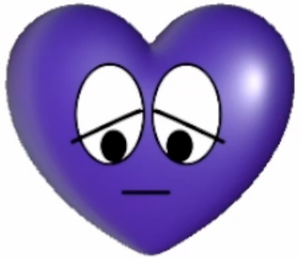
Self-Synchronized Oscillators

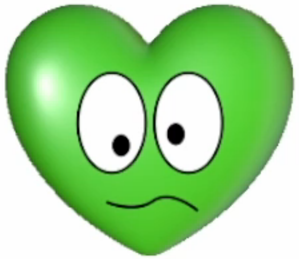
For ***Human-Swarm Interactive Music Systems***

By *Pedro Lucas*

Self-Synchronized Oscillators

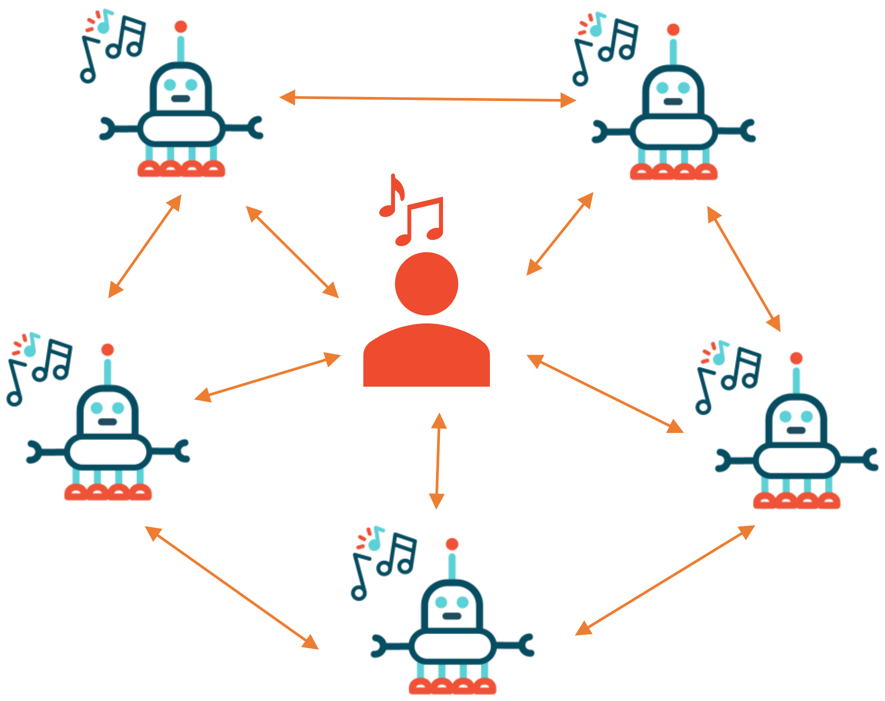
For ***Human-Swarm Interactive Music Systems***





Intention

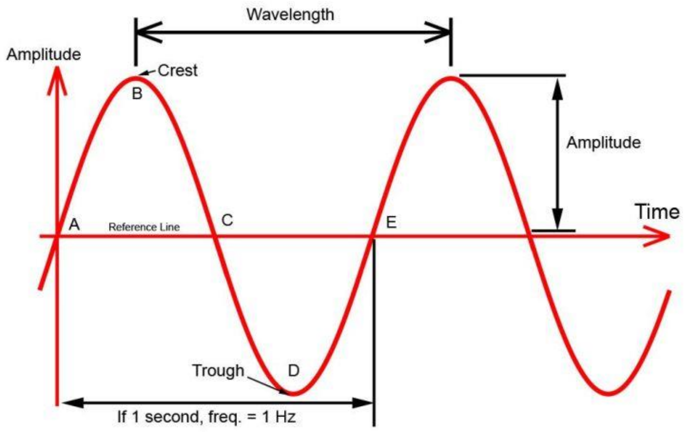
Use self-synchronized oscillators as a mechanism for “automatic tempo synchronization” in Human-Swarm Interactive Music Systems, that is, a **“decentralized clock”.**



What is an “*oscillator*”?

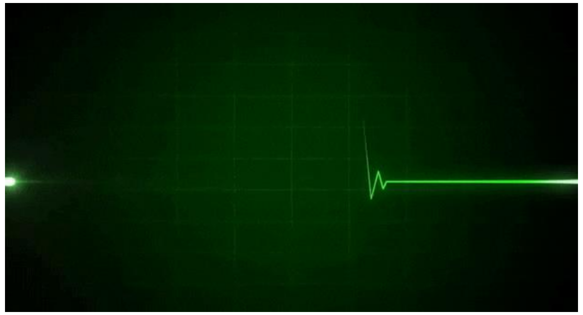
What is an “*oscillator*”?

- Any instrument for producing **oscillations[1].**
- An **oscillation** is a periodic motion about an equilibrium position[2].



https://mathematicalmysteries.org/sine-wave/

https://pudgypenguins.com/



https://giphy.com/gifs/heartbeat-VGK2WUT3amXjG

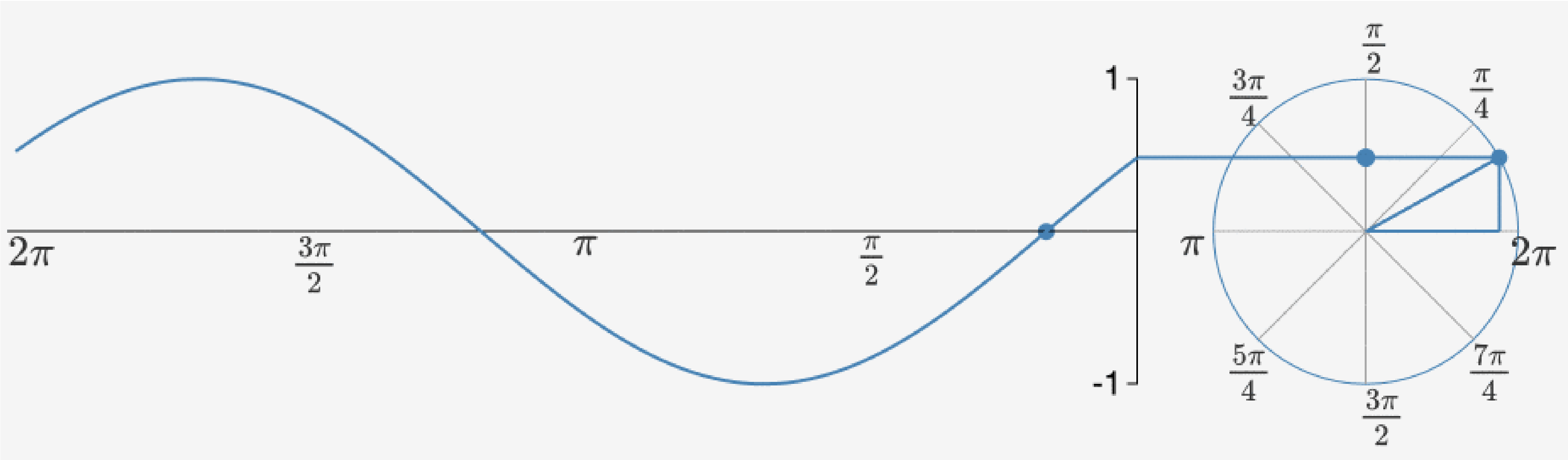
https://www.sonypictures.de/

[1]https://www.collinsdictionary.com/dictionary/english/oscillator

[2]https://www.oxfordreference.com/display/10.1093/acref/9780199233991.001.0001/acref-9780199233991-e-2171?rskey=M2crUs&result=2441

What is an *"oscillator"*?

