

# Aula 16

## Roteiro

- Os três problemas
- *Monte Carlo to the Rescue*
- Caminho trilhado
- Monte Carlo como *Building Block*
- Desafios

# Problemas de Somatório

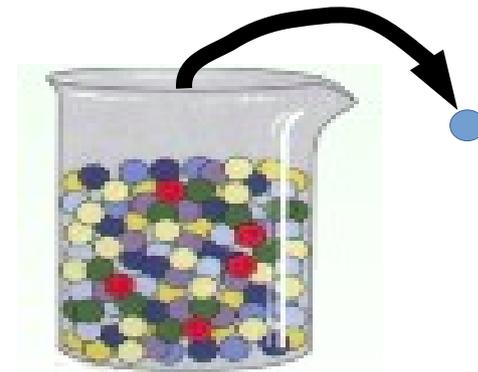
- Calcular o valor de um somatório (ou integral) de uma função  $g$ 
  - calcular valor esperado de v.a.

$$G_N = \sum_{i=1}^N g(i) \quad E_f[g(X)] = \sum_{i=1}^N g(i) f(i)$$

- Quando isso é difícil?
  - $N$  é muito grande (ex.  $> 10^{20}$ )
  - $g$  leva tempo para computar (ex. mais que linear)
- Exemplo
  - número de permutações de 52 cartas cuja soma das 10 primeiras cartas é menor do que 20

# Problemas de Amostragem

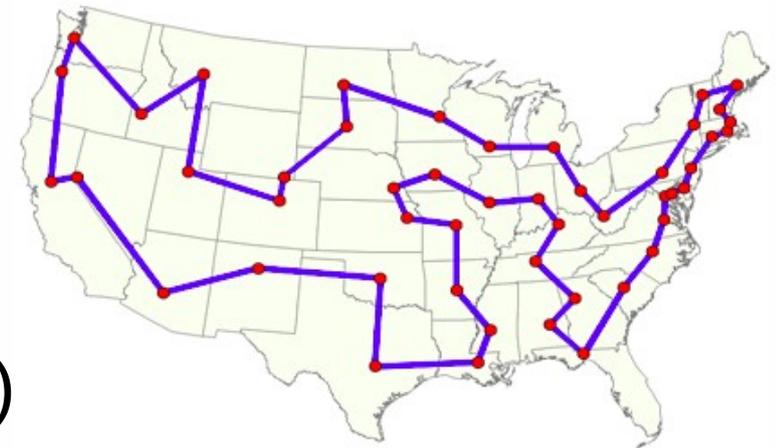
- Gerar amostra de um espaço amostral  $S$  com função de probabilidade  $f$ 
  - espaço simples (ex. números naturais) ou complicado (ex. grafos conexos)



- Quando isso é difícil?
  - função de probabilidade  $f$  é complicada
  - $|S|$  é muito grande (ex.  $> 10^{20}$ )
  - amostras são difíceis de gerar iterativamente
- Exemplo
  - gerar um conjunto independente (modelo *hardcore*) de grafo com probabilidade uniforme

# Problemas de Otimização

- Encontrar elemento de valor mínimo em espaço de estado discreto
  - cada elemento de  $S$  tem valor  $f(s)$
- Quando isso é difícil?
  - $|S|$  é muito grande (ex.  $> 10^{20}$ )
- Exemplo
  - encontrar menor percurso entre  $n$  cidades (caixeiro viajante)
  - $S =$  todos os percursos,  $|S| = n!$



# Monte Carlo to the Rescue!

## Algoritmos de Monte Carlo atacam todos estes problemas!

- Conjunto de técnicas para atacar problemas destes tipos (cálculo, amostragem e otimização)
  - atacar = resolver de forma aproximada
- Muitas técnicas e algoritmos diferentes
  - surgiu na década de 40, grande expansão de uso na computação nas últimas duas décadas
- Ideia central da técnica: **aleatoriedade**
  - usar o acaso (a chance) para computar!

