AGA 511

Métodos Computacionais em Astronomia

Segundo semestre de 2020

Método de Monte Carlo

* Referência básica: caps. 2 e 3 da minha tese de doutorado (no Stoa)

* Origem na II Guerra Mundial
  ➜ propagação de nêutrons em meios materiais
  ➜ John von Neumann e Stanislas Ulam

* **Princípio básico do MMC**: simular um grande número de vezes a propagação de nêutrons individuais, levando em conta todos os eventos possíveis

* Pode ser aplicado sempre que cada processo físico envolvido possa ser descrito em termos de uma distribuição de probabilidades

Princípio Fundamental do MMC

* Seja um processo físico associado a uma variável x, definida no domínio [a, b]

* Exemplo: espalhamento da luz, definido entre 0 e 180 graus

* Função distribuição de probabilidades:

➜ p(x) dx é a probabilidade de x estar entre x e x + dx

➜ normalizada:

$$\int_a^b p(x')dx' = 1$$

➜ Função cumulativa de probabilidades:

$$P(x) = \int_a^x p(x')dx', \qquad 0 \leq P \leq 1$$

Princípio Fundamental do MMC

* A probabilidade de que P esteja no intervalo [P, P+dP] é a mesma que a variável x esteja em [x, x+dx]:

$$\wp(P)dP = p(x)dx$$

Como p(x) = dP / dx, temos que

$$\wp(P)dP = dP, \;\therefore\; \wp(P) = 1$$

Valores da função cumulativa de probabilidades estão distribuídos uniformemente entre 0 e 1.

Princípio Fundamental do MMC

* Segue-se que, para amostrar um valor de P, pode-se usar a seguinte relação:

$$P(x) = \xi$$

onde xi é um número aleatório entre 0 e 1. Portanto, para amostrar um valor x_s da variável x devemos resolver a seguinte equação:

