

Frustum Culling Híbrido Utilizando CPU e GPU

Aluno: Eduardo Telles Carlos

Orientador: Alberto Raposo

Co-Orientador: Marcelo Gattass

Agenda

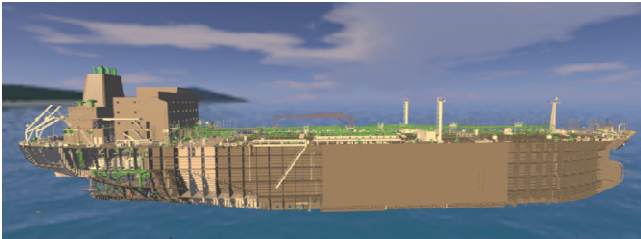
- ▶ **Objetivos**
- ▶ **Motivação**
- ▶ **Algoritmos de visibilidade**
- ▶ **Frustum Culling em CPU**
- ▶ **Frustum Culling em GPU**
- ▶ **Frustum Culling híbrido**
- ▶ **Conclusão**
- ▶ **Trabalhos Futuros**

Objetivos

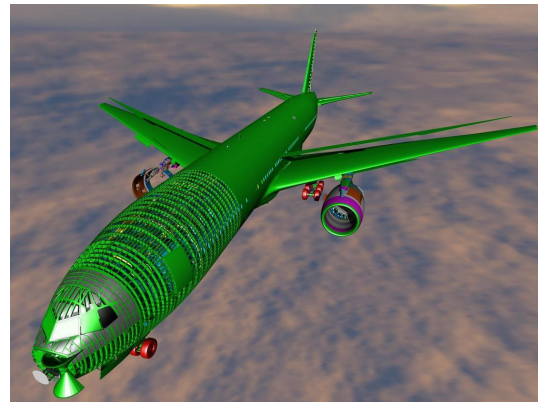
- ▶ Estado da arte do frustum culling
- ▶ Propor alternativas para utilizar GPU
- ▶ Apresentar algoritmo híbrido

Motivação

- ▶ Modelos massivos
- ▶ Lei de Moore
- ▶ GPGPU (General-purpose computing on graphics processing units)
- ▶ Poucos estudos na área



