Contents

Simulation and modeling of natural processes

by University of Geneva

About this Course

This course gives you an introduction to modeling methods and simulation tools for a wide range of natural phenomena. The different methodologies that will be presented here can be applied to very wide range of topics such as fluid motion, stellar dynamics, population evolution, ... This course does not intend to go deeply into any numerical method or process and does not provide any recipe for the resolution of a particular problem. It is rather a basic guideline towards different methodologies that can be applied to solve any kind of problem and help you pick the one best suited for you. The assignments of this course will be made as practical as possible in order to allow you to actually create from scratch short programs that will solve simple problems. Although programming will be used extensively in this course we do not require any advanced programming experience in order to complete it.

Show less

- 

  Taught by:[Bastien Chopard](), Full Professor

  Computer Science

Syllabus

Week 1

Introduction and general concepts

This module gives an overview of the course and presents the general ideas about modeling and simulation. An emphasis is given on ways to represent space and time from a conceptual point of view. An insight of modeling of complex systems is given with the simulation of the grothw and thrombosis of giant aneurysms. Finally, a first class of modeling approaches is presented: the Monte-Carlo methods.

7 videos, 1 reading

1. **Reading:** Course slides
2. **Video:** Objectives and background
3. **Video:** Modeling and Simulation
4. **Video:** Modeling Space and Time
5. **Video:** Example of bio-medical Modeling
6. **Video:** Monte Carlo methods I

7. **Video:** Monte Carlo methods II
8. **Video:** Monte Carlo methods III

Graded: Introduction and general concepts

Week 2

Introduction to programming with Python 3

This module intends to provide the most basic concepts of high performance computing used for modeling purposes. It also aims at teaching the basics of Python 3 which will be the programming language used for the quizzes in this course.

Graded: Introduction to programming with Python 3
Graded: Project - Piles
Graded: Project - Class:Integration

Week 3

Dynamical systems and numerical integration

Dynamical systems modeling is the principal method developed to study time-space dependent problems. It aims at translating a natural phenomenon into a mathematical set of equations. Once this basic step is performed the principal obstacle is the actual resolution of the obtained mathematical problem. Usually these equations do not possess an analytical solution and advanced numerical methods must be applied to solve them. In this module you will learn the basics of how to write mathematical equations representing natural phenomena and then how to numerically solve them.

Graded: Dynamical systems and numerical integration
Graded: The implicit Euler scheme
Graded: Project - Lotka-Volterra

Week 4

Cellular Automata

This module defines the concept of cellular automata by outlining the basic building blocks of this method. Then an insight of how to apply this technique to natural phenomena is given. Finally the lattice gas automata, a subclass of models used for fluid flows, is presented.

Graded: Cellular Automata
Graded: Project - The Parity Rule

Week 5

Lattice Boltzmann modeling of fluid flow

This module provides an introduction to the lattice Boltzmann method, a powerful tool in computational fluid dynamics. The lesson is practice oriented and show, step by step, how to write a program for the lattice Boltzmann method. The program is used to showcase an interesting problem in fluid dynamics, the simulation of a vortex street behind an obstacle.

Graded: Lattice Boltzmann modeling of fluid flow
Graded: Project - Flow around a cylinder
Graded: Collision Invariant

Week 6

Particles and point-like objects

A short review of classical mechanics, and of numerical methods used to integrate the equations of motions for many interacting particles is presented. The student will learn that the computational expense of resolving all interaction between particles poses a major obstacle to simulating such a system. Specific algorithms are presented to allow to cut down on computational expense, both for short-range and large-range forces. The module focuses in detail on the Barnes-Hut algorithm, a tree algorithm which is popular a popular approach to solve the N-Body problem.

Graded: Particles and point-like objects
Graded: Project - Barnes-Hut Galaxy Simulator

Week 7

Introduction to Discrete Events Simulation

In this module, we will see an alternative approach to model systems which display a trivial behaviour most of the time, but which may change significantly under a sequence of discrete events. Initially developed to simulate queue theory systems (such as consumer waiting queue), the Discrete Event approach has been apply to a large variety of problems, such as traffic intersection modeling or volcanic hazard predictions.

Graded: Introduction to Discrete Event Simulation
Graded: Project - Simple modelling of traffic lights

Week 8

Agent based models

Agent Based Models (ABM) are used to model a complex system by decomposing it in small entities (agents) and by focusing on the relations between agents and with the environment. This approach is derived from artificial intelligence research and is currently used to model various systems such as pedestrian behaviour, social insects, biological cells, etc.

Graded: Agent based models
Graded: Project - Multi-agents model

Week 3 Quiz

Week 3 Quiz 1

Friday, January 20, 2023
7:40 PM

Skip to Main Content

SEARCH IN COURSE
Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 3
- Dynamical systems and numerical integration

PreviousNext

- **Dynamical systems and numerical integration**
- **General introduction to dynamical systems**
- **Two examples of dynamical systems**
- **Balance equations**
- **Ordinary differential equations**
- **Numerical integration of partial differential equations**
- **Assignments**
- **Quiz:** Dynamical systems and numerical integration
  4 questions
- **Quiz:** The implicit Euler scheme
  9 questions
- **Quiz:** Project - Lotka-Volterra
  4 questions

Dynamical systems and numerical integration

Quiz30 minutes • 30 min

Submit your assignment

Due February 5, 11:59 PM ISTFeb 5, 11:59 PM IST
Attempts 100 every 8 hours
Try again

Receive grade

To Pass 60% or higher

Your grade

50%

View Feedback

We keep your highest score

Like

Dislike

Report an issue

Back

Dynamical systems and numerical integration

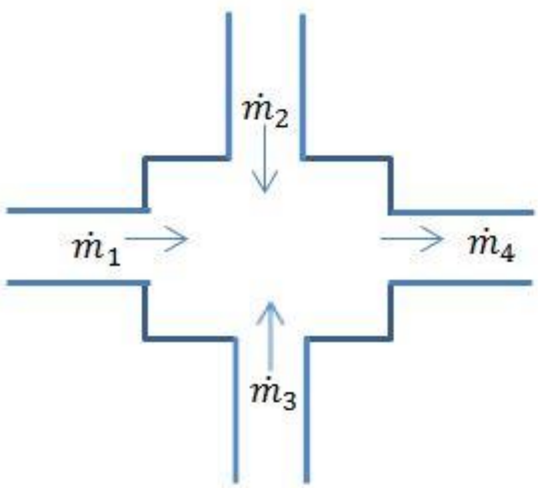
Graded Quiz. • 30 min. • 8 total points available.8 total points

DueFeb 5, 11:59 PM IST

1.

Question 1

Pipe flow balance:



You are a mechanical engineer and you have been asked to design a water pipe system. You know by your formation that the mass rate $\cdot m \cdot$ across a flat surface is well estimated by ρVA where V is the average fluid velocity on the surface of area A. The density ρ is constant and is the density of the water.

Your colleagues impose some constraints on the system, i.e. $1= 2= 3=1A1=A2=A3=1\ 2m2$ because all input pipe diameters are the same in order to reduce manufacturing cost. The flow comes in and goes out at constant average velocities $1=0.5V1=0.5\ /\ m/s$, $2=0.5V2=0.5\ /\ m/s$, $3=1V3=1\ /\ m/s$ and $4=1V4=1\ /\ m/s$.

You also know that the design only need to work in a steady flow environment, i.e. the state of the system do not change with time. It is also clear to you that in such a simplified system, there is no destruction or creation of mass.

Use a balance equation (i.e. a simplified continuity equation) to find the area $4A4$ (in meter square) that the design needs to work successfully.

Find the answer of the previous question with 4 digits of accuracy.

Answer: 2

=> m1 + m2 + m3 = m4

=> A4 = A * (V1+V2+V3) / V4 = 2A = 2*1 = 2

2 points

2.

Question 2

Assume that the order of convergence of your numerical method is equal to 3. If you divide your discretization step by a factor 2, by what factor would you divide the error?

Answer: 8

=> E ~ O(dt^k); where "k" is order of convergence

=> for k=3, dt=dt/2 => E' ~ O((dt/2)^3) ~ O(dt^3/8) ~ 1/8 * E

3 points

3.

Question 3

Consider the differential equation: = + $dtdy=t+y$ with $(0)= 0y(0)=y0$. If we integrate it with the Explicit Euler Scheme, and $\Delta = 0.1\Delta t=0.1$, then which equation in the following is false?

2 points

$(0.1)= 0+0.1(0.1+ 0)y(0.1)=y0+0.1(0.1+y0)$

$(0.3)- (0.2)0.1=0.2+ (0.2)0.1y(0.3)-y(0.2)=0.2+y(0.2)$

$(+\Delta)= ( )+\Delta [+ ( )]y(tn+\Delta t)=y(tn)+\Delta t[tn+y(tn)]$

Answer: Option 1

=> y(tn+dt) = y(tn) + dt * (tn + y(tn)) OR [y(tn+dt) - y(tn)]/dt = tn + y(tn)

=> Therefore FIRST option is incorrect. It should have been: y(0+0.1) = y0 + 0.1 * (**0** + y0)

4.

Question 4

Refer to the slides of page 72 and try to figure out which points would be used for the space discretization of $\partial 4\ \ \partial\ 4 \partial t 4 \partial 4 Ci$?

1 point

$+1,\ ,\ -1 Ci{+}1, Ci, Ci{-}1$

$+2,\ +1,\ ,\ -1,\ -2 Ci{+}2, Ci{+}1, Ci, Ci{-}1, Ci{-}2$

$+3,\ +2,\ +1,\ ,\ -1,\ -2,\ -3 Ci{+}3, Ci{+}2, Ci{+}1, Ci, Ci{-}1, Ci{-}2, Ci{-}3$

Answer: Option 2

d^2 = Ci+1 - 2Ci + Ci-1

=> d^4 = (Ci+2 - 2Ci+1 + Ci) - 2 * (Ci+1 - 2Ci + Ci-1) + (Ci - 2Ci-1 + Ci-2)
=> Coeff used: Ci+2, Ci+1, Ci, Ci-1, Ci-2

Week 3 Quiz 2

Friday, January 20, 2023
7:54 PM

Skip to Main Content

SEARCH IN COURSE
Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 3
- The implicit Euler scheme

PreviousNext

- **Dynamical systems and numerical integration**
- **General introduction to dynamical systems**
- **Two examples of dynamical systems**
- **Balance equations**
- **Ordinary differential equations**
- **Numerical integration of partial differential equations**
- **Assignments**
- **Quiz:** Dynamical systems and numerical integration
  4 questions
- **Quiz:** The implicit Euler scheme
  9 questions
- **Quiz:** Project - Lotka-Volterra
  4 questions

The implicit Euler scheme

Quiz30 minutes • 30 min

Submit your assignment

Due February 5, 11:59 PM ISTFeb 5, 11:59 PM IST
Attempts 3 every 8 hours
Start assignment

Receive grade

To Pass 50% or higher

Your grade

-Not available
Like
Dislike

Report an issue

Back

The implicit Euler scheme

Graded Quiz. • 30 min. • 16 total points available.16 total points

DueFeb 5, 11:59 PM IST

1.