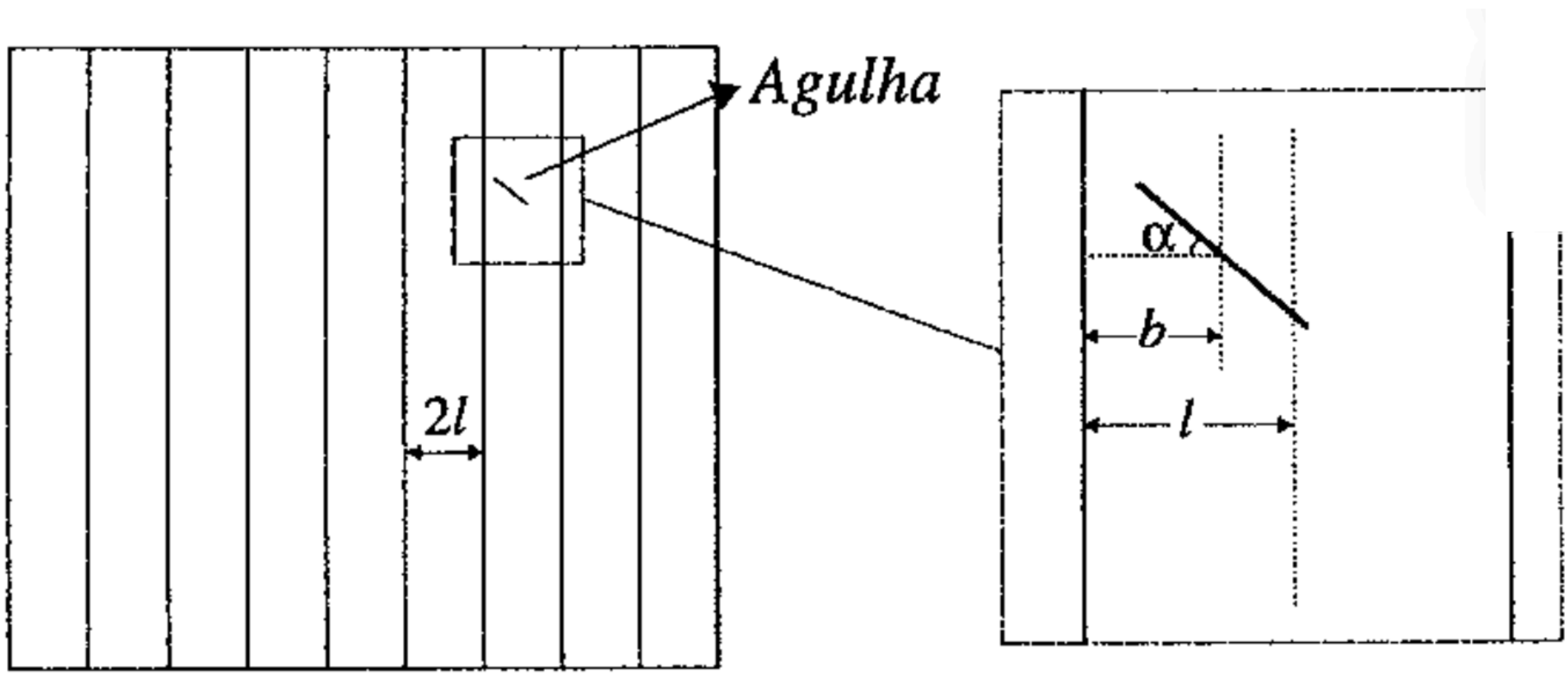
$$\xi = \int_a^{x_s} p(x)dx$$

Exemplo 1 - A agulha de Buffon

* Buffon’s needle: https://en.wikipedia.org/wiki/Buffon%27s_needle
* Agulha de comprimento l lançada N vezes sobre uma cartolina grande, com linhas paralelas separadas de 2l
* É possível mostrar que a probabilidade de um lançamento cair sobre uma linha é $1/\pi$
* Seja N_1 o número de vezes que a agulha cai sobre uma linha. É possível mostrar que

$$r = \frac{N_1}{N} = \frac{1}{\pi}$$

Exemplo 1 - A agulha de Buffon



(a) (b)

Exemplo 1 - A agulha de Buffon

* Implementação MC

* Modelo do sistema:

➜ 2 entidades: agulha e cartolina

➜ 1 processo: lançamento

➜ 2 eventos: distribuição discreta

* Amostragem de um evento:

➜ $b \in [0, l[\rightarrow b = \xi l$

➜ $\alpha \in [0, \pi/2[\rightarrow \alpha = \xi \pi/2$

* Condição para verificar se a agulha está sobre uma linha:

$$\frac{l}{2} \cos(\alpha) \geq b$$

Exemplo 1 - A agulha de Buffon

* Erro de Monte Carlo
* Seja r_e a razão experimental obtida a partir de uma simulação com N lançamentos.
* O erro cometido será da ordem de: $\sigma_{r_e} = \frac{r_e}{\sqrt{N}}$

Tabela 2.1 – Resultados para simulações da experiência da agulha.

N° de lançamentos	*N/N₁*	*Erro da Simulação*	*Erro real*
10	2,5	0,8	0,6
1.000	3,27	0,10	0,13
10.000	3,09	0,03	0,05
1.000.000	3,134	0,003	0,007
100.000.000	3,1419	0,0003	0,0003
500.000.000	3,1415	0,0001	0,0001

Exemplo 1 - A agulha de Buffon

* Curiosidade.
* Lazzarini (1901) - 34080 lançamentos
* Obteve: 3.1415929. Bom demais para estar correto!

Tabela 2.1 – Resultados para simulações da experiência da agulha.

N° de lançamentos	*N/N_1*	*Erro da Simulação*	*Erro real*
10	2,5	0,8	0,6
1.000	3,27	0,10	0,13
10.000	3,09	0,03	0,05
1.000.000	3,134	0,003	0,007
100.000.000	3,1419	0,0003	0,0003
500.000.000	3,1415	0,0001	0,0001

Método de Monte Carlo

Esse exemplo, apesar de muito simples, é útil para identificarmos os principais "ingredientes" de um código de Monte Carlo. Começa-se com a formulação de um modelo para o sistema estudado. Em seguida, identificam-se as *entidades* do modelo e os *processos* aos quais essas entidades estarão sujeitas. Finalmente, definem-se as *configurações* das entidades que são relevantes ao problema.

EP 1 - A agulha de Buffon

* Fazer um programa em C ou Fortran que resolva o problema da agulha de Buffon usando o método de MC
* Entrada: número de lançamentos a serem simulados
* Saída: estimativa de pi e seu erro respectivo
* Entregar:

1. Código fonte do programa por email
2. Gráfico mostrando como o erro experimental se compara com o erro real como função de N. O erro real se comporta como o esperado? Discuta.