Double eagle tanker
80 milhões triângulos



Boeing 777
350 milhões de triângulos



Sunflowers
1 bilhão de triângulos

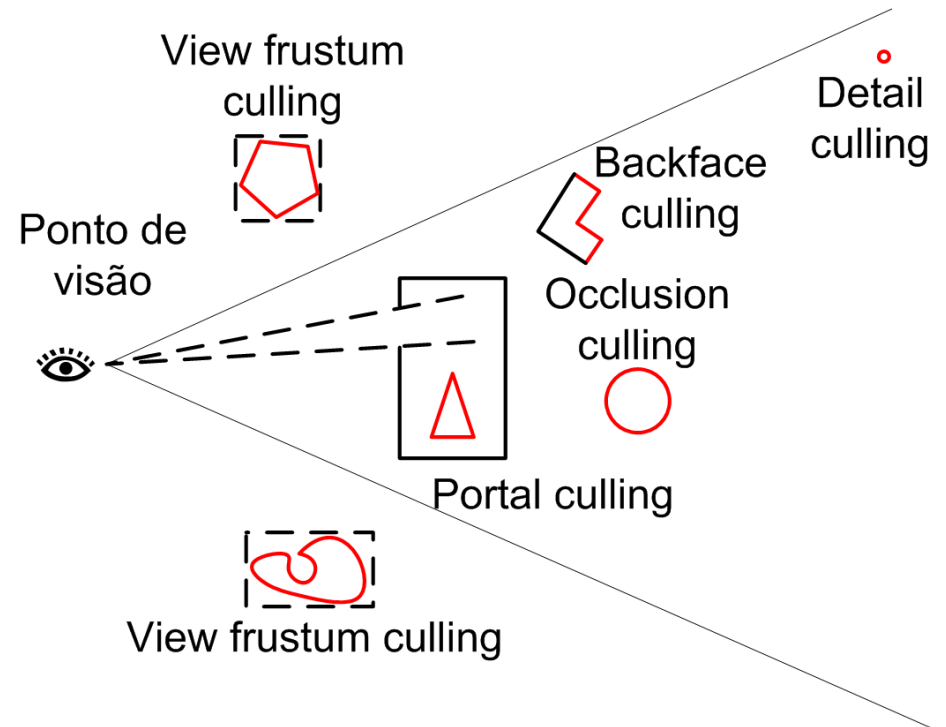
Agenda

- ▶ Objetivos
- ▶ Motivação
- ▶ **Algoritmos de visibilidade**
 - ▶ Algoritmos de culling
 - ▶ Pipeline gráfico
- ▶ Frustum Culling na CPU
- ▶ Frustum Culling na GPU
- ▶ Frustum Culling híbrido
- ▶ Conclusão
- ▶ Trabalhos Futuros

Algoritmos de visibilidade (1)

▶ Culling

- ▶ Back-face culling
- ▶ View-frustum culling
- ▶ Occlusion culling
- ▶ Portal culling
- ▶ Detail culling



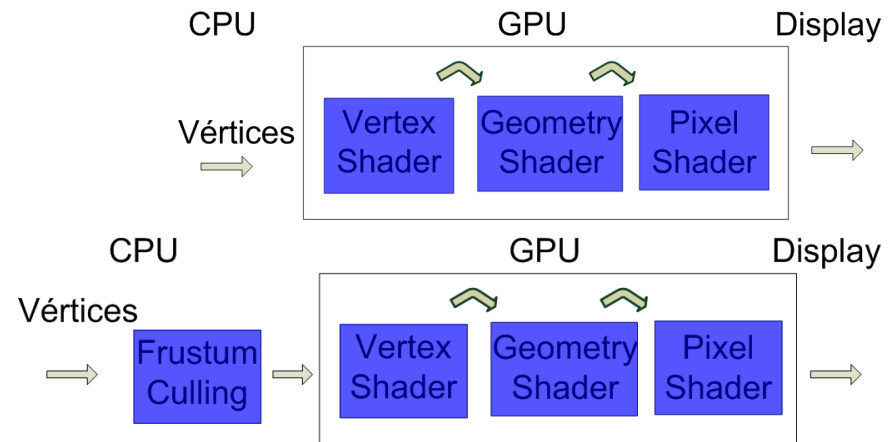
Algoritmos de visibilidade (2)

- ▶ Classificação dos algoritmos [coh03]
 - ▶ Exato X Conservativo X Aproximado
 - ▶ Pré-processados X Online
 - ▶ 2-D X 3-D
 - ▶ Software X Hardware
 - ▶ Cenas Dinâmicas X Cenas Estáticas

