedback). Here $\phi$ denotes the oscillator state, $s$ the available sensory information, and $\Theta$, $\Psi$, $\Gamma$ represent trainable parameters. Since this parametrization is fully differentiable, the Tegotae feedback could be computed as in (2) Feedback by replacing $\frac{C(\phi)}{d\phi}$ with the backward pass through the A network, with respect to the input variables. Since the Tegotae value does not appear in the control equations, we directly parameterize the derivative of the Action Term (A'), leading to the updated feedback (4). Additionally, we extend the standard Tegotae concept by the equally parameterized, active Tegotae gain (G) that receives inputs from oscillators and sensors. We use the Kuramoto phase oscillator [17] as a model for the CPG oscillators. The oscillator activity is synchronized via the Tegotae feedback f according to equation (5).

$$T = A(\phi, \Psi)R(s, \Gamma) \tag{3}$$

$$f = G(\phi, s, \Theta)\frac{\delta T}{\delta \phi} = G(\phi, s, \Theta)A'(\phi, \Psi)R(s, \Gamma) \tag{4}$$

$$\dot{\phi} = \omega + \sigma f \tag{5}$$

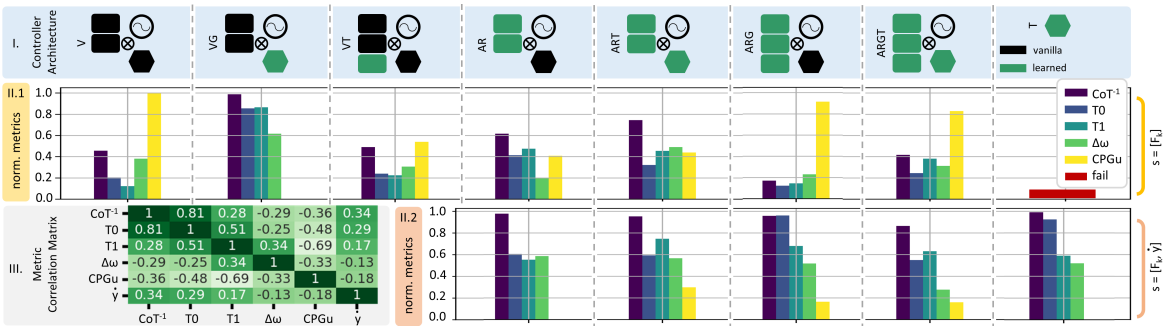


Fig. 2. Subplot I: illustrates the evaluated controller architectures. The pictograms follow the depiction of Fig. 1. Components colored black are modeled after the vanilla Tegotae formulation II-A, whereas green indicates generalization by a dedicated MLP Network: V = Vanilla Tegotae Feedback and static gain; G = learn-able Tegotae Gain; AR = generalized Action and Reaction terms; T = learn-able actuation function. Subplots II.1 and II.2 contain mean and globally normalized performance metrics of the architectures above. II.1 receives $s = [F_k]$ as observation for the Reaction term and single network policy and II.2 $s = [F_k, \dot{y}]$. Subplot III shows the global correlation matrix of the performance metrics described in section IV-A.

Previous studies building upon the Tegotae approach utilized different strategies to calculate actuator signals from the CPG networks:

- oscillator states as continuous target variables for secondary position controllers [11], [12], [14], [20]–[23].
- fixed amplitude $\alpha$ and duration step-function initiated by preset oscillator phase thresholds [30] (Sector actuation: equation 6).
- reflexive response to decaying Tegotae based on Tegotae feedback [30] (Reflex actuation: equation (7))

$$F_a(\phi) = \alpha[\sigma(t - 1.75\pi) - \sigma(t - 1.85\pi)] \tag{6}$$

$$F_a(f) = -min(0, f) \tag{7}$$

In this work, we first investigate the characteristics of the Sector and Reflex actuations in unison with learned A and R terms. Finally, we replace the actuation function in Fig. 1 with an additional MLP network (T) resulting in a unified, end-to-end trainable Tegotae policy modeled after the task distribution of Tegotae control.