Question 1

Let take back the example in the course in the slide page 65 (consult the offline slides). With a simple calculator computed the state evolution of the system. Use $0=1s0=1$ and $0=0t0=0$ for the numerical evaluation.

What is the value of $4s4$ when using the explicit scheme and Δ $=0.05\Delta t=0.05$?

Find the answer to the question with at least 3 digits of accuracy.

8 points

Answer: 0.0625

Sj+1 = Sj * (1-10*0.05) = Sj * 0.5 => Sk = S0/(2^k) => S4 = 1/16

2.

Question 2

What is the value of $4s4$ when using the implicit scheme and Δ $=0.05\Delta t=0.05$?

Find the answer to the question with at least 3 digits of accuracy.

1 point

Answer: 0.1975

Sj+1 = Sj/(1+10*0.05) = Sj/(3/2) => Sk = S0 / (3/2)^k => S4 = 16/81

3.

Question 3

What is the value of $4s4$ when using the explicit scheme and Δ $=0.1\Delta t=0.1$?

Find the answer to the question with at least 3 digits of accuracy.

1 point

Answer: 0

Sj+1 = Sj * (1 - 10*0.1) = 0

4.

Question 4

What is the value of $4s4$ when using the implicit scheme and Δ =$0.1\Delta t$=0.1?

Find the answer to the question with at least 3 digits of accuracy.

1 point

Answer: 0.0625

Sj+1 = Sj /(1+10*0.1) = Sj/2 => Sk = S0/(2^k) => S4 = 1/16

5.

Question 5

What is the value of $4s4$ when using the explicit scheme and Δ =$0.2\Delta t$=0.2?

Find the answer to the question with at least 3 digits of accuracy.

1 point

Answer: 1

Sj+1 = Sj * (1 - 10*0.2) = -Sj => Sk = (-1)^k * S0 => S4 = 1

6.

Question 6

What is the value of $4s4$ when using the implicit scheme and Δ =$0.2\Delta t$=0.2?

Find the answer to the question with at least 3 digits of accuracy.

1 point

Answer: 0.012345

Sj+1 = Sj /(1+10*0.2) = Sj/3 => Sk = S0/(3^k) => S4 = 1/81

7.

Question 7

What is the value of $4s4$ when using the explicit scheme and Δ =$0.25\Delta t$=0.25?

Find the answer to the question with at least 3 digits of accuracy.

1 point

Answer: 5.0625

Sj+1 = Sj * (1 -10*0.25) = (-3/2) * Sj => Sk = (-3/2)^k * s0 => S4 = 81/16

8.

Question 8

What is the value of $4s4$ when using the implicit scheme and Δ $=0.25\Delta t=0.25$?

Find the answer to the question with at least 3 digits of accuracy.

1 point

Answer: 0.00666389

Sj+1 = Sj/(1+10*0.25) = (2/7) * Sj => Sk = (2/7)^k * S0 => S4 = 16/(49*49)

9.

Question 9

Can you guess the time step stability limit of the explicit method in this situation? That is the step time Δ Δt which the solution starts to oscillate and does not converge to a unique solution.

1 point

Answer: 0.2

At dt = 0.2, Explicit values start to oscillate as 1 and -1
At dt -.025, Explicit values not only oscialte between positive and negative value but also become increasing large!

Coursera Honor Code [Learn more]

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

SubmitSave draft

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/aoRxY/the-implicit-euler-scheme/attempt>

Week 3 Quiz 3

Friday, January 20, 2023
9:01 PM

Skip to Main Content

SEARCH IN COURSE

Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 3
- Project - Lotka-Volterra

PreviousNext

- **Dynamical systems and numerical integration**
- **General introduction to dynamical systems**
- **Two examples of dynamical systems**
- **Balance equations**
- **Ordinary differential equations**
- **Numerical integration of partial differential equations**
- **Assignments**
- **Quiz:** Dynamical systems and numerical integration
  4 questions
- **Quiz:** The implicit Euler scheme
  9 questions
- **Quiz:** Project - Lotka-Volterra
  4 questions

Project - Lotka-Volterra

Quiz30 minutes • 30 min

Submit your assignment

Due February 5, 11:59 PM ISTFeb 5, 11:59 PM IST

Attempts 3 every 8 hours

Resume assignment

Receive grade

To Pass 60% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Project - Lotka-Volterra

Graded Quiz. • 30 min. • 4 total points available.4 total points

DueFeb 5, 11:59 PM IST

1.

Question 1

The attached code "lotkaVolterra.py" is a program on Lotka-Volterra model mentioned in the course. In that program, time is discretized by Explicit-Euler method and Runge-kutta2 method respectively. Explicit-Euler model has been explained fully in the course. The second order of Runge-Kutta model is expressed as below:

1=Δ (,) $k1=\Delta t f(\phi n,tn)$

2=Δ (+ 12, +Δ 2) $k2=\Delta t f(\phi n+2k1,tn+2\Delta t)$

+1= + 2 $\phi n+1=\phi n+k2$

Δ = +1− $\Delta t=tn+1-tn$

(A more detailed explanation about Runge-Kutta2 is shown at http://lpsa.swarthmore.edu/NumInt/NumIntSecond.html)

If necessary, install the dependencies of the program (numpy and matplotlib). Read and run the program. The first figure generated plots the population of antelopes and cheetahs versus time by Runge-Kutta2 model and Explicit-Euler model respectively. The second figure shows the population of cheetahs versus the population of antelopes in time. Observe these two figures, and select the phenomena caused by large numerical error.

lotkaVolterra

PY File

1 point

In the first figure, the maximum population of antelopes computed by Runge-Kutta2 method keeps a constant value.

In the first figure, the maximum population of antelopes computed by Explicit-Euler method increases with time.

In the seceond figure, the population of cheetahs computed by Runge-Kutta2 method changes with the population of antelopes periodically.

In the second figure, the population of cheetahs computed by Explicit-Euler model expands each period.

2.

Question 2
In this program, Runge-Kutta2 model with N=16 000 is used as the exact solution to compute the error of Runge-Kutta2 method and Explicit-Euler method.

Try to find out the error of Runge-Kutta2 method at N = 4000 (with 6 digits of accuracy) from the output of the program.

1 point

Answer: 0.00010598640665514971

3.

Question 3
The error of Explicit-Euler method at N = 4000 is : (6 digits of accuracy)

1 point

Answer: 0.001287294870538054

4.

Question 4
According to the third figure, which order is the Runge-Kutta2 scheme of?

1 point

Answer: 2

Coursera Honor Code [Learn more]

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Submit **Save draft**

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/WnRtB/project-lotka-volterra/attempt>

Week 4 Quiz

Monday, January 23, 2023
10:49 PM

Skip to Main Content

SEARCH IN COURSE
Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 4
- Cellular Automata

PreviousNext

- **Cellular Automata**
- **Definition and basic concepts**
- **Historical background**
- **A mathematical abstraction of reality**
- **Cellular Automata Models for Traffic**
- **Complex systems**
- **Lattice-gas Automata**
- **Assignments**
- **Quiz:** Cellular Automata
  4 questions
- **Quiz:** Project - The Parity Rule
  3 questions
- **Reading:** ReadingNotes on the Parity Rule
  . Duration: 10 minutes10 min

Cellular Automata

Quiz30 minutes • 30 min

Submit your assignment

Due February 12, 11:59 PM ISTFeb 12, 11:59 PM IST
Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 60% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Cellular Automata

Graded Quiz. • 30 min. • 7 total points available.7 total points

DueFeb 12, 11:59 PM IST

1.

Question 1

For the parity rule example, what is the correct lookup table?

The index is computed as follows:

index = s_(i-1,j) + 2*s_(i+1,j) + 4*s_(i,j-1) + 8*s_(i,j+1)

2 points

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Rule(index)	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Rule(index)	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Rule(index)	1	0	0	1	0	1	1	0	0	1	1	0	1	0	0	1

Answer:

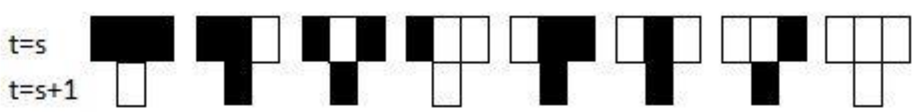
index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Rule(index)	0	1	1	0	1	0	0	1	1	0	0	1	0	1	1	0

None of the above.

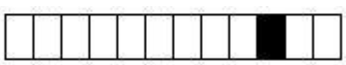
2.

Question 2

Wolfram cellular automata rule 110 specifies the color of the cell on the next time step depending on the color of its neighbors.



At time =0t=0 the initial condition is:



Using fixed BC on both side of the domain such that color of a cell outside of the domain is always white, what is the state of the system at =8t=8?

2 points

Answer: t=8

Solution: The current cell with two of its neighbors can be modeled as 3 bits with dark state represented as 1.

=> Next state is "1" (Dark) when pattern is 001,010,011,101,110 else it is "0" (White) for other patterns

So when we iterate with given initial position of a dark cell at 10th position (1..12):

[0]000000000100[0]

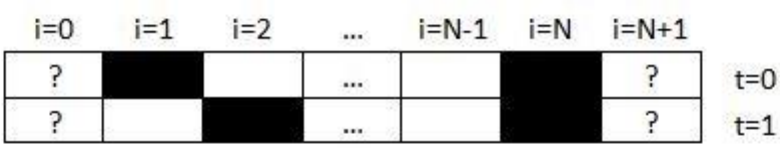
we get the following pattern at t=8

011111110100

3.

Question 3

Note that in this exercise a black cell corresponds to a cell with a car and that the flow of cars is from left to right. Also, BC stands for boundary condition. For a one-dimensional car simulation to be well defined a BC must be applied at each end of the domain. This is illustrated below where computational information is missing at the cell positions =0$i=0$ and = +1$i=N+1$.

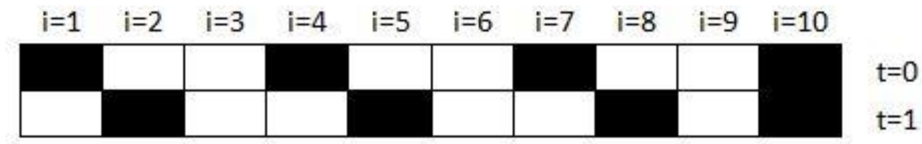


The BC can be constant in time, i.e. the cells at =0$i=0$ and = +1$i=N+1$ may always be with a car or never be with a car. The BC can as well be time dependent where the presence of a car depends on the time step number and/or others rules.