We can describe them mathematically (e.g. ***y = sin(x)***):

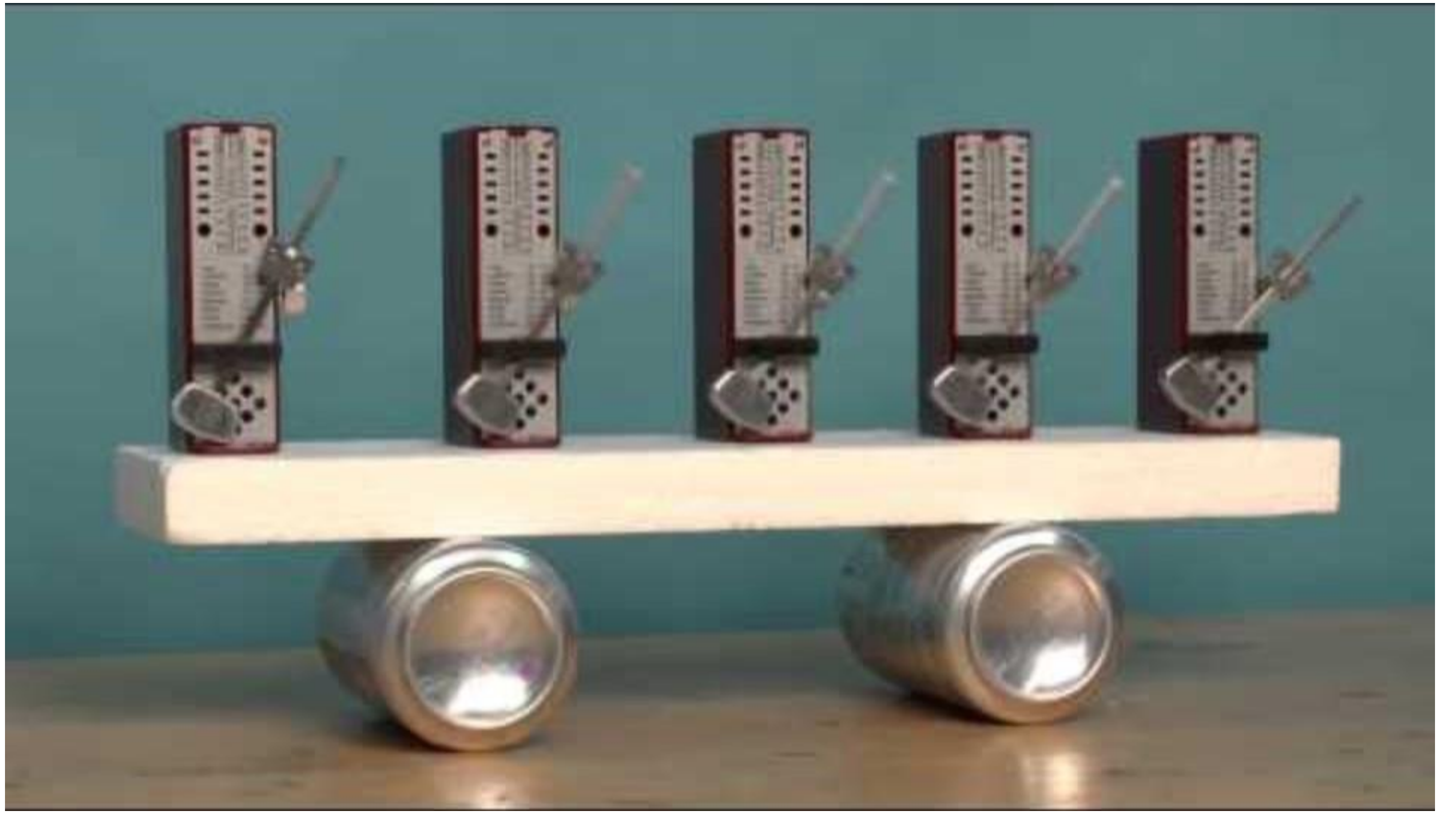


https://mathematicalmysteries.org/sine-wave/

What are *“coupled oscillators”*?

What are “*coupled oscillators*”?

Coupled oscillators are oscillators connected in such a way that energy can be transferred between them[1]



[1]http://teacher.pas.rochester.edu/PHY235/LectureNotes/Chapter12/Chapter12.pdf

What are “*coupled oscillators*”?

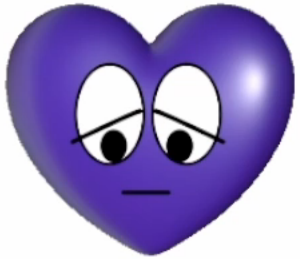
Fireflies Synchronization

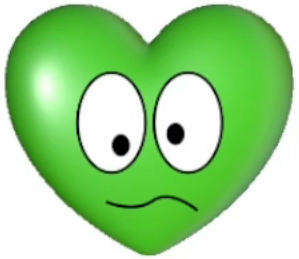
http://sevenworldsoneplanet.com/

Crowd Synchronization

https://www.youtube.com/watch?v=eAXVa__XWZ8

What are "*coupled oscillators*"?





Coupled Oscillators Models

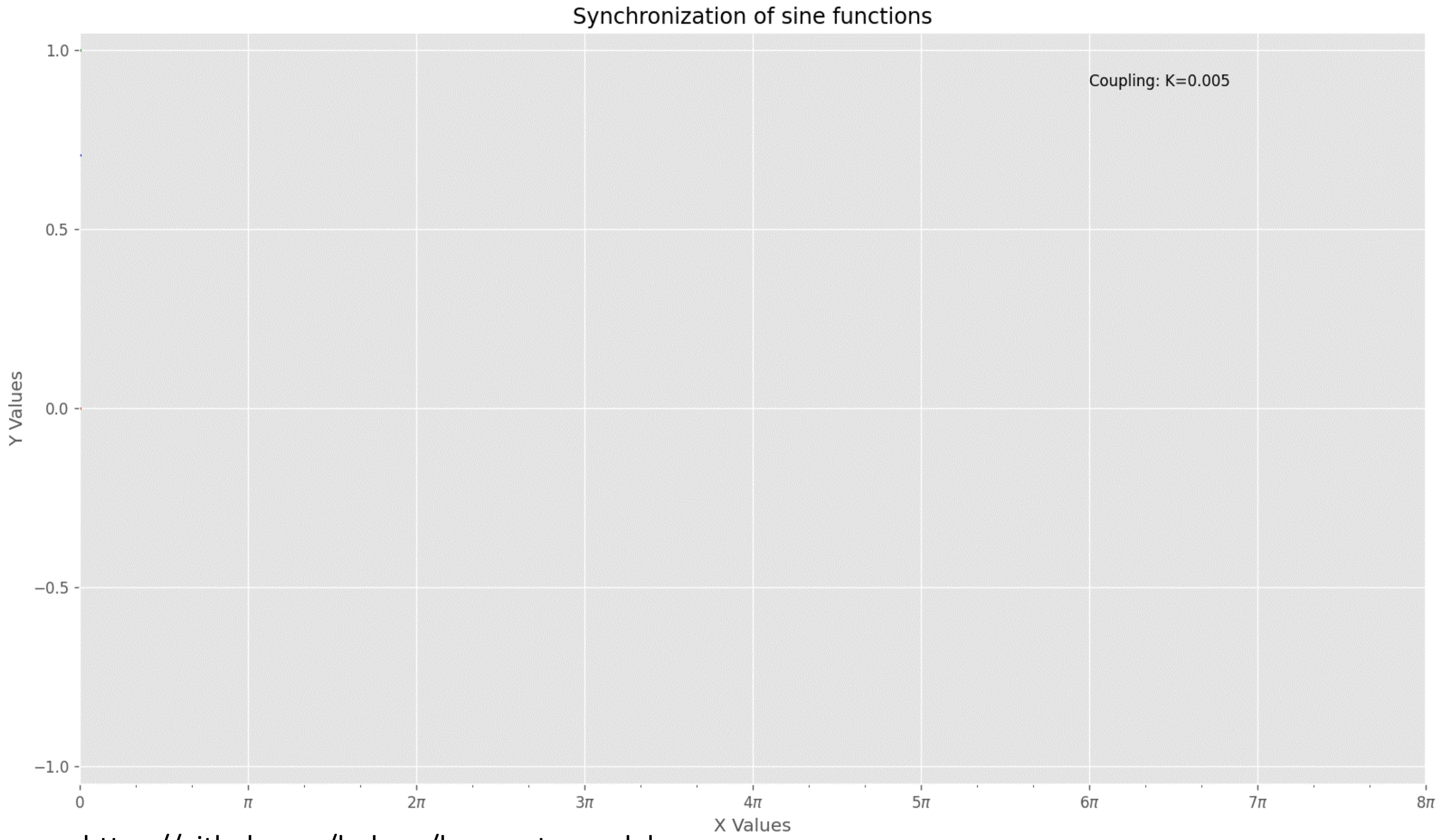
Coupled Oscillators Models

Kuramoto Model[1]

$$\dot{\theta}_i(t) = \omega_i + \frac{K}{N}\sum_{j=1}^{N}[\sin(\theta_j(t) - \theta_i(t))]$$

“Phase (θ)-Coupled Oscillators”

“Continuous interaction”



https://github.com/k-donn/kuramoto-model

[1] Y. Kuramoto, “Self-entrainment of a population of coupled non-linear oscillators,” in *International Symposium on Mathematical Problems in Theoretical Physics*, H. Araki, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 1975, pp. 420–422.