Pipeline do Frustum Culling

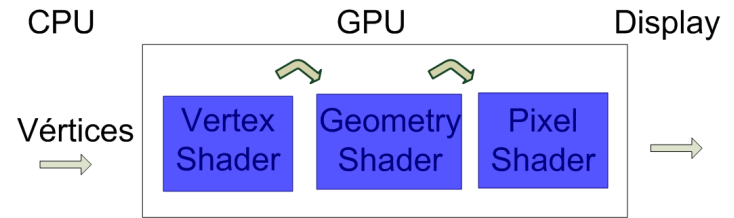
- ▶ Pipeline da GPU

- ▶ Em CPU

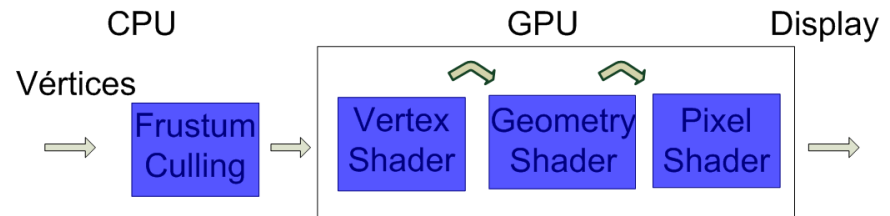


Pipeline do Frustum Culling

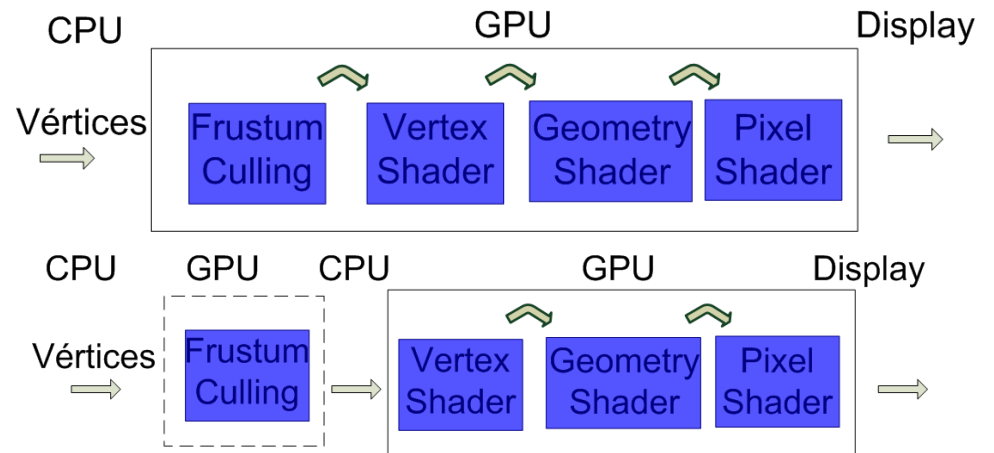
- ▶ Pipeline da GPU



- ▶ Em CPU

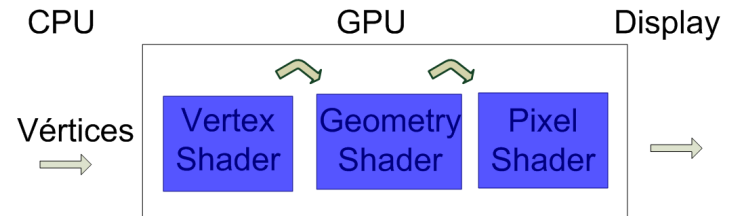


- ▶ Em GPU

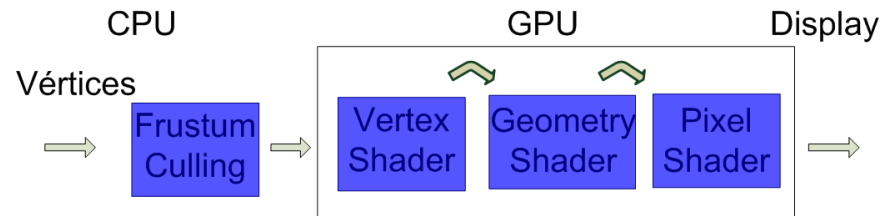


Pipeline do Frustum Culling

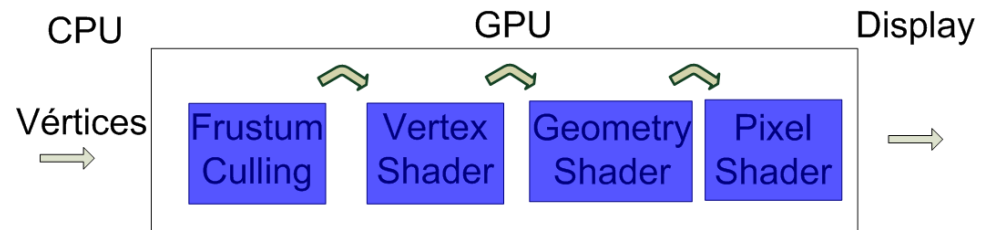
- ▶ Pipeline da GPU



