



UNIVERSITÀ DEGLI STUDI DI MILANO



Laboratorio di Simulazione Numerica a.a. 2023/2024

Prof. Davide Emilio Galli

Davide.Galli@unimi.it



1

Who am I?

My name is Davide Emilio Galli and I work as associate professor at Dipartimento di Fisica, Università degli Studi di Milano

office: ground floor – LITA building

e-mail: Davide.Galli@unimi.it

web-site: ariel.unimi.it



God does not play dice, but ... you will!

Research fields:

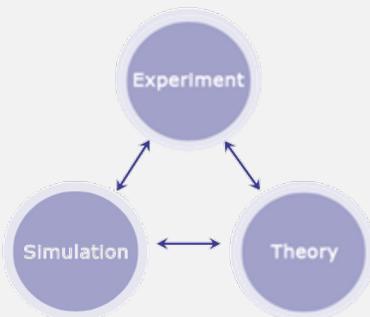
- Quantum gases, liquids and solids
- Statistical Mechanics & Complex systems
- Monte Carlo, Quantum Monte Carlo, Molecular Dynamics methods
- Statistical analysis of inverse problems
- Applications of stochastic optimization methods, Computational Intelligence, Machine Learning and Neural Networks, Quantum Machine Learning
- Quantum Simulation on real quantum computers



2

The third way of Physics: Simulation

Simulation = essential tool in the study of complex systems (Physics, & other fields), often used to anticipate, complete and reinforce experimental and theoretical analyzes.



Numerical Simulation Laboratory: introduction and application of various techniques (Monte Carlo and others) for the simulation of complex systems and the solution of complex problems.

3

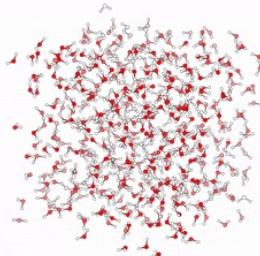
Houston, we have a problem! : Computability

$$\vec{F} = m\vec{a}$$

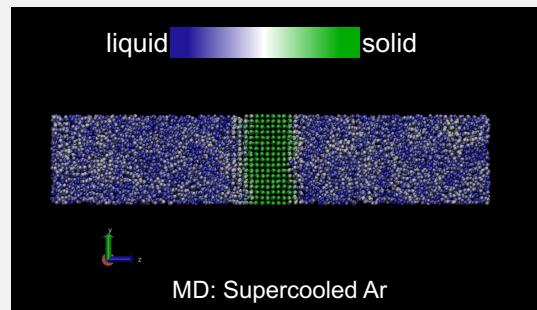


$$\left. \begin{aligned} \dot{q}_i &= \frac{\partial H(q_i, p_i)}{\partial p_i} \\ \dot{p}_i &= -\frac{\partial H(q_i, p_i)}{\partial q_i} \end{aligned} \right\} i = 1, 2, \dots, 3N$$

Molecular
Dynamics
 H_2O



liquid solid



4

2

Emerging phenomena

More is different!

[P.W. Anderson, Science 177, 393 (1972)]

P.W. Anderson

"TOMORROW HELL!
WE NEED IT NOW"
PUT ALL YOUR POWER IN THE JOB!

UNIVERSITÀ DEGLI STUDI DI MILANO

5

Advanced Monte Carlo methods

Special random walks
(stochastic processes,
Markov chains)
to select
configurations distributed
with a specific
probability

Markov chains

Posterior density

UNIVERSITÀ DEGLI STUDI DI MILANO

6



Monte Carlo ... random = accurate!

$$\langle O \rangle_p = \int_{R^n} d\vec{r} \ p(\vec{r}) O(\vec{r})$$

$$\langle O \rangle_p \cong \frac{1}{N} \sum_{i=1}^N O(\vec{r}^i) \Big|_{p(\vec{r})}$$



$$\text{Errore} \propto \frac{1}{\sqrt{N}}$$



Monte Carlo simulations

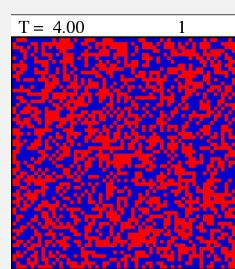
The Metropolis algorithm generates "photos" (configurations) of the system to calculate

$$\langle O \rangle_p \cong \frac{1}{N} \sum_{i=1}^N O(\vec{r}^i) \Big|_{p(\vec{r})}$$