EP 1 - A agulha de Buffon

3. Estimativa de quantos lançamentos são necessários para se calcular pi com 14 casas decimais, e quanto tempo isso leva em um CPU convencional

Data da entrega: 14/09

Será aceito o EP com atraso, mas será descontado 0,5 ponto por dia de atraso

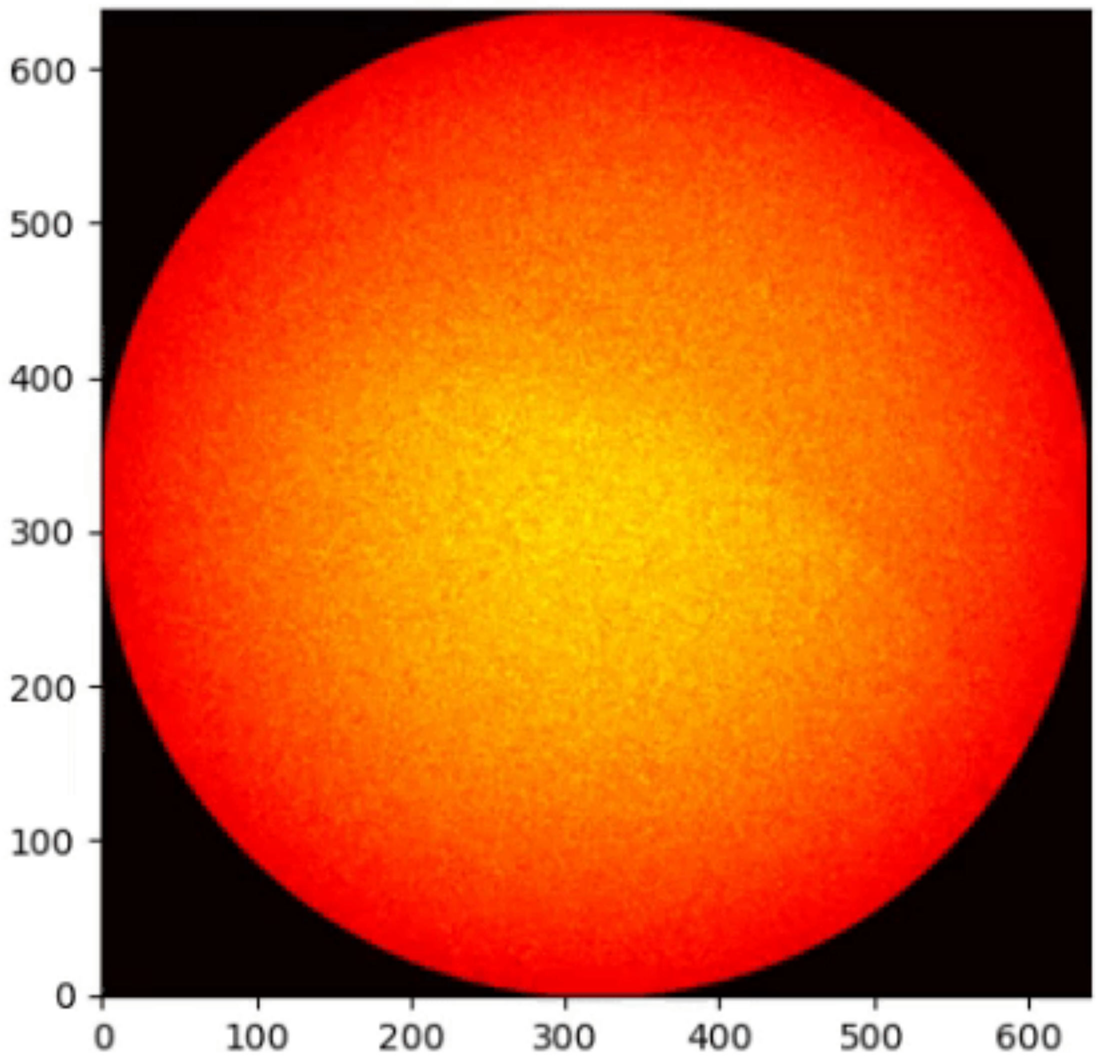
Projeto - Imagem de uma Estrela

Fazer um programa que calcule a imagem de uma estrela com manchas estelares em sua superfície usando o método de Monte Carlo. O programa deve simular o lançamento de fótons da superfície estelar e o registro destes fótons por observadores.

