Chapter 1

dummy Introduction

Chapter 2

dummy Background

Chapter 3

Baseline

Chapter 4

dummy Tools and software

Chapter 5

dummy Implementation

Chapter 6

Experiments and results

This chapter presents the experiments and results set up and performed in the novel synchronization simulator in Unity, as presented in Chapter 5, for certain configurations of musical robot collectives. Effects of the individual musical robots's hyperparameters on the collective achievement and performance of achieving harmonic synchrony are presented. Some examples are hyperparameters which determines how much each musical robot will adjust itself after hearing a transmitted fire signal from a neighbouring robot; α for phase adjustment and β for frequency adjustment.

The main performance scores presented in this chapter will consist of synchronization times / simulation time (s) (i.e. how long it takes robots to reach the state of harmonic synchrony if they ever do), accompanied by the respective and corresponding error rates during the belonging simulation runs (i.e. the percentage of robot collectives out of e.g. 30 runs failing to reach harmonic synchrony before the maximum time limit of e.g. 5 minutes).

6.1 Solving the ϕ synchronization problem

This is the section where experiments attempting to solve the first and simpler problem, namely synchronizing only the phases ϕ_i of all agents i, are presented and analyzed. These are then experiments where all musical robots have an equal and fixed frequency, only adjusting phases, in order to entrain to synchronize their phases to each other until reaching harmonic synchrony.

6.1.1 Reproducing K. Nymoen's phase synchronization

In order to see that our developed synchronization simulator in Unity yields more or less the same results as K. Nymoen et al.'s firefly system [1], similar experiments as reported in their paper are performed. These tell us whether differences in performance, in terms of synchronization times (s), is simply due to implementation differences, or actually because of the utilized synchronization methods and hyperparameters in question.

First off, and as performed and reported in K. Nymoen et al.'s paper [1] (c.f. their Fig. 7), Mirollo & Strogatz's phase adjustment method (as presented in

2.4.2.1) is experimented with for varying phase coupling constants α, as seen in Figure 6.1.

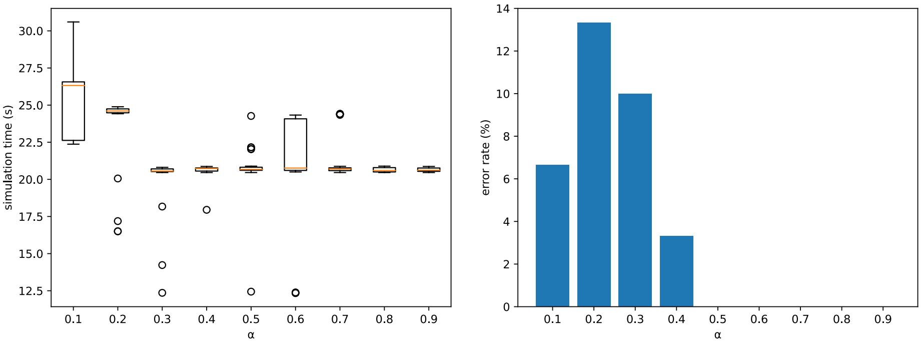

Figure 6.1: Synchronization times (s) for 6 robots with initially random and unsynchronized phases but equal and fixed frequencies (1Hz), for varying phase coupling constants α. 30 simulation runs per α. $\beta = 0$ and $t_{ref} = 50ms$. Maximum time limit was 5 minutes.

6.1.2 Hyperparameter tuning

Tuning hyperparameters α and $t_{ref\_percentage\_of\_period}$ for several robot collective sizes, according to the performance scores of how long it takes the robot collective to achieve harmonic synchrony. Exactly these two specific hyperparameters are experimented with since they empirically seem to be the most important ones to set correctly before starting the simulator; that is, in terms of the robots actually managing to achieve harmonic synchrony.

6.1.3 Comparing phase adjustment methods

Now, an experiment follows where we investigate the validity of the claimed [] benefits of performing bi directional phase adjustments—both inhibitory and excitatory—compared to simply adjusting phases in an excitatory way (i.e. only "pushing" other ocillators's phase further or higher when firing, never "holding" or "dragging" it back).

6.2 Solving the ϕ&ω synchronization problem

This is the section for the experiments attempting to solve the second and harder problem of synchronizing both phases ϕ_i, as well as frequencies ω_i, for all agents i. These are then experiments where all musical robots originally have unequal and ever-changing phases and frequencies, adjusting both phases and frequencies in order to entrain to synchronize their phases and frequencies until reaching harmonic synchrony.

6.2.1 Reproducing K. Nymoen's phase and frequency synchronization

Here we present attempts made in the novel Unity synchronization simulator at recreating K. Nymoen et al.'s first results with their novel frequency synchronization method, which is utilizing, amongst other aspects, self awareness [1].

