

Licenciatura em Engenharia Informática

Sistemas Multimédia

Introdução

Telmo Reis Cunha

Departamento de Eletrónica, Telecomunicações e Informática

Universidade de Aveiro – 2021/2022

1. Objetivos

- Apresentar e dar a entender o contexto e relevância dos sistemas multimédia;
- Indicar a necessidade e aprender as técnicas de representação eficiente de informação, com particular ênfase na informação considerada em sistemas multimédia;
- Apresentar e dar a entender as principais tecnologias e normas para a representação eficiente e processamento da informação inerente aos sistemas multimédia;
- Dar a conhecer os principais fatores relacionados com o desempenho de sistemas multimédia.

2. Contextualização

Sistema Multimédia... o que é?

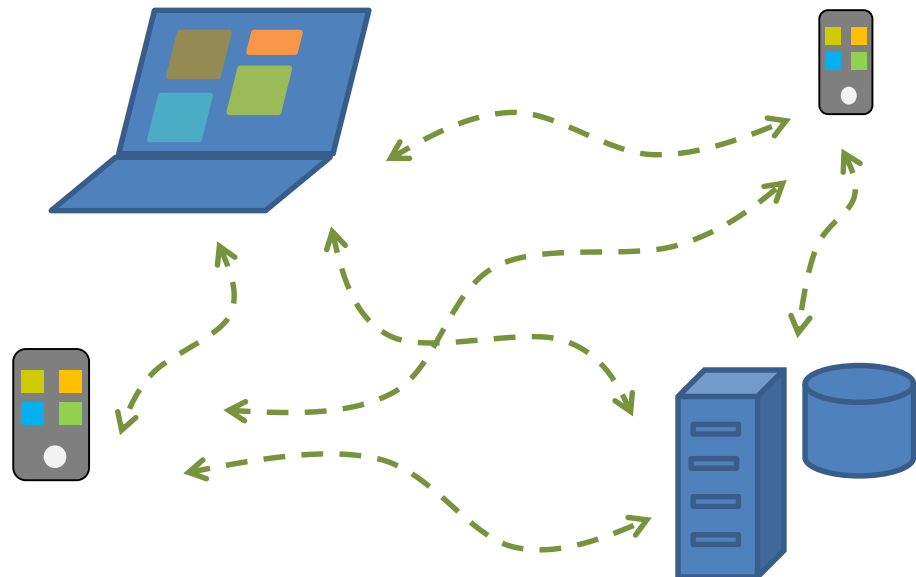
- Aplicação e/ou dispositivo que faz uso (tira partido) simultâneo de **informação** em diversos formatos:
 - Texto
 - Imagens
 - Desenhos / gráficos
 - Animação
 - Vídeo
 - Áudio (incluindo voz)
 - Interfaces interativas
 - ...

2. Contextualização

Exemplos de Sistemas Multimédia

- Videoconferência
- Telemedicina
- Motores de pesquisa em bases de dados de imagem
- Realidade aumentada
- Páginas web
- ...

Aquisição, transmissão e processamento de grandes quantidades de **informação!**



2. Contextualização

Exemplo: Transmissão de imagens digitais

- Resolução: 3264 x 1836 (≈ 6 Mpixel)
- Imagem a cores RGB de 3 Bytes/pixel (≈ 18 MB)
- Transferir 100 imagens (≈ 1.8 GB)
- Transferência via Wi-Fi, norma g (≈ 4 MB/s)
- **Tempo de transferência: 7.5 minutos**
- Comprimir imagem?
- JPEG: ≈ 2.6 MB/imagem (exemplo)
- **Tempo de transferência: 65 seg.**
- O que se dá em troca?



Wi-Fi



2. Contextualização

Exemplo: Seleção do Formato de Compressão

Áudio:

- MP2, MP3, SBC, True Audio (TTA), WavPack, WAV, Speex, ...

Voz:

- GSM, CELL, G.711, G.721, ...

Imagem:

- JPEG, TIFF, GIF, BMP, PNG, MetaFile, ...