# Monte Carlo está na Moda

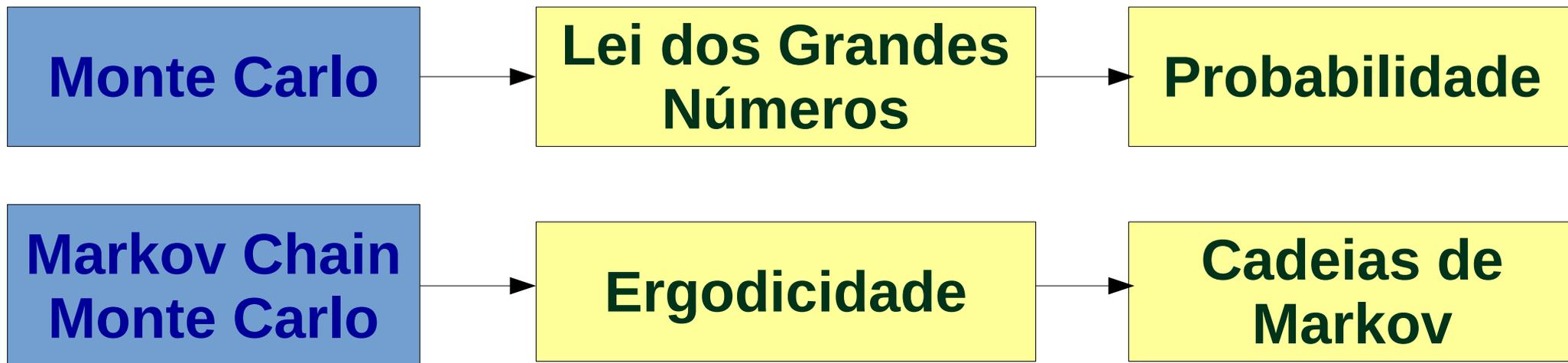
- Computação clássica
  - encontrar algoritmos provadamente eficientes para solução exata
- Computação moderna
  - encontrar algoritmos em geral eficientes que em geral dão boas soluções

## Mudança de paradigma!

- Fomentado pela mudança na complexidade dos problemas sendo atacados
  - ex. algoritmo que dirige um carro
- **Aleatoriedade:** técnica central da computação moderna
  - algoritmos de Monte Carlo visam usar aleatoriedade com rigor matemático

# Monte Carlo nesta Disciplina

- Objetivo: descobrir e se familiarizar com conceitos teóricos e técnicas fundamentais
  - entender o que está por de trás de Monte Carlo
  - sem aprofundar em nenhuma aplicação específica (em seu projeto sim)



Estudamos o amarelo para  
melhor entender o azul!

# Caminho Trilhado (1/2)

- Logística, apresentação e motivação (1 aula)
- Revisão de probabilidade (2 aulas)
  - eventos, independência, Bayes, v.a., binomial, valor esperado, conjunta
- Desigualdades e Lei dos grandes números (2 aulas)
  - Markov, Chebyshev, Chernoff, limitante da união, erro e confiança
- Monte Carlo para estimar somatório/integral (1 aula)
  - variância, erro
- Monte Carlo para gerar amostras (2 aulas)
  - transformada inversa, *rejection sampling*, *importance sampling*, simulação

# Caminho Trilhado (2/2)

- Cadeia de Markov (3 aulas)
  - distribuição no tempo, propriedades, distribuição estacionária, convergência, reversibilidade, decomposição em autovetores, tempo de mistura, vão espectral
- Markov Chain Monte Carlo (3 aulas)
  - caminho amostral, simulação, teorema ergódico, gerando amostras, Metropolis-Hastings, Gibbs Sampling
- *Simulated Annealing* (1 aula)
  - Distribuição de Boltzmann, definição, convergência, estratégia de resfriamento
- Encerramento (1 aula)
  - aula de hoje

# Monte Carlo e Você

- Muitos conceitos, resultados, e técnicas
  - visão geral da teoria sem focar em problemas específicos
- Ainda há muito a ser explorado
  - tanto na teoria quanto na prática
  - muitas aplicações com soluções específicas
- Monte Carlo é cada vez mais usado (e ainda pouco compreendido)
  - fração de pessoas que entendem MCMC está diminuindo
- Buscar entender e aplicar no seu problema
  - quando for o caso!

# MC como *Building Block*

- Cada vez mais aplicações vem empregando métodos de aprendizado de máquina
  - *AI everywhere!*
- Técnicas de aprendizado de máquina como *building blocks* para solução de problemas
  - fatorização de matrizes, clusterização de pontos, redes neurais, aprendizado por reforço
  - *frameworks* prontos para uso: *scikit-learn*, TensorFlow
- Monte Carlo é empregado como *building blocks* destas técnicas
  - estimação, amostragem, otimização
  - *Building block do building block!*

# Muitos Desafios

- Quantas passos/amostras são necessários (para garantir um erro menor que  $\varepsilon$ )?
  - ex. tempo de mistura em CM, ou geração de amostras para aproximar somatório
- Qual CM construir para gerar amostras ou buscar por solução ótima?
  - ex. para aplicar Metropolis-Hastings ou *simulated annealing*

## Perguntas importantes e difíceis!

- na teoria e na prática, de forma generalizada ou em problemas específicos

# Fim



- Perguntas e comentários?