

Aula 1

Roteiro

- Três (tipos de) problemas
- *Monte Carlo to the Rescue*
- História da origem do nome
- Monte Carlo na Computação

Problema 1: Otimização

- Considere um espaço de estado discreto e bem grande
 - $S = \{1, 2, 3, \dots, n\}$, onde n é enorme
- Considere uma função que avalia a qualidade de cada estado
 - $f: S \rightarrow R$

Encontrar estado que maximiza a função

- $s_o = \operatorname{argmax}_{s \text{ em } S} f(s)$
- Força bruta: avaliar todos elementos de S é proibitivo

Problema 1: Exemplo 1

- $S = \text{todos os grafos rotulados com } n \text{ vértices}$
- Cada elemento de S é um grafo
 - $|S| = 2^{n(n-1)/2}$
- $f(s) = \text{distância média de } s$
- Encontrar grafo s_o que maximiza f
- Força bruta (testar todos) é proibitivo!
 - 1ms para cada grafo \rightarrow 1130 anos, se $n=10$

Como encontrar s_o de forma mais eficiente?

Problema 1: Exemplo 2

- $S =$ todas as permutações de n cartas
- Cada elemento de S é uma ordenação das cartas
 - $|S| = n!$
- $f(s) =$ número de subsequências crescentes em s
- Encontrar permutação s_o que maximiza f
- Força bruta (testar todos) é proibitivo!
 - 1ms para cada ordenação $\rightarrow 10^{57}$ anos, se $n=52$

Como encontrar s_o de forma mais eficiente?

Problema 2: Somatório

- Considere um somatório com muitos termos

$$G = \sum_{i=1}^n w_i$$

- w_i = valor da i -ésima parcela
- n = total de parcelas é enorme

**Calcular o valor do somatório
(de forma aproximada)**

- Força bruta: somar todos os termos é proibitivo

Problema 2: Exemplo 1

- Considere um jogo de cartas (determinístico)
 - ganhar ou perder depende apenas da sequência de cartas
- De quantas formas podemos vencer o jogo?

$$G = \sum_{i=1}^n w_i \leftarrow \quad w_i = 1 \text{ se sequência } i \text{ vence, ou 0 c.c.}$$

- Se baralho tem k cartas, $n = k!$
- Força bruta (todos os termos) é proibitivo
 - 1ms para cada ordenação $\rightarrow 10^{57}$ anos, se $k = 52$

Problema 2: Exemplo 2

- Considere todas sequências binárias com k bits
- Seja p_i a probabilidade associada a sequência i
- Qual é o valor esperado de P ?

$$E[P] = \sum_{i=1}^n p_i v_i \leftarrow \begin{array}{l} \text{definição de valor esperado,} \\ v_i \text{ é o valor da sequência } i \end{array}$$

- Com k bits temos, $n = 2^k$ sequências
- Força bruta (todos os termos) é proibitivo
 - 1ms para cada sequência \rightarrow 35 anos, se $k = 40$

Problema 3: Amostragem

- Considere um espaço de estado discreto
 - $S = \{1, 2, \dots, n\}$, com n enorme
- Considere uma função que associa a cada objeto um peso
 - $w_i > 0$ para todo i em S

**Escolher aleatoriamente um objeto
com probabilidade proporcional ao peso**

- Força bruta: enumerar todos para escolher é proibitivo!

Problema 3: Exemplo

- Todos os trios de amizade do Facebook
- Função que associa a cada trio o número de mensagens trocadas pelo trio

Escolher aleatoriamente um trio com probabilidade proporcional ao peso

- FB: 3 bilhões de perfis ativos
- Total de trios $\sim 10^{27}$
- 1 ms para gerar cada trio, 10^{16} anos

Monte Carlo to the Rescue

Resolvem todos estes problemas!

- Conjunto de técnicas para atacar problemas como estes (otimização, somatório, amostragem)
 - algoritmos ou métodos de Monte Carlo
 - não há uma definição geral bem aceita
- Muitas técnicas e algoritmos diferentes
 - ideia e aplicação surgiu na década de 40
- Fundamento de Monte Carlo: **aleatoriedade**
 - usar o acaso (chance) para computar!

Ideia central na computação moderna!

História da Origem

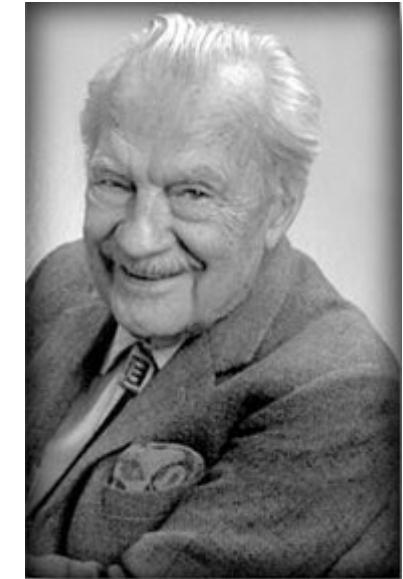
- **S. Ulam:** se recuperando de uma doença, pondera sobre as chances de ganhar o jogo de paciência (em 1946)
- Depois de fazer contas, sugere a ideia de jogar várias vezes para estimar