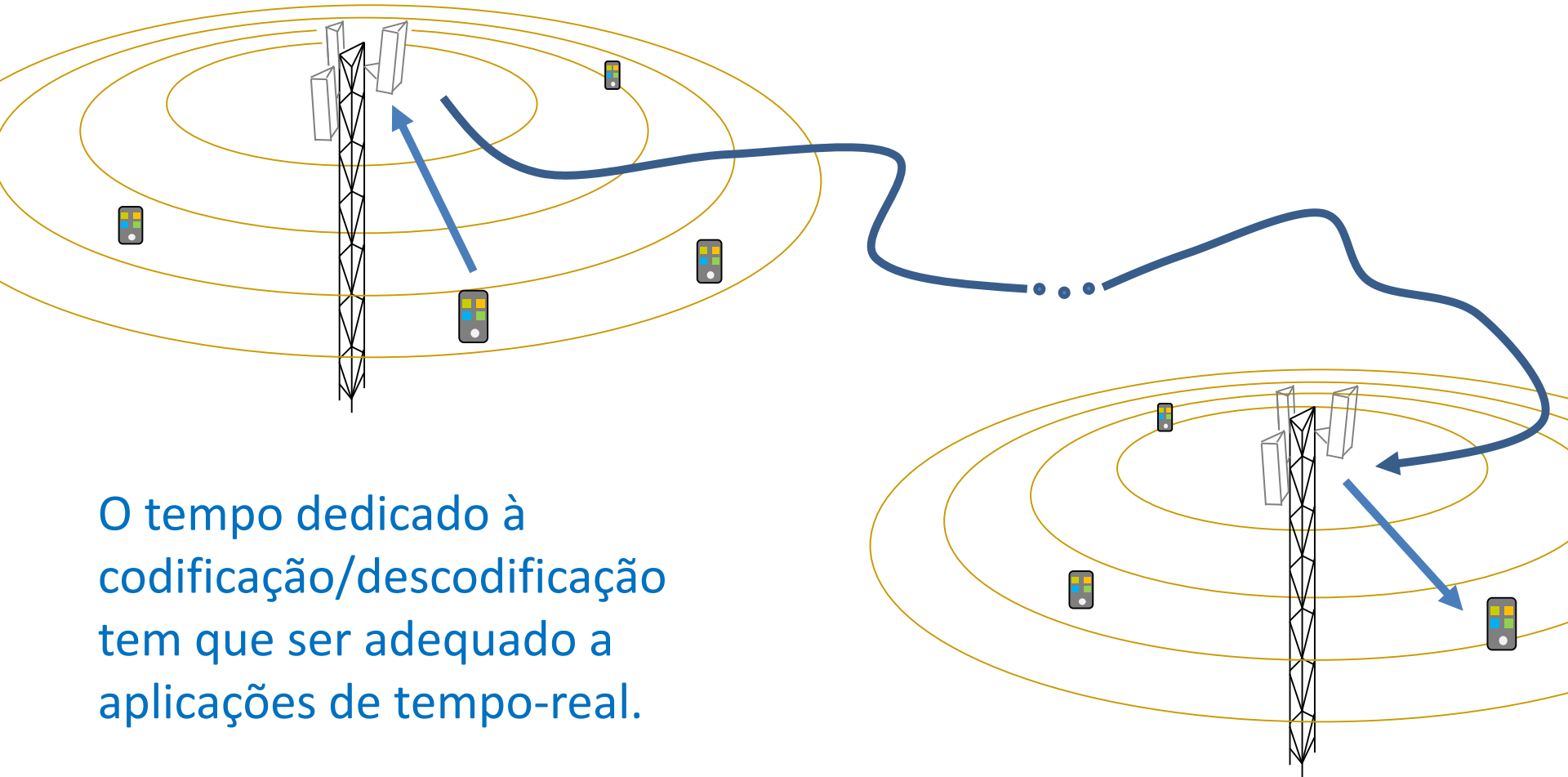
Vídeo:

- MPEG-2, MPEG-4, H.120, VC-2, ...

Vantagens / Desvantagens?
Qual escolher?