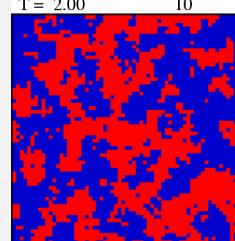


$$P_{acc} = \min \left[1, \frac{p(new)}{p(old)} \right]_{random \ moves}$$

2D Ising model



High T:
para-magnetic
phase



Low T:
magnetic
phase

$$p = \frac{e^{-\frac{H}{k_B T}}}{Z}$$

Stochastic optimization: e.g. Genetic Algorithms

Unknown target optimization problem: 😎

i-th generation: 😊 😞 😞 😞 😊 😂 😌 😊 😊 😈 😈 😈 😈

sorting : 😊 😂 😊 😊 😊 😊 😊 😞 😈 😈 😈 😈

Decreasing probability to be selected as parents

Selection+crossover: (😊+😊) (😊+😊) (😊+😊) (😊+😊) (😊+😊)
 (😊+😊) (😊+😊) (😊+😊) (😊+😊) (😊+😊)

=

😊 😊 😊 😊 😊 😊 😊 😊 😊 😊 😊 😊

Mutation : 😊 💀 😊 😊 😊 😊 😊 😊 😊 😊 💀

(i+1)-th generation

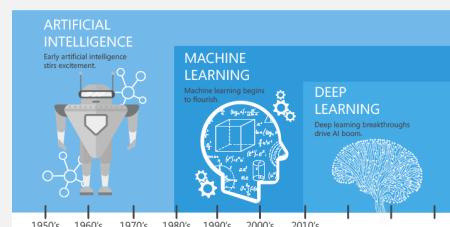
... after N>>1

generations: 😎 😎 😎 😊 😊 😊 😊 😊 😊 😊 😊

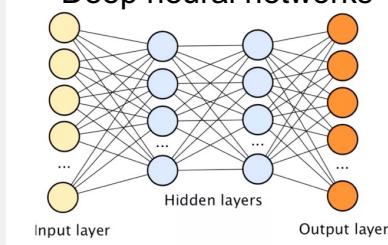
Deep learning



- We are in the age of Big Data... even in Science...
- numerical approaches that learn from data are increasingly important



Deep neural networks



Laboratorio di Simulazione Numerica

PROGRAM

- Probability theory, stochastic processes, mathematical statistics
- Sampling of random variables and Monte Carlo integration
- Markov Chains, Metropolis Algorithm
- Numerical simulations in classical and quantum statistical mechanics
- Stochastic calculus and stochastic differential equations with applications
- Computational intelligence, optimization, stochastic optimization
- Introduction to Machine learning & Neural Networks
- Introduction to parallel programming (MPI, Open MP)

When?: II Semester

- 24 hours Theoretical (12 Lectures)
- 36 hours Practice (12 Computing Laboratory sessions)

11

Numerical Simulation Laboratory goals

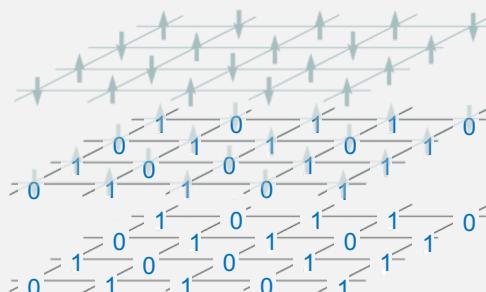
- Learn random sampling techniques (Monte Carlo) from basics to advanced applications
- “Learn” mostly means “practically” not (only) “theoretically”, so the emphasis will be on the “how-to-do”
- ... learn the basics of something that hopefully could be:
 - useful in your studies
 - useful in your future research activities
 - useful in your future activities (whatever they will be!)
- Learn/improve/practice in coding ... by yourself! ... So, you will play dice alone!
- Code languages (Simulation: C++, Data analysis: Python)
- Generally, exercises will be simple ... the goal is to learn “how-to-do”, not to complicate (too much) your life!

