

Trabalhando com arquivos de dados

Derivada numérica de dados

Matheus Roos

Universidade Federal de Santa Maria

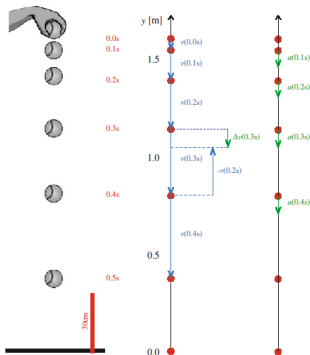
20 de julho de 2022

- 1 Motivação
- 2 O Problema
 - O desafio
- 3 Arquivo de dados
 - Definição
 - Declarações de i/o
 - Atributos da declaração OPEN
- 4 Retornando para a cinemática
 - Definindo a velocidade
 - Definindo a aceleração
 - Derivada numérica de dados

- Situação onde apenas dados são fornecidos;
- Áreas como a astrofísica recebem dados dos observatórios;
- Então como trabalhar com os arquivos de dados de terceiros?
- Como calcular a derivada numérica sem um função previamente conhecida?

O Problema

- Vamos tomar como base o exemplo apresentado no livro Elementary Mechanics Using Python, Ander Malthe-Sørensen:



- Onde uma bola de tênis é solta de uma altura de aproximadamente 1,5m a partir do repouso;
- Vamos primeiramente fazer um plot destes dados;

- Nosso interesse é a partir destes dados, tirar informações da física do fenômeno/experimento;
- Começamos descrevendo o movimento;
- Definimos o deslocamento $\Delta x(t_0)$ no intervalo de $t = t_0$ a $t = t_0 + \Delta t = t_1$ como:

$$\Delta x(t_0) = x(t_0 + \Delta t) - x(t_0) = x(t_1) - x(t_0)$$

- Vamos então calcular o deslocamento;

Arquivos de dados

- Quando estivermos trabalhando com um grande volume de dados, devemos optar pelo uso do recurso dos arquivos de dados em detrimento a imprimir os valores na tela;
- O arquivo de dados é armazenado na memória secundária (a memória RAM é primária);
- A memória secundária é mais lenta que a memória principal do computador, mas ainda assim, permite acesso rápido aos dados;
- Nos primórdios da computação o dispositivo de armazenamento secundário eram as fitas magnéticas;
- Estas fitas magnéticas armazenavam dados de forma semelhante as fitas cassete;



Fonte: <https://www.ehow.com.br>

- As fitas magnéticas eram lidas de forma sequencial, sem a possibilidade da leitura saltar para outro ponto;
- Acesso sequencial vs acesso restrito;
- Por razões históricas o Fortran trabalha com acesso sequencial.
- A unidade de input/output é às vezes chamada de unidade lógica ou simplesmente unidade;
- E corresponde ao primeiro asterisco de `WRITE(*,*)` ou `READ(*,*)`;
- O asterisco corresponderá a unidade padrão de escrita/leitura;
- O número correspondente a unidade deverá ser um inteiro;
- Existem várias instruções Fortran para manipular a leitura/gravação dos dados nestes arquivos de dados;

Declarações de i/o

Declaração i/o	Descrição
OPEN	Associa um arquivo com um n ^o específico de unidade
CLOSE	Finaliza a associação de um arquivo com uma unidade
READ	Lê dados de uma unidade
WRITE	Escreve dados numa unidade
REWIND	Move para o início de um arquivo de dados
BACKSPACE	Volta um registro de um arquivo de dados.

- A declaração

`OPEN(lista_argumentos)`

- possui diversos atributos disponíveis, que serão acessados através da lista de argumentos;
- O primeiro destes atributos especifica a unidade, sendo obrigatório;
- O segundo designa um nome do arquivo de dados, apesar de não ser obrigatório é altamente recomendado;
- Os demais especificarão como se dará a leitura. Todos opcionais.

Atributos da declaração OPEN

Os atributos acessíveis através da lista de argumentos, que já conhecemos, são:

- UNIT: indica o número da unidade ao qual associar o arquivo de dados, aceitando apenas valores inteiros:

UNIT=numero_inteiro

- FILE: especifica o nome do arquivo a ser aberto, aceitando valores do tipo character, escrevemos da seguinte forma:

FILE= nome_do_arquivo

Temos como novidade o atributo STATUS: define o status do arquivo a ser aberto, aceitando valores do tipo character previamente estabelecidos, tal que,

STATUS=expressao_status

onde expressao_status poderá assumir os valores:

- 'OLD': especifica que o arquivo a ser aberta já existe;
- 'NEW': especifica que o arquivo que será aberto não existe. Padrão;
- 'REPLACE': se o arquivo já existir ele substituirá, caso não exista ele irá criar um novo arquivo;
- 'SCRATCH': rascunho, sem nome de arquivo, sendo de leitura e escrita;
- 'UNKNOWN': Desconhecido.

