Real Time Movement Classification in Versatile CPG Control

Melanie Jouaiti, Patrick Henaff

To cite this version:

Melanie Jouaiti, Patrick Henaff. Real Time Movement Classification in Versatile CPG Control. Workshop on Robust Artificial Intelligence for Neurorobotics, Aug 2019, Edinburgh, United Kingdom. hal-02291647

HAL Id: hal-02291647
https://hal.archives-ouvertes.fr/hal-02291647
Submitted on 19 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Abstract

Everyday human tasks are composed of a succession of discrete and rhythmic movements. If we want robots to be able to interact appropriately, it appears paramount for them to be able to perform both types of movements too. Though Central Pattern Generators (CPGs) are usually employed for rhythmic movement generation, they are also able of producing discrete movements. In this paper, we present a classification method to distinguish rhythmic and discrete movements so that the CPG can switch from one mode to the other. Moreover, we introduce several new plasticity rules more suitable for discrete movements.

1 Introduction

Discrete movements are defined as singularly occurring events preceded and followed by a period without motion; rhythmic movements are continuous and recurring periodically. It was proposed that rhythmic movements might be a concatenation of discrete movements [Baratto et al., 1986, Feldman, 1980] or on the contrary, discrete movements might be aborted rhythmic movements [Mottet and Bootsma, 1999, Guiard, 1993]. A fMRI study by [Schaal et al., 2004] revealed that rhythmic movements involve only a subset of the cortical activity observed in discrete movements, effectively demonstrating that the neural mechanisms differ for rhythmic and discrete movements.

The distinction between rhythmic and discrete movements is still an ongoing problematic in motor neuroscience. Complex tasks involve a closely intertwined association of both discrete and rhythmic movements (cleaning, piano playing...). While humans are naturally able to seamlessly combine rhythmic and discrete movements, this is a bigger issue in robotics. Robot controllers need to be able to realize both types of movements and easily transition from one mode to the other. Several such structures have already been proposed, they are mainly a combination of rhythmic and discrete pattern generators. [Degallier et al., 2011] introduced a model associating the Hopf (rhythmic) and VITE (discrete) [Bullock and Grossberg, 1988] models. In [Sternad et al., 2000], the Matsuoka (rhythmic) [Matsuoka, 1985] and VITE (discrete) models were combined together. [de Rugy and Sternad, 2003, Sternad, 2008] proposed a model based solely on the Matsuoka oscillator but the discrete movement generated is merely the abortion of a rhythmic movement. As underlined by [Degallier and Ijspeert, 2010], the Matsuoka oscillator is indeed intrinsically rhythmic and can’t generate real discrete movements.

In [Jouaiti and Henaff, 2018], we showed that the Rowat-Selverston oscillating neuron model [Rowat and Selverston, 1993] is able to generate both rhythmic and discrete movements and seamlessly switch from one mode to the other. A high-level rationale to switch from one mode to the other was however lacking. This paper is thus an attempt to answer that question.

A CPG is a biological structure found in the spinal cord of vertebrates. It is responsible for the generation of rhythmic patterns which can be modulated by sensory feedbacks. CPG models are based on a pair of mutually inhibitory oscillating neurons, called half-center [Grillner and Wallen, 1985, Rybak et al., 2006], controlling the extensor and flexor muscles. Non-linear models of CPG, also called relaxation oscillators, can synchronize with an external input or with a coupled CPG, thus ensuring coordination. The model of CPG used here has four layers: Rhythm Generator, Pattern Formation, Sensory Neurons and Motoneurons and incorporate hebbian plasticity rules allowing versatility and adaptation. CPGs provide an control which allows to imitate movements but not copy them, so that each system (human and robot in Human-Robot Interactions) can retain their own characteristics while...
coordinating. We observe the same phenomenon with humans who can imitate each other without exactly copying (e.g. different amplitude, different kind of movement). Moreover, CPGs are quite robust to aberrant values (e.g. punctual error in the input signal).

In this work, we revise the plasticity rules introduced in [Jouaiti et al., 2018] to make them effective for discrete movements too. We also propose a classification algorithm for rhythmic and discrete movements which has been implemented into the CPG so that the CPG can automatically switch from one mode to the other. We will demonstrate the effectiveness of this system using human motion data. In the first section, we will recall the CPG architecture and introduce the new plasticity rules and the classification algorithm. Then, we will present results demonstrating the effectiveness of the new learning mechanisms and classification. Finally, we will discuss our results.

2 Material and Method

In this section, we will first recall the CPG architecture. Then, we will introduce the new plasticity rules for the discrete mode. Finally, we will present our classification rule for rhythmic and discrete movements.

2.1 CPG Architecture

The general architecture for the CPG is represented in Fig. 3. See [Jouaiti et al., 2018] for extensive details.