12



Numerical Simulation Laboratory EXAM

- Complete all the given **exercises**
- For each group of exercises produce a [jupyter-notebook](#) in which you describe the results obtained with description, comments and figures
- **About one week before the exam:** submit on [GitHub](#) your own working codes (with comments & indentation) and, if needed, detailed instruction to compile
- **Then** I will evaluate your work and **let you attend** the **oral exam** (questions about your exercises and only key concepts)
- That's all folks!



UNIVERSITÀ DEGLI STUDI DI MILANO

13



Theoretical Lectures: E room **Wednesday: 16:30-18:30**
Numerical exercises: Computer Lab. **Thursday: 14:30-17:30**

UNIVERSITÀ DEGLI STUDI DI MILANO

Theoretical lectures, 2024 dates:

February	28	(1 lecture)
March:	6 – 13 – 20	(3 lectures)
April:	3 – 10 – 17	(3 lectures)
May:	8 – 15 – 22 – 29	(4 lectures)
June:	5 – 12	(1 lecture + backup lecture)

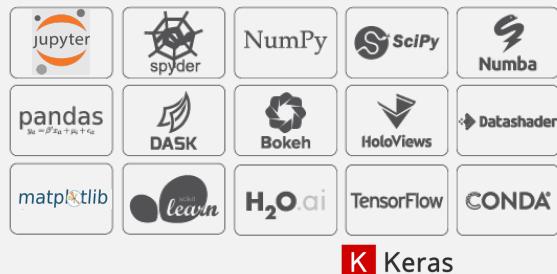
Numerical exercises, 2024 dates:

February	29	(1 lab.)
March:	7 – 14 – 21	(3 labs.)
April:	4 – 11	(2 labs.)
May	2 – 9 – 16 – 23 – 30	(5 labs.)
June:	6 – 13	(1 lab. + 1 backup lab.)

14

Software needed on your laptop

For jupyter-notebooks and a Python 3.* interpreter



Go to: <https://www.anaconda.com/distribution/>

Download available for Linux, macOS & Win ...

Download & install the [Anaconda Distribution](#)

Python already installed? Don't worry, it won't be overwritten!

15

Other useful (on-line) tools

REPL.IT: for C++ (& Python but not Jupyter) coding & test runs
(you can try also long run but maybe it is better to use the Laboratory's PC:
your_username@tolab.fisica.unimi.it)



Go to:

<https://replit.com>

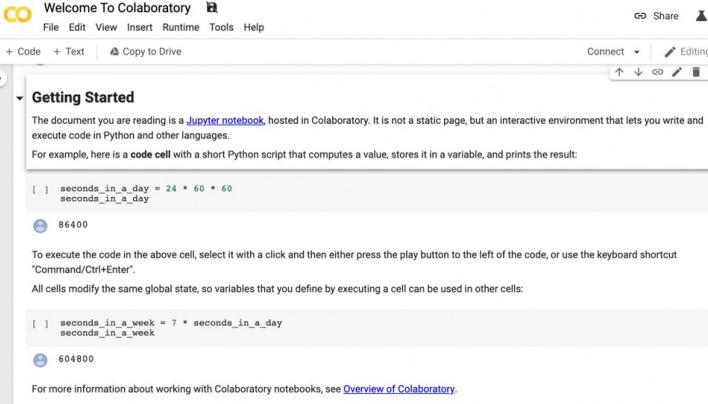
The screenshot shows a Jupyter Notebook interface with the following details:

- File Explorer:** Shows files like `main.py`, `gdp-life.txt`, `GDPlife.png`, and `us-cities.txt`.
- Packager files:** Shows `poetry.lock` and `pyproject.toml`.
- Code Cell:** The `main.py` file contains Python code for data analysis and visualization. It includes imports for `requests`, `seaborn`, `pandas`, `matplotlib`, and `numpy`. It reads data from `gdp-life.txt` and `data_to_viz/master/us-cities.txt`, performs correlation analysis, and creates a scatter plot comparing GDP per capita and life expectancy.
- Run Button:** A green "run" button is at the top right.
- Share Button:** A red arrow points to the "share" button in the top right corner.
- Output Area:** Shows the URL `https://04-data-science-and-visualisation-with-repl.lean3vijayen94.repl.run` and the Python version `Python 3.8.2 (default, Feb 26 2020, 02:56:10)`.
- Help and Settings:** Includes a question mark icon and a gear icon.