Fazer um video mostrando a rotação da estrela

Elementos do problema:

* Estrela, fótons, observadores
* Processo físicos: emissão dos fótons, escurecimento de bordo
* Configurações: número de manchas, tamanho relativo em relação à estrela, intensidade relativa, posições dos observadores


600
500
400
300
200
100
0
0
100
200
300
400
500
600

Projeto - Imagem de uma Estrela

Procedimento básico:

1. Amostrar um ponto aleatório na superfície da estrela
2. Amostrar uma direção aleatória de propagação do fóton emitido
3. Registrar o fóton como visto por um observador
4. Repetir este procedimento um número suficiente de vezes para produzir uma imagem com razão sinal ruído suficiente

Projeto - Imagem de uma Estrela

Sistema de coordenadas.

Qual usar? Esférico? Cartesiano?

É indiferente…

Projeto - Imagem de uma Estrela

Sistema de coordenadas:

* Posição do fótons na estrela:

(x_e, y_e, z_e)

* Direção de propagação:

A direção $\hat{s}$ é caracterizada pelas coordenadas direcionais (u, v, w)

$$\begin{cases} u = \hat{i} \cdot \hat{s} \\ v = \hat{j} \cdot \hat{s} \\ w = \hat{k} \cdot \hat{s} \end{cases}$$

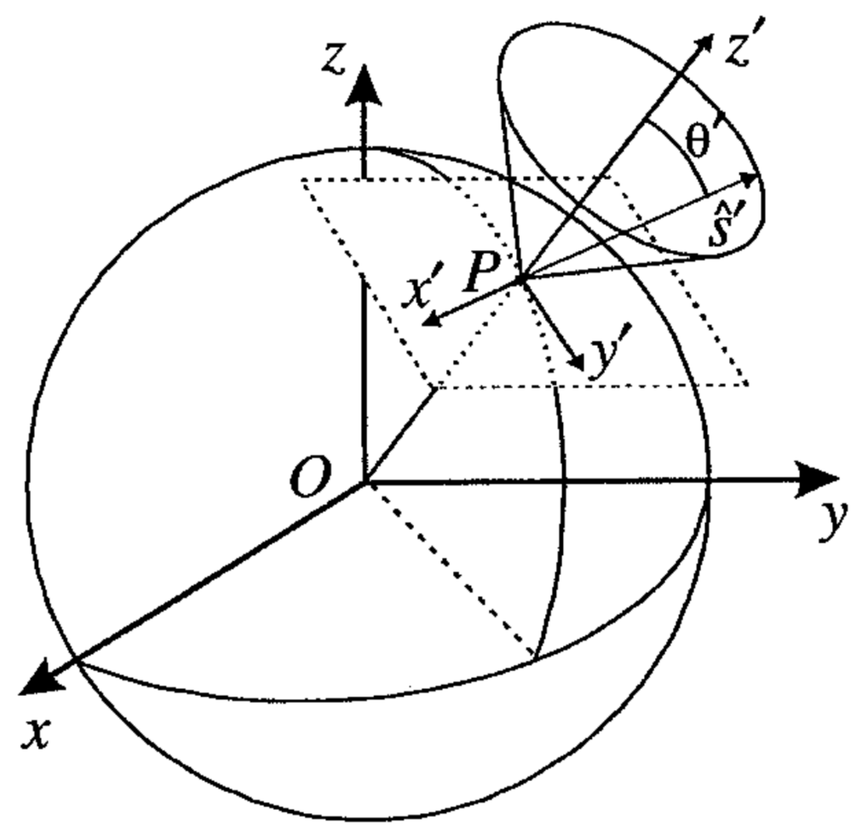


Figura 3.1 - Emissão da radiação por uma fonte esférica.