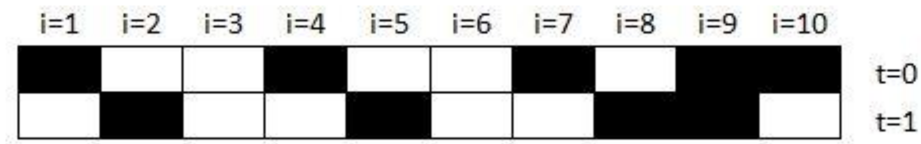
Which time evolution does not correspond to the traffic rule?

2 points

Fixed BC on the left-side domain (with no car) and fixed BC on the right-side domain (with a car).

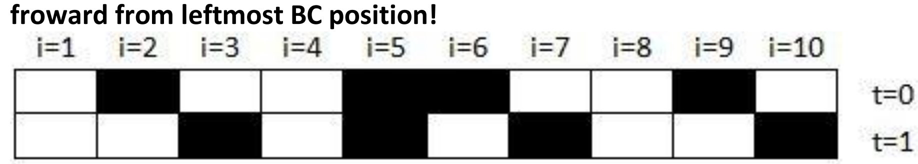


Fixed BC on the left-side domain (with a car) and fixed BC on the right-side domain (with no car).

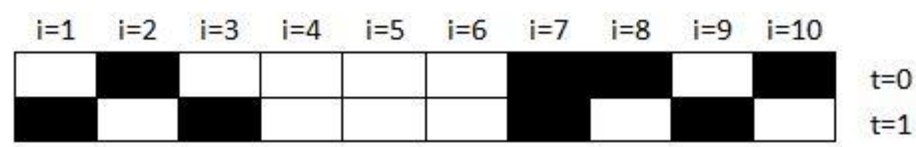


Time dependent BC on the left-side domain (with a car only if the time step is even) and fixed BC on the right-side domain (with no car).

Answer: At t=1, first cell should have a car as previous state would have had a car to move froward from leftmost BC position!



Periodic BC on both domain sides.



4.

Question 4

Which statement is false concerning LGA and HPP?

1 point

LGA is a fully discrete dynamical system: fluid is discretized as lots of particles; Space is discretized into regular lattices; time is discretized into time steps.

In LGA, The lattice sites change their states synchronously at discrete time steps.

Answer: 1 particle per direction!
In HPP, each lattice site only can store up one moving particle.

In HPP, there are 2^4=16 possible states for each lattice site.

Coursera Honor Code [Learn more]

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Submit **Save draft**

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/mQPhB/cellular-automata/attempt>

Skip to Main Content

SEARCH IN COURSE
Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 4
- Project - The Parity Rule

PreviousNext

- **Cellular Automata**
- **Definition and basic concepts**
- **Historical background**
- **A mathematical abstraction of reality**
- **Cellular Automata Models for Traffic**
- **Complex systems**
- **Lattice-gas Automata**

- **Assignments**
- **Quiz:** Cellular Automata
  4 questions
- **Quiz:** Project - The Parity Rule
  3 questions
- **Reading:** ReadingNotes on the Parity Rule
  . Duration: 10 minutes10 min

Project - The Parity Rule

Quiz30 minutes • 30 min

Submit your assignment

Due February 12, 11:59 PM ISTFeb 12, 11:59 PM IST

Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 60% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Project - The Parity Rule

Graded Quiz. • 30 min. • 4 total points available.4 total points

DueFeb 12, 11:59 PM IST

1.

Question 1

An incomplete parity rule simulator is provided for this project, i.e. "parityRule.py". Also, three monochrome bitmap files are also provided, i.e. "image1.bmp", "image2.bmp" and "image3.bmp".

parityRule

ZIP File

This simulator has two inputs, the file name of a monochrome ".bmp" file ("imageName" at the line 24) and the number of iterations to execute the parity rule on the image ("maxIter" at the line 25). One important output of the simulator is the number of white pixels in the final image which is printed to the screen.

Your task is first to complete the python code by implementing the core of the parity rule algorithm. This core algorithm should start at the line 49 of "parityRule.py". That algorithm is given in the slide 4 of the cellular automata presentation. You can either implement the algorithm with an on the fly calculation or with a look-up table. However, for this project, periodic boundary conditions are imposed in all four directions.

Once you correctly have implemented the parity rule algorithm, the first question is:

What is the number of white pixels after 32 iterations of the parity rule with "image1.bmp"?

Answer: **33504**

From <http://localhost:8888/notebooks/Downloads/Modeling/modeling.ipynb>

2 points

2.

Question 2

What is the number of white pixels after 32 iterations of the parity rule with "image2.bmp"?

Answer: 0

1 point

3.

Question 3

What is the number of white pixels after 32 iterations of the parity rule with "image3.bmp"?

Answer: 0

1 point

Coursera Honor Code [Learn more]()

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Submit **Save draft**

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/BbbjZ/project-the-parity-rule/attempt>

Week 5

Wednesday, January 25, 2023
6:54 PM

Skip to Main Content

SEARCH IN COURSE
Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 5
- Optional - Equations and challenges

PreviousNext

- **Lattice Boltzmann modeling of fluid flow**
- **Introduction**
- **The Lattice Boltzmann Algorithm**
- **Application**
- **Assignments**
- **Practice Quiz:** Optional - Equations and challenges
  1 question
- **Quiz:** Lattice Boltzmann modeling of fluid flow
  2 questions
- **Quiz:** Project - Flow around a cylinder
  1 question
- **Quiz:** Collision Invariant
  3 questions

Optional - Equations and challenges

Practice Quiz30 minutes • 30 min

Submit your assignment

Resume assignment

Receive grade

To Pass 80% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Optional - Equations and challenges

Practice Quiz. • 30 min. • 1 total point available.1 total point

1.

Question 1

Let say that there is a laminar flow between two infinitely large horizontal plates, the top plate (at = $y=H$) is moving with a positive velocity U in the x direction and the bottom plate (at =0$y=0$) is immobile. Assume that there is no gravity, that the flow is steady and that vertical velocity =0$v=0$.

In 2D and with the differential operator expand the continuity and Navier-Stokes equations are:

∂ ∂ +∂ ∂ =0$\partial x \partial u+\partial y \partial v=0$

∂ ∂ + ∂ ∂ + ∂ ∂ =−1 0∂ ∂ + (∂2 ∂ 2+∂2 ∂ 2)$\partial t \partial u+u \partial x \partial u+v \partial y \partial u=-\rho 0 1 \partial x \partial P+v(\partial x 2 \partial 2 u+\partial y 2 \partial 2 u)$

∂ ∂ + ∂ ∂ + ∂ ∂ =−1 0∂ ∂ + (∂2 ∂ 2+∂2 ∂ 2)$\partial t \partial v+u \partial x \partial v+v \partial y \partial v=-\rho 0 1 \partial y \partial P+v(\partial x 2 \partial 2 v+\partial y 2 \partial 2 v)$

Which velocity profiles are incorrect (i.e. cannot be a good solution because it do not respect the continuity equation, Navier-Stokes equations or the boundary conditions)? (Note that one of the profiles is actually the good solution)

1 point

( )= 0∂ ∂ (−)+ $u(y)=\rho 0 v \partial x \partial P y(y-H)+H U y$

( )=1 0 ∂ ∂ (−)+ $u(y)=\rho 0 v 1 \partial x \partial P y(y-H)+H U y$

( )= 0∂ ∂ (−)+ $u(y)=\rho 0 v \partial y \partial P y(y-H)+H U y$

( )=1 0 ∂ ∂ (−)+ $u(x)=\rho 0 v 1 \partial x \partial P x(x-H)+H U x$

( )= 0∂ ∂ 2+ $u(y)=\rho 0 \nu \partial x \partial P y 2+HUy$

Coursera Honor Code [Learn more]

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Submit **Save draft**

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/quiz/tWOCf/optional-equations-and-challenges/attempt>

Optional - Equations and challenges

Practice Quiz30 minutes • 30 min

Submit your assignment

Try again

Receive grade

To Pass 80% or higher

Your grade

100%

View Feedback

We keep your highest score

Like

Dislike

Report an issue

Back

Optional - Equations and challenges

Practice Quiz. • 30 min. • 1 total point available.1 total point

Congratulations! You passed!

Grade received 100%

To pass 80% or higher

Go to next item

1.

Question 1

Let say that there is a laminar flow between two infinitely large horizontal plates, the top plate (at = $y=H$) is moving with a positive velocity U in the x direction and the bottom plate (at =0$y=0$) is immobile. Assume that there is no gravity, that the flow is steady and that vertical velocity =0$v=0$.

In 2D and with the differential operator expand the continuity and Navier-Stokes equations are:

∂ ∂ +∂ ∂ =0$\partial x \partial u+\partial y \partial v=0$

∂ ∂ + ∂ ∂ + ∂ ∂ =−1 0∂ ∂ + (∂2 ∂ 2+∂2 ∂ 2)$\partial t \partial u+u \partial x \partial u+v \partial y \partial u=-\rho 01 \partial x \partial P+\nu(\partial x 2 \partial 2u+\partial y 2 \partial 2u)$

∂ ∂ + ∂ ∂ + ∂ ∂ =−1 0∂ ∂ + (∂2 ∂ 2+∂2 ∂ 2)$\partial t \partial v+u \partial x \partial v+v \partial y \partial v=-\rho 01 \partial y \partial P+\nu(\partial x 2 \partial 2v+\partial y 2 \partial 2v)$

Which velocity profiles are incorrect (i.e. cannot be a good solution because it do not respect the continuity equation, Navier-Stokes equations or the boundary conditions)? (Note that one of the profiles is actually the good solution)

1 / 1 point

( )= 0∂ ∂ (−)+ $u(y)=\rho 0\nu\partial x\partial Py(y-H)+HUy$

Correct

( )=1 0 ∂ ∂ (−)+ $u(y)=\rho 0\nu 1\partial x\partial Py(y-H)+HUy$

( )= 0∂ ∂ (−)+ $u(y)=\rho 0\nu\partial y\partial Py(y-H)+HUy$

Correct

( )=1 0 ∂ ∂ (−)+ $u(x)=\rho 0\nu 1\partial x\partial Px(x-H)+HUx$

Correct

( )= 0∂ ∂ 2+ $u(y)=\rho 0\nu\partial x\partial Py2+HUy$

Correct

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/quiz/tWOCf/optional-equations-and-challenges/attempt?redirectToCover=true>

1. Let say that there is a laminar flow between two infinitely large horizontal plates, the top plate (at $y = H$) is moving with a positive velocity U in the x direction and the bottom plate (at $y = 0$) is immobile. Assume that there is no gravity, that the flow is steady and that vertical velocity $v = 0$.