16

Other useful (on-line) tools

Google colab: For Python, Jupyter notebooks, Machine Learning (Tensorflow & Keras)




Go to: <https://colab.research.google.com/>

17

Other useful (on-line) tools

OPEN AI: ChatGPT, go to: <https://chat.openai.com/chat>



ChatGPT

Examples	Capabilities	Limitations
"Explain quantum computing in simple terms" →	Remembers what user said earlier in the conversation	May occasionally generate incorrect information
"Got any creative ideas for a 10 year old's birthday?" →	Allows user to provide follow-up corrections	May occasionally produce harmful instructions or biased content
"How do I make an HTTP request in Javascript?" →	Trained to decline inappropriate requests	Limited knowledge of world and events after 2021

ChatGPT Feb 13 Version. Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.

18



Other useful (on-line) tools



Ciao a tutti gli studenti del Laboratorio di Simulazione Numerica!

Sono ChatGPT, un assistente virtuale addestrato per aiutare le persone ad apprendere e risolvere problemi in vari campi. Sono stato creato per supportare la vostra esperienza di apprendimento nella programmazione e nella simulazione numerica, ma è importante che mi utilizzi in modo critico per ottenere i massimi benefici.

In particolare, potete utilizzarmi per migliorare le vostre capacità di scrittura e di espressione, oltre che per ottenere spiegazioni più chiare ed esaustive sui concetti che state studiando. Tuttavia, è importante che non vi limitiate a copiare e incollare il mio output o ad affidarvi esclusivamente a me per risolvere i problemi. Al contrario, utilizzatemi come un supporto per approfondire la vostra comprensione dei concetti e sviluppare la vostra abilità nella programmazione.

Vi auguro un grande successo nel vostro percorso di apprendimento e spero di poter essere un utile alleato per voi in questa avventura.

Ways ChatGPT can save hours of work every day, and why you'll never outcompete those who use AI effectively:

- **Tracking down bugs:** If you are having trouble finding a bug in your code, ask ChatGPT for help
- **Writing documentation/comments:** Ask ChatGPT to write the documentation/comments for a piece of code, and it usually does a great job
- **Improve/simplify existing code:** Ask ChatGPT to improve/simplify existing code by describing what you want to accomplish. It will give you instructions about how to do it, including the modified code
- **Explaining code:** take some code you want to understand and ask ChatGPT to explain it



Sito Ariel



Scienze e Tecnologie
Laboratorio di Simulazione Numerica



Lezioni e materiali didattici a.a. 2023/2024



Jupyter Notebooks

Il file compresso contiene i seguenti 7 jupyter notebooks:

- SOS Unix shell
- Introduction to jupyter notebooks (00)
- Introduction to Python (01/02/03)
- Introduction to GitHub for exercise delivery (04)
- Introduction to Keras (05)

Per essere visualizzati sul vostro laptop vi consiglio di installare l'interprete Python associato alla distribuzione Anaconda (<https://www.anaconda.com/distribution/>)



File allegati

[jupyter_2021_06_10.tgz](#)

Parallel Random Number Generator

Parallel Random Number Generator

La routine è RANDom New York University (RANINYU). Scritta negli anni 80 con un moltiplicatore suggerito nel 2 volume di Knuth "Art of scientific computing" p 102. Riadattata poi a inizio anni 90 nel gruppo di M.H. Kalos (da parte del Dr. Panoff).

Si tratta di un Linear Congruential Generator con prime addenda a 48 bit. La relazione che genera x_{n+1} da x_n è la seguente: $x_{n+1} = (a x_n + c) \bmod m$ dove a è il moltiplicatore, c è l'addendo (numero primo con certe proprietà), m è il modulo, in questo caso $m=2^{48}$.