Projeto - Imagem de uma Estrela

1. <u>Amostragem de um ponto aleatório na superfície da estrela</u>

São necessários dois números aleatórios. O primeiro amostra z entre
-*R* e R e o segundo amostra o ângulo azimutal:

$$\phi = 2\pi\xi_2$$

DICA: neste problema, o tamanho da estrela é irrelevante. Então R = 1, sem perda de generalidade

$$\begin{cases} z_e = R(2\xi_1 - 1) \\ x_e = R\sqrt{1 - z_e^2}\cos(\phi) \\ y_e = R\sqrt{1 - z_e^2}\sin(\phi) \end{cases}$$

Projeto - Imagem de uma Estrela

2. Amostragem de uma direção aleatória de propagação do fóton emitido

Dada uma posição na superfície, sorteia-se uma direção de propagação dada por

$$\hat{s}' = (\mu', \phi')$$

Inicialmente sorteia-se μ' a partir do escurecimento de bordo

Depois, fazemos

$$\phi' = 2\pi\xi$$

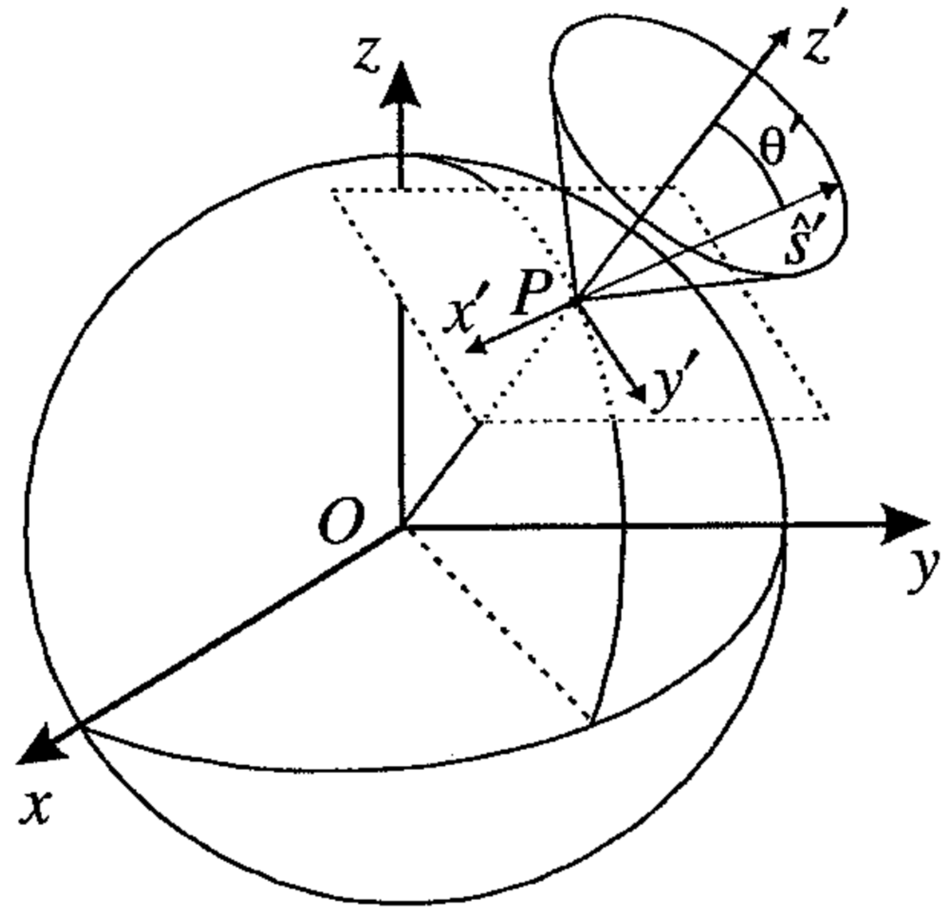
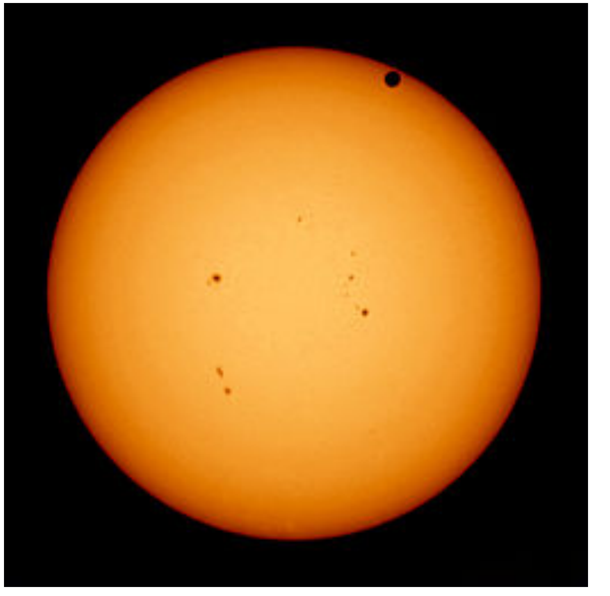


Figura 3.1 - Emissão da radiação por uma fonte esférica.