Ordering by phase couplings Given that K. Nymoen et al. do not mention their β value in their frequency synchronization experiment where they order for different phase coupling values α, an *empirically decent* β value of 0.4 is chosen for this experiment. What *empirically decent* refers to in this case are K. Nymoen et al.'s findings in the results of their last experiment [1] where synchronization times for various β values were evaluated; deeming $\beta = 0.4$ to be a good value, with no further improvement in synchronization performance when β is increased further. Also, when empirically observing the visual and aural signals being transmitted through the synchrony simulator in Unity, this value of $\beta = 0.4$ seemed to lead to relatively stable—given original instability—simulation runs where the robot collective does manage to achieve harmonic synchrony, which is desirable when testing how fast they can synchronize.

Here, K. Nymoen et al.'s self aware frequency adjustment, as implemented in the novel synchrony simulator in Unity, is experimented with for varying phase coupling constants α as in Figure 6.1 — only that this time not only phases are initially un-synchronized; robot frequencies are also initially un-synchronized. See the results in Figure 6.2.

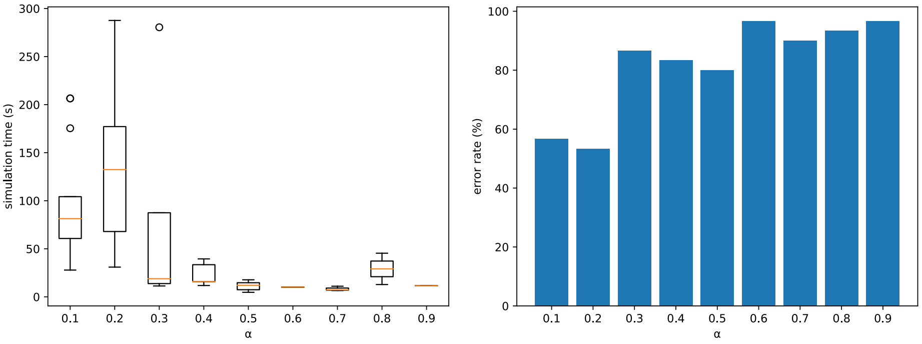

Figure 6.2: Synchronization times (s) for 6 robots with both initially random and unsynchronized phases, and frequencies, for varying phase coupling constants α. 30 simulation runs per α. $t_{ref} = 50ms$ and $m = 5$. Maximum time limit was 5 minutes.

Ordering by frequency couplings Now, we perform the same phase & frequency synchronization experiment as in Figure 6.2, except that this time we will fix the phase coupling constant α and instead test how the individual musical robots's various frequency coupling constants β affect the performance of the musical robot collective. These results are shown in Figure 6.3

Again, K. Nymoen et al. does not specify exactly the phase coupling constant they use in this latter experiment when testing their firefly-inspired synchronization system for various β values. Hence, the now fixed phase coupling constant α is here selected by reusing the α value found in the similar experiment presented in Figure 6.2 to yield the lowest error rate in the musical robot collective when synchronizing to each other. This then gives us a fixed $\alpha = 0.2$.

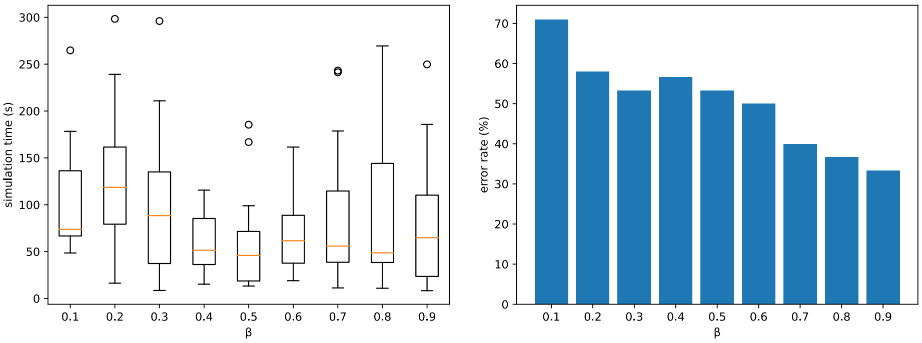

Figure 6.3: Synchronization times (s) for 6 robots with both initially random and unsynchronized phases, and frequencies, for varying frequency coupling constants β. 30 simulation runs per β. $t_{ref} = 50ms$ and $m = 5$. Maximum time limit was 5 minutes.

6.2.2 Hyperparameter tuning

Tuning hyperparameters β and m for several robot collective sizes according to performance scores. The reason for tuning exactly these two hyperparameters, the frequency coupling constant β and the error score list length (or memory length) m, is again since they seem the most significant for the synchronization performance.

Bibliography

[1] Kristian Nymoen et al. “Decentralized harmonic synchronization in mobile music systems”. In: *2014 IEEE 6th International Conference on Awareness Science and Technology (iCAST)*. IEEE, 2014, pp. 1–6.