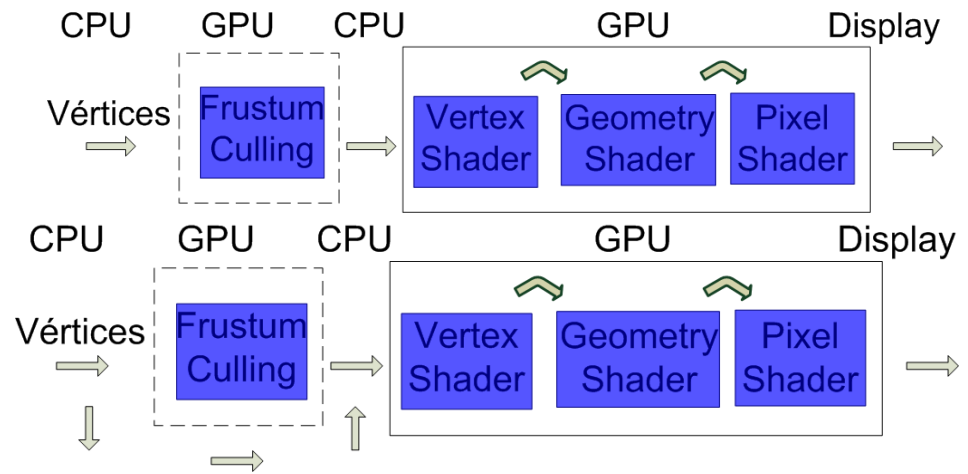
- ▶ Em CPU



- ▶ Em GPU



- ▶ Híbrido

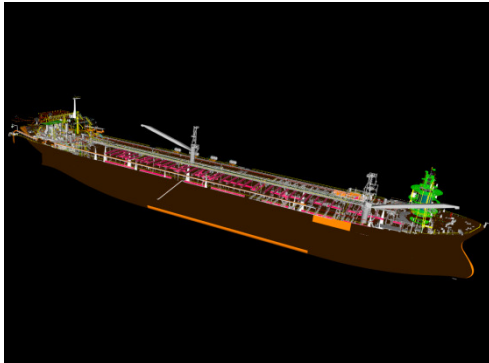


Agenda

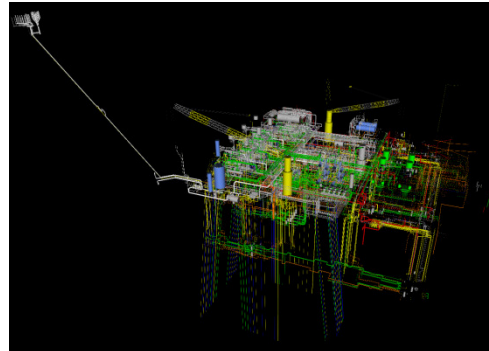
- ▶ Objetivos
- ▶ Motivação
- ▶ Algoritmos de visibilidade
- ▶ **Frustum Culling na CPU**
 - ▶ Modelos / Ambiente de Benchmark
 - ▶ Técnicas / Resultados
 - ▶ Hierarquia / Resultados
 - ▶ Otimizações / Resultados
- ▶ Frustum Culling na GPU
- ▶ Frustum Culling híbrido
- ▶ Conclusão
- ▶ Trabalhos Futuros