### B. Mechanical platform

We analyze and train our considerations in simulation on a one-dimensional hopper Robot (Monopod) and a single Kuramoto oscillator. The hopper is constructed of a point-mass (COM) and a mass-less leg made up of a parallel arrangement of a spring, a damper, and a linear actuator (Fig. 1 (Environment)). The COMs height trajectory over time $y(t)$ is described by the ODE

$$\ddot{y} = \frac{1}{m}(F_k(y) + F_c(\dot{y}) + F_a(\phi, f) + mg) \tag{8}$$

$$F_k(y) = k(l_0 - y + h) \tag{9}$$

$$F_c(\dot{y}) = c\dot{y} \tag{10}$$

The hopper parameters are the mass m, the spring and damper coefficients k and c, and the resting length of the leg $l_0$. Since the hopper is hopping on the spot, only the height of the CoM appears in the dynamic system, while h denotes the surface level. The spring $F_k$, damper $(F_c)$ and actuation forces $(F_a|0 \leq F_a \leq \alpha)$ are switched off during the flight-phase, which is characterized by $y > l_0 + h$.

### C. Learning Setup

Previous works have optimized the parameters of Tegotae control for known and static environmental conditions [21]–[23]. In this work, we aim to provide a platform, that allows Tegotae control to be augmented with adaptive and learnable components that can be trained for optimal behavior under changing environmental conditions. As such our learning setup is implemented under the reinforcement learning regime.

Tegotae control relies on entrainment between body dynamics and oscillators for synchronization. Modern algorithms, such as SAC [9] and PPO [27], inject exploratory noise at much short step intervals, which has been shown to disrupt the much slower entrainment processes [12]. As a remedy, the authors of [11], [12] optimized their CPG policies by Policy Gradients with Parameter-based Exploration (PGPE) [28], which eliminates noise injection during operation, through episodic parameter updates.

To select the best-suiting algorithm, we conduct a preliminary analysis of SAC, PPO, and PGPE. The testbed is a Tegotae controller, consisting of a learnable Tegotae Feedback structure with fixed gain, and an actuation function of either (6) or (7). Our target metric for this analysis is the maximization of the inverse Cost of Transport ($CoT^{-1}$) (12) of an episode. The step-wise reward signal of SAC and PPO is either 0 or calculated as the $CoT^{-1}$ (12) for the last hopper jump, at the instant, the jump reaches its apex. This reward is only provided if energy was expended to discourage solutions that minimize energy expenditure by inactivity. At the end of each episode, the average $CoT^{-1}$ of all jumps is provided. PGPE requires an episodic quantification of the fitness of a policy. In this preliminary analysis, we provide the $CoT^{-1}$ (12) as fitness value. In subsequent experiments utilizing PGPE, we compute fitness as the combination of the $CoT^{-1}$ in (12) and the divergence of the normalized CoM trajectory spectrum from the oscillator frequency: $(CoT^{-1} + \Delta\Omega)$ in (13). The spectral component has an optimal value of 0 and should facilitate finding solutions that comply with prescribed oscillator frequencies.

$$fitness = CoT^{-1} + \Delta\Omega \tag{11}$$

$$CoT^{-1} = \frac{\sum y_{flight}(t)}{E} \tag{12}$$

$$\Delta\Omega = -(w_{osci} - w_{peak})^2 - (1 - A_{peak})^2 \tag{13}$$

## IV. Experiments

### A. Experimental Setup

To analyze the viability of our approach, we train a range of controllers of varying degrees of generalization (Fig. 2 I.) on a 1D Hopper and compare their performance to the results of [30]. The inputs to the Tegotae Feedback components are the oscillator phase $\phi$ (Action Term) and sensory observations $s \in \{[F_k], [F_k, \dot{y}\}]$ (Reaction Term), while the actuation signal is computed from $\{\phi, \dot{y}\}$. A final comparison is made to a standard, single-network MLP policy without internal oscillators, that shares the same number of parameters, but not the structure of our Tegotae control policy. Actuator signals are instead directly calculated based on observations $\in \{F_k, \dot{y}\}$. Specifically, we analyze the following metrics:

- ($CoT^{-1}$) inverse Cost of transport after limit cycle formation according to (12)
- (T0) Initial transient period from hopper release to limit cycle formation
- (T1) Stability to novel and unexpected environmental disturbances in the form of a sudden lowering of the surface level. We quantify this stability as the transient period between the initial hopper touchdown after the surface event and new limit cycle formation, if any.
- ($\Delta w$) is the weighted sum of the differences between all peak frequency components $w_p$ $p \in P$ in the spectrum of the CoM trajectory and the oscillator frequency $w_{osci}$: P = $\{p|A_p \geq \frac{1}{2}A_{max}\}$

$$\Delta w = \sum_{p \in P} |w_p - w_{osci}| A_p \tag{14}$$

- (CPGu) indicator to what degree the CoT depends on a progressive oscillator state. The oscillator phase is clamped to 6 equidistant values $\in [0, 2\pi]$, and the policy subsequently evaluated for limit cycle formation. CPGu = 1 corresponds to no limit cycle formation for any of the fixed values, while CPGu = 0 corresponds to utter independence of model performance from the oscillator.