Projeto - Imagem de uma Estrela



Escurecimento de Bordo

Normalmente há uma dependência entre a intensidade emitida, I, e a direção radial (normal à superfície da estrela).

$$I = I(\cos\theta) = I(\mu)$$

Uma aproximação frequentemente usada para I é

$$\frac{I(\mu)}{I_0} = 1 + a\mu$$

Pode-se amostrar mu facilmente usando o princípio fundamental.

Projeto - Imagem de uma Estrela

2. Amostragem de uma direção aleatória de propagação do fóton emitido

Finalmente, basta escrever os cosenos diretores no sistema (x, y, z)

$$\hat{s} = \begin{cases} u_0 = -\sqrt{1-\mu'^2}\,\mathrm{sen}\,\phi\,\mathrm{sen}\,\phi' + \cos\phi\left[\sqrt{1-\mu'^2}\,z_e/R\cos\phi' + \mu'\sqrt{1-(z_e/R)^2}\right] \\ v_0 = \sqrt{1-\mu'^2}\cos\phi\,\mathrm{sen}\,\phi' + \mathrm{sen}\,\phi\left[\sqrt{1-\mu'^2}\,z_e/R\cos\phi' + \mu'\sqrt{1-(z_e/R)^2}\right] \\ w_0 = \mu' z_e/R - \sqrt{1-(z_e/R)^2}\sqrt{1-\mu'^2}\cos\phi' \end{cases}$$

Projeto - Imagem de uma Estrela

Definição dos observadores

Os observadores são definidos dividindo-se uma esfera imaginária em torno da estrela em “caixinhas” com o seguinte tamanho

$$\Omega_{\rm caixa} = \frac{4\pi}{N_\mu N_\phi}$$

N_μ é o número de divisões em latitude

N_ϕ é o número de divisões em azimute

Portanto, o número total de "observadores" é $N_\phi N_\mu$

Um dado observador é caracterizado por dois índices: imu, iphi

Projeto - Imagem de uma Estrela

3. Registrar o fóton como visto por um observador

Para registrar o fóton emitido, inicialmente determina-se os índices imu e icphi:

```
imu = 0.5*(w+1)*nmu + 1.

tmp = ATAN2(v,u)

IF (tmp < 0.) tmp = tmp + 2.*pi

iphi = tmp*nphi/2./pi+1.
```

NOTA: a variável `pi` precisa ser definida pelo usuário

Projeto - Imagem de uma Estrela

3. <u>Registrar o fóton como visto por um observador</u>

Depois determina-se o parâmetro de impacto da posição do fóton sobre o disco da estrela como visto pelo observador. Inicialmente define-se o sistema x', y', z' girando o sistema x, y, z em torno do eixo z por um ângulo phi. Neste sistema, a direção do observador fica no plano x'- z'.

Em seguida obtém-se o sistema x'', y'', z'' girando y' por 90-acos(w), de forma que x'' se alinhe com a direção do observador.

As coordenadas y'', z'' correspondem ao parâmetro de impacto procurado.

Projeto - Imagem de uma Estrela

```
Código em FORTRAN para calcular o parâmetro de impacto (xpp
e ypp). x: array de 3 números contendo a posição do fóton

raiz = SQRT(1.-w*w)
IF (raiz == 0.) THEN
   zpp = -x(1)
   ypp =  x(2)
ELSE
   st = raiz
   raiz = 1.0/raiz
   cf = u*raiz
   sf = v*raiz
   ct = w

   ypp = -sf*x(1) + cf*x(2)
   dum =  cf*x(1) + sf*x(2
   zpp = -dum*ct + x(3)*st
   xpp =  dum*st + x(3)*ct
END IF
```

IMPORTANTE: no plano do céu, x -> ypp e y -> zpp

Projeto - Imagem de uma Estrela

3. <u>Registrar o fóton como visto por um observador</u>

Finalmente, dado o parâmetro de impacto, deve-se determinar os índices do “pixel" da imagem a que este parâmetro de impacto corresponde: ix e iy.

