

# Reconstrução Paralelizada de Dados de Ressonância Magnética de Fluxo em Processadores Multi-núcleo

Rosana Ribeiro Lima (rosanariblim@gmail.com)  
João Luiz Azevedo de Carvalho (joaoluiz@pgea.unb.br)

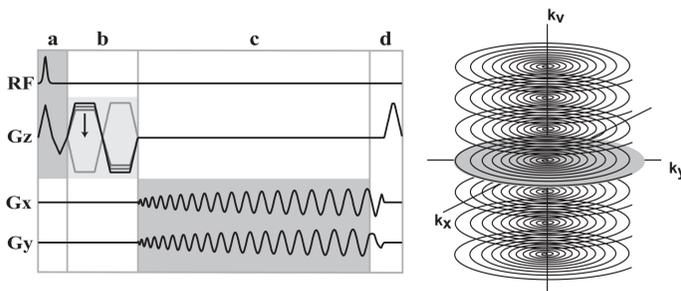
Grupo de Processamento de Sinais e Imagens Médicas  
Departamento de Engenharia Elétrica  
Universidade de Brasília, Brasília-DF, Brasil

## Objetivo

- ▶ Ressonância de fluxo pode ser usada para apoiar o diagnóstico de doenças valvulares [1] e tensão de cisalhamento na parede da carótida [2].
- ▶ Contraste de fase: rápido, mas tem problemas com a questão do volume parcial
- ▶ Codificação de velocidades por Fourier (FVE): robusto para volume parcial, mas longa aquisição
- ▶ Tempo de aquisição no FVE pode ser reduzido por trajetórias espirais em  $k_x$ - $k_y$  (codificação espacial) [1]
- ▶ FVE espiral: longo tempo de reconstrução, devido à alta dimensionalidade e à amostragem não-Cartesiana
- ▶ Tempo de reconstrução pode ser reduzido por meio de computação paralelizada!

## FVE espiral

- ▶ Dados multidimensionais:  $m(x, y, z, v, t)$



Sequência de pulso FVE espiral [1]

Trajectoria no espaço-k do FVE espiral: uma pilha de espirais em  $k_x$ - $k_y$ - $k_v$  [1]

- ▶ Reconstrução:
  - ▶ NUFFT [3] ao longo de  $k_x$ - $k_y$
  - ▶ FFT ao longo de  $k_v$

## Reconstrução paralelizada em Matlab

- ▶ Uso do laço “parfor” no lugar do laço “for”
- ▶ Tipos de variáveis permitidas: “temporary”, “broadcast”, “loop”, “sliced” e “reduction” [4]
- ▶ Implementação específica
- ▶ Tempo fixo de inicialização: funciona melhor para processamento em larga escala

**errado**

```
X = zeros(3,12);
parfor m = 1:3
    for n = 1:12
        X(m,n) = m + n;
    end
    disp(X(m,1))
end
```

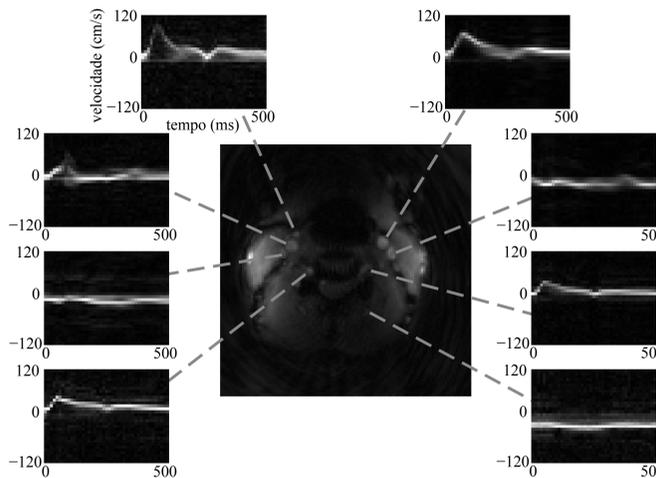
**certo**

```
X = zeros(3,12);
parfor m = 1:3
    p = zeros(1,12);
    for n = 1:12
        p(n) = m + n;
    end
    disp(p(1))
    X(m,:) = p;
end
```

