System configuration setups:

# Makefile\_setup --- make file for compiling and creating the executable sysrun.exe; To be used as: make –f Makefile\_setup

# This class contains the following files:

# demo\_setup.f90

This file is initializing the SYSTEM\_CLASS, which is the module used for reading/creating time series for the benchmark cases. It contains the subroutines/functions used to read/create dynamical variables for the case studies.

#

# This subroutine calls one of the following functions (which are part of the SYSTEM\_CLASS in syssetup.f90:

# IF (benchmark == 1) then

# CALL SETUP\_DRIVER(frmt, Nframes, Ndim, Natoms, Offset, iStart, iSkip, iStop, debug, benchmark, Ax, Cxy, Sigma\_x, Sigma\_y)

# else

# CALL SETUP\_DRIVER(Nframes, Ndim, Natoms, Offset, iStart, iSkip, iStop, debug, benchmark)

# endif

# subroutine mdTrajEnergy(Istart, iSkip, iStop) --- It reads the energy from a Molecular dynamics simulations for each frame

# subroutine mdTrajPDBVmd(Istart, iSkip, iStop) --- It reads the molecular dynamics trajectory in PDB format as created using VMD software, each residue is represented by

# --- CA atom coordinates, or

# --- Collective coordinates representing essential degrees of freedom for each amino acid computed using Machine Learning Approach

# subroutine Benchmark1(Ax,Cxy,Sigmax,Sigmay,iStart,iSkip,iStop) --- Benchmark 1

!! The Multivariate Gaussian Process

!!

# subroutine Benchmark2(Istart, Iskip, Istop):

!! Mutual information in this Experiment:

!! Y = X+Z

!! X uniformly distributed in [-1/2:1/2]

!! Z uniformly distributed in [-a/2:a/2]

!! f\_y(y) = convolution(X,Z)

!! I(X;Y) = h(Y) - h(Z)

!! In Nats: I(X;Y) = a/2 - ln(a) if a <= 1; otherwise I(X;Y)=1/(2\*a) if a>=1.

!! In Bits: I(X;Y) = [a/2 - ln(a)]/ln(2) if a <= 1; otherwise I(X;Y)=[1/(2\*a)]/ln(2) if a>=1.

!! Example: a=0.5, then I(X;Y) = 1.36067 (bits); h(Y)=0.49075 (bits)

!!

# subroutine Benchmark3(Istart, Iskip, Istop)

!! Mutual information in the this Experiment:

!! Y = X+Z

!! X uniformly distributed in [-1/2:1/2]

!! Z1 uniformly distributed in [-a/2:a/2]

!! Z2 uniformly distributed in [-a/4:a/4]

!! f\_y(y) = convolution(X,Z1,Z2)

!!

# subroutine Benchmark4(Istart, Iskip, Istop)

!!

!! Mutual information in the this Experiment:

!! Throwing a fair die:

!! X -> Variable representing the value of the face closeer to us

!! Y -> Variable representing the value of the top of the die

!! Possible values are 1, 2, 3, 4, 5, 6

!! Theoretical value:

!! I(T,B) = H(B) - H(B|T) = log(6) - log(4) = log(3) - 1

!! H(B) = 2.584962501; H(B|T) = 2.0

!!

# subroutine Benchmark5(Istart, Iskip, Istop)

!!

!! Mutual information in the this Experiment:

!! Flipping a fair Coin:

!! X -> Variable representing the value of the bottom of the coin

!! Y -> Variable representing the value of the top of the coin

!! Head = 1 and Tail = 0

!! Theoretical value:

!! I(T,B) = H(B) - H(B|T) = log(2) - 0 = 1

!!

# subroutine write\_xyz() --- It writes in output the real variable trajectory coordinates

# subroutine write\_xyzs() --- It writes in output the symbolic trajectory coordinates

# subroutine read\_xyz -- It reads trajectory coordinates

# subroutine Allocate\_System(Benchmark) --- It allocates memory for global variables of the SYSTEM\_CLASS

# subroutine deAllocate\_System() --- It frees allocated memory for global variables of the SYSTEM\_CLASS

#

# The Input Parameters of this MODULE incude:

Time to start trajectory - iStart

Number of steps to skip - iSkip

Time to stop trajectory - iStop

Offset residue numbering - Offset

Number of time series - Natoms

Dimensionality of the problem - Ndim

Set debugging flag value - Debug

Set Benchmark flag value - Benchmark

Chose the format of Input/Output data – frmt

A bash shell script is given below:

!#/bin/bash

root=../sifm-master

natoms=2

ndim=1

Benchmark=1

Debug=1

iStart=10001

iSkip=1

iStop=20000

Offset=0

Ax=0.5

Cxy=1.0

Sigma\_x=0.5

Sigma\_y=1.0

frmt=’csv’

if [ $Benchmark == 1 ]; then

${root}/sysrun.exe ${iStart} ${iSkip} ${iStop} ${Offset} ${natoms} ${ndim} ${Debug} ${Benchmark} ${Ax} ${Cxy} ${Sigma\_x} ${Sigma\_y} ${frmt}

else

${root}/sysrun.exe ${iStart} ${iSkip} ${iStop} ${Offset} ${Natoms} ${Ndim} ${Debug} ${Benchmark} ${frmt}

fi