Isto é deixado como exercício. 🙌🏻

Como cada fóton corresponde a uma unidade (arbitrária, neste caso) de energia, o “registro" de um fóton corresponde a somar 1 a um array definido como:

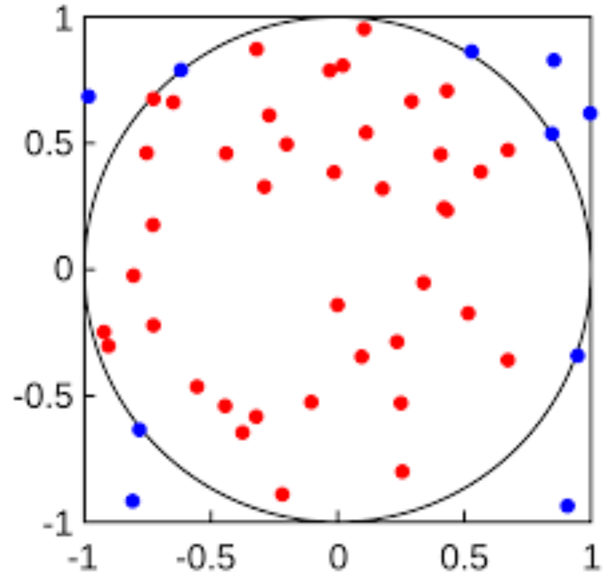
```
REAL, ALLOCATABLE :: image(:,:,:,:)
…
ALLOCATE (image(nx,ny,nmu,nphi))
```

Projeto - Imagem de uma Estrela

Discussão em aula: como implementar uma mancha estelar?

Outras ideias para projetos

* <u>Integração Multi-Dimensional usando MC</u>



Bibliografia:

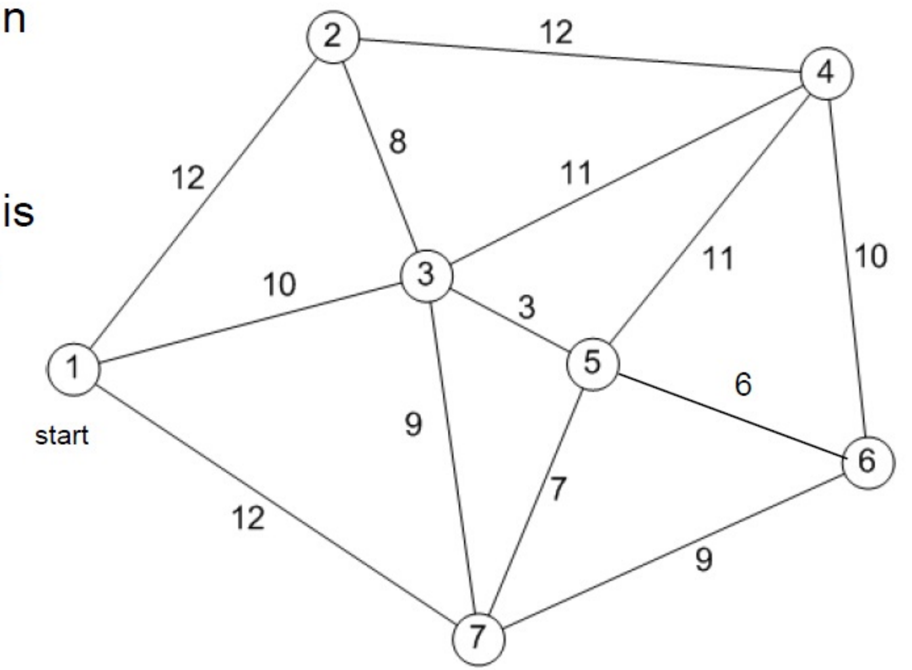
https://www.ime.usp.br/~jstern/miscellanea/LabSimulacao/EricVeach2.pdf