The Tegotae Feedback f of the baseline model is described by (2), which acts as feedback signal to a Kuramoto oscillator. The actuator signal is either calculated based on the state of the oscillator (6) or reflexively coupled to the Tegotae Feedback as in (7). When the controller intends to be in the stance phase ($-\sin(\phi) > 0$) and observes a non-zero ground reaction force, Tegotae is positive. In the opposite case, Tegotae is negative and the controller seeks to improve it by accelerating or decelerating the oscillator.

### B. Learning procedures

In this work, we use RL to tune the free parameters of the Action, Reaction, Gain, and Actuation networks to given environmental conditions. We implemented our training environment in OpenAI Gym [4] and made use of the functionalities of the PGPElib [29] (PGPE) as well as Stable-Baselines3 [26] (SAC, PPO) for policy optimization. The hyperparameters of SAC, PPO, and PGPE were tuned with Optuna [1]. First, we verify the feasibility of PGPE against SAC and PPO for controllers relying on entrainment by comparing their training performance according to (12) consistently during training. Each algorithm is analysed on 2 policies following the structure in Fig. 1. The policy Action and Reaction terms are MLPs, while the gain is a constant scalar. Policy (I) uses (6) to calculate actuator signals, and Policy (II) relies on (7). The resulting 6 models are trained 3 times each, 1000 episodes for SAC and PPO and 80 episodes for PGPE, with the results averaged for each algorithm (Fig. 3). Associated reward signals are described in Section III-C.

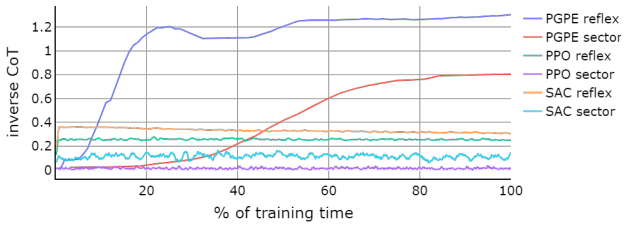


Fig. 3. $CoT^{-1}$ evaluated during the training process of SAC, PPO and PGPE for identical policies.

To investigate the influence of different levels of generalization on the Tegotae feedback structure, varying controller models are constructed and evaluated. A model is defined by the controller architecture, the oscillator frequencies $\omega_t \in \{4, 8\}$ the observation provided to the Tegotae feedback $\in \{[F_k]; [F_k, \dot{y}]\}$ and the observation provided to the actuation function $\in \{[\phi], [f], [\phi, f]\}$. Non-learnable Actuation functions follow (6) and (7) for their respective observations. The Action, Reaction and Gain Networks receive observations $\in \{[F_k], [F_k, \dot{y}]\}$. Each architecture in (Fig. 2 I.) is paired with all feasible permutations of Tegotae feedback and actuation observations, resulting in a total of 218 distinct models. Each model is trained a total of 15 times, starting from randomly initialized weights. For this optimization, we use the PGPE algorithm and the fitness function (11). The final model is sampled from the best-performing parameter distribution. Model metrics are subsequently calculated as the mean performance of the final model, across 5 episodes with oscillator frequencies of $\omega \in [\omega_{t-2}, \omega_{t+2}]$.

A crucial aspect of our analysis is that we intentionally deprive the training process of any environmental disturbances and thus do not embed knowledge of such events in the learned parameters. Additionally, the utilization of the CPG feedback structure and oscillator states is not specifically encouraged. Stability to unexpected events and automatic pattern generation in the absence of sensory information are core properties of CPGs and are expected to emerge from the intrinsic properties of the controller.

