**Laboratorium Nr.1**

**CHAOTIC SYSTEMS**

1. Zapoznaj się z PyCX: Przeczytaj najpierw artykuł: PYCX\_paper.pdf. Oglądnij tutorial PYCX\_tutorial. Zapoznaj się z biblioteką: <https://github.com/hsayama/PyCX>
2. W notatniku udostępnionym przez moich studentów (Monika Darosz & Tomasz Michalec - <https://mybinder.org/v2/gh/Chimstaz/PyCXlab/master> ) znajdują się przykłady związane z uruchamianiem procedur PYCX. (Jeżeli coś nie będzie działać użyj pliku lab.ipynb)
   1. Równanie logistyczne z rysowaniem wykresu bifurkacyjnego (ds-chaotic-behavior-logistic-map.py)
   2. Dynamiką układu Lorenza (ds-Lorenz-equations.py)

oraz symulacjami związanymi z automatami komórkowymi i systemami agentowymi.

* 1. Punkty a i b dotyczą zakresu aktualnego Laboratorium. Ale zadania są inne.

1. W tym zadaniu proszę znaleźć inne odwzorowanie także dające jako wynik ciekawe wykresy bifurkacyjne (patrz Wikipedia pochodna Schwartza dla układów chaotycznych).

Text

Description automatically generated with medium confidence

1. Zrealizuj następujące zadania:
   1. Dokonaj symulacji na zaburzonym odwzorowaniu (punkt 3). Zaburzenie polega na włączeniu w odwzorowanie szumu (np. Gaussowskiego). Zwiększaj szum zwiększając sigma w rozkładzie Gaussa. Przedyskutuj wyniki. Dlaczego są takie, a nie inne?
   2. Zrób to samo z atraktorem Lorenza. Za każdym krokiem losuj z Gaussa każdy z parametrów systemu Lorenza. Wartość oczekiwana dla rozkładu jest równa parametrowi Lorenza i zmienia się sigma.
   3. Jaki ogólny wniosek płynie z wyników.

**Komentarz**: Ze względu, że oryginalna biblioteka pycxsimulator korzysta z okienek GUI tworzonych za pomocą Tkinter, nie może być ona bezpośrednio użyta w Jupyterze. W związku w tym w repozytorium <https://github.com/Chimstaz/PyCXlab> przygotowana została wersja pycxsimulator.py, która implementuje część funkcjonalności za pomocą widgetów Jupytera.