2. Contextualização

Exemplo: Atraso de Codificação



3. Programa

Capítulo 1 – Fundamentos:

- Conceito de sinal e sistema. Exemplos de sinais: de som e de imagem.
- Sinais sinusoidais.
- Revisão da série de Fourier.
- Números complexos (revisão) e sua utilização na representação de sinais.
- Conteúdo de frequência de um sinal. Exemplos em sinais de som e imagem.
- Definição de DFT (*Discrete Fourier Transform*) e DCT (*Discrete Cosine Transform*), e suas aplicações (1D e 2D).

3. Programa

Capítulo 2 – Amostragem de Sinais:

- Fundamentos da teoria de amostragem de sinais.
- Reconstrução de sinais amostrados.
- *Aliasing*: definição e exemplos.
- Quantização da amplitude e ruído de quantização.
- Sistemas de conversão A/D e D/A – arquiteturas e exemplos de aplicação.
- Número de bits efetivos (ENOB) de um conversor.

Capítulo 3 – Codificação de Sinais:

- Fundamentos e noção de código.
- Exemplos de códigos usados na codificação de sinais.
- Noção e definição de Entropia.
- Códigos de Huffman.

3. Programa

Capítulo 4 – Compressão de Informação:

- Compressão com e sem perdas.
- Técnicas de compressão para sinais multimédia.
- Algoritmos de compressão de imagem: raw, PNG, TIFF, JPEG, JPEG2000, etc.
- Compressão de sinais de voz e música: MPEG 3.
- Compressão de sinais de vídeo: HDTV, PAL, MPEG 2.

4. Avaliação

Avaliação discreta na componente prática (50% da nota final):

- 3 trabalhos a realizar durante 3 aulas práticas (15-19/Nov; 13-17/Dez; 19-24/Jan).
- Resolvidos individualmente (ou em grupos de 2 alunos, se as restrições relativas à pandemia o permitirem).
- Cotação igual para todos os trabalhos.

Avaliação por exame na componente T/TP (50% da nota final):

- Realizado na época normal de exames.
- Requer a obtenção de nota mínima de 8.0 valores (em 20).

As classificações da componente prática obtidas na edição de 2020/2021 são consideradas válidas para a corrente edição (desde que tenham sido positivas, naturalmente).

5. Bibliografia

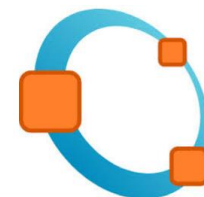
- James McClellan et al., “Signal Processing First”, Prentice Hall, 2003 (SDUA 621.391G.68)
- Nuno Ribeiro e José Torres, “Tecnologias de Compressão Multimédia”, FCA, 2009 (SDUA 004.9G.115)
- Ze-Nian Li and Mark S. Drew, “Fundamentals of Multimedia”, Pearson Prentice Hall, 2004.
- José M. N. Vieira, “Matlab num Instante”

6. MATLAB / Octave

- **MATLAB** – Aplicação disponível na UA para simulação numérica, cálculo, operação sobre sinais, etc.