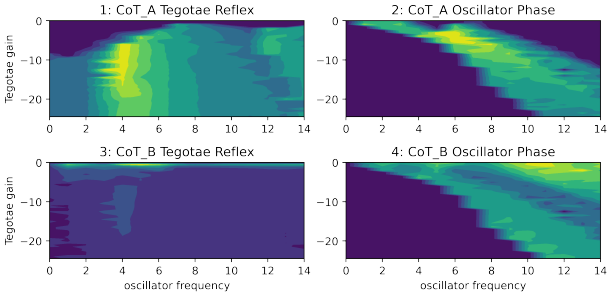


Fig. 4. Relationship between the inverse Cost of Transport ($CoT^{-1}$) for Tegotae Reflex and oscillator phase-based actuation functions for Tegotae gain and oscillator frequency pairs. $CoT^{-1}$ A is for positively climbed heights during the flight phase only, while $CoT^{-1}$ B considers total positive upwards movement of the CoM.

## V. Results and Discussion

In this work, we proposed a generalization of the Tegotae framework, by systematically replacing the individual terms in the Tegotae feedback formulation (1) and (2) and subsequent actuation signal generators through dedicated, general function approximators. The resulting Tegotae, CPG control policy can be automatically tuned to a variety of platforms and scenarios while retaining the distributed control structure of the Tegotae approach. The free model parameters were tuned in an RL setting. Since the entrainment mechanisms fundamental to Tegotae control have been shown to be disrupted by modern algorithms like SAC and PPO, we compared their performance against PGPE, which avoids the injection of explorative noise, through episodic parameter updates [11], [12]. Our results support these findings, with the agent performance of SAC, PPO, and PGPE during training summarized in Fig. 3. The purpose of the controller is to modulate the oscillator velocity, with actuation signals coupled to oscillator feedback and state. While SAC and PPO produce periodic feedback patterns, their actions are based on too short time horizons, which do not lead to fruitful entrainment. At the same time, PGPE was able to converge to solutions, that operate at the resonance of the controlled system. A drawback of PGPE is its limitation by the curse of dimensionality, which requires sufficiently small parameter spaces to remain feasible. We found that policies exceeding 128 parameters, were untrainable. The results presented in the following utilize PGPE policies with 96 trainable parameters.

The generalization of individual components of the Tegotae feedback structure entails specific allocations of model metrics, which are summarized for different controller architectures in Fig. 2. To compare our results to previous works, we first constructed baseline controllers (V), which implement the Tegotae architectures of [24], [30]. Contrary to this work, they rely on handcrafted Action and Reaction terms and an empirically tuned, static Tegotae gain $\sigma$. The resulting control strategies are strongly connected to the state of the oscillators and achieved medium $CoT^{-1}$ performance. Meanwhile, their stability to unforeseen environmental disturbances are among the best (small T0 and T1) of all tested architectures. Fig. 5 shows the actuation surface for selected, illustrative controller configurations. The surfaces correspond to the behavior of the entire controller, including the oscillator dynamics, Tegotae feedback mechanism, and actuation function for all possible $(\phi, F_k)$ pairs. Additionally, the Tegotae feedback and actuator force are plotted for one revolution of the oscillator in the limit cycle, for each surface respectively. The actuation behavior in the limit cycle for the reflex and sector actuation are depicted in Fig. 5 (I and II). It is visible how the oscillator is slowed down during actuation in both instances, to synchronize power injection with the stance phase. The actuation magnitude is either gradually modulated by $F_k$ (reflex) or a step function (sector).

These results also provide a comparison of subsequent evaluations to traditional approaches. In [30] it was shown that the same Tegotae feedback mechanism, which was generalized in this work, leads to a similar controller behavior as obtained by optimal control. The learnable parameters introduced in this work enable the automatic generation of comparably performing controllers, without the need for carefully hand-engineered feedback terms, which is crucial for more complex systems involving a large number of oscillators.