Con questo moltiplicatore e questo modulo il periodo del generatore risulta $P=2^{48-10^{14}}$. Gli x_n sono numeri interi e corrispondono allo stato salvato nella routine. Il numero random reale in $[0,1]$ si ottiene con x_n/m .

Notare che per ogni c si ha un generatore diverso. In un lavoro di Percus e Kalos (che vi ho aggiunto) sono state mostrate le proprietà che due diversi $(c1 \text{ e } c2)$ devono avere affinché le serie generate (x_1, \dots, x_m) siano effettivamente indipendenti. I 384 numeri primi che già avevo in Primes soddisfano questo criterio. Segundo quel criterio, un mio collaboratore (Dr. G. Bertaina) me ne ha ottenuti 32001. Li trovate in primes32001.in, in ordine decrescente.

Notate che nella routine lo stato a 48 bit è salvato in 4 interi minori di 4096 ($=2^{12}$). Il tutto, comprese le operazioni, è scritto in base 4096 (anche in numeri primi nel file Primes, per questo sono copie).

File allegati

- [main.cpp](#)
- [Makefile](#)
- [percus1989.pdf](#)
- [Primes](#)
- [primes32001.in](#)
- [random.cpp](#)
- [random.h](#)
- [seed.in](#)



Introductory lecture: supplementary material



21

A short history of Monte Carlo

- Perhaps the earliest documented use of random sampling to find the solution to an integral is that of Comte de Buffon. In 1777 he described the following experiment: A needle of length L is thrown at random onto a horizontal plane ruled with straight lines a distance d ($d > L$) apart. What is the probability P that the needle will intersect one of these lines? Comte de Buffon performed the experiment of throwing the needle many times to determine P . He also carried out the mathematical analysis of the problem and showed that

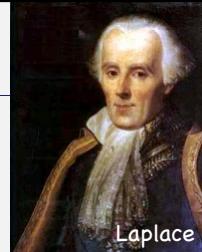
$$P = \frac{2L}{\pi d}$$



Fig. 1.6 Georges Louis Leclerc, Count of Buffon (1707–1788), performing the first recorded Monte Carlo simulation, in 1777. (Published with permission of *Le Monde*.)

A short history of Monte Carlo

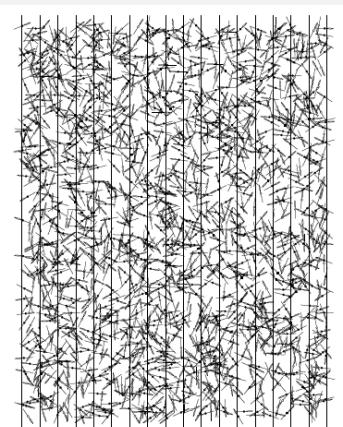
- Some years later (1820), Laplace suggested that this idea could be used to evaluate π from throws of the needle: if the needle is thrown down N times and is observed to land on a line M of those times, we can make an estimate of π from



Laplace

$$\pi = \frac{2L}{Pd} = \lim_{N \rightarrow \infty} \frac{2LN}{Md}$$

- A number of investigators made use of this method over the years to calculate approximate values for π . The most famous of these is Mario Lazzarini, who in 1901 announced that he had calculated a value of 3.1415929 for π from an experiment in which a needle was dropped 3408 times onto a sheet of paper ruled with lines.
- This value, accurate to better than three parts in ten million, would be an impressive example of the power of the statistical sampling method were it not for the fact that it is almost certainly faked.



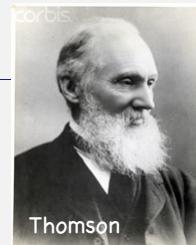
A short history of Monte Carlo

- Badger (1994) has demonstrated extremely convincingly that the chances of finding the results Lazzarini did were poorer than three in a million; Lazzarini was imprudent enough to publish details of the progress of the experiment through the 3408 castings of the needle, and it turns out that the statistical "fluctuations" in the numbers of intersections of the needle with the ruled lines are much smaller than one would expect in a real experiment. All indications are that Lazzarini forged his results!
- However, other, less well known attempts at the experiment were certainly genuine, and yielded reasonable figures for π : 3.1596 (Wolf 1850), 3.1553 (Smith 1855). Apparently, performing the Buffon's needle experiment was for a while quite a sophisticated pastime amongst Europe's intellectual gentry!
- With the advent of mechanical calculating machines at the end of the nineteenth century, numerical methods took a large step forward. These machines increased enormously the number and reliability of the arithmetic operations that could be performed in a numerical "experiment", and made the application of statistical sampling techniques to research problems in physics a realistic possibility for the first time.