Coupled Oscillators Models

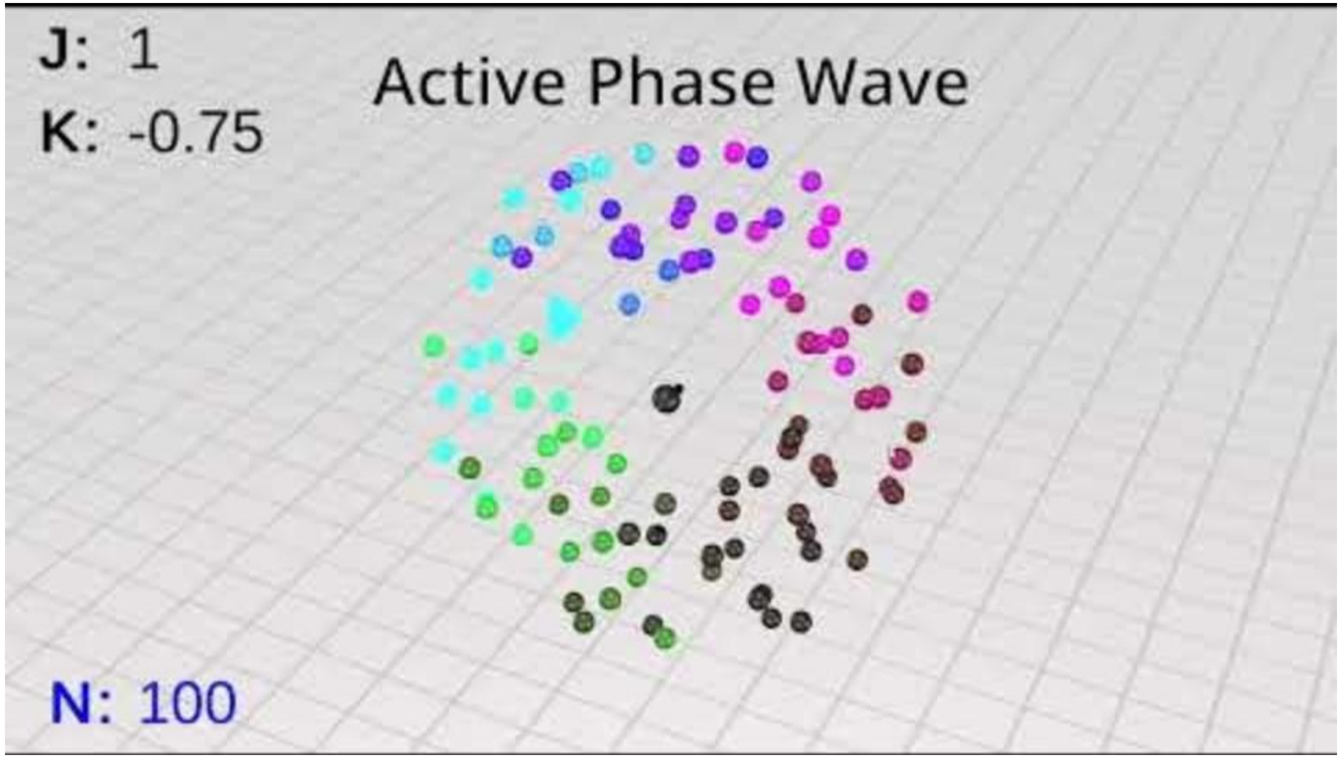
Swarmalator Model[1]

$$\dot{\mathbf{x}}_i = \mathbf{v}_i + \frac{1}{N}\sum_{j=1}^{N}\left[\mathbf{I}_{\mathrm{att}}(\mathbf{x}_j - \mathbf{x}_i)F(\theta_j - \theta_i) - \mathbf{I}_{\mathrm{rep}}(\mathbf{x}_j - \mathbf{x}_i)\right], \quad (1)$$

$$\dot{\theta}_i = \omega_i + \frac{K}{N}\sum_{j=1}^{N}H_{\mathrm{att}}(\theta_j - \theta_i)G(\mathbf{x}_j - \mathbf{x}_i) \quad (2)$$

"Phase (θ) and Space ($\mathbf{x}$) - Coupled Oscillators"

"Also continuous interaction"



https://youtu.be/sDsLuTV6qLw

Entrainment Workshop presentation: https://osf.io/stx3e

[1]**K. P. O'Keeffe, H. Hong, and S. H. Strogatz, "Oscillators that sync and swarm,"** ***Nature Communications*****, vol. 8, no. 1, pp. 1–12, 2017**

Coupled Oscillators Models

Mirollo-Strogatz Model[1]

When a fire event is received:

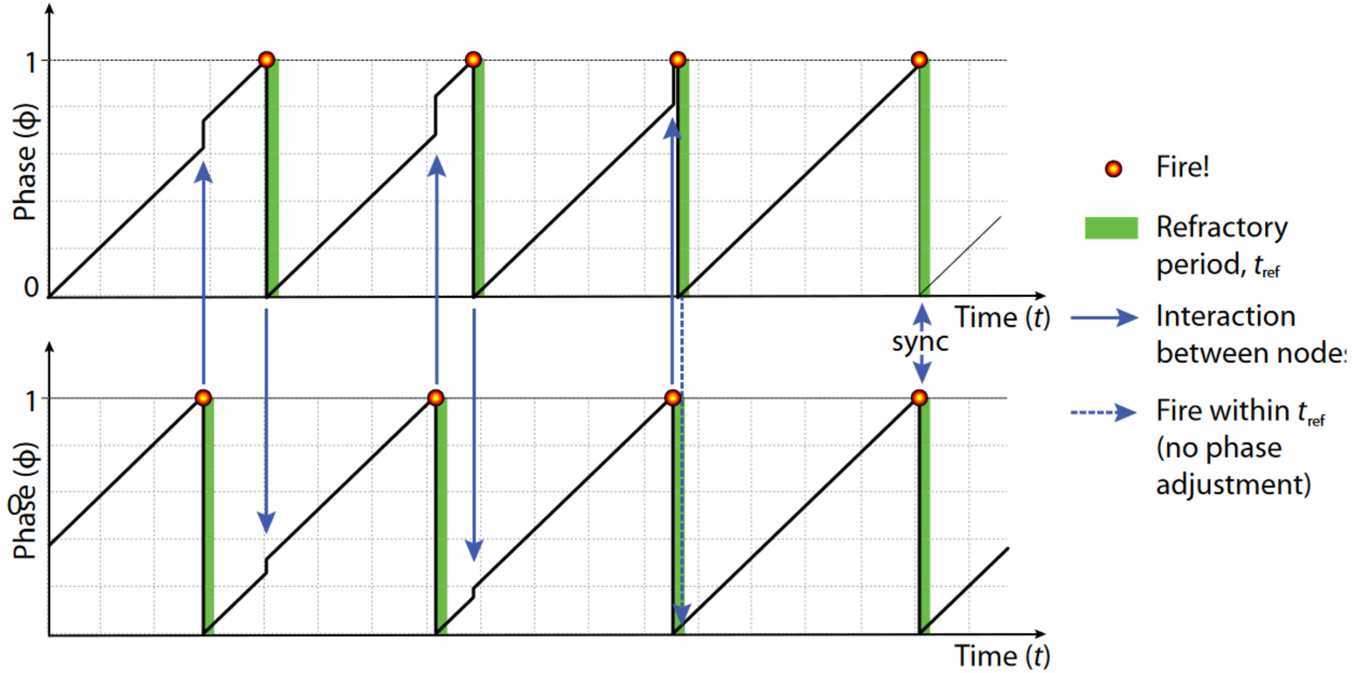
$$\phi_j(t) = 1 \Rightarrow \begin{cases} \phi_j(t^+) = 0 \\ \phi_i(t^+) = P(\phi_i(t)) \quad \forall i \neq j \end{cases}$$

Phase update from (Nymoen et al., 2014)[2]:

$$P(\phi) = \phi - \alpha \cdot \sin 2\pi\phi \cdot |\sin 2\pi\phi|$$

"Pulse-Coupled Oscillators"

"Event-based interaction (Fire Event)"

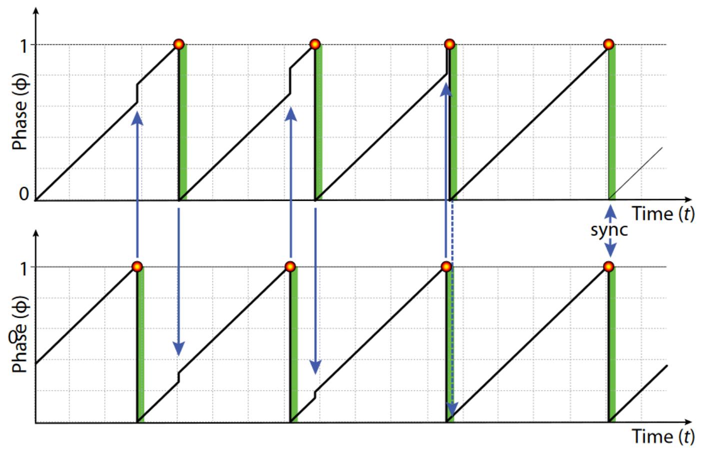


Two equal pulse-coupled oscillators synchronize using Mirollo-Strogatz algorithm[2]