- **J. von Neumann:** vê o potencial e desenvolve programas para implementar a ideia
- Difusão de neutrinos e outros em reação nuclear, projeto Manhattan (origem da bomba atômica)

História da Origem

- **N. Metropolis:** sendo um projeto secreto, sugere um codinome para a técnica
- Monte Carlo: cassino em Mônaco onde jogava o tio de Ulam



- Algoritmos de Monte Carlo tem papel central no projeto Manhattan
- Resolvem problemas de otimização e cálculo de somatórios
- Implementado em um dos primeiros computadores: ENIAC

Ponto de Inflexão

Equation of State Calculations by Fast Computing Machines

The Journal of Chemical Physics **21**, 1087 (1953); <https://doi.org/10.1063/1.1699114>

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, and Augusta H. Teller

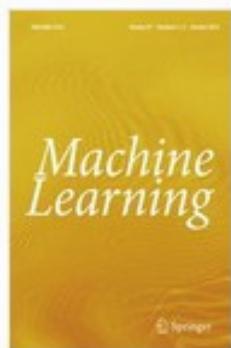
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

Edward Teller

Department of Physics, University of Chicago, Chicago, Illinois

- Metropolis e coautores (maridos e mulheres!) publicam uma técnica geral em 1953
 - artigo tem hoje 45 mil citações!
- Ideia é generalizada e amplamente utilizada em diversos ramos da ciência
 - algoritmo de Metropolis-Hasting
- Top-10 mais influentes algoritmos do Século XX pela revista IEEE Comput in Science & Engineering

Monte Carlo na Computação



[Machine Learning](#)

January 2003, Volume 50, [Issue 1–2](#), pp 5–43 | [Cite as](#)

An Introduction to MCMC for Machine Learning

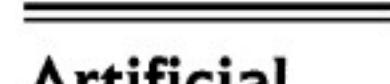
Authors

[Authors and affiliations](#)

Christophe Andrieu, Nando de Freitas, Arnaud Doucet, Michael I. Jordan

IEEE TRANSACTIONS ON COMPUTATIONAL INTELLIGENCE AND AI IN GAMES, VOL. 4, NO. 1, MARCH 2012

A Survey of Monte Carlo Tree Search Methods



Member, IEEE,
Subscriptions

IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 24, NO. 5, MAY 2002

Image Segmentation by Data-Driven Markov Chain Monte Carlo

Zhuowen Tu and Song-Chun Zhu

Sebastian Thrun^{a,*}, Dieter Fox^b, Wolfram Burgard^c, Frank Dellaert^a

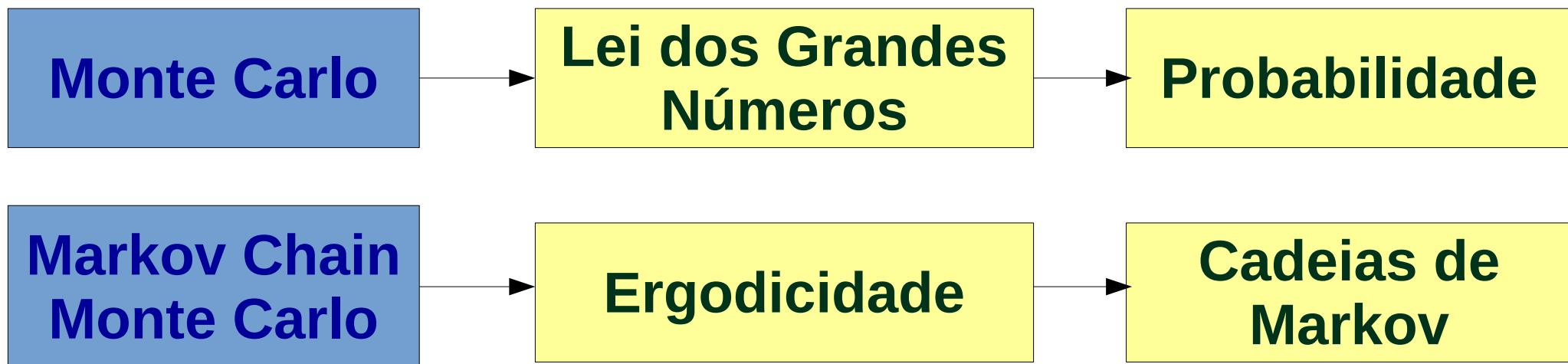
- Todos com centenas de citações!

Monte Carlo na Computação

- Amplamente empregado na computação
 - geralmente para resolver problemas (todos artigos anteriores)
- Computação também propõe algoritmos eficientes
 - modelagem do problema para melhor desempenho de algoritmos específicos
 - ex. variações do *simulated annealing*
- Monte Carlo está na moda na Computação
 - mudança de paradigma: computar usando dados e aleatoriedade
 - Monte Carlo tem tudo a ver com este novo paradigma

Monte Carlo nesta Disciplina

- Objetivo é aprender conceitos teóricos e técnicas fundamentais
 - permitem entender a base que sustenta Monte Carlo
 - não iremos aprofundar em nenhuma aplicação específica (mas seu projeto sim)



Estudaremos o amarelo para melhor entender o azul!