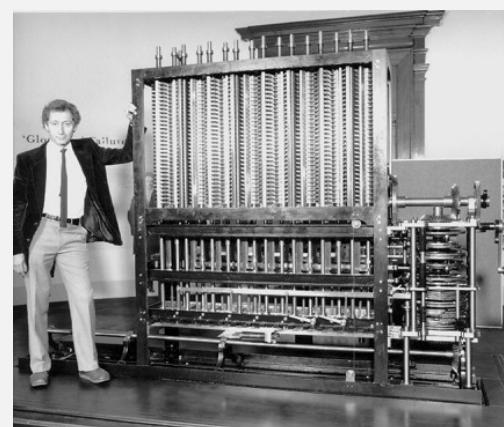


A short history of Monte Carlo

- An early example of what was effectively a Monte Carlo calculation of the motion and collision of the molecules in a gas was described by William Thomson (later Lord Kelvin) in 1901.
- Thomson's calculations were aimed at demonstrating the truth of the equipartition theorem for the internal energy of a classical system.
- However, after the fashion of the time, he did not perform the laborious analysis himself, and a lot of the credit for the results must go to Thomson's secretary, William Anderson, who apparently solved the kinetic equations for more than five thousand molecular collisions using nothing more than a pencil and a mechanical adding machine.



25



A short history of Monte Carlo

- The first real applications of the statistical sampling method to research problems in physics seem to have been those of Enrico Fermi, who was working on neutron diffusion in Rome in the early 1930s.
- Fermi never published his numerical methods; apparently he considered only the results to be of interest, not the methods used to obtain them, but according to his influential student and collaborator Emilio Segrè those methods were, in everything but name, precisely the Monte Carlo methods later employed by Ulam and Metropolis and their collaborators in the construction of the hydrogen bomb (Segrè 1980).
- In studying the behavior of the newly discovered neutron, he carried out sampling experiments about how a neutral particle might be expected to interact with condensed matter. These led to substantial physical insight and to the more analytical theory of neutron diffusion and transport.



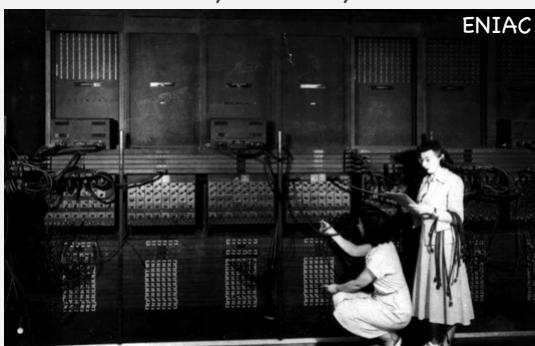
26

A short history of Monte Carlo

- So it was that when the Monte Carlo method finally caught the attention of the physics community, it was again as the result of armed conflict. The important developments took place at the Los Alamos National Laboratory in New Mexico, where Nick Metropolis, Stanislaw Ulam and John von Neumann gathered in the last months of the Second World War to collaborate on numerical calculations to be performed on the new **ENIAC** (Electronic Numerical Integrator And Computer) electronic computer, a mammoth, room-filling machine containing some 18000 triodic valves, whose construction was nearing completion at the University of Pennsylvania.