Modelos de Testes

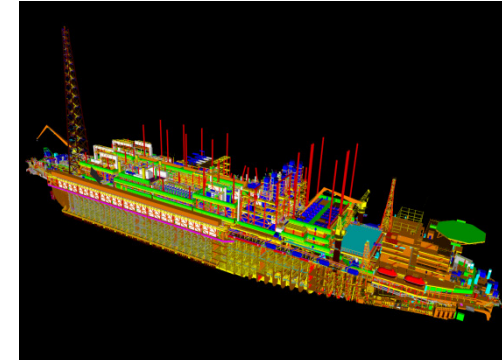
P-38



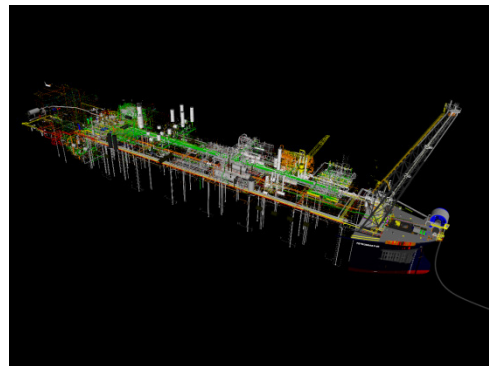
P-40



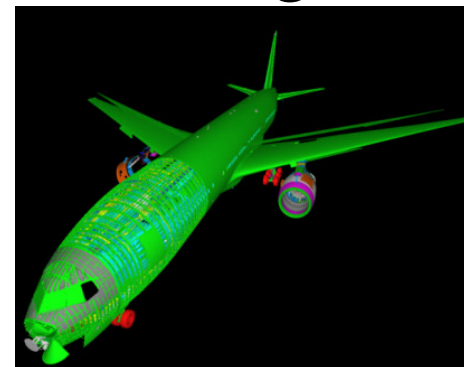
P-43



P-50



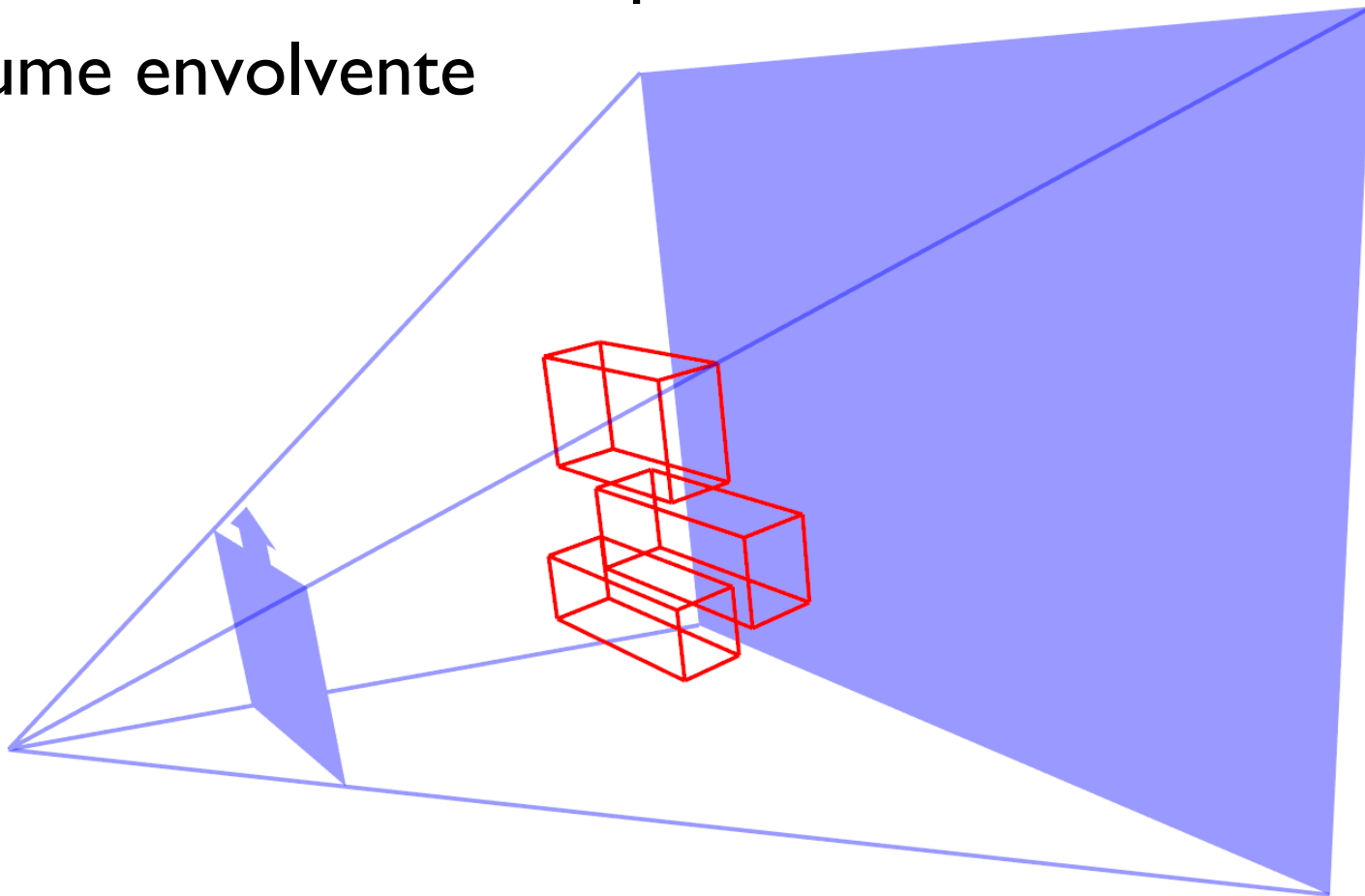
Boeing 777



- Quad-Core X9650 @ 3.00 GHz, 8 GB DDR2 800, 280 GTX 1GB, Windows XP 64 bits
-

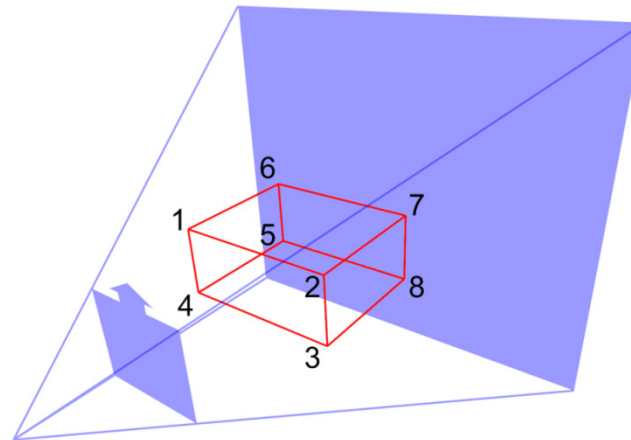
Frustum Culling em CPU

- ▶ Algoritmo clássico com ponto
- ▶ Volume envolvente

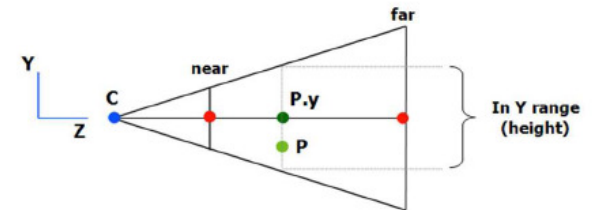