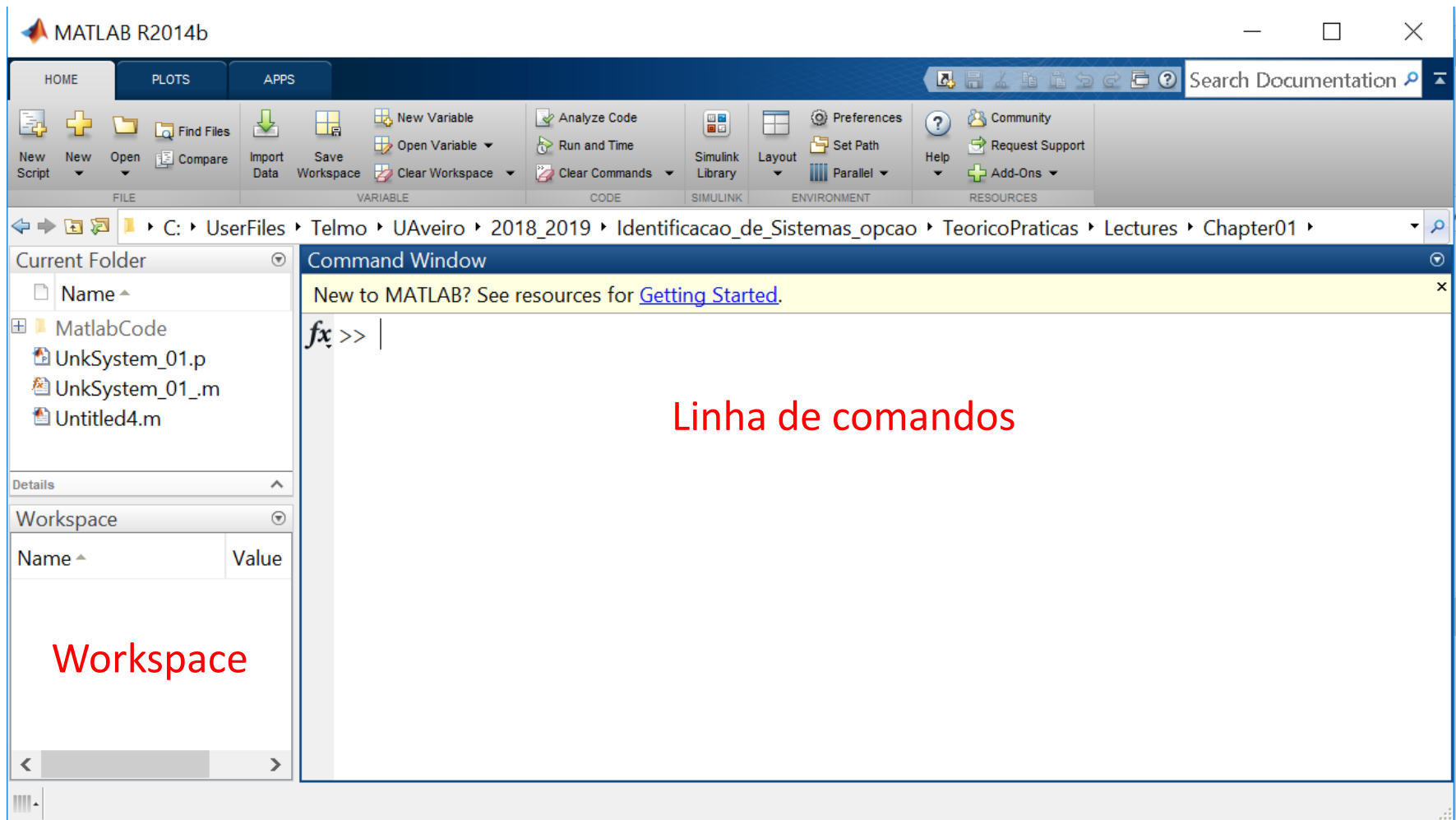


- Possui ferramentas que permitem:
 - Desenvolver novas funções e algoritmos;
 - Operar facilmente sobre sinais (de voz, de imagem, ...);
 - Visualizar graficamente a informação;
 - ...
- **Octave** – aplicação *open-source* semelhante ao MATLAB, com sintaxes semelhantes.



6. MATLAB / Octave

- Interface



6. MATLAB / Octave

- **Declarar variáveis:**

```
>> z = 2;      % Cria a variável z no workspace e atribui-lhe o valor 2.
```

```
>> y = 3*z^2+5; % A variável y é criada, com o valor resultante da operação.
```

```
>> v1 = [1 2 3 4 5]; % Vetor linha de 1x5 elementos.
```

```
>> v2 = [1; 2; 3; 4; 5]; % Vetor coluna de 5x1 elementos.
```

```
>> v3 = [1 2 3 4 5]'; % Vetor coluna igual a v2 ( ' produz a matriz transposta).
```

```
>> A1 = [1 2 3; 4 5 6]; % Matriz 2x3.
```

```
>> A2 = [A1; [7 8 9]]; % Matrix 3x3.
```

```
>> A3 = v2*v1; % Matrix 5x5.
```

```
>> a = v1*v2; % Matrix 1x1 (escalar).
```

```
>> v4 = v2.*v3; % Vetor coluna 5x1 (. indica operação elemento-a-elemento).
```