In 2D and with the differential operator expand the continuity and Navier-Stokes equations are:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho_0}\frac{\partial P}{\partial x} + \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$

$$\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{1}{\rho_0}\frac{\partial P}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right)$$

Which velocity profiles are incorrect (i.e. cannot be a good solution because it do not respect the continuity equation, Navier-Stokes equations or the boundary conditions)? (Note that one of the profiles is actually the good solution)

☐ $u(y) = \frac{\nu}{\rho_0}\frac{\partial P}{\partial x} y(y - H) + \frac{U}{H} y$

☐ $u(y) = \frac{1}{\rho_0 \nu}\frac{\partial P}{\partial x} y(y - H) + \frac{U}{H} y$

☐ $u(y) = \frac{\nu}{\rho_0}\frac{\partial P}{\partial y} y(y - H) + \frac{U}{H} y$

☐ $u(x) = \frac{1}{\rho_0 \nu}\frac{\partial P}{\partial x} x(x - H) + \frac{U}{H} x$

☐ $u(y) = \frac{\nu}{\rho_0}\frac{\partial P}{\partial x} y^2 + \frac{U}{H} y$

Answer: 1,3,4,5 are incorrect. Only option 2 (u(y) = (1/(p0*v)) * dP/dx * y * (y-H) + U/H * y can be a solution.

Solution:
v = 0, i.e. from continuity=> du/dx = 0 > d^2u/dx^2) = 0, also u(x) = 0, u(y) not zero
There is no time variant, so du/dt = 0
Therefore (2) NS equation becomes 0 = -1/p0 * dP/dx + v * d^2u/dy^2
Which leads to: d^2u/dx^2 = 1/(p0*v) * dP/dx

(1) incorrect as "v" viscocity term is in numerator
(2) correct
(3) incorrect ha sterm of dP/dy instead of dP/dx
(4) incorrect as u(x) = 0
(5) incorrect as it does not satisfy boundary condition of u(y=h) = H

- Lekhraj Sharma

Chat with us

Lattice Boltzmann modeling of fluid flow

Quiz30 minutes • 30 min

Submit your assignment

Due February 19, 11:59 PM ISTFeb 19, 11:59 PM IST

Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 50% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Lattice Boltzmann modeling of fluid flow

Graded Quiz. • 30 min. • 2 total points available.2 total points

DueFeb 19, 11:59 PM IST

1.

Question 1

Select the statements which are true.

1 point

The LBM runs the simulation with Boolean variables and extracts macroscopic variables by taking averages at the end. [False; LBM runs with continuous variable from start which are derived from statistics like particle density]

Both the collision step and streaming step take the system forward in time. [False; only streaming does!]

In the LBM, pressure is computed from the density by applying the ideal gas law. [TRUE]

Equilibrium depends on local velocity, density and viscosity. [False; not on viscosity]

The fluid viscosity depends on the inverse of the relaxation parameter ω *[TRUE]*

2.

Question 2

Which code of the following represents inflow velocity boundary condition?

1 point

```
fout = fin - omega*(fin-eq)
```

```
for i in range(9) # Streaming
    fin[i,:,:] = roll(roll(fout[i,:,:],v[i,0],axis=0),v[i,1],axis=1)
```

```
for i in range(9) # obstacle
    fout[i,obstacle] = fin[8-i,obstacle]
```

```
# inflow boundary condition
fin[[0,1,2],0,:]=feq[[0,1,2],0,:]+fin[[8,7,6],0,:]-feq[[8,7,6],0, :]
```

1

```
# outflow boundary condition
fin[col3,-1, :]= fin[col3,-2, :]
```

Coursera Honor Code [Learn more]

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

SubmitSave draft

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/ufqQ8/lattice-boltzmann-modeling-of-fluid-flow/attempt>

Skip to Main Content

SEARCH IN COURSE
Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 5
- Project - Flow around a cylinder

PreviousNext

- **Lattice Boltzmann modeling of fluid flow**
- **Introduction**
- **The Lattice Boltzmann Algorithm**
- **Application**

Project - Flow around a cylinder

Quiz30 minutes • 30 min

Submit your assignment

Due February 19, 11:59 PM ISTFeb 19, 11:59 PM IST

Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 60% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Project - Flow around a cylinder

Graded Quiz. • 30 min. • 1 total point available.1 total point

DueFeb 19, 11:59 PM IST

1.

Question 1

Advice:

Please note that finding the solution to this project may take a while and it also may require that you run the Python code for several hours on your personal computer. You are therefore advised to start this project as soon as possible.

Use the Python code provided in the course, i.e. “lbmFlowAroundCylinder.py”, to **<u>approximately determine the critical Reynold number, i.e. the lowest Reynold number for which the flow around the cylinder enters an unsteady regime after a sufficient number of iterations</u>**.

lbmFlowAroundCylinder

PY File

For the purpose of this project, it is sufficient to find the critical Reynolds number within an accuracy of ±5. You will need to run the Python code several time at different Reynolds numbers, but in each run, you are not required to run the code for more than 200 000 time steps.

You may want to consult the following Wikipedia article to make sure you understand the difference between a steady and an unsteady flow:
https://en.wikipedia.org/wiki/Fluid_dynamics#Steady_vs_unsteady_flow

For this exercise, we consider a flow configuration to be steady if the velocity norm $|| ^{\rightarrow}||||u||$ is time independent after a sufficient number of iterations. Otherwise, we assume that it remains unsteady forever. Note that the cylinder is spatially immobile and that the frame of reference is stationary with respect to the cylinder.

It is also interesting to point out that the critical Reynolds number depends on the flow configuration. In an exterior flow, in which the flow is not, as in our example, restricted by lateral walls, the critical Reynolds number would be larger.

The Python code automatically generates “.png” images of the velocity norm every 100 time steps. To help visualize the time dependence on the velocity field, it is useful to create a video clip from the images.

To create a video clip:

Under Windows, the freeware “IrfanView” can easily be used for that purpose.

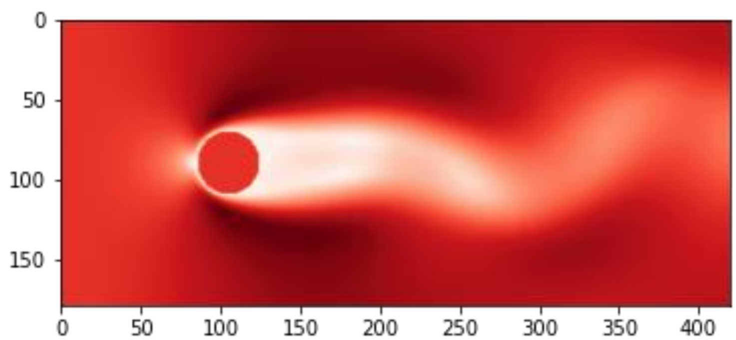
Under Linux, “avconv” or “ffmpeg” can be used with a command line of this type:

avconv -r 12 -i vel.%04d.png -q:v 0 vel.avi

Under Mac OS X, you can also use “avconv” or “ffmpeg”, but you first need to install “brew”.

Answer: **30.0**, We start to unsteadyness at RE=30.0. See it higher level as well (Re=35, 40, 50) but do NOT see it lower levels (Re=27, 25, 20, 15, 10, 5).

Unorm plot for Re=30.0 at iteration of 200,000 (30.0_vel.1999)



1 point
Coursera Honor Code Learn more

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Submit **Save draft**

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/NgXON/project-flow-around-a-cylinder/attempt>

Skip to Main Content

SEARCH IN COURSE
Search

Collision Invariant

Quiz30 minutes • 30 min

Submit your assignment

Due February 19, 11:59 PM ISTFeb 19, 11:59 PM IST

Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 60% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Collision Invariant

Graded Quiz. • 30 min. • 3 total points available.3 total points

DueFeb 19, 11:59 PM IST

1.

Question 1

A collision invariant is a quantity that does not change during collision, i.e. which has the same value when computed from the pre-collision or post-collision populations. By running numerical tests with your code, find out if the density is collision invariant.

1 point

Yes.

No.

2.

Question 2

A collision invariant is a quantity that does not change during collision, i.e. which has the same value when computed from the pre-collision or post-collision populations. By running numerical tests with your code, find out if the velocity is collision invariant.

1 point

Yes.

No.

3.

Question 3

A collision invariant is a quantity that does not change during collision, i.e. which has the same value when computed from the pre-collision or post-collision populations. By running numerical tests with your code, find out if the difference between $1f1$ and $7f7$ is collision invariant.

1 point

Yes.

No.

Coursera Honor Code [Learn more]

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Submit **Save draft**

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/6RFvG/collision-invariant/attempt>

Week 6 Quiz

Sunday, January 29, 2023
9:06 PM

Skip to Main Content

SEARCH IN COURSE
Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 6
- Particles and point-like objects

PreviousNext

- **Particles and point-like objects**
- **Numerical modeling of point-like objects**
- **Short-range interactions**
- **Long-range interactions**
- **Assignments**
- **Quiz:** Particles and point-like objects
3 questions
- **Quiz:** Project - Barnes-Hut Galaxy Simulator
1 question

Particles and point-like objects

Quiz30 minutes • 30 min

Submit your assignment

Due February 26, 11:59 PM ISTFeb 26, 11:59 PM IST
Attempts 3 every 8 hours
Start assignment

Receive grade

To Pass 40% or higher

Your grade

-Not available
Like
Dislike

Report an issue

Back

Particles and point-like objects

Graded Quiz. • 30 min. • 5 total points available.5 total points

DueFeb 26, 11:59 PM IST

1.

Question 1

Suppose you have two particles numbered 1 and 2 respectively at the initial positions 1(0)=0x1(0)=0 m and 2(0)=106x2(0)=106 m. Their initial velocities are also null, i.e. 1=0v1=0 m/s and 2=0v2=0 m/s. The particle masses are 1=1020m1=1020 kg and 2=1012m2=1012 kg. Note that the gravitational constant G is equal to 6.67408×10−116.67408×10−11 m3/(kg s2)m3/(kg s2). We model these particles as point-wise particles that interact only through the gravitational force (which is given in the slide 12). The numerical time step is equal to Δ =1000Δt=1000 s.

The leap-frog algorithm cannot be used to find the position of the particle at the next time step (=Δ t=Δt) because it requires the velocity at =Δ /2t=Δt/2 which the initial state of the problem does not provide. The explicit Euler scheme, however, can be used to find a first approximation of the position of the particles at =Δ t=Δt, then a simple centered finite difference formula can be used to approximate the velocity at =Δ /2t=Δt/2. Subsequently, the leap-frog algorithm has all the information needed to continue the time integration of the equations (slide 25-28).

The explicit Euler scheme gives an approximation of 1(Δ)=6.67408×10−5x1(Δt)=6.67408×10−5 m and 2(Δ)=993325.92x2(Δt)=993325.92 m . Using the Euler approximation, the centered finite difference leads to an approximation of 1(Δ /2)=6.67408×10−8v1(Δt/2)=6.67408×10−8 m/s and 2(Δ /2)=−6.67408v2(Δt/2)=−6.67408 m/s.

Now that you know these initial conditions, use the leap-frog algorithm to find an approximation of 2(2Δ)x2(2Δt) (in meter).

Find the approximation with at least 6 digits of accuracy.