Our first proposed extension to the Tegotae framework is motivated by the dependency of the optimal Tegotae Gain setting on the dynamics of the system, which leads to quick performance degradation when the oscillator frequency is varied. In Fig. 4 the relationship between $\sigma$ and a given oscillator velocity ($\omega$) is depicted for the reflex and the sector actuation. For the latter a near linear relationship between $\sigma$ and $\omega$ can be found, which correlates with the findings of [20]. The dynamics of the reflex actuation are noticeably more complex and the closed-loop resonance cannot be influenced by fixed Tegotae gains. We thus introduce a third term $G(\phi, s)$ into the Tegotae feedback law, which receives the combined inputs of the Action and Reaction terms, to actively modulate the gain setting (4). The following results retain the chosen sensory input of s = $F_k$ of the baseline [24], [30]. Fig. 2 (II.2) contains selected, architecture-dependent performance metrics, introduced in Section IV-A. Adding active gain modulation to the otherwise unchanged model (VG) vastly improves the $CoT^{-1}$, which correlates with the expected behavior of our reward formulation. However, due to the exploitation of a closed-loop resonance, initial and perturbation recovery transients are substantially prolonged. Interestingly, gain modulation effectively switches off the oscillator feedback mechanism, which is visible in the erratic Tegotae feedback in Fig. 5 (IV). This effect is further expressed in the independence of the actuation instant from the oscillator state, while the actuation force is solely modulated by $F_k$. A learned actuation function (VT) on the other hand, does on average not lead to large deviations from the baseline performance but produces smoother actuation signals (compare Fig. 5 (V) and (VT)).

Learning the Tegotae feedback mechanism (AR), has the potential to improve the efficiency of the system, while significantly lowering the transient spikes encountered for mere gain modulation, without specifically encouraging such behavior. At the same time, the jumping frequency can be

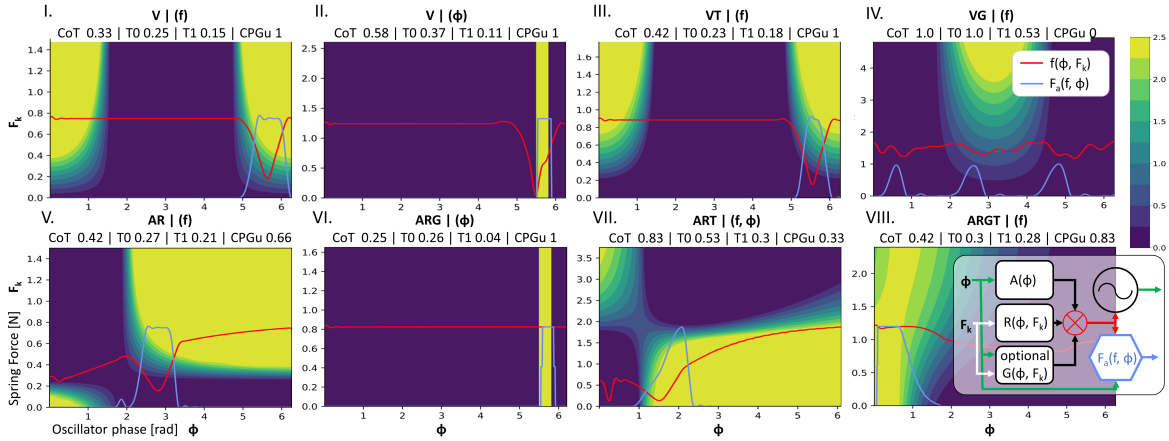


Fig. 5. The surface plots visualize the actuation signal as the output of the entire Tegotae policy, dependent on all possible oscillator state $\phi$ [rad] and spring-force $F_k$ [N] pairs. Actuation and oscillator feedback trajectories for a single oscillator rotation in the limit cycle, are depicted in blue and red respectively. Policy architectures, actuation function inputs, and resulting, normalized performance metrics are indicated above each plot.

easily controlled through its tight coupling to the oscillator frequency. Paring an AR controller with a learnable actuation function (ART) leads to further gains in locomotion efficiency. In Fig. 5 (VII) the feedback mechanism of the selected ART policy exhibits the stance phase extension behavior of standard Tegotae, while the power injection strategy is more finely modulated. The resulting smoother actuator signal has the potential to reduce actuator tears. Contrary to VG, added gain modulation to the generalized Tegotae feedback structure (ARG and ARGT), reduces the efficiency, which is inherently correlated with lower transient periods. For ARG this results in below-baseline performance. Fig. 5 (VI and VIII) re-demonstrate, that gain modulation switches off the Tegotae feedback mechanism. This leads to little time available for power injection in (VI) as the actuation instant is coupled to the oscillator state, while in (VIII) feedback reflexive actuation decays to force reflexive actuation.

The power of the distributed, CPG-based approach becomes apparent when the zero performance of the single network policy (T) is considered in the eye of the strongly limited available sensory information. While it shares the same number of learnable parameters, it is not capable of beating or even approaching the baseline. This underlines the viability of our distributed control architecture, even in RL environments.