For the rhythm generator neurons, Rowat-Selverston cells are used, the neuron model can be written as follows:

\[
\dot{V}_{i(E,F)} = y_{i(E,F)} - W \frac{y_{i(E,F)}}{1 + e^{-4y_{i(E,F)}}} + \epsilon_i F_i 
\]

(1)

\[
\dot{y}_{i(E,F)} = \left( \sigma_f - \frac{\tau_m}{\tau_s} - 1 - \sigma_f \tanh^2 \left( \frac{\sigma_f}{A_{f_i(E,F)}} V_{i(E,F)} \right) \right) \cdot \frac{y_{i(E,F)}}{\tau_m} - \frac{y_{i(E,F)}}{\tau_s \tau_m} V_{i(E,F)} + \frac{A_{f_i(E,F)}}{\tau_s \tau_m} \tanh \left( \frac{\sigma_f y_{i(E,F)}}{A_{f_i(E,F)}} \right)
\]

(2)

with \( V \) the membrane potential and \( \tau_m \) and \( \tau_s \) time constants, \( A_f \) influences the output amplitude, while \( \sigma_f \) determines whether the neuron is able to oscillate or not. \( \sigma_s \) influences the intrinsic frequency, \( i \in \mathbb{N} \), designating the joint id. \( F_i \) the CPG input, \( \epsilon \) a synaptic weight designed to scale the input and the term in \( W \) models the mutual inhibition between the extensor and flexor rhythmic cells.

2.2 CPG Discrete Mode

The CPG can seamlessly switch from one mode to the other by modifying the values of \( \sigma_f \) and \( \sigma_s \) [Jouaiti and Henaff, 2018]. Setting \( \sigma_f = 0.0 \) and \( \sigma_s \leq 1.0 \) will yield a discrete mode and the CPG become equivalent to a PID controller. The rhythmic mode requires \( \sigma_f \geq 1.0 \) and \( \sigma_s > 1.0 \). However, the plasticity rules we introduced in [Jouaiti et al., 2018] were exclusively designed for the rhythmic mode and are not appropriate for discrete movements. Consequently, we modify the plasticity equations as follows:

The frequency learning rule which was previously defined follows:

\[
\dot{\sigma}_{s_i(E,F)} = \frac{2 \epsilon_i (1 - \sigma_f) F_i \sqrt{\tau_m \tau_s (1 + \sigma_s) - \sigma_f}}{y_{i(E,F)} \sqrt{V_{i(E,F)}^2 + y_{i(E,F)}^2}}
\]

(3)

becomes:

\[
\dot{\sigma}_{s_i(E,F)} = \frac{\sigma_f \cdot 2 \epsilon_i (1 - \sigma_f) F_i \sqrt{\tau_m \tau_s (1 + \sigma_s) - \sigma_f}}{y_{i(E,F)} \sqrt{V_{i(E,F)}^2 + y_{i(E,F)}^2}}
\]

(4)
This rule now ensures that frequency learning is disabled in discrete mode, i.e. when $\sigma_f = 0.0$.

The learning rule for $A_F$ remains unchanged:

$$A_{f_i(E,F)} = -\mu \left( \frac{\nu \sigma_f V_{f_i(E,F)}}{A_{f_i(E,F)}} \right)^2 - F_i^2 \quad (5)$$

The equation for $\epsilon$ previously defined as:

$$\dot{\epsilon}_i(E,F) = \lambda \tanh^2(\xi F_i) \left( 1 - (\epsilon_i(E,F) F_i)^2 \right) \quad (6)$$

becomes:

$$\dot{\epsilon}_i(E,F) = \begin{cases} \tanh^2(\xi F_i) \
\left[ (1 - \sigma_f) \cdot \lambda F_i^2 - \text{output}_i^2 \right] + \
\sigma_f \cdot \lambda \left[ (1.0 - (\epsilon \cdot F_i)^2) \right] \end{cases} \quad (7)$$

This ensures that the amplitude of $\epsilon F$ remains 1.0 (optimal functioning mode) in the rhythmic mode, while forcing the output amplitude to match the input amplitude in the discrete mode.

### 2.3 Rhythmic - Discrete Classification

To classify the signal as rhythmic or discrete, we use the Fast Fourier Transform (FFT) over $n$ sliding windows of size $\{1,...,n\}$ seconds. If the difference between the maximum of the FFT and the mean is large, then we assume that there is a frequency peak. We also consider the spread of the peak. Combining several window sizes also allows to take a wider range of frequency into account. This yields the following equation for $C(x,t)$ the classification result at time $t$ with $x$ the array of all input values:

$$C(x,t) = \sum_{N=1}^{n} P(\{x_{t-N}...x_t\}) + \alpha \cdot S(\{x_{t-N}...x_t\}) \quad (8)$$

$$P(x) = \max(\text{FFT}(x)) - \frac{\text{FFT}(x) - \text{stdDev}(\text{FFT}(x))}{\text{len}(\text{FFT} > \text{FFT})} \quad (9)$$

$$S(x) = \text{len}(\text{FFT}(x) > \text{FFT}(x)) \quad (10)$$

with $n$ the number of iterations, $\text{FFT}(x)$ the FFT of $x$, $\text{stdDev}(x)$ the standard deviation of $x$ and $\text{len}(\text{FFT} > \text{FFT})$ the number of values greater than the average FFT.