Answer: 979887.77 m

Solution:

a2(t+dt) = F2/m2 = G * m1 * [x1(t+dt) - x2(t+dt)]/ [x1(t+dt) - x2(t+dt)]^3

=> a2 = 6.67408 * 10^-11 * 10^20 * -(993325.92 - 6.67408 * 10^-5)^-2 = -0.0067640665

v2(t+3/2*dt) = v2(t+1/2*dt) + dt * a2(t+dt)

=> v2 = -6.67408 + 10^3 * 6.7640665 * 10^-3 = -13.43814655

x2((t+dt)+dt) = x2(t+dt) + (t+dt) * v2(t+3/2*dt)

=> x2 = 993325.92 + 10^3 * -13.43814655 = 979887.77345 (Answer)

3 points

2.

Question 2
Which statement is true?

1 point

Long range forces are characterized by the fact that they drop to zero very fast.

The grid method used in this course to compute the terms of short-range forces can also be applied to the n-body problem.

For the evaluation of short-rang forces, the smaller the cutoff length is, the more accurate the algorithm is.

The computation of an electrostatic interaction between a large number of electrically charged particles can be simplified by the Barnes-Hut algorithm. [TRUE]

3.

Question 3
Concerning the algorithm of insertion of a new body into a node *n*, which statement is false?

1 point

If node *n* does not contain any body, put the new body there.

If node *n* is an internal node, update the center-of-mass of *n* and total mass of *n*, Recursively insert the new body in the appropriate quadrant.

If node *n* is an external node, update the center-of-mass of *n* and total mass of *n*, and put the new body there. [FALSE, need to first subdivide into four regions and proceed recursively!]

If node n is an external node, subdivide the region further by creating four children.

Coursera Honor Code [Learn more]

I, **Lekhraj Sharma**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

SubmitSave draft

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/HFO5I/particles-and-point-like-objects/attempt>

1. Suppose you have two particles numbered 1 and 2 respectively at the initial positions $x_1(0) = 0$ m and $x_2(0) = 10^6$ m. Their initial velocities are also null, i.e. $v_1 = 0$ m/s and $v_2 = 0$ m/s. The particle masses are $m_1 = 10^{20}$ kg and $m_2 = 10^{12}$ kg. Note that the gravitational constant G is equal to 6.67408×10^{-11} $\mathrm{m}^3/(\mathrm{kg\ s}^2)$. We model these particles as point-wise particles that interact only through the gravitational force (which is given in the slide 12). The numerical time step is equal to $\Delta t = 1000$ s.

The leap-frog algorithm cannot be used to find the position of the particle at the next time step ($t = \Delta t$) because it requires the velocity at $t = \Delta t/2$ which the initial state of the problem does not provide. The explicit Euler scheme, however, can be used to find a first approximation of the position of the particles at $t = \Delta t$, then a simple centered finite difference formula can be used to approximate the velocity at $t = \Delta t/2$. Subsequently, the leap-frog algorithm has all the information needed to continue the time integration of the equations (slide 25-28).

The explicit Euler scheme gives an approximation of $x_1(\Delta t) = 6.67408 \times 10^{-5}$ m and $x_2(\Delta t) = 993325.92$ m . Using the Euler approximation, the centered finite difference leads to an approximation of $v_1(\Delta t/2) = 6.67408 \times 10^{-8}$ m/s and $v_2(\Delta t/2) = -6.67408$ m/s.

Now that you know these initial conditions, use the leap-frog algorithm to find an approximation of $x_2(2\Delta t)$ (in meter).

Find the approximation with at least 6 digits of accuracy.

[Skip to Main Content]

SEARCH IN COURSE

Search

- Lekhraj Sharma

Chat with us

- Simulation and modeling of natural processes
- Week 6
- Project - Barnes-Hut Galaxy Simulator

PreviousNext

- **Particles and point-like objects**
- **Numerical modeling of point-like objects**
- **Short-range interactions**
- **Long-range interactions**
- **Assignments**
- **Quiz:** Particles and point-like objects
  3 questions
- **Quiz:** Project - Barnes-Hut Galaxy Simulator
  1 question

Project - Barnes-Hut Galaxy Simulator

Quiz30 minutes • 30 min

Submit your assignment

Due February 26, 11:59 PM ISTFeb 26, 11:59 PM IST

Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 60% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Project - Barnes-Hut Galaxy Simulator

Graded Quiz. • 30 min. • 1 total point available.1 total point

DueFeb 26, 11:59 PM IST

1.

Question 1

Generalize the two-dimensional Barnes-Hut galaxy simulator provided in the course, i.e. “barnes_hut.py”, into a three-dimensional Barnes-Hut galaxy simulator.

barnes_hut

[PY File]

In order to be consistent between the various possible generalizations, take into account the following constraints:

1) Do not change the simulation parameters of the main program, such as the random seed, theta, mass, ini_radius, etc.;

2) The x-, y- and z-pos should be initialized within a cube with side-length 2*ini_radius;

3) Keep only the bodies inside a sphere of radius ini_radius;

4) For the purpose of this project, you do not have to generalize the function “plot_bodies”. A three-dimensional “plot_bodies” function is given below:

```
def plot_bodies(bodies, i):
# Write an image representing the current position of the bodies.
# To create a movie with avconv or ffmpeg use the following command:
# ffmpeg -r 15 -i bodies3D_%06d.png -q:v 0 bodies3D.avi
    ax = plt.gcf().add_subplot(111, projection='3d')
    ax.scatter([b.pos()[0] for b in bodies], \
    [b.pos()[1] for b in bodies], [b.pos()[2] for b in bodies])
    ax.set_xlim([0., 1.0])
    ax.set_ylim([0., 1.0])
    ax.set_zlim([0., 1.0])
    plt.gcf().savefig('bodies3D_{0:06}.png'.format(i))
```

5) For simplicity, keep the angular momentum in the x-y-plane, but assume a null initial z-momentum for all the bodies. Use this code for initializing the momentum:

```
for body in bodies:
    r = body.pos() - array([0.5, 0.5, body.pos()[2] ])
    body.momentum = array([-r[1], r[0], 0.]) * \
```

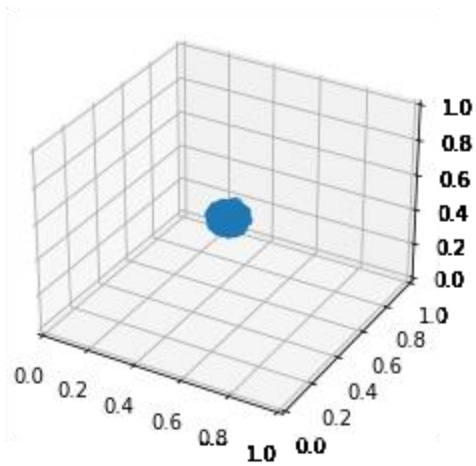
```
    mass*inivel*norm(r)/ini_radius
```

To validate your three-dimensional implementation, what is at iteration i=500 the z-component of the position of the first body in the list “bodies”?

Find this approximation accurate to fourth digits.

Answer: 0.50996268

At iteration: 500 First Body Position: [0.491613   0.47023082 0.50996268]



From <http://localhost:8888/notebooks/Downloads/Modeling/modeling.ipynb>

The Python code automatically generates “.png” images of the position of the bodies every 20 time steps. Similarly as in the LBM project, it is useful to create a video clip from the images in order to help visualize the time dependence on the positions. The parameters by default of the algorithm should generate a nice looking “galaxy” simulation

Additional information for learning purposes:

If you want to continue working on this topic, a C++ implementation of the Barnes-Hut galaxy simulator is also available in the course material.

barnes_hut_cpp

ZIP File

If you play further with the Python galaxy simulator and increase, for example, the number of bodies, you will rapidly find out that the code does not run very fast. As a matter of fact, it turns out to be up to three orders of magnitude slower than the corresponding C++ implementation. In case you intend to run further simulations on your own, you should therefore consider switching to the

C++ implementation provided in the course materials, or search the Internet for other software libraries in this field.

We did not have such a problem with the lattice Boltzmann code of the previous chapter. Although in this case the Python code was also slower than a corresponding C++ implementation, it was still reasonably fast, thanks to an efficient use of the NumPy through an array-based notation. This effect can unfortunately not be reproduced for the Barnes-Hut code, as NumPy doesn't offer any efficient quad-tree data structure.

1 point
Coursera Honor Code [Learn more](#)

I, **Lekhraj Sharma**, understand that submitting work that isn’t my own may result in permanent failure of this course or deactivation of my Coursera account.

SubmitSave draft

Like
Dislike
Report an issue

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/DVYhH/project-barnes-hut-galaxy-simulator/attempt>

Week 7 Quiz

Wednesday, February 1, 2023
11:08 PM

[PreviousNext]

- **Introduction to Discrete Events Simulation**
- **Introduction to Discrete Events**
- **Definition of Discrete Events Simulations**
- **Optimisation problems**
- **Implementation matters**
- **Two examples**
- **Assignments**
- [**Quiz:** Introduction to Discrete Event Simulation
3 questions]
- [**Quiz:** Project - Simple modelling of traffic lights
1 question]

Introduction to Discrete Event Simulation

Quiz30 minutes • 30 min

Submit your assignment

Due March 5, 11:59 PM ISTMar 5, 11:59 PM IST

Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 50% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Introduction to Discrete Event Simulation

Graded Quiz. • 30 min. • 6 total points available.6 total points

DueMar 5, 11:59 PM IST

1.

Question 1

Concerning the Discrete Event Simulation (DES), which statement is false?

1 point

The system is simulated by jumping from one event to the next in time.

It is required that the state of the system can be computed analytically at any time between two events.

By DES, we could know the state of each entity at any time.

More than one event can occur simultaneously in DES. [FALSE, only one event can occur at a given time]

2.

Question 2
Following the traffic intersection example showed in the class, at what time the last event happen in the following simulation if the model parameters are a=25 and b=15.

Time	Event	C	F	Queue
0		0	R	Car(10) Car(15) Car(30) Car(60)
10	CAR	1	R	Car(15) Car(30) R2G(10+25) Car(60)
15	CAR	2	R	Car(30) R2G(35) Car(60)
30	CAR	3	R	R2G(35) Car(60)
35	R2G	0	G	Car(60) G2R(35+3*15)
60	CAR	0 (pass)	G	G2R(80)
80	**G2R**	**0**	**R**	**Empty**

Answer: **80** (last event time after which queue become empty!)

2 points

3.

Question 3
During a volcano eruption, three of the spherical bombs have the following characteristic:

	Bomb 1	Bomb 2	Bomb 3
v(0) (/)(m/s)	[-2 0 2]	[1 1 40]	[-1 1 20]
r(0) ( )(m)	[0 0 100]	[-10 -5 30]	[-5 -5 60]
Radius ( )(m)	2	3	3

Assume that the gravitational acceleration is downward and equal to 9.81 / $2m/s2$. The ground is at z=0.

If there is a collision between the bombs, which collision events happen first?