[1] **R. E. Mirollo and S. H. Strogatz, "Synchronization of Pulse-Coupled Biological Oscillators," *SIAM J. Appl. Math.*, vol. 50, no. 6, pp. 1645–1662, Dec. 1990**

[2] **K. Nymoen, A. Chandra, K. Glette, and J. Torresen, "Decentralized harmonic synchronization in mobile music systems," *2014 IEEE 6th International Conference on Awareness Science and Technology, iCAST 2014*, vol. 257906, no. 257906, 2014**

Pulse-Coupled Oscillators



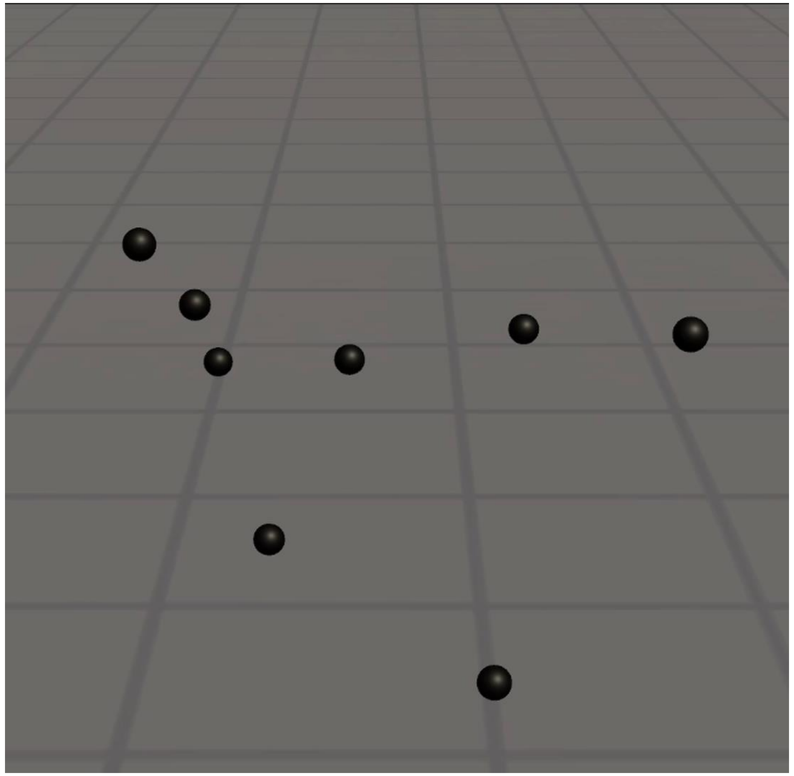
Pulse-Coupled Oscillators

The intention is to: Use pulse-coupled oscillators as a mechanism for “automatic tempo synchronization” in Human-Swarm Interactive Music Systems, that is, a “decentralized clock”.

Simulation in a 3D environment
(Unity3D implementation: PC Platform)

$$P(\phi) = \phi - \alpha \cdot \sin 2\pi\phi \cdot |\sin 2\pi\phi|$$

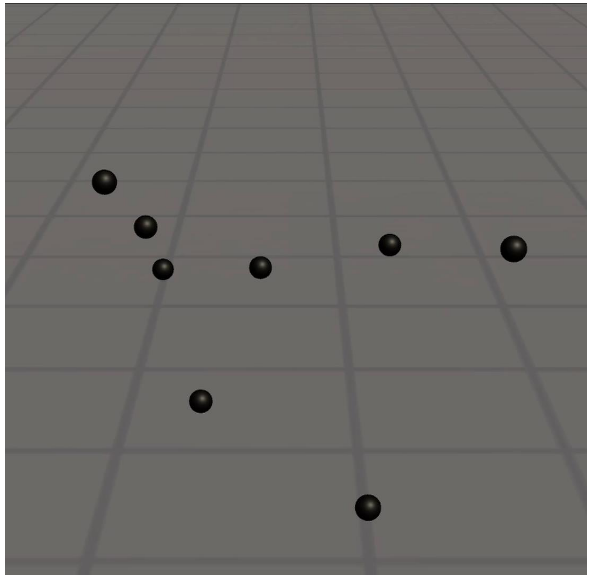
(K. Nymoen et al. , 2014)



K. Nymoen, A. Chandra, K. Glette, and J. Torresen, “Decentralized harmonic synchronization in mobile music systems,” *2014 IEEE 6th International Conference on Awareness Science and Technology, iCAST 2014*, vol. 257906, no. 257906, 2014

Pulse-Coupled Oscillators

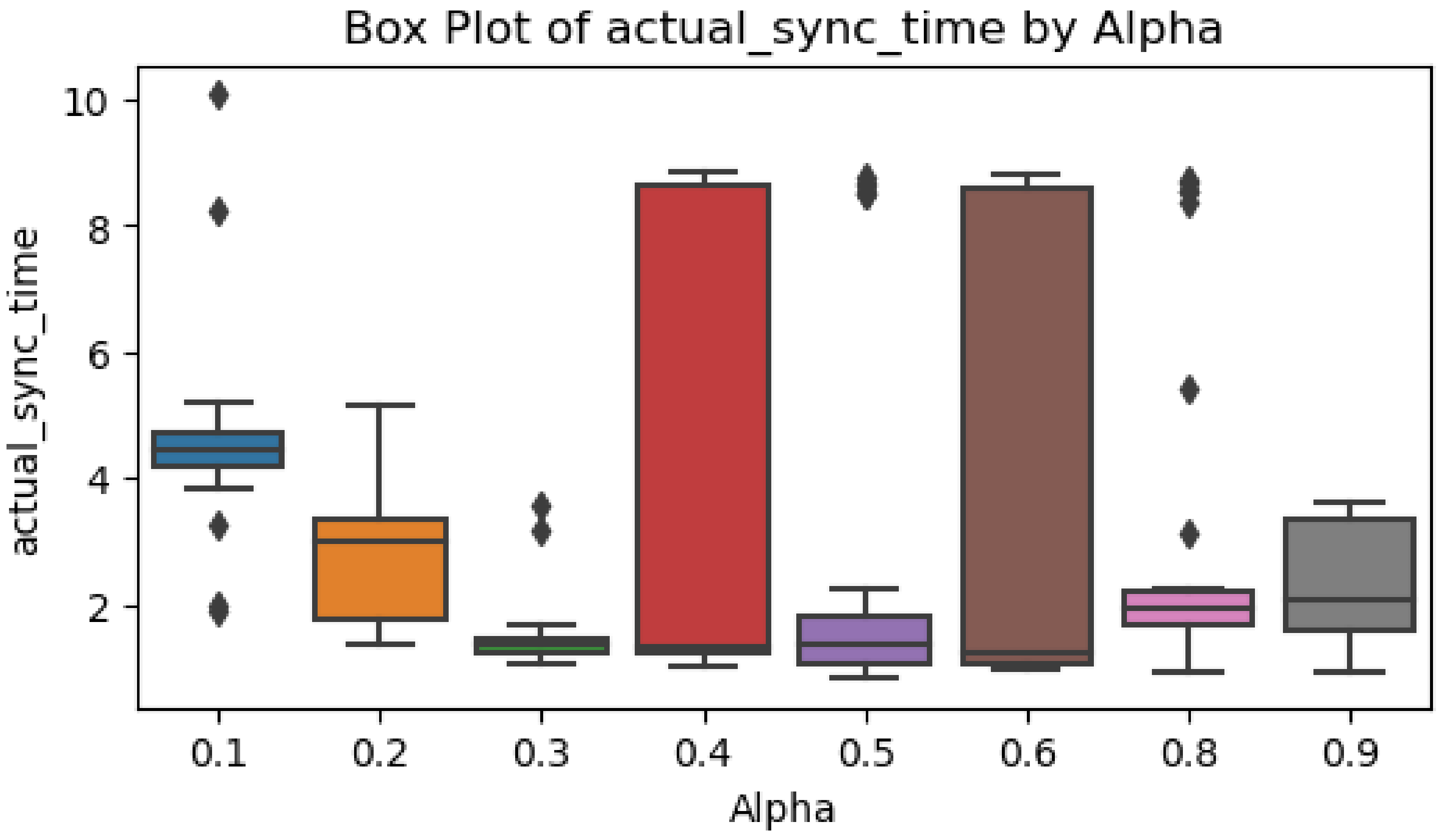
Simulation



Pulse-Coupled Oscillators

Simulation: Synchronization Time Results (6 agents)