An advantage of our generalized Tegotae formulation is the trivial integration of additional sensory information. In this case, the vertical velocity of the CoM is appended to the previously used spring force in the extended observation vector $s = [F_k, \dot{y}]$. Overall, providing $\dot{y}$ to the Tegotae feedback structure, makes the controller less dependent on the oscillator activity (low CPGu metric) and enables large gains in efficiency, but entails an equally large degradation of stability to disturbances (Fig. 2 (II.2) (AR), (ART), (ARG), (ARGT)). This is consistent with the works of [8], [15], where feedback that modulates the CPG activity (actuation signals are strongly coupled to the CPG state), was described as most stable to locomotion in uncertain terrain. Increasing the information content in the sensory feedback also makes the standard MLP policy viable (Fig. 2 II.2 T) and raises its efficiency on par with that of Tegotae control, but requires very long transient periods to achieve a stable limit cycle.

Among all permutations of model architectures and observations, the individual performance criteria are not independently achievable. Fig. 2 (III) shows the global correlation matrix of the collective performance of all models discussed above. Generally, high efficiency leads to a trade-off with slow convergence to a limit cycle and reduced stability to disturbances. Meanwhile, the high dependency of the actuation dynamics on the CPG activity greatly improves the convergence behavior for unexpected perturbations. While this entails lowered efficiency, the tight coupling between oscillators and actuation offers elegant means of velocity modulation, since the velocity of the CPG oscillators can easily be controlled by high-level control signals.

In future works, we plan to replace the MLPs with learnable, neural oscillator models such as [2], where the constraints of the Matsuoka's neural oscillator are differentiably embedded in an Artificial Neural Network. Further, we intend to implement our controller design for a network of decentralized oscillators, investigating its ability to elicit entrainment mechanisms on multifaceted platforms. Finally, the RL foundation of our setup enables the investigation of the ability of learned Tegotae feedback structures, to generalize environmental conditions encountered during training to similar, but yet unknown domains.

## VI. Conclusion

In this study, we proposed a task-distributed, closed-loop CPG-based policy modeled after Tegotae control. In simulation experiments, our end-to-end learnable Tegotae policy demonstrated major efficiency gains, compared to previous, manually designed Tegotae-based methods, while retaining the characteristics of Tegotae control. The distributed policy architecture proved to be especially robust to reduced information environments. Under equal conditions, a classic monolithic RL policy failed to learn successful locomotion patterns and only achieved performance parity with additional sensory information.

## REFERENCES


[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

[2] Prof. Amar Nath Singh and. Utility study of neural network and back propagation algorithm in the field of learning and computing data in mines area. *International Journal Of Engineering And Computer Science*, 11 2016.

[3] D M Armstrong. The supraspinal control of mammalian locomotion. *The Journal of Physiology*, 405(1):1–37, nov 1988.

[4] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[5] Luigi Campanaro, Siddhant Gangapurwala, Daniele De Martini, Wolfgang Merkt, and Ioannis Havoutis. Cpg-actor: Reinforcement learning forcentral pattern generators. In *Towards Autonomous Robotic Systems*, pages 25–35. Springer International Publishing, 2021.

[6] Yvain de Viragh, Marko Bjelonic, C. Dario Bellicoso, Fabian Jenelten, and Marco Hutter. Trajectory optimization for wheeled-legged quadrupedal robots using linearized ZMP constraints. *IEEE Robotics and Automation Letters*, 4(2):1633–1640, apr 2019.

[7] Gen Endo, Jun Morimoto, Takamitsu Matsubara, Jun Nakanishi, and Gordon Cheng. Learning CPG-based biped locomotion with a policy gradient method: Application to a humanoid robot. *The International Journal of Robotics Research*, 27(2):213–228, feb 2008.

[8] Yasuhiro Fukuoka, Hiroshi Kimura, and Avis H. Cohen. Adaptive dynamic walking of a quadruped robot on irregular terrain based on biological concepts. *The International Journal of Robotics Research*, 22(3-4):187–202, mar 2003.

[9] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. January 2018.

[10] Auke Jan Ijspeert. Central pattern generators for locomotion control in animals and robots: A review. *Neural Networks*, 21(4):642–653, 05 2008.

[11] Matthew Ishige, Takuya Umedachi, Tadahiro Taniguchi, and Yoshihiro Kawahara. Exploring behaviors of caterpillar-like soft robots with a central pattern generator-based controller and reinforcement learning. *Soft Robotics*, 6(5):579–594, 10 2019.