3 points

Collision between bomb 1 and bomb 2 (Around 1.5 sec)

Collision between bomb 1 and bomb 3 (around 1.8 sec)

There is no collision between the bombs.

Collision between bomb 2 and bomb 3 [around 1.1 sec]

Answer: **Collision between bomb 2 and 3**

Solution:

Projectile Motion equation: r(t) = 1/2 * g * t^2 + v(0) * t + r(0)

Therefore for two projectile "I" and "j" to collide at time "t"

|1/2 * g * t^2 + vi(0) * t + ri(0) - 1/2 * g * t^2 + vj(0) + rj(0)| < |Ri + Rj|

=> t^2 + 2 * (dv.dr)/dv^2 * t + (dr^2 - (Ri+Rj)^2)/dV^2 = 0, where dv = vi-vj, dr =ri-rj

=> comparing it to quadratic equation: x^2 + 2bx + c = 0, where b = (dv.dr)/dv^2, c = (dr^2 - (Ri+Rj)^2)/dv^2

=> t =(-b +- sqrt(b^2 - c))

=> for t to have real solution, b^2-c >= 0

For Bomb 1, 2: dv = 85.1, dr.dv = -2695, dr = 70.0, R1+R2=5 => t = 1.87+-0.322 > t(min) = 1.54 sec

For Bomb 1, 3: dv = 18.05, dr.dv = -730, dr = 40.6, R1+R2=5 => t = 2.253+-0.46 > t(min) = 1.79 sec

For Bomb 2, 3: dv = 20.1, dr.dv = -610, dr = 30.0, R1+R2=6 => t = 1.525+-0.407 > t(min) = 1.12sec

=> Bomb 2 and 3 will collide earliest!

[vectors - What is the general approach to calculating time of impact in 3D? - Physics Stack Exchange]

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/nf6R2/introduction-to-discrete-event-simulation/attempt>

Project - Simple modelling of traffic lights

Quiz30 minutes • 30 min

Submit your assignment

Due March 5, 11:59 PM ISTMar 5, 11:59 PM IST

Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 60% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Project - Simple modelling of traffic lights

Graded Quiz. • 30 min. • 1 total point available.1 total point

DueMar 5, 11:59 PM IST

1.

Question 1

An incomplete traffic lights simulator is provided for this project, i.e. “trafficLights.py”.

trafficLights

PY File

The model has two simple parameters Tc and Tp. The two parameters represent unit of time in seconds. The first parameter is the time latency to change the traffic lights from red to green once a car is found waiting in the queue. The second parameter Tp is the time required for a car to cross the intersection. So once the lights change from red to green, the lights will stay green for an amount of time equals to Tp times the number of cars waiting. The model also assumes that once the lights change from red to green, all cars waiting pass through immediately and at the same time.

This project has two parts.

First, you need to complete the line of code 77, 79, 87 and 95. Indication on what to do on these lines is given into the code. Once you have correctly inserted the Python code on these lines. You can run the program and it will solve the “Evolution example” shown in the class at the course’s slide 45. In particular, the last event print should be a G2R event at time 115.

For the second part of this project, you will need to uncomment the lines 134 to 139. This section of the code will then add randomly an additional hundred cars into the queue event. Also, change the passage time Tp to 15 seconds. The question is at what time the last event is now taking place?

Answer: **712** [G2R(712)]

Basic Case:
CAR(10)
CAR(25)
CAR(35)
R2G(40)
CAR(60)
G2R(70)
CAR(75)

R2G(105)
G2R(115)
Green light =False, cars=0
Additional Random Cars Case:
CAR(10)
CAR(25)
CAR(35)
R2G(40)
CAR(60)
CAR(75)
G2R(85)
CAR(86)
CAR(95)
CAR(101)
CAR(102)
CAR(103)
CAR(105)
CAR(113)
R2G(116)
CAR(120)
CAR(123)
CAR(128)
CAR(134)
CAR(137)
CAR(142)
CAR(145)
CAR(150)
CAR(158)
CAR(166)
CAR(168)
CAR(176)
CAR(177)
CAR(184)
CAR(192)
CAR(199)
CAR(201)
CAR(202)
CAR(204)
CAR(213)
G2R(221)
CAR(222)
CAR(226)
CAR(235)
CAR(243)
CAR(247)
R2G(252)
CAR(254)
CAR(260)
CAR(262)
CAR(266)

CAR(271)
CAR(280)
CAR(282)
CAR(287)
CAR(288)
CAR(292)
CAR(295)
CAR(296)
CAR(301)
CAR(304)
CAR(312)
CAR(320)
G2R(327)
CAR(329)
CAR(336)
CAR(340)
CAR(348)
CAR(356)
R2G(359)
CAR(361)
CAR(367)
CAR(371)
CAR(378)
CAR(387)
CAR(388)
CAR(391)
CAR(399)
CAR(407)
CAR(415)
CAR(419)
CAR(420)
CAR(429)
G2R(434)
CAR(437)
CAR(445)
CAR(447)
CAR(449)
CAR(453)
CAR(454)
CAR(463)
R2G(467)
CAR(470)
CAR(475)
CAR(481)
CAR(488)
CAR(491)
CAR(497)
CAR(505)
CAR(514)
CAR(519)

CAR(524)
CAR(532)
CAR(540)
CAR(545)
CAR(546)
CAR(549)
CAR(550)
CAR(558)
CAR(560)
CAR(568)
G2R(572)
CAR(577)
CAR(582)
CAR(583)
CAR(585)
CAR(594)
CAR(597)
CAR(601)
R2G(607)
G2R(712)
Green light =False, cars=0

From <http://localhost:8888/notebooks/Downloads/Modeling/modeling.ipynb>

1 point

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/7kcyH/project-simple-modelling-of-traffic-lights/attempt>

An incomplete traffic lights simulator is provided for this project, i.e. “trafficLights.py”.

trafficLights
PY File

The model has two simple parameters T_c and T_p. The two parameters represent unit of time in seconds. The first parameter is the time latency to change the traffic lights from red to green once a car is found waiting in the queue. The second parameter T_p is the time required for a car to cross the intersection. So once the lights change from red to green, the lights will stay green for an amount of time equals to T_p times the number of cars waiting. The model also assumes that once the lights change from red to green, all cars waiting pass through immediately and at the same time.

This project has two parts.

First, you need to complete the line of code 77, 79, 87 and 95. Indication on what to do on these lines is given into the code. Once you have correctly inserted the Python code on these lines. You can run the program and it will solve the “Evolution example” shown in the class at the course’s slide 45. In particular, the last event print should be a G2R event at time 115.

For the second part of this project, you will need to uncomment the lines 134 to 139. This section of the code will then add randomly an additional hundred cars into the queue event. Also, change the passage time T_p to 15 seconds. The question is at what time the last event is now taking place?

Enter answer here

Basic Case:
CAR(10)
CAR(25)
CAR(35)
R2G(40)
CAR(60)
G2R(70)
CAR(75)
R2G(105)
G2R(115)
Green light =False, cars=0
Additional Random Cars Case:
CAR(10)
CAR(25)
CAR(35)
R2G(40)
CAR(60)
G2R(70)
CAR(75)
CAR(86)
CAR(95)
CAR(101)
CAR(102)
CAR(103)
R2G(105)
CAR(105)

CAR(113)
CAR(120)
CAR(123)
CAR(128)
CAR(134)
CAR(137)
CAR(142)
CAR(145)
CAR(150)
CAR(158)
G2R(165)
CAR(166)
CAR(168)
CAR(176)
CAR(177)
CAR(184)
CAR(192)
R2G(196)
CAR(199)
CAR(201)
CAR(202)
CAR(204)
CAR(213)
CAR(222)
CAR(226)
CAR(235)
CAR(243)
CAR(247)
CAR(254)
G2R(256)
CAR(260)
CAR(262)
CAR(266)
CAR(271)
CAR(280)
CAR(282)
CAR(287)
CAR(288)
R2G(290)
CAR(292)
CAR(295)
CAR(296)
CAR(301)
CAR(304)
CAR(312)
CAR(320)
CAR(329)
CAR(336)
CAR(340)
CAR(348)

CAR(356)
CAR(361)
CAR(367)
G2R(370)
CAR(371)
CAR(378)
CAR(387)
CAR(388)
CAR(391)
CAR(399)
R2G(401)
CAR(407)
CAR(415)
CAR(419)
CAR(420)
CAR(429)
CAR(437)
CAR(445)
CAR(447)
CAR(449)
CAR(453)
CAR(454)
G2R(461)
CAR(463)
CAR(470)
CAR(475)
CAR(481)
CAR(488)
CAR(491)
R2G(493)
CAR(497)
CAR(505)
CAR(514)
CAR(519)
CAR(524)
CAR(532)
CAR(540)
CAR(545)
CAR(546)
CAR(549)
CAR(550)
G2R(553)
CAR(558)
CAR(560)
CAR(568)
CAR(577)
CAR(582)
CAR(583)
CAR(585)
R2G(588)

CAR(594)
CAR(597)
CAR(601)
G2R(658)
Green light =False, cars=0

From <http://localhost:8888/notebooks/Downloads/Modeling/modeling.ipynb>

Multi Agent Systems

Thursday, February 2, 2023
11:49 PM

Last module, I detailed what agent were.
And now we will see how we will build a system by
simply putting several agent in the same environment.
So usually we talk about multi-agent systems, and it can look like that.
So we have a collection of agents.
Like before, they are in the environments.
But now the agent can interact with each other.
Play video starting at ::44 and follow transcript0:44
There are several ways of understanding and ending this interaction.
Play video starting at ::49 and follow transcript0:49
Here, I try to detail the connection, the interaction between the agents.
Play video starting at ::57 and follow transcript0:57
But in fact, of course an agent
can simply see the other agent as part of it's environment perception.
So again, from an agent again, all the rest can be seen as a black box.
This property is really interesting for modeling because it means that we can
focus on a spot of the model and consider that all the rest is uniform black box.
And so we can refine the agent rules without caring about the rest.
And hoping that doing that and putting all the agent [INAUDIBLE] inside,
together, emergent behavior will occur.
Play video starting at :1:38 and follow transcript1:38
So I will discuss a bit now about what
are the aspects involved with multi-agent system.
So, the first thing that we have to say is most agent based systems,
they use a multi-agent system because one single agent is just a function.
So it's just a program,
we don't need any smart framework to do that.
So most, the interesting thing about it is the fact that we will
put several models together, several agents.
And we hope that by the interaction or the cumulative
effect of all these agents, something rich will happen.
And here we are discussing about simulation and
modeling of natural processes and of course about agent-based modeling.