$$P(\phi) = \phi - \alpha \sin 2\pi\phi \cdot |\sin 2\pi\phi|$$

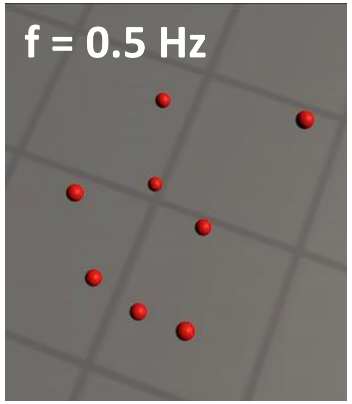


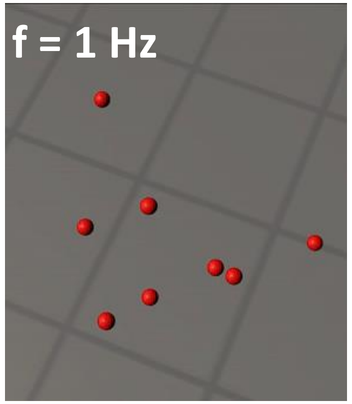
Alpha (α)	Mean Time(Sec)
0.1	4.59
0.2	2.76
0.3	**1.47**
0.4	3.74
0.5	2.75
0.6	3.44
0.8	3.04
0.9	**2.31**

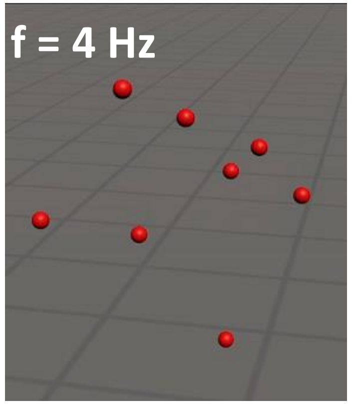
Explored previously in: K. Nymoen, A. Chandra, K. Glette, and J. Torresen, "Decentralized harmonic synchronization in mobile music systems," *2014 IEEE 6th International Conference on Awareness Science and Technology, iCAST 2014*, vol. 257906, no. 257906, 2014

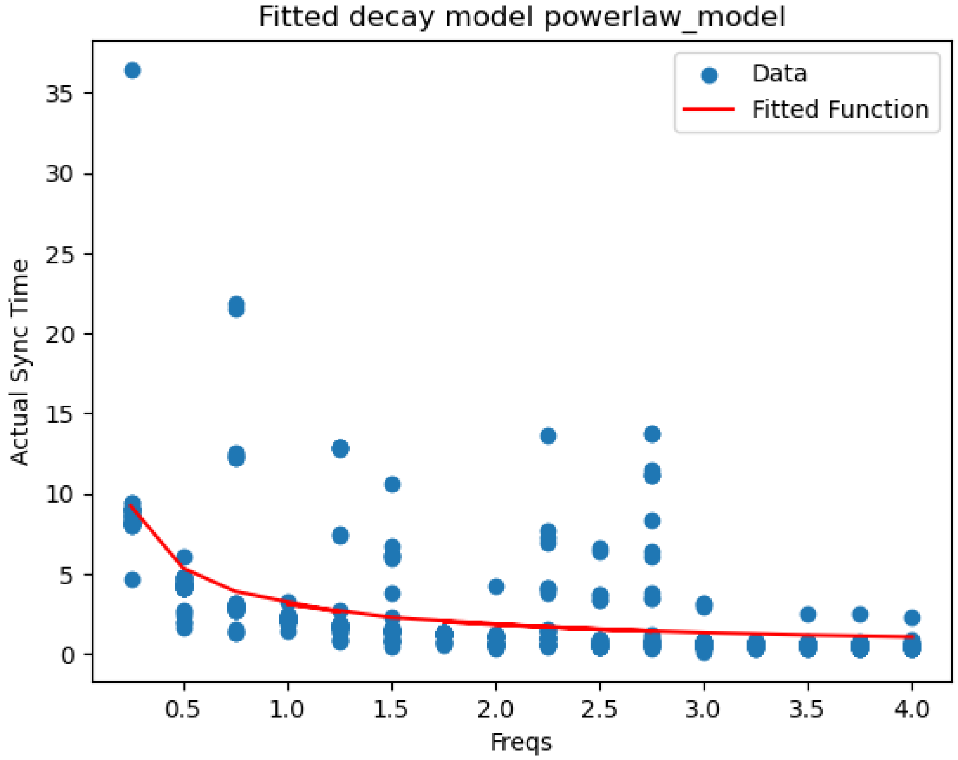
Pulse-Coupled Oscillators

Simulation: Synchronization Time Results per frequency (8 agents, alpha = 0.9)









Power Law Model

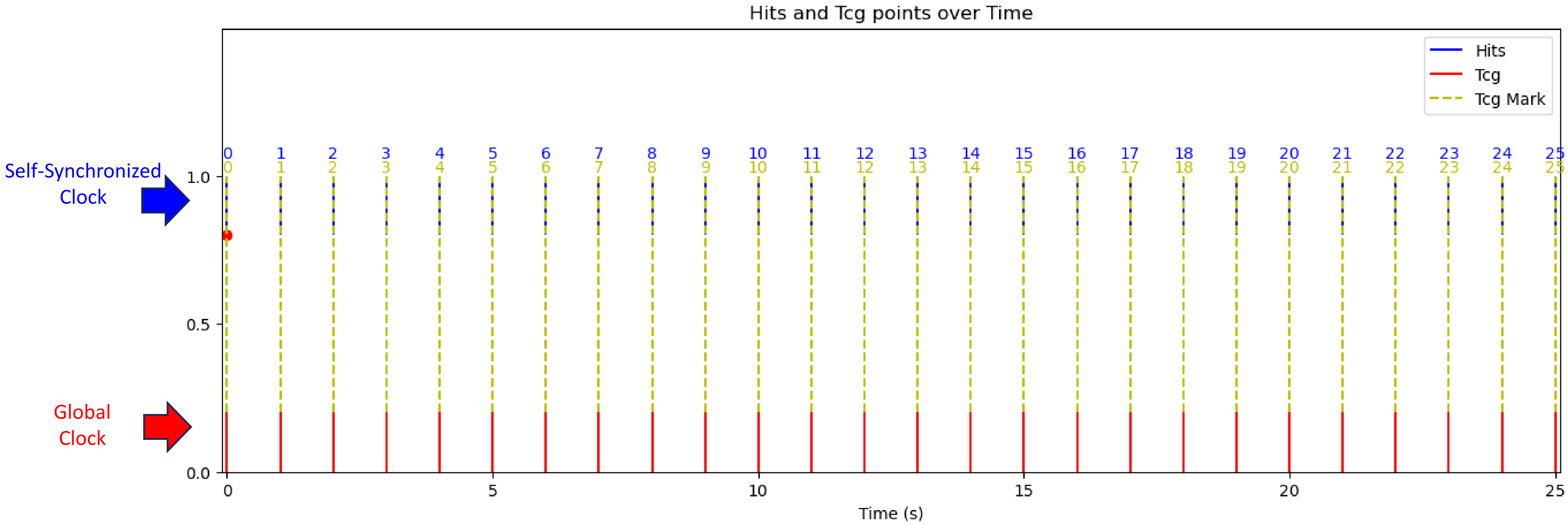
$$y = \frac{a}{x^b} \Rightarrow y = \frac{3.09}{x^{0.78}}$$

Coefficient of determination (R^2): 0.3416

Pulse-Coupled Oscillators

Simulation: Stability Results (8 agents, alpha = 0.9)