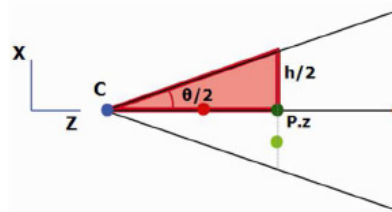


Planos X Radar

▶ Planos



▶ Radar



From Sunar, M.S.; Zin, A.M.; Sembok, T.M.T., "Range Detection Approach in Interactive Virtual Heritage Walkthrough," *Artificial Reality and Telexistence--Workshops*, 2006. Nov. 2006

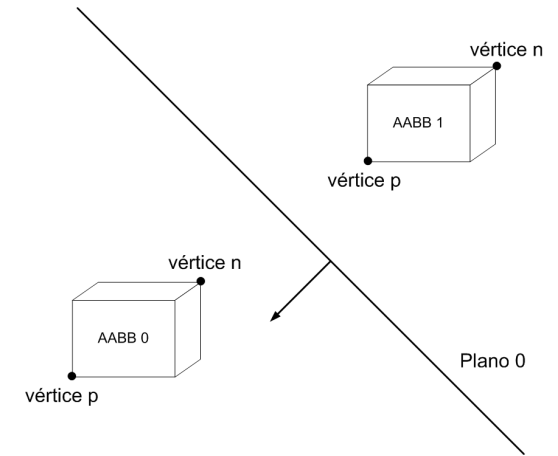
Método	Atualização	Melhor caso	Pior caso
Planos	316 operações	7 operações	336 operações
Radar	112 operações	8 operações	192 operações