But pay attention because multi-agent systems are, in fact,
used in lots of other fields like optimization for example.
The fact that lots of optimization methods they use,
agent which will try to find the optimized solution.
I won't speak about it.
There are really interesting methods of network security where some agents
are mobile and will jump from one computer to the other and
try to monitor what's going on.
Play video starting at :3: and follow transcript3:00
I won't talk about it.
And finally the videogames is perhaps the most interesting example
from the system that are multi-agents but not agent-based modeling, per se.
In the videogames, for example, every opponent or
every character in the game that is not controlled
by a player is programmed using multi-agent system approach.
Bad guys and shooters, for example, and so on.
But I won't talk about it.
We will restrict ourselves to agent-based modeling.
But, all I will say here about multi-agent system are valid for all applications.
So first of all, often the agents are specialized.
They have a special location.
Their environment is a special domain can be 2D, it can be 3D,
or it can have whatever topology you want, for example, a graph.
If you are interested in epidemics spreading,
you are not only interested in where the people are, because they will move.
What is interesting is the graph of the relation.
For example, every agent, they have a higher chance of infecting
Play video starting at :4:24 and follow transcript4:24
their coworkers, their family, and so on, so what is the real
special domain here is a graph with the relation between the agents.
But, really often, they are in 2D and 3D, and
the example I will show you is the case.
Play video starting at :4:41 and follow transcript4:41
The agent of a location, they could move, for example, animals, pedestrians,
Play video starting at :4:47 and follow transcript4:47
can move if you have a traffic model, the agent will move and so on.
And this location is really interesting, is really important.
Because it could affect two things.
First of all, the perception awareness of the agents, of the environment.
Instead of seeing everything, he will just see a little bit of it.
And then the current position of an agent can
change the interaction network between other agents.
So two small examples here.
At left, so here we have an environment, black walls like a kind of maze.
And agent are the black dots.
And in green, you see the environment awareness of every agent.
So here, I present you two kind of approach.
The first one agent is aware of its immediate surrounding with a radius.
Play video starting at :5:44 and follow transcript5:44
For example, it can be a radio signal, and

the agent can perceive another thing that are in the same radius or
it can listen to noise around it and so on.
But we can have other approaches.
For example, in that case, the agent have a line of sight.
They have a direction where they point and
they can just see what is in front of them.
Play video starting at :6:11 and follow transcript6:11
So you can decide as you want to what would be the environment
awareness of the agent, what exactly they will perceive.
And in lots of interesting models that would change with their spatial position.
Play video starting at :6:29 and follow transcript6:29
The other parameter which is affected is the interaction topology,
because here again, the agent are black dots.
And if you see a blue line between two dots mean that these two
agent can interact.
Play video starting at :6:45 and follow transcript6:45
At right, you can see an example of an agent
interaction network where every agent can interact with the others.
That would be the case for example with the radio system with
every agent can decode or send a message to the other one.
And I don't know, telephone or email where everybody can reach all people.
But you can also interaction other interaction topology.
For example, you can see that the interaction happen only at contacts.
So when two agents are in contact, they can exchange something.
Or you can have, for example,
the topology can depend on the special awareness of every agent.
In that example, for example, the agent can communicate
only if they are aware of the same region of the space.
So, these three agents can communicate but
Play video starting at :7:47 and follow transcript7:47
not though too far away because they don't see the same zone.
But they can use the middle agent to relay.
Or these three agents, these four agent here can communicate with each
other because they are quite close enough and so on.
So you have to decide if they are interaction, what's the topology?
Is it only a shorter, one short contact of nodes?
And so that would change a lot, the output and the dynamic of your network.
So here, we will talk in the rest of this lecture about agent based models.
Agent based model again is the use of a multi-agent system
to model a natural phenomenon.
And again we model the agent, the basic individual entities,
but we observe a global complex behavior, and
that's the behavior that we want to validate with
the nature that we want to compare to experiments.
Play video starting at :8:56 and follow transcript8:56
Now that I define what a Multi-Agent System, I will spend the next module to
explain you how usually they are implemented in a computer system.

Agent Based Models: Implementation

Thursday, February 2, 2023
11:46 PM

After describing what are agent and multi-system agent,
we'll now say some words about how to implement them in a computer system.
Of course, Agent Based Models,
multi-agent systems are interesting only when implemented in the computer
because there's no way of solving them analytically, or so on.
So there's several ways of course of implementing it, in fact you can use
a whole new approach in the language you want a example in almost all languages.
It's always possible to implement them.
But there are some better match than other and they think that for agent and
Object-Oriented definition can be really good match.
Because this kind of correspondence of equivalence between the concept of
agent based, modeling of agents, and the concept of object in object oriented.
Play video starting at :1:15 and follow transcript1:15
For example, you have an immodity of several type of agent.
For example, okay, if you have ant model, you can have an ant with different castes,
like a worker ant, the soldier ant, the queen and so on.
So you can see the class in the object-oriented
programming as a kind of agent.
And then every class has to be instantiated.
So you can have, you can see the instance of this class is simply the agent.
And you have one instance per agent.
If you need to spawn an agent during the model, you just instantiate the class.
And really important concept of object-oriented,
is the encapsulation with private and public members.
And here it's interesting,
because the private member will be doing internal state of the agent.
The memories, the part of it that makes it kind of black box from the out of sight.
And the behavior will be a public method or a collection of public method.
It's how the agent will react to the stimulus.
So here in [INAUDIBLE] Java, an example of an agent.
So, you have a class for agents and here you have
a sort of Java because of course you have to assign them using a constructor.
But here you have an ID, so it's really important, it's good advice
that every one of your agents has a different ID.
So that will allow you to debug, to track, to extract a single trajectory.
Play video starting at :2:58 and follow transcript2:58
But then we have private member, we describe the state.
And then you have a behavior function.
That is even a perception, root and an action.
And of course for every model, the perception.
What is a perception?
What is an action?

Would be different, it really depends of what you are trying to model.
Play video starting at :3:19 and follow transcript3:19
And given that, we can now try to see how to relate every pieces together.
Because again, if I go back, it's quite easy to define that for every agent.
Then of course, you, knowledge in the field will
help you to feel what's in the dot and to choose whether the state will be.
But it's quite easy to do that part.
And then you will have to put all the object together to make them evolve and
then we have two basic approach.
We can update the system in an asynchronous or
synchronous way, like for cellular automata, for example.
Play video starting at :4:3 and follow transcript4:03
So I show you here in sort of Python how to,
Play video starting at :4:11 and follow transcript4:11
the general form of the code, a kind of universal loop, universal update loop.
For a sync update and synchronous update.
So first, asynchronous.
Asynchronous means that at every time iteration,
I will take the agent one after the other and update it.
And once I finished update everybody I will jump to the next iteration.
So if we go a little bit in details.
So first we need to initialize a system.
So here we have initial function which returns a list of agents.
And then an initial environment.
Play video starting at :4:48 and follow transcript4:48
The time is the timing, the initial time, and
we will define a max time as a stop condition.
Could be other stop conditions, of course, but here,
we will continue to iterate until the time reach the max time.
At each iteration, we will consider every agent individually.
First, we'll compute the perception that the agent we receive.
Knowing the environment and the list of all the other agents.
We'll call it p, then we can use the agent to compute this behavior and
you wait on an action.
And finally, we update the environment using this action.
Play video starting at :5:30 and follow transcript5:30
So the environment will be updated, and we will go to the next agent and
we compute the perception for next agent and so on.
When we have seen all the agent, when we have iterated over all the agent, we can
increment time, can be often continuous time, but other solution are possible.
And if the time is still smaller than the max allowed time,
we'll concentrate again to see all the agents.
That approach seems natural, but we have to pay attention here.
Because here we consider always the agents in order and that can introduce a bias
where some agents are favored because their decision will take place before.
So one of the ways to over come that is, instead of taking the listing the agent
always in the same order, we shuffle the agent list at each iteration.
But the problem is that some phenomena are really synchronous
because they are operating at the same time.
And the, another way to implement it is using a synchronous of that scheme.

So here the code looks simpler, but it isn't necessarily.
Play video starting at :6:43 and follow transcript6:43
Here we initialize again the system and receive a list of agent and
environment for the time is the same.
We'll loop until we reach a small [INAUDIBLE].
The difference is that every step is done on all the agents at the same time.
For example, here we will compute
the same step, but agent per agent.
Here, first of all, we will compute all perception, so for
that we have the environment, the agent, and we receive a list of perception.
Which has the same length than the list of agents and every element on this list is
what the agent at this list index will perceive of the environment.
Once we have all the perceptions, we can use this function or behaviors.
Play video starting at :7:32 and follow transcript7:32
That will call the behavior function of every agent on this perception,
and return a list of action with the action that all the agents wants
to effect.
Yeah, to effect an environment.
So now we have to update the environment.
And the environment will be updated using the least of all actions.
So if there are conflicts because two agent want to do the same thing,
like going at the same special position,
you get a conflict that will be handled at this level.
So we can use our rule to explain all this conflict must be handled.
And then we increment the time in the same way as we done in the synchronous update.
The interesting part about synchronous update also,
is that it is easier to prioritize.
Because you know that, for example, all behaviors, what you have to do is,
you have a list of perception, a list of agent.
You have to take both lists to put them side, every element aside.
And then use the call the function behavior,
and then every agent on the corresponding perception.
And that you can do in parallel because they won't depend on each other.
Play video starting at :8:51 and follow transcript8:51
It's the same for computing the perceptions.
You can do that in parallel, but
if you have to handle every agent one after the other,
you cannot because you will have conflict and you will have to end on them.
Play video starting at :9:4 and follow transcript9:04
So the synchronous update is interesting for that, but
again in lots of systems where action is slow or uncommon,
it is not realistic to put in that everything happened at the same time.
You have to choose depending on the problem that you have.
Play video starting at :9:22 and follow transcript9:22
I think really interesting is that you have to choose
If you want to Lagrangian or Eulerian approach.
So usually people use a Lagrangian.
It mean that every agent has a location it's aware, in some sense,
of its location.
Play video starting at :9:41 and follow transcript9:41

And we can then compute for every agent since we know the location of every agent,
for example, we can compute if the two agent are in contact or
if two agents are inside a communication radius.
What every agent can see of their environment.
A typical representation for that is a list of agent's.
Where every agent has an idea in positioning in the to d plus or
the other variable state that it could have.
And so that's quite common to see that.
The problem is that it can be really slow, because naively,
if you have to compute I don't know, web interaction topology,
knowing the communication radius of every agent.
You will have to consider all pair of
agent to understand if they are close enough to communicate.
It means something of the order of n squared and
that can be really bad if you have huge number of agents.
But we can have something which is sub quadratic by using
specialized data structure.
So for example, k-d tree, which will recursively split the space
Play video starting at :11:2 and follow transcript11:02
among all the direction.
Usually, it cost O(n log n) to
build the kd tree out of end point.
But once you have one you can in log and modify it.
So basic log in which is quite small, and do all the rand search,
like and finding all the agent which are in the are of agent of interest in
o of n, the square root of n which helps you a lot in large system.
There are other interesting structure like r trees and
I once spent time in that but if you ever realistic system with lots of agent and
you need to be fast, you have to think about the that you will use.
An alternative is to use a Eulerian approach,
in that case the environment is a regular grid of cells,
a bit like is really similar, but
every cell contains a list of agents It's a list of agent like a showed you earlier.
The agent don't have to know their exact position, because they can pretend that
they are roughly at the same spots every cell is a spot and all the agents.
It's like every cell is a well stirred mixture.
And so it's fast to compute the interaction communication before,
because you can say, okay first that's every agent can communicate with every
other agent in the same place that they are in contact.
But for
longer communication agents you can include the neighboring cells and so on.
Movement is easy.
You just move and then jump to the adjusting cell in the correct direction.
And it allows to nice parallism.
It allows to compute the interaction really fast.
To couple it with other model, like cellular automata, finite differences.
And of course you will lose some special
spatial precision because in this approach, the agent is
unnecessarily has a discrete position instead of having a continuous one.