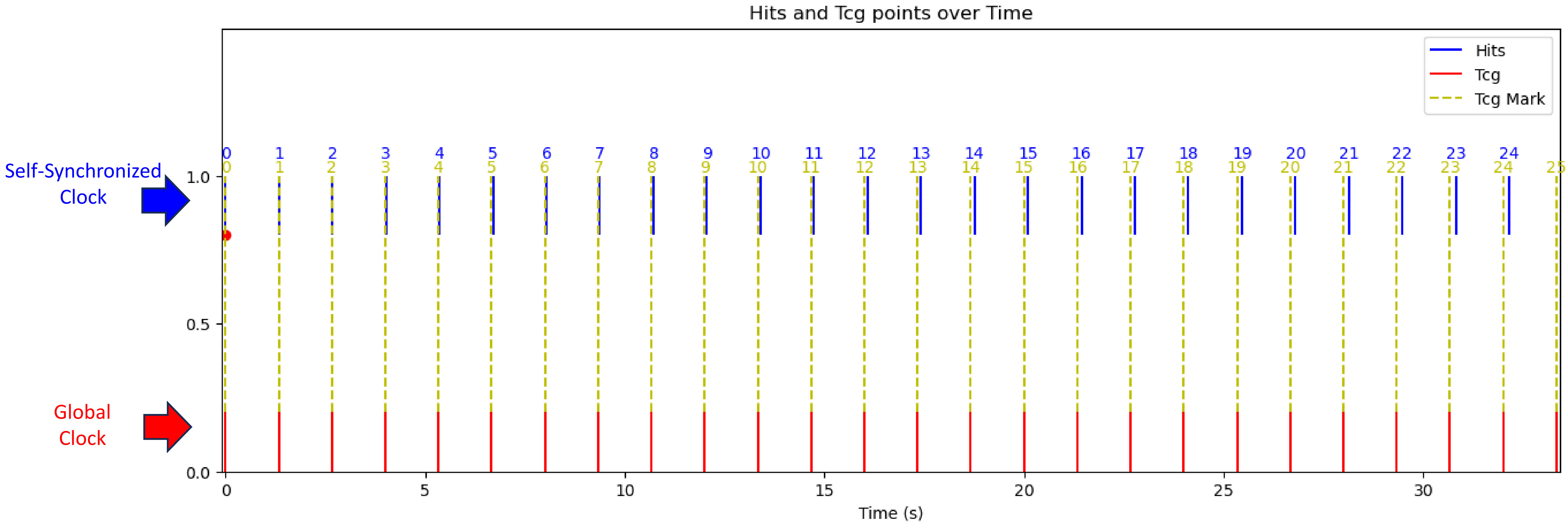
f = 1 Hz, T = 1 sec



Pulse-Coupled Oscillators

Simulation: Stability Results (8 agents, alpha = 0.9)

f = 0.75 Hz, T = 1.333333333333 sec

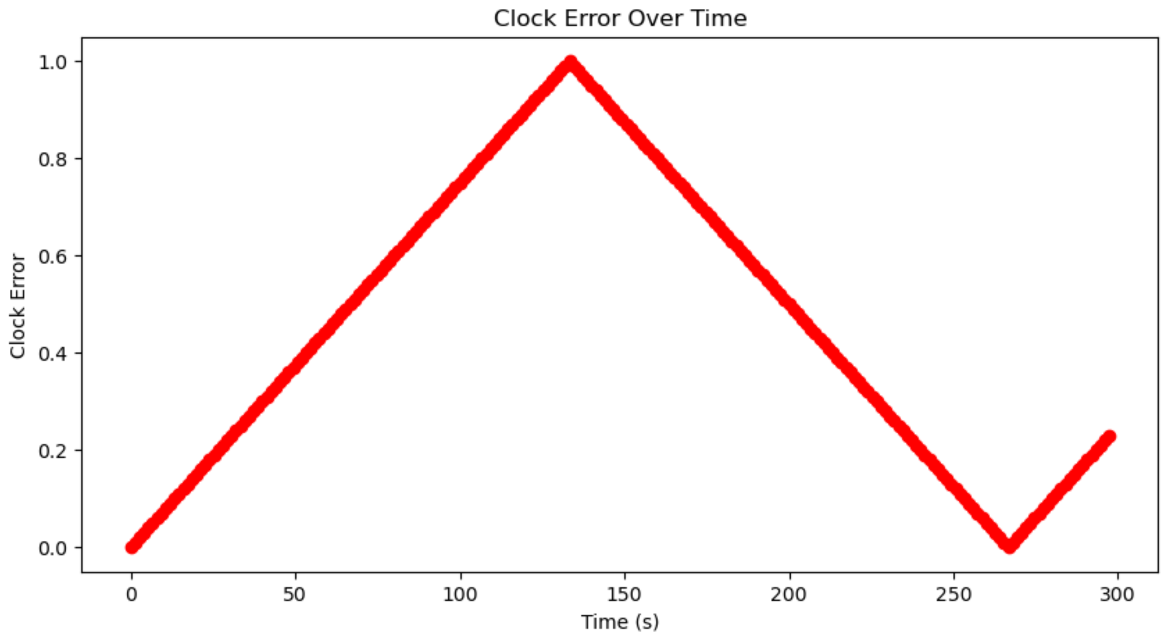


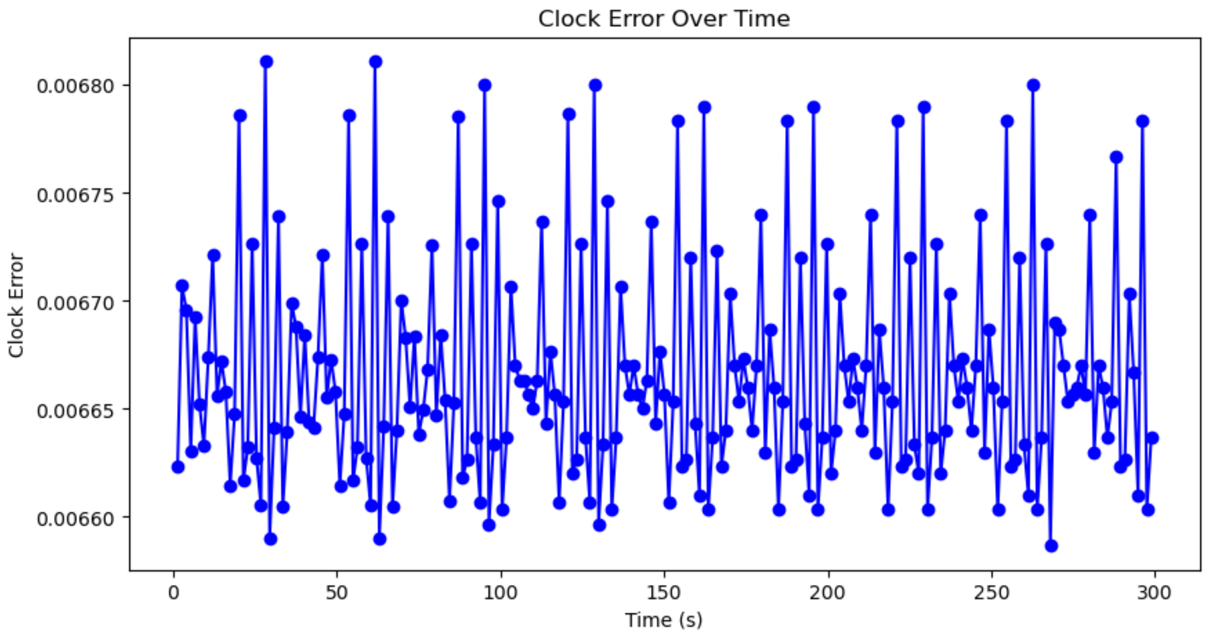
Float Imprecision. It might be fixable

Pulse-Coupled Oscillators

Simulation: Stability Results (8 agents, alpha = 0.9)

f = 0.75 Hz, T = 1.333333333333 sec

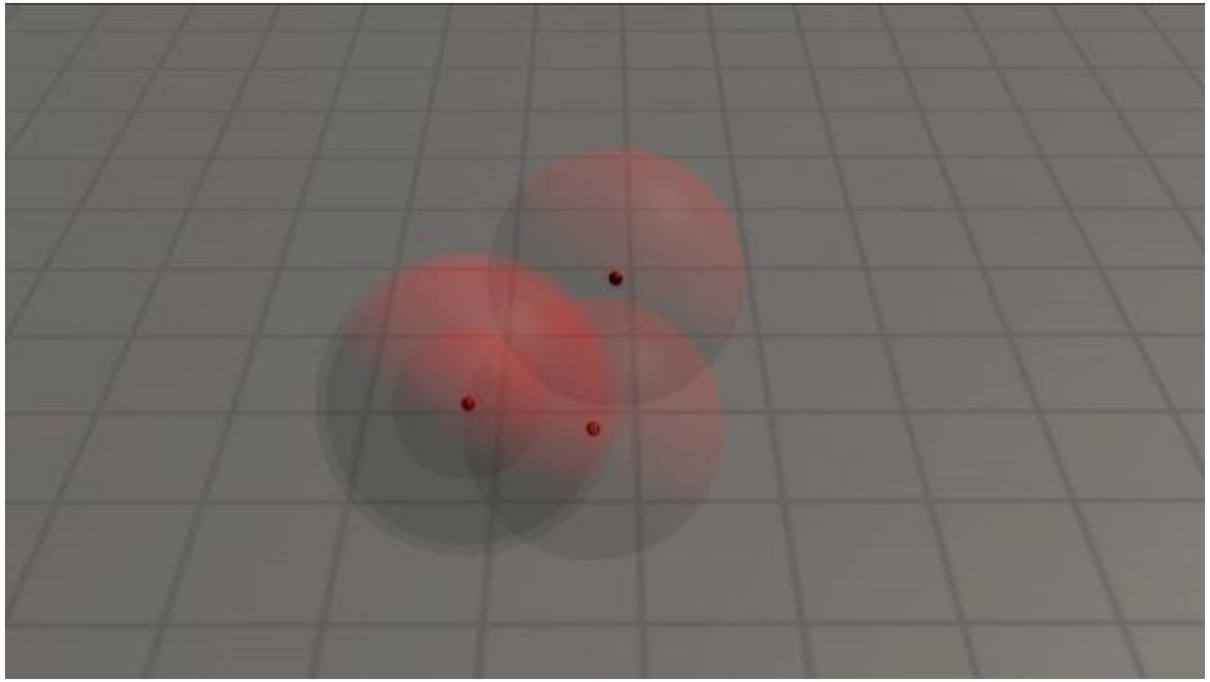




***How important are those types of errors for performers and audiences regarding a Human-Swarm IMS?

Pulse-Coupled Oscillators

A Delayed Model



Pulse-Coupled Oscillators

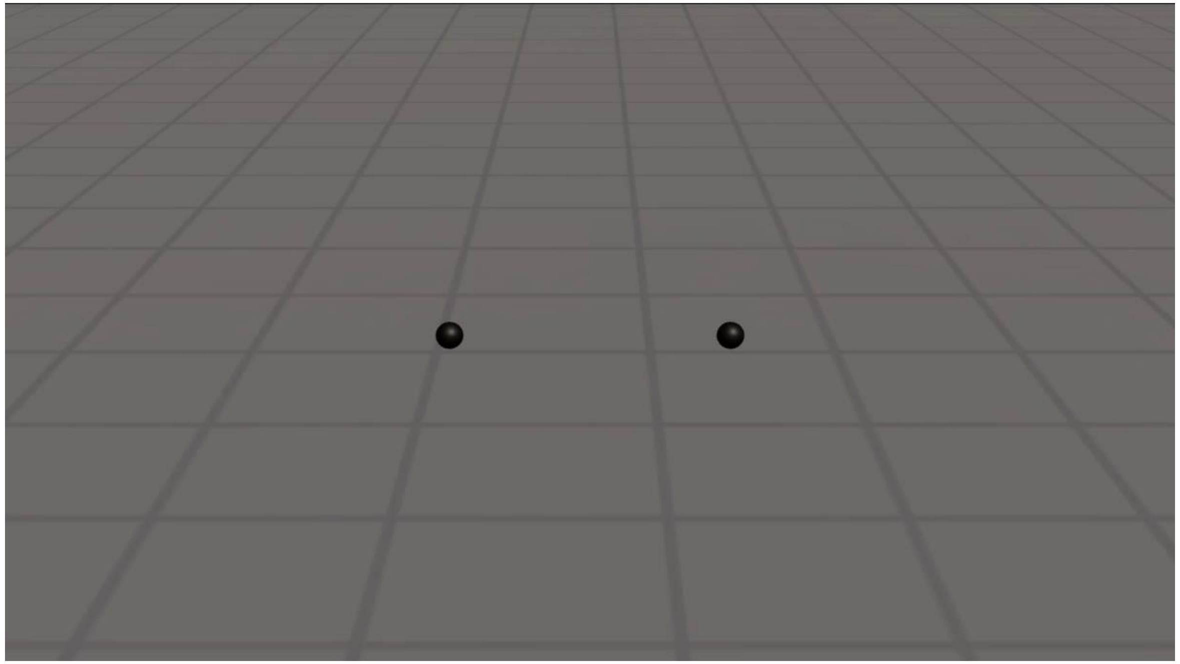
Delayed Model Algorithm

Considering T as the oscillators' period and $\boldsymbol{L_{ij}}$ *a constant latency from agent* ***i*** *to agent* ***j****, and* $\boldsymbol{L_{ij} = L_{ji}}$

When a fire event is received by agent ***i*** *from agent* ***j****:*

1. *Wait a time* $\boldsymbol{t_L = T - mod(L_{ij}, T)}$
2. *After waiting, apply:*

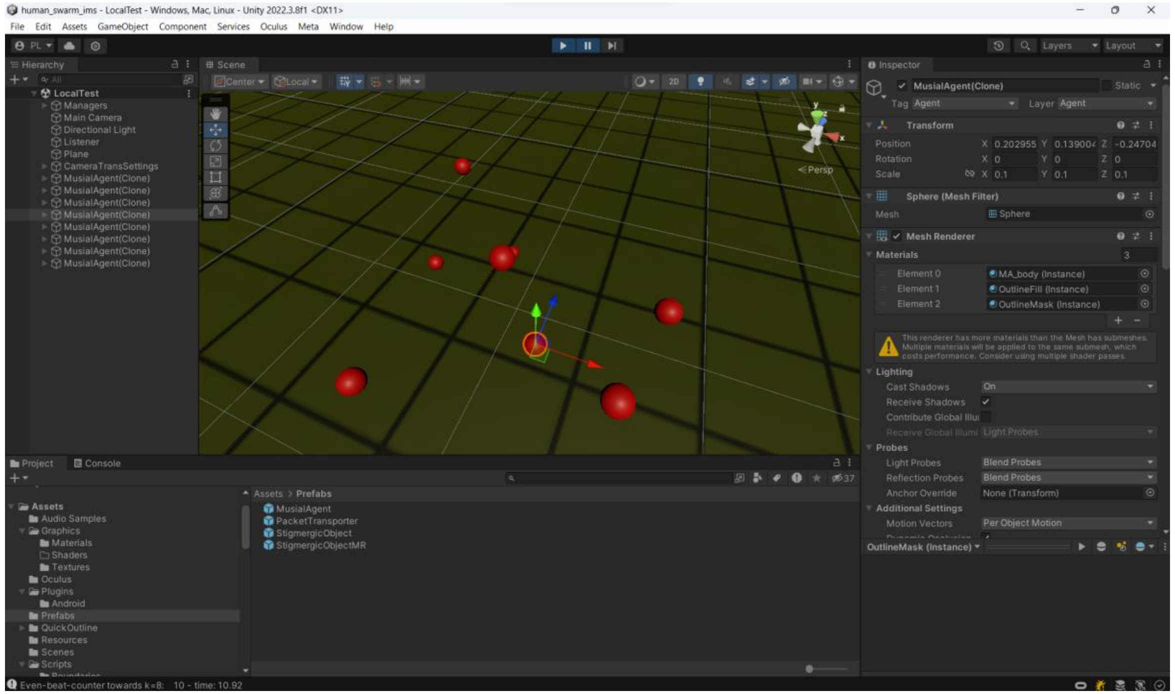
$$P(\phi) = \phi - \alpha \cdot \sin 2\pi\phi \cdot |\sin 2\pi\phi|$$



We need peer-to-peer "awareness"

Pulse-Coupled Oscillators

Implementations



Unity3D Game Engine

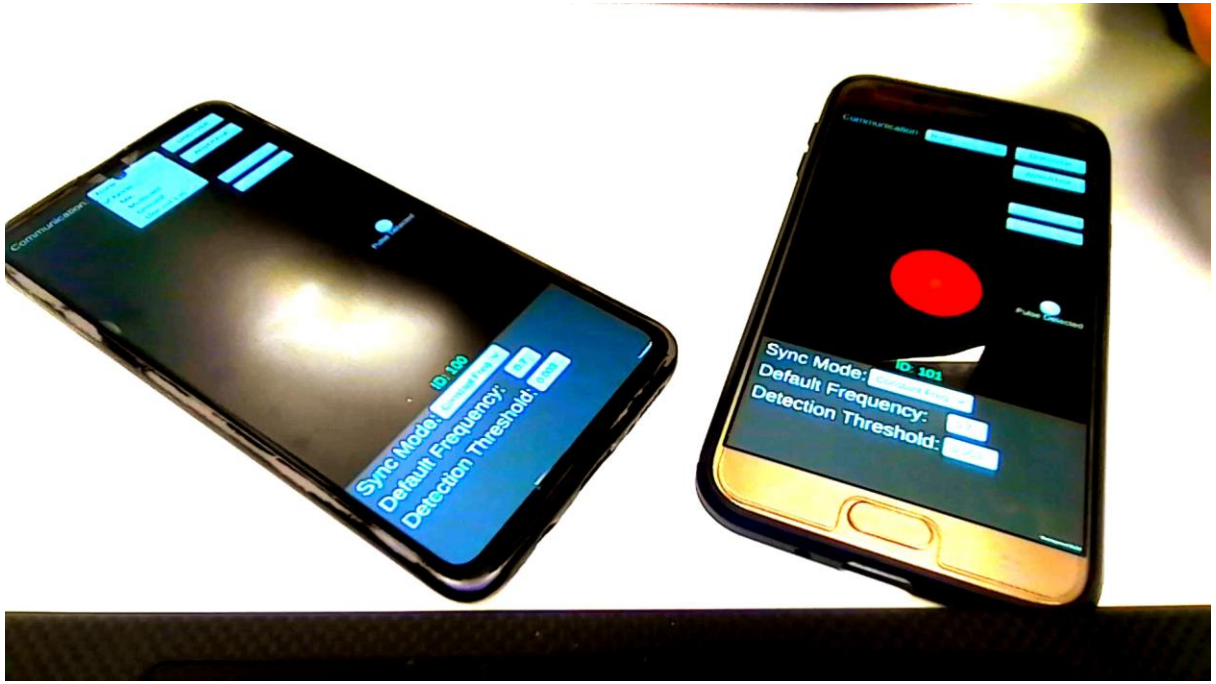
Pulse-Coupled Oscillators

Implementations

The same source code is used for all implementations through “The Artifact”:

1. Simulation (Unity3D – Windows Platform)
2. Physical Devices (Unity3D – Android Platform)
    1. Sound Trigger (Beat detection algorithm)
    2. Network Multicast Message (Broadcast)
    3. Network Unicast Message (peer-to-peer) (Non-latency and Latency model)

Android Implementation
Unicast with Latency Model



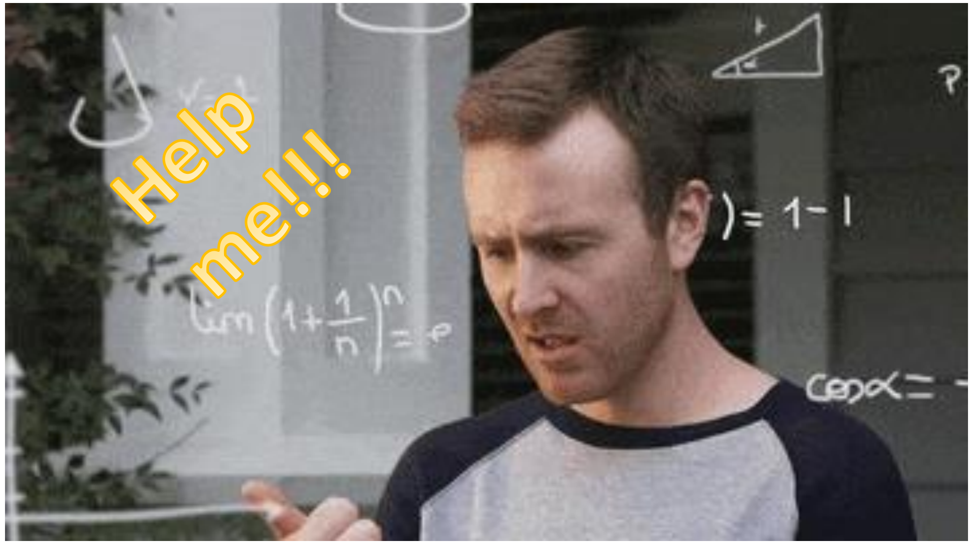
Pulse-Coupled Oscillators

Implementations

Technical Challenge:

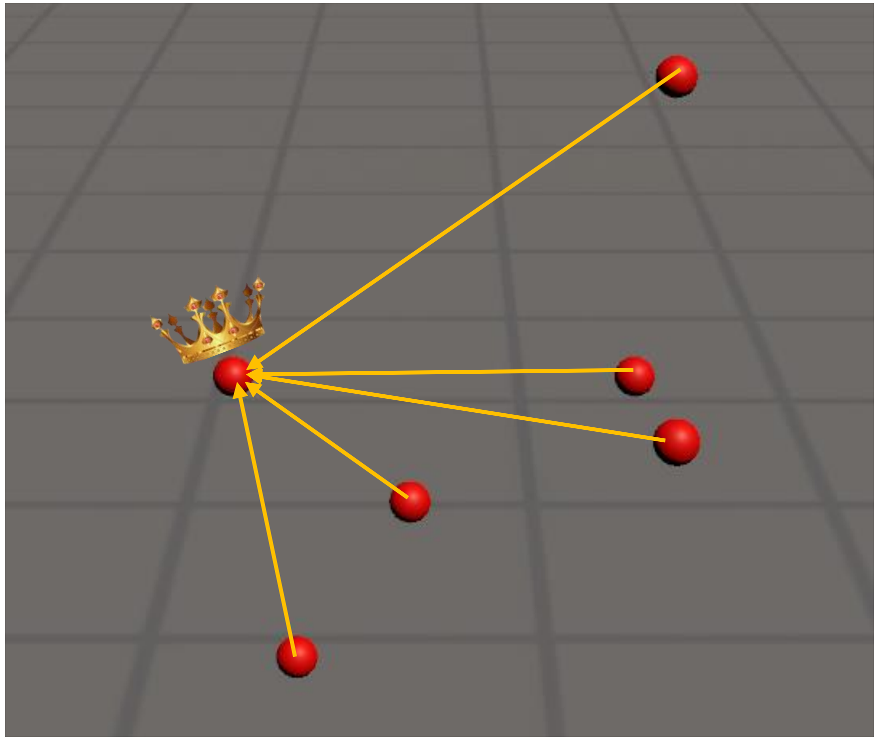
How do we calculate a good estimation of the latency L_{ij} without saturating the network or having another potential performance issue?

- Statistics (mean, median, etc.)?
- Machine Learning?



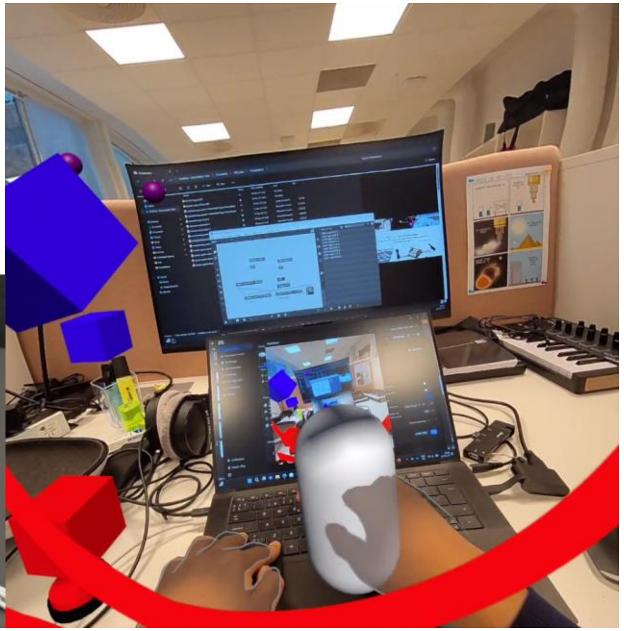
Alternative to Self-Synchronized Oscillators

Central clock selection… executed through Multi-Agent Systems Methods (e.g. voting procedures). The goal is to increase time precision and stability.



Future Work

- Complete the study of **“self-synchronized oscillators”** and **“global clock selection”** (simulation and physical devices)**.**
- Testing out swarming algorithms together with audio and visual mappings.
- Development of **virtual, physical-virtual, and physical platforms (based on “The Artifact”).**



XR. Meta Quest 3

ITS Lab. Robot Swarm

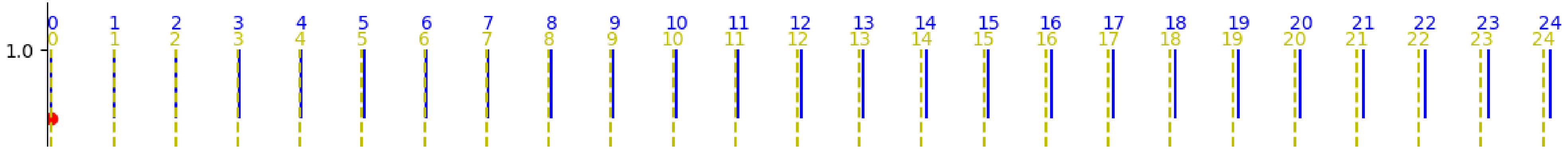
Questions….

for you all…

Questions from me to you

1. Is there a known tolerance value (in milliseconds) for synchronization in musical performances? What constitutes a "mistake"?

2. Are stability errors, as the ones shown below, perceivable or affect in some way "musical performances that rely on high temporal precision (e.g. electronic music)"?



3. What if these errors are not constantly accumulating, but happen randomly over time?

4. If we can find the model for errors like the one above, can we use these models as part of a music performance?

… now I can answer your questions ❤️