6. MATLAB / Octave

- **Declarar variáveis e funcionalidades:**

```
>> t = 1 : 5;      % Resultado: t = [1 2 3 4 5].
```

```
>> t = [1 : 5];    % Resultado: t = [1 2 3 4 5].
```

```
>> t = [1 : 0.5 : 3]; % Resultado: t = [1  1.5  2  2.5  3].
```

```
>> size(t) % Retorna as dimensões de t (ans= 1 5)
```

```
>> length(t) % Comprimento do vetor (ans= 5)
```

Para se ver como usar uma determinada função, executar:

```
>> help função
```


6. MATLAB / Octave

- **Indexar elementos de vetores e matrizes:**

```
>> A = [1 2 3; 4 5 6];
```

```
>> a1 = A(2,3); % Resultado: a1 = 6.
```

```
>> i = 1;
```

```
>> j = 2;
```

```
>> a2 = A(i,j); % Resultado: a2 = 2.
```

```
>> b1 = A(:,2); % Resultado: b1 = [2; 5].
```

```
>> b2 = A(1,2:end); % Resultado: b2 = [2 3].
```

```
>> B3 = A(1:2;2:3); % Resultado: B3 = [2 3; 5 6].
```

6. MATLAB / Octave

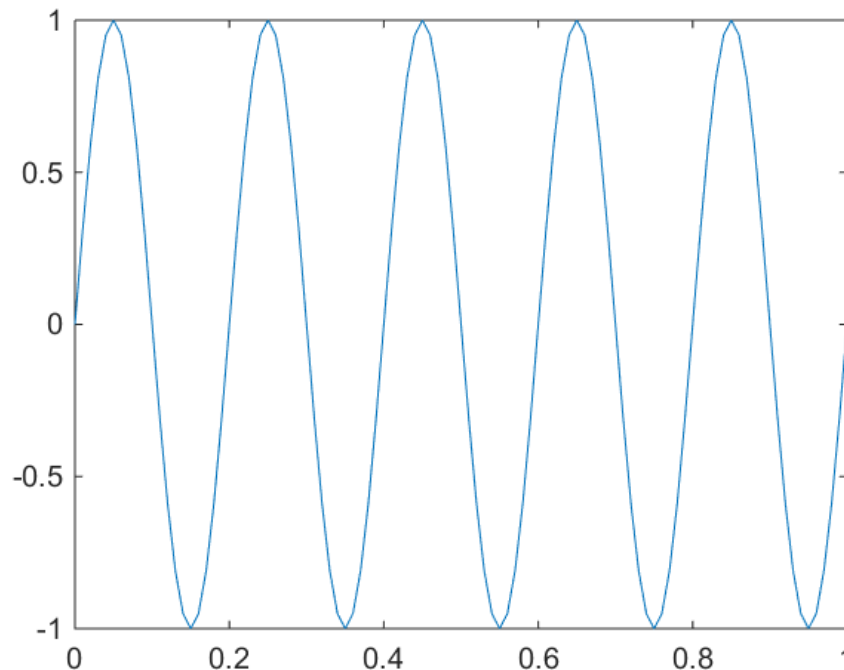
- **Gráficos:**

```
>> t = [0 : 0.01 : 1]'; % Vetor de instantes de tempo (segundos).
```

```
>> f = 5; % Frequência de uma senoide (em Hz).
```

```
>> x = sin(2*pi*f*t); % Vetor de amostras do sinal sinusoidal.
```

```
>> plot(t,x)
```



6. MATLAB / Octave

- **Gráficos:**

```
>> y = 0.5*cos(2*pi*f*t); % Vetor de amostras de outra sinusóide.
```

```
>> plot(t,x,'-.',t,y)
```

```
>> xlabel('Time (sec)');
```

```
>> ylabel('Amplitude');
```

```
>> title('Example of Graphics');
```

```
>> legend('x','y');
```

```
>> grid;
```