So the physique may not be correct if you have a dynamic.
Of course you can combine both approaches.
The variable time is really interesting to consider it also.
You don't have to use continuous time, because most of the time,
Play video starting at :13:35 and follow transcript13:35
model will use continuous time where the time stamp increase
Play video starting at :13:40 and follow transcript13:40
the time by a small amount every temporary iteration.
You can also use discrete time like we've seen last week.
Play video starting at :13:53 and follow transcript13:53
And use the ABM, the agent base model, to update the Discreet Event System state.
And here the behavior is just function with dates and event.
Play video starting at :14:5 and follow transcript14:05
The state of the system.
And will return new state plus collection of event.
So if you look at it from the point of view of Discrete Event Systems,
Play video starting at :14:19 and follow transcript14:19
every time an event happens you will find the concerned agent,
give the event as a percept to the agent.
And the agent reacted by changing the states, and pushing new events.
And so you can use EBM inside the discrete event system without any problem.
Play video starting at :14:42 and follow transcript14:42
So we discussed about some concern.
About how to implement agent based modules.
Of course, if you want to understand better how to do it,
the best way is to try to implement a system yourself.
Or to consider an existing agent bis module framework and
try to look at the source code to see how they did it.
Play video starting at :15:7 and follow transcript15:07
That concludes the most general part about this lecture.
Next module, I will try to take two examples coming from biology and
explain them properly and how agent based module are used to understand them better

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/lecture/196XP/implementation-of-agent-based-models>

Week 8 Quiz

Thursday, February 2, 2023
11:36 PM

[Skip to Main Content]()

SEARCH IN COURSE

Agent based models

Quiz30 minutes • 30 min

Submit your assignment

Due March 12, 11:59 PM ISTMar 12, 11:59 PM IST

Attempts 3 every 8 hours

Start assignment

Receive grade

To Pass 50% or higher

Your grade

-Not available

Like

Dislike

Report an issue

Back

Agent based models

Graded Quiz. • 30 min. • 4 total points available.4 total points

DueMar 12, 11:59 PM IST

1.

Question 1

Which statements are right:

1 point

In the Eulerian approach, it is possible to know the exact position of each agent at a given time. [False]

Multi-agent systems are synonymous of agent based models [False]

The Lagrangian approach is precise in space, but becomes lack of efficiency when there are a large number of agents. [TRUE]

In the implementation of ABM, if the synchronous update algorithm is adopted, the actions between different agents might conflict with each other. [TRUE]

2.

Question 2

In the Deneubourg's Model, what happen if the two the model parameters $1k1$ and $2k2$ are equal to 0. Assume initially that no ants carry or hold a corpse.

=> Pp = (k1/k1+f)^2 = 0, Pd = (f/f+k2)^2 = 1

=> Ants do NOT pickup but will always deposit! But irony is if they do NOT pickup anything what would they deposit especially when initially no ants carry/hold a corpse!

1 point

The workers will always pick up or deposit a corpse.

The workers will always pick up or but never deposit a corpse.

The workers will never pick up but always deposit a corpse.

The workers will never pick up and never deposit a corpse. [Answer]

3.

Question 3

In the example of bacteria chemotaxy, the code of behaviour function is as below:

```
def behaviour(rho,m_i,d_i):
if rho<=m_i:
   p=p_d
else
   p=p_i
if random()<=p:
   return rho,randomDirection()
else:
   return rho,d_i
```

If rho=0.7, m_i=0.6, d_i=S, p_i=0.4 and p_d=0.6, along which direction the bacterium will most probably move?

2 points

N

S (Answer)

E

W

Any direction above

Solution: Since for rho > m_i case, p=p_i=0.4, i.e there is only 0.4 probability that it will tumble (change direction). So it is more likely (with prob of 0.6) that it will keep going in same direction (d_i). Since d_i is "S", it will more likely to continue in "S" direction.

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/wJWip/agent-based-models/attempt>

The characteristics of the model

The characteristics of the model

There are 10 bacteria whose length is 1 micrometer scattered in 2D square with sides of 100 micrometer. The domain is p
simplify the problem, we consider that the bacteria could exist on a same place simultaneously.

The bacteria move successively by a time step of $\Delta_t = 0.2s$. They can either continuously advance in a straight line wit
or take a random direction (with a uniform probability) and advance at the speed v.

The rule for their movement of every time step is described as following:

1. If $\rho(x(t)) > \rho(x(t - \Delta_t))$, the bacterium continues to move in a straight line with a probability of 90%.

2. If not. it continues to move in a straight line with a probability of 50%.

Conentration field

The concentration is equal to:

$\rho(x) = \frac{1}{1+d(x,C)}$

Where C is the center of the domain, and $d(x, C)$ is the distance between the point x and C.

bacteria.py implements the above model. In the class of bacteria, there are some blanks which are needed to be filled. Re
blanks.

Questions:

The blank at the place of **Question 1** in the code should be filled with:

1 point

1

P1 (Answer)

1-P1

P2

1-P2

2.

Question 2

The blank at the place of **Question 2** in the code should be filled with:

1 point

P1

1-P1

P2 [Answer]

1-P2

3.

Question 3

The blank in the place of **Question 3** in the code should be filled with:

1 point

```
self.vx = 0.

self.vy = 0.
```

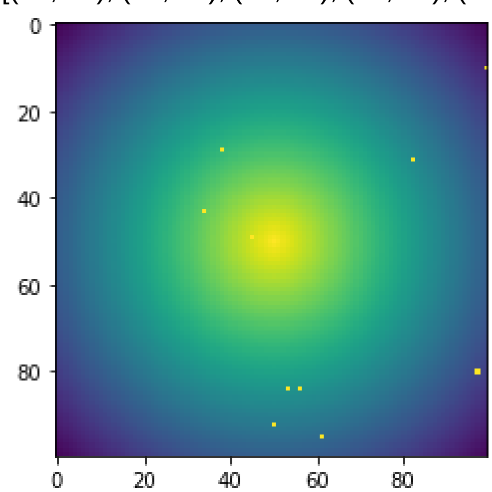
```
self.randomize_velocity()  [Answer]
```

```
self.vx = -self.vx
self.vy = -self.vy
```

From <https://www.coursera.org/learn/modeling-simulation-natural-processes/exam/mWYtT/project-multi-agents-model/attempt>

Bacteria positions after first 10 iterations: Most of them are away from center

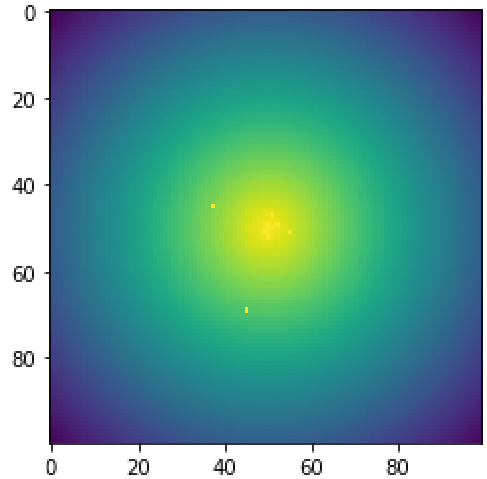
[(92, 50), (95, 61), (29, 38), (84, 53), (43, 34), (49, 45), (10, 99), (84, 56), (80, 97), (31, 82)]



From <http://localhost:8888/notebooks/Downloads/Modeling/modeling.ipynb>

Bacteria positions after last iterations (200 rounds): Most bacteria are now in center!

(50, 50), (50, 50), (49, 52), (52, 50), (47, 51), (49, 50), (45, 37), (51, 49), (69, 45), (51, 55)]



From <http://localhost:8888/notebooks/Downloads/Modeling/modeling.ipynb>

Code:

```
# Bacetria Simulation

from __future__ import print_function, division #compatibility py2 - py3
import random, math, numpy
import matplotlib.pyplot as plt

V = 2e-6
DT = 0.2
L = 100e-6
P1 = 0.9
P2 = 0.5
N = 10

def get_density(x,y): #version A
   return 1./(1.+math.hypot(x-L/2.,y-L/2.))

##def get_density(x,y): #version B
##    return float(math.hypot(x-L/2.,y-L/2.) < 15e-6)


def draw(b_list, n, t):
```

```
    m = numpy.zeros((n,n))
    b = []
    for x in range(n):
        for y in range(n):
            m[x,y] = get_density(x*L/n,y*L/n)
    for bacteria in b_list:
        x,y = int(bacteria.x*n/L), int(bacteria.y*n/L)
        b.append((x,y))
        m[x,y] = 1.
    print(b)
    plt.imshow(m, interpolation='None') #add interpolation='None' for non-smoothed image
    #plt.savefig("bacteria"+str(t)+".png")
    plt.show() #directly show the image

class Bacteria(object):

    def __init__(self, x, y):
        self.x = x
        self.y = y
        self.vx = None
        self.vy = None
        self.randomize_velocity()
        self.old_density = get_density(self.x, self.y)

    def randomize_velocity(self):
        alpha = random.random()*math.pi*2
        self.vx = math.cos(alpha) * V
        self.vy = math.sin(alpha) * V
        assert (math.hypot(self.vx, self.vy) - V) < 0.0000001

    def update(self):
        current_density = get_density(self.x, self.y)
        go_forward = False # go_forward "True" means, keeping going in same direction!
        if current_density > self.old_density:
            ######  Question 1 #########
            if random.random() < P1: # We want it to keep going in same dir with P1 prob
                go_forward = True
        else:
            ######  Question 2 #########
            if random.random() < P2: # We want it to keep moving in same direction with prob of P2
                go_forward = True
        if not go_forward: # Need to tumble to a random direction
            ######  Question 3 #########
            self.randomize_velocity()

        self.x += self.vx * DT
        self.y += self.vy * DT
        #domain periodicity:
```

```
        self.x %= L
        self.y %= L
        self.old_density = current_density

b_list = [Bacteria(random.random()*L, random.random()*L) for i in range(N)]
#print(b_list)
t_max = 301 # 200 orig
t_draw = 20 # 40 orig
for t in range(t_max):
    if t%t_draw == 0:
        draw(b_list, 100, t)
    for bacteria in b_list:
        bacteria.update()
```