Menor = Melhor

Planos + otimização 1 X Radar

► Planos + otimização I (Planos NP)

n_x	n_y	n_z	p -vertex	n -vertex
+	+	+	$[x_{max}, y_{max}, z_{max}]$	$[x_{min}, y_{min}, z_{min}]$
+	+	-	$[x_{max}, y_{max}, z_{min}]$	$[x_{min}, y_{min}, z_{max}]$
+	-	+	$[x_{max}, y_{min}, z_{max}]$	$[x_{min}, y_{max}, z_{min}]$
+	-	-	$[x_{max}, y_{min}, z_{min}]$	$[x_{min}, y_{max}, z_{max}]$
-	+	+	$[x_{min}, y_{max}, z_{max}]$	$[x_{max}, y_{min}, z_{min}]$
-	+	-	$[x_{min}, y_{max}, z_{min}]$	$[x_{max}, y_{min}, z_{max}]$
-	-	+	$[x_{min}, y_{min}, z_{max}]$	$[x_{max}, y_{max}, z_{min}]$
-	-	-	$[x_{min}, y_{min}, z_{min}]$	$[x_{max}, y_{max}, z_{max}]$



► Radar

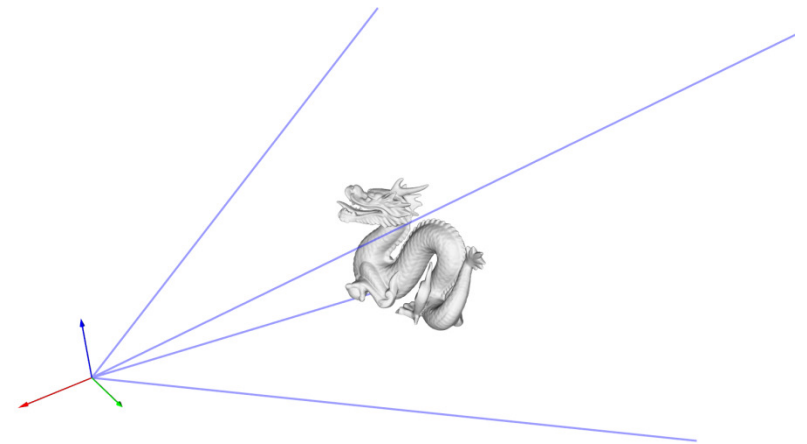
Método	Atualização	Melhor caso	Pior caso
Planos NP	316 operações	10 operações	120 operações
Radar	112 operações	8 operações	192 operações

Menor = Melhor

Planos NP X Planos NP_[Res08]

- ▶ Planos NP

- ▶ Planos + otimização 2_[Res08]



Método	Atualização	Melhor caso	Pior caso
Planos NP	316 operações	10 operações	120 operações
Planos NP _[Res08]	340 operações	11 operações	11 operações

Menor = Melhor

Resultados

▶ Caminho de câmera pelos modelos

Métodos	P-38	P-40	P-43	P-50	Boeing 777
Planos	40.678	9.16	4.95	4.106	7.836
Radar	40.678	9.16	9.626	9.399	16.096
Planos NP	237.279	62.008	32.516	27.197	34.171

Maior = Melhor

Hierarquia de volume envolventes (1)

▶ Construção da hierarquia

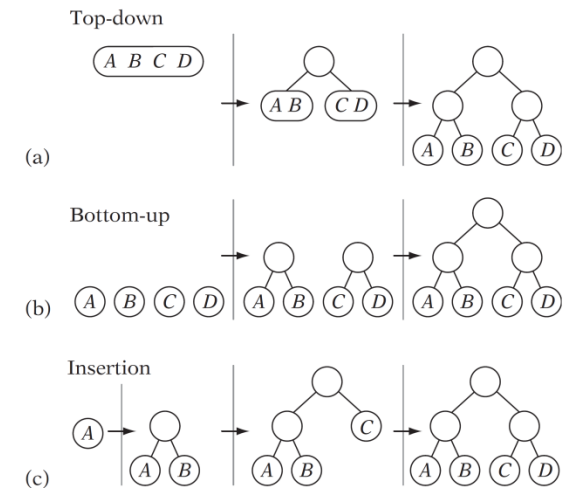
- ▶ Top-down
- ▶ Bottom-up
- ▶ Insertion

▶ Estratégia de particionamento

- ▶ Minimizar a soma dos volume dos filhos
- ▶ Dividir primitivas igualmente entre os filhos
- ▶ Maximizar a separação dos volumes dos filhos

▶ Plano de corte