Metropolis



Ulam

A short history of Monte Carlo

- It seems to have been Stan Ulam who was responsible for reinventing Fermi's statistical sampling methods. He tells of how the idea of calculating the average effect of a frequently repeated physical process by simply simulating the process over and over again on a digital computer came to him whilst huddled over a pack of cards, playing patience one day.
- With his mind never far from the exciting new prospect of the ENIAC computer, the thought immediately crossed his mind that he might be able to get the machine to play these games for him far faster than he ever could himself, and it was only a short conceptual leap to applying the same idea to some of the problems of the physics of the hydrogen bomb that were filling his work hours at Los Alamos
- He later described his idea to John von Neumann who was very enthusiastic about it, and the two of them began making plans to perform actual calculations.
- The war ended before the first Monte Carlo calculations were performed on the ENIAC. There was some uncertainty about whether the Los Alamos laboratory would continue to exist in peace-time, and Edward Teller, who was leading the project to develop the hydrogen bomb, was keen to apply the power of the computer to the problems of building the new bomb, in order to show that significant work was still going on at Los Alamos.

A short history of Monte Carlo

- Von Neumann developed a detailed plan of how the Monte Carlo method could be implemented on the ENIAC to solve a number of problems concerned with neutron transport in the bomb, and throughout 1947 worked with Metropolis on preparations for the calculations.
- They had to wait to try their ideas out however, because the ENIAC was to be moved from Philadelphia where it was built to the army's Ballistics Research Laboratory in Maryland. For a modern computer this would not be a problem, but for the gigantic ENIAC, with its thousands of fragile components, it was a difficult task, and there were many who did not believe the computer would survive the journey. It did, however, and by the end of the year it was working once again in its new home.
- Before von Neumann and the others put it to work on the calculations for the hydrogen bomb, Richard Clippinger of the Ballistics Lab suggested a modification to the machine which allowed it to store programs in its electronic memory. Previously a program had to be set up by plugging and unplugging cables at the front of the machine, an arduous task which made the machine inflexible and inconvenient to use. Von Neumann was in favour of changing to the new "stored program" model, and Nick Metropolis and von Neumann's wife, Klari, made the necessary modifications to the computer themselves.

A short history of Monte Carlo

- It was the end of 1947 before the machine was at last ready, and Metropolis and von Neumann set to work on the planned Monte Carlo calculations. The early neutron diffusion calculations were an impressive success, but Metropolis and von Neumann were not able to publish their results, because they were classified as secret.
- Over the following two years however, they and others, including Stan Ulam and Stanley Frankel, applied the new statistical sampling method to a variety of more "mundane" problems in physics, such as the calculation of the properties of hard-sphere gases in two and three dimensions, and published a number of papers which drew the world's attention to this emerging technique.
- The calculations received a further boost in 1948 with the arrival at Los Alamos of a new computer, humorously called the MANIAC. (Apparently the name was suggested by Enrico Fermi, who was tiring of computers with contrived acronyms for names; he claimed that it stood for "Metropolis and Neumann Invent Awful Contraption").
- Apart from the advantage of being in New Mexico rather than Maryland, the MANIAC was a significant technical improvement over the ENIAC. It was faster and contained a larger memory (5 kilobytes!!!).

A short history of Monte Carlo

- The 1949 paper by Metropolis and Ulam on statistical techniques for studying integro-differential equations is of interest because it contained in its title the first use of the term "Monte Carlo" to describe this type of calculation.

JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION

Number 247

SEPTEMBER 1949

Volume 44

THE MONTE CARLO METHOD

NICHOLAS METROPOLIS AND S. ULAM
Los Alamos Laboratory

We shall present here the motivation and a general description of a method dealing with a class of problems in mathematical physics. The method is, essentially, a statistical approach to the study of differential equations, or more generally, of integro-differential equations that occur in various branches of the natural sciences.

- Also in 1949 the first conference on Monte Carlo methods was held in Los Alamos, attracting more than a hundred participants. It was quickly followed by another similar meeting in Gainesville, Florida.



A short history of Monte Carlo

- Of particular note to us is the publication in 1953 of the paper by Nick Metropolis, Marshall and Arianna Rosenbluth, and Edward and Mici Teller, in which they describe for the first time the Monte Carlo technique that has come to be known as the Metropolis algorithm. This algorithm was the first example of a thermal "importance sampling" method, and it is to this day easily the most widely used such method.

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*
(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANTAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

- Nowadays, thanks to the improvement of computational power of computers, Monte Carlo methods are used in many fields of Physics, Engineering, applied Mathematics, Theoretical Chemistry, Statistics, Quantitative Finance, Machine learning ...