[12] Matthew Ishige, Takuya Umedachi, Tadahrio Taniguchi, and Yoshihiro Kawahara. Learning oscillator-based gait controller for string-form soft robots using parameter-exploring policy gradients. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, oct 2018.

[13] Takeshi Kano, Ryu Wakimoto, Mitsutoshi Sato, Ayumi Shinohara, and Akio Ishiguro. Designing higher fourier harmonics of ¡i¿tegotae¡/i¿ function using genetic algorithm—a case study with an earthworm locomotion. *Bioinspi. Biomim.*, 14(5):054001, 08 2019.

[14] Takeshi Kano, Ryo Yoshizawa, and Akio Ishiguro. Tegotae-based control scheme for snake-like robots that enables scaffold-based locomotion. In *Biomimetic and Biohybrid Systems*, pages 454–458. Springer International Publishing, 2016.

[15] Hiroshi Kimura, Seiichi Akiyama, and Kazuaki Sakurama. *Autonomous Robots*, 7(3):247–258, 1999.

[16] Scott Kuindersma, Robin Deits, Maurice Fallon, Andrés Valenzuela, Hongkai Dai, Frank Permenter, Twan Koolen, Pat Marion, and Russ Tedrake. Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot. *Autonomous Robots*, 40(3):429–455, jul 2015.

[17] Yoshiki Kuramoto. Self-entrainment of a population of coupled nonlinear oscillators. In *International Symposium on Mathematical Problems in Theoretical Physics*, pages 420–422. Springer-Verlag.

[18] Eve Marder and Dirk Bucher. Central pattern generators and the control of rhythmic movements. *Current Biology*, 11(23):R986–R996, 11 2001.

[19] T. Matsubara, J. Morimoto, J. Nakanishi, M. a. Sato, and K. Doya. Learning CPG-based biped locomotion with a policy gradient method. In *5th IEEE-RAS International Conference on Humanoid Robots, 2005.* IEEE.

[20] Rui Filipe Morais. Cpg and tegotae-based locomotion control of quadrupedal modular robots. 2016.

[21] Dai Owaki, Masashi Goda, Sakiko Miyazawa, and Akio Ishiguro. A minimal model describing hexapedal interlimb coordination: The tegotae-based approach. *Frontiers in Neurorobotics*, 11, 06 2017.

[22] Dai Owaki, Shunya Horikiri, Jun Nishii, and Akio Ishiguro. Tegotae-based control produces adaptive inter- and intra-limb coordination in bipedal walking. *Frontiers in Neurorobotics*, 15, 05 2021.

[23] Dai Owaki and Akio Ishiguro. A quadruped robot exhibiting spontaneous gait transitions from walking to trotting to galloping. *Scientific Reports*, 7(1), 03 2017.

[24] Dai Owaki, Takeshi Kano, Ko Nagasawa, Atsushi Tero, and Akio Ishiguro. Simple robot suggests physical interlimb communication is essential for quadruped walking. *Journal of The Royal Society Interface*, 10(78):20120669, 01 2013.

[25] Brahayam Ponton, Alexander Herzog, Andrea Del Prete, Stefan Schaal, and Ludovic Righetti. On time optimization of centroidal momentum dynamics. September 2017.

[26] Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021.

[27] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. July 2017.

[28] Frank Sehnke, Christian Osendorfer, Thomas Rückstieß, Alex Graves, Jan Peters, and Jürgen Schmidhuber. Policy gradients with parameter-based exploration for control. In *Artificial Neural Networks - ICANN 2008*, pages 387–396. Springer Berlin Heidelberg.

[29] Nihat Engin Toklu, Paweł Liskowski, and Rupesh Kumar Srivastava. Clipup: A simple and powerful optimizer for distribution-based policy evolution. In *International Conference on Parallel Problem Solving from Nature*, pages 515–527. Springer, 2020.

[30] Riccardo Zamboni, Dai Owaki, and Mitsuhiro Hayashibe. Adaptive and energy-efficient optimal control in cpgs through tegotae-based feedback. *Frontiers in Robotics and AI*, 8, 05 2021.

[31] Liu Ziyin, Tilman Hartwig, and Masahito Ueda. Neural networks fail to learn periodic functions and how to fix it. June 2020.