- ACTION: Especifica a ação a ser tomada ao abrir o arquivo, apenas leitura, apenas escrita, ou então leitura e escrita (padrão), ele aceitará valores do tipo character pré-estabelecidos, na forma;

ACTION=expressao_action

expressao_action poderá assumir apenas os seguintes valores:
'READ', 'WRITE' ou então 'READWRITE' (padrão);

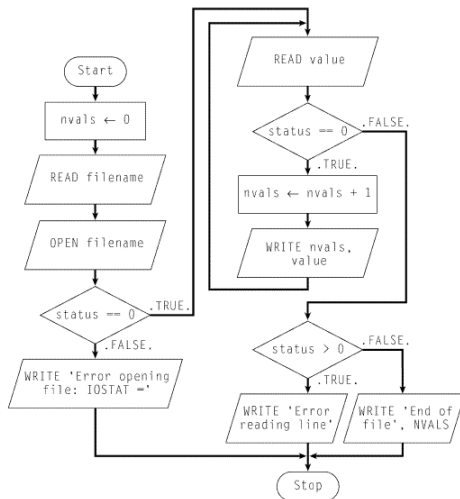
- IOSTAT: Muito importante para sinalizar erros. Se a abertura do arquivo for concluída com êxito, IOSTAT retornará um 0, caso haja algum erro retornará um inteiro diferente de zero.
- Obs: Depois que o arquivo é fechado, a unidade de i/o que estava associada a ele fica livre para ser reatribuído a qualquer outro arquivo em uma nova instrução OPEN.

A cláusula IOSTAT

- Podemos utilizar a cláusula IOSTAT também em declarações READ;
- Se a leitura for bem sucedida, IOSTAT retornará zero;
- Caso haja algum erro, ele retornará um inteiro positivo diferente de zero;
- Caso o final do arquivo seja atingido, IOSTAT retornará um inteiro negativo¹;
- No FORTRAN 2003 foi incluído o IOMSG, que ao invés de retornar um número, ele devolverá uma texto informando do que se trata o erro.
- O programa será abortado caso a declaração READ não tenha o atributo IOSTAT;

¹Apesar de existirem outras formas de fazer isso

Fluxograma



Fonte: Stephen Chapman, Fortran 95, 2003 for scientists and engineers (2007).

Definindo a velocidade

- A questão agora é determinar o quão rápido o objeto cai, mas não podemos utilizar apenas o deslocamento;
- O motivo disso é que os deslocamentos se tornam menores a medida que diminuimos o intervalo de tempo;
- Então definimos a velocidade média por:

$$v_m(t_1) = \frac{\Delta x}{\Delta t} = \frac{x(t_0 + \Delta t) - x(t_0)}{\Delta t} = \frac{x(t_1) - x(t_0)}{\Delta t}.$$

- E por conseguinte, definimos a velocidade instantânea:

$$v = \lim_{\Delta t \rightarrow 0} \frac{x(t_0 + \Delta t) - x(t_0)}{\Delta t} = \frac{dx}{dt}$$

- Vamos agora calcular a velocidade média a partir dos dados fornecidos.

Definindo a aceleração

- Analogamente ao que fizemos no caso da velocidade média, definimos a aceleração média como o quão rápido a velocidade muda, escrevemos então,

$$a_m(t_1) = \frac{\Delta v}{\Delta t} = \frac{v(t_0 + \Delta t) - v(t_0)}{\Delta t} = \frac{v(t_1) - v(t_0)}{\Delta t},$$

- E por associação, definimos a aceleração instantânea:

$$a = \lim_{\Delta t \rightarrow 0} \frac{\Delta v}{\Delta t} = \frac{dv}{dt}.$$

- Dada a natureza do problema, modelamos ele por:

$$y(t) = y_0 + v_0 t + \frac{1}{2}gt^2,$$

- cuja equação para a velocidade será

$$v(t) = v_0 + gt,$$

- com aceleração igual a

$$a = g = 9,81m/s^2.$$

Esboçando o algoritmo

- Devemos transformar o desenvolvimento teórico feito anteriormente em código, então definimos a "velocidade média numérica" como:

$$\bar{v}_i = \frac{y_i - y_{i-1}}{t_i - t_{i-1}} = \frac{y_i - y_{i-1}}{\Delta t},$$

- e a aceleração média numérica

$$\bar{a} = \frac{\bar{v}_i - \bar{v}_{i-1}}{t_i - t_{i-1}} = \frac{\Delta \bar{v}}{\Delta t}$$

- Devemos nos recordar que não conhecemos $y(t)$ para todos os valores de t .
- mas apenas os valores que foram medidos de $y(t_i)$.
- Não podemos encontrar uma expressão **analítica exata** para a derivada de $y(t)$.
- Mas podemos aproximar a velocidade instantânea da velocidade média, de t_{i-1} a t_i :

$$\bar{v}_i(t_i) = \frac{y(t_i + \Delta t) - y(t_i)}{\Delta t} \simeq v(t_i) = \frac{dy}{dt}.$$

- Adotamos o mesmo procedimento para a aceleração:

$$\bar{a}_i(t_i) = \frac{v(t_i + \Delta t) - v(t_i)}{\Delta t} \simeq a(t_i) = \frac{dv}{dt}.$$

- Ao final deveremos comparar o modelo teórico com o experimental.