Exemplo de usos corretos e incorretos do parfor, executando a mesma função

## Dados reconstruídos

- ▶ Exame de FVE espiral CINE de múltiplos cortes
- ▶ Resolução espacial:  $1,4 \times 1,4 \times 5 \text{ mm}^3$
- ▶ 8 leituras em espiral com densidade variável (4 ms cada)
- ▶ Resolução de velocidade: 5 cm/s (32 codificações de velocidade)
- ▶ Resolução temporal: 12 ms (43 fases cardíacas)
- ▶ 5 cortes axiais
- ▶ Tempo de aquisição: 2,4 min/corte (256 batimentos cardíacos a 105 bpm)

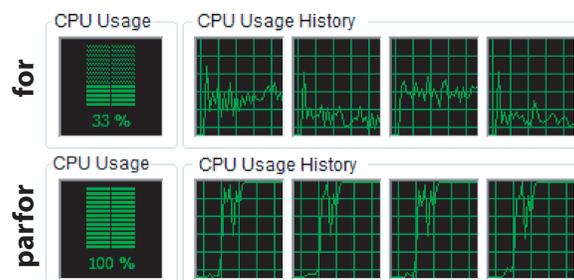


Distribuições tempo-velocidade de um corte axial do pescoço de um voluntário saudável

- ▶ Tempo de reconstrução medido para:
  - ▶ Matlab R2008a e R2011a
  - ▶ Processadores quad-core (4 núcleos) e dual-core (2 núcleos)
  - ▶ Dados 2D ( $x, y$ ), 4D ( $x, y, v, t$ ) e 5D ( $x, y, z, v, t$ )

## Laços Parfor

- ▶ Restrições do Parfor → alterações no algoritmo
  - ▶ Externalização do laço parfor
  - ▶ Mudança da ordem dos laços de fase e corte (corte passa a ser variável tipo “loop”)
  - ▶ Redefinição de  $m(x, y, k_v, c)$  dentro do laço parfor (passa a ser variável tipo “temporary”)
- ▶ Alocação do código em laços mais externos sempre que possível



Uso dos núcleos de um processador quad-core enquanto executa uma implementação sequencial do algoritmo de reconstrução (com “for”), e uma implementação paralelizada (com “parfor”)

## Resultados e discussão

Tempo de reconstrução (em segundos) para o algoritmo sequencial (laço “for”) e a abordagem paralelizada (laço “parfor”)

	MATLAB R2008a					
	processador dual-core		redução	processador quad-core		redução
	for	parfor		for	parfor	
$m(x, y)$	0.8	1.1	não	0.3	0.9	não
$m(x, y, v, t)$	171.2	107.8	37%	87.5	49.8	43%
$m(x, y, z, v, t)$				435.7	275.5	37%

	MATLAB R2011a					
	processador dual-core		redução	processador quad-core		redução
	for	parfor		for	parfor	
$m(x, y)$	0.7	1.0	não	0.3	0.9	não
$m(x, y, v, t)$	107.1	90.3	16%	65.2	40.1	38%
$m(x, y, z, v, t)$				334.9	218.3	35%

- ▶ O uso do laço “parfor” não foi capaz de reduzir o tempo de reconstrução nos conjuntos de dados pequenos, devido ao seu tempo de inicialização fixo
- ▶ O Matlab 2011a apresentou tempos de reconstrução consideravelmente menores
- ▶ A rapidez conquistada por usar a reconstrução paralelizada no Matlab 2011 foi menos expressiva.
  - ▶ Versões mais novas têm, para muitas de suas funções de fábrica, capacidade de execução multi-núcleo
  - ▶ Elevado uso da CPU mesmo com os laços “for” tradicionais

## Conclusão

- ▶ A reconstrução paralelizada é uma abordagem simples e prática para acelerar a reconstrução de imagens de ressonância magnética
- ▶ Pode ser especialmente útil quando se lida com dados multidimensionais, amostragem não-Cartesiana e/ou reconstrução iterativa (por exemplo, compressed sensing)

## Referências

- [1] Carvalho JLA and Nayak KS. MRM 57:639, 2007
- [2] Carvalho JLA et al. MRM 63:1537, 2010
- [3] Fessler JA and Sutton BP. IEEE TSP 51:560, 2003
- [4] <http://www.mathworks.com/help/>

## Apoio financeiro

- ▶ PIBIC/UnB/CNPq
- ▶ DEG/UnB
- ▶ PROAP/CAPES
- ▶ PGEA/ENE/FT/UnB
- ▶ Edital MCT/CNPq 014/2010 – Universal