This allows us to introduce a new synaptic rule for $\sigma_f$ related to the rhythmic / discrete movement detection:

$$\sigma_f = \gamma \cdot \left( \frac{\tanh(100 \cdot C(x,t)) + 1.0}{2.0} - \sigma_f \right) \quad (11)$$

with $\gamma$ a learning step.

The classification results are positive for rhythmic movements and negative for discrete movements. The classification is amplified with a gain and smoothed and normalized using a sigmoid. According to this rule, $\sigma_f$ converges towards 1 when the classification result is positive and towards 0 when it is negative.

### 3 Results

First, we will demonstrate the effectiveness of the new plasticity rule for $\epsilon$. Then we will evaluate our classification algorithm. And finally, we will present results for $\sigma_f$ learning mechanism.

#### 3.1 Discrete Movements

To demonstrate the effectiveness of the new rule, we tested the model with systematic data: a step, a ramp and a sinusoid. We can observe on Figure 2 that although the CPG without learning effectively reproduces the signal, it doesn’t match the amplitude of the signal. However, with the new learning rule for $\epsilon$, the CPG also adapts to the input amplitude.

[Figure 2: In blue: input signal of the CPG; in orange: CPG output. Left: CPG without any learning rule. Right: CPG with the new learning rule]

#### 3.2 Rhythmic - Discrete Movements Classification

The classification algorithm runs in average in 4 ms for each time step, allowing real time classification. The classification algorithm has been tested on human motion data acquired with the T-Sens Motion Sensors [tea,]. The participants were performing series of nine ten seconds waving followed each time by four to six seconds of rest. Sampling frequency was
Real Time Movement Classification in Versatile CPG Control

64Hz. 4457.53 seconds of data were manually labelled as rhythmic or discrete and the classification was evaluated on this data.

Let us define precision, recall and F1-score as follows:

\[
\text{precision} = \frac{TP}{TP + FP} \quad (12)
\]

\[
\text{recall} = \frac{TP}{TP + FN} \quad (13)
\]

\[
F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (14)
\]

with \(TP\) the number of true positives, \(FP\) the number of false positives and \(FN\) the number of false negatives.

Overall, we obtained 65% (±5%) of precision, 100% (±6.2e−6%) of recall and a F1 score of 0.79 (±0.04). Failures of classification mostly come from data which looks like noise and fails to be classified as rhythmic. Moreover, although the classification immediately detects a switch from discrete to rhythmic, there is a delay for the transition between rhythmic and discrete.

Failures of classification mostly come from data which looks like noise and fails to be classified as rhythmic. Moreover, although the classification immediately detects a switch from discrete to rhythmic, there is a delay for the transition between rhythmic and discrete.

\[
\text{PLV}(t) = \frac{1}{N} \sum_{i=0}^{N} e^{j(\phi_1(i) - \phi_2(i))} \quad (15)
\]

with \(N\) the sliding window size, \(j = \sqrt{-1}\), \(\phi_k\) the instantaneous phase of signal \(k\).

It can be observed on Figure 5 that, while the PLV struggles at transitions, the signals are coordinated (high PLV score) most of the time.

Evaluating the performance on all the data yields an average PLV of 0.85 (±0.04), 62% (±3%) precision, 98% (±7%) recall and a F1-score of 0.76 (±0.04).

4 Conclusion

In this work, we revised the plasticity rules introduced in [Jouaiti et al., 2018] to make them effective for discrete movements too. We also proposed a classification rule for rhythmic and rhythmic movements which has been implemented into the CPG so that the CPG can automatically switch from one mode with the other. We showed that the CPG is effectively able to adapt to both discrete and rhythmic movements using human motion data.

Though the results are not perfect, they are highly satisfactory, especially considering the variability in the input signals. In future work, we will validate this in real human-robot interactions where the robot will imitate the human movements.

This work also demonstrates the versatility of the Rowat-Selverston model which can incorporate a wide range of plasticity mechanisms. These mechanisms can be freely modified to take into account the constraints or goals of the interactions and expand the adaptive capabilities of the CPG.

References


Figure 4: Left: Example of classification result. In blue, the CPG input, in magenta, the evolution of $\sigma_f$, i.e. classification result (1 for rhythmic and 0 for discrete). Right: Zoom over the results. In blue, the CPG input, in green the CPG output, in magenta, the evolution of $\sigma_f$.

Figure 5: Coordination performance of the CPG (PLV between the CPG input and output). In blue, CPG input, in red, PLV score.