Outras ideias para projetos

* O problema do Caixeiro Viajante

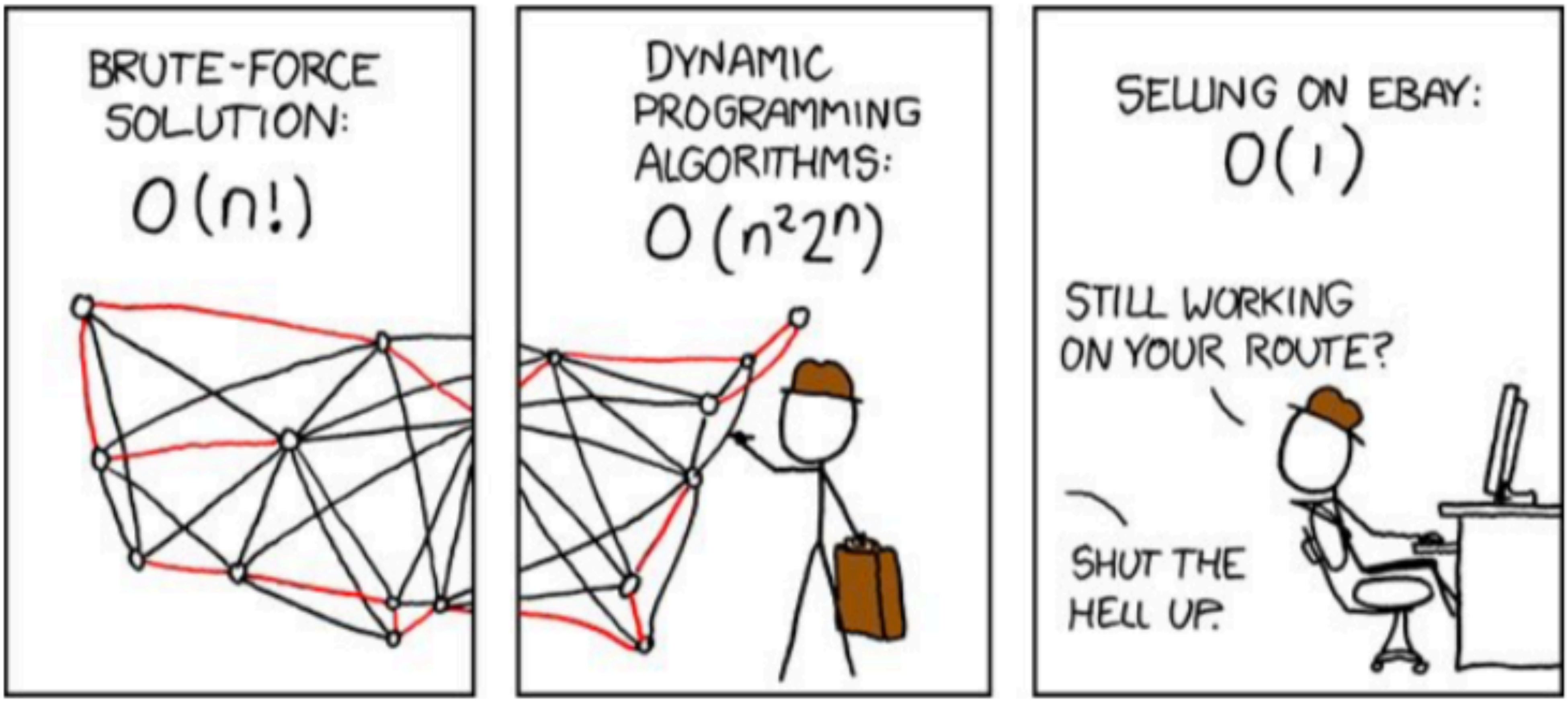
The Traveling Salesman Problem

- Starting from city 1, the salesman must travel to all cities once before returning home
- The distance between each city is given, and is assumed to be the same in both directions
- Only the links shown are to be used
- Objective - Minimize the total distance to be travelled



Outras ideias para projetos

* O problema do Caixeiro Viajante



Outras ideias para projetos

* Algum problema interessante da sua IC?
    * Automação de tarefas
    * Repetição de procedimentos
* Hidrodinâmica
* Transporte Radiativo
    * Espalhamento eletrônico da luz polarizada :-)
* Markov Chain Monte Carlo usando o algoritmo de metrópolis

EP 2 (Projeto Individual)

Data de entrega: 12/10

De preferência em C ou Fortran

Fazer estudo de desempenho

Lição de Casa para Semana que vem!

1) Estudar os problemas propostos em aula e definir qual será o tema do seu EP2
2) Estudar as quatro ferramentas propostas para o curso, e escolher com qual delas vai trabalhar:
    a) MPI - C++ ou FORTRAN
    b) OpenMP - C++ ou FORTRAN
    c) CUDA - C++
    d) OpenACC

Semana que vem: definição dos projetos e formação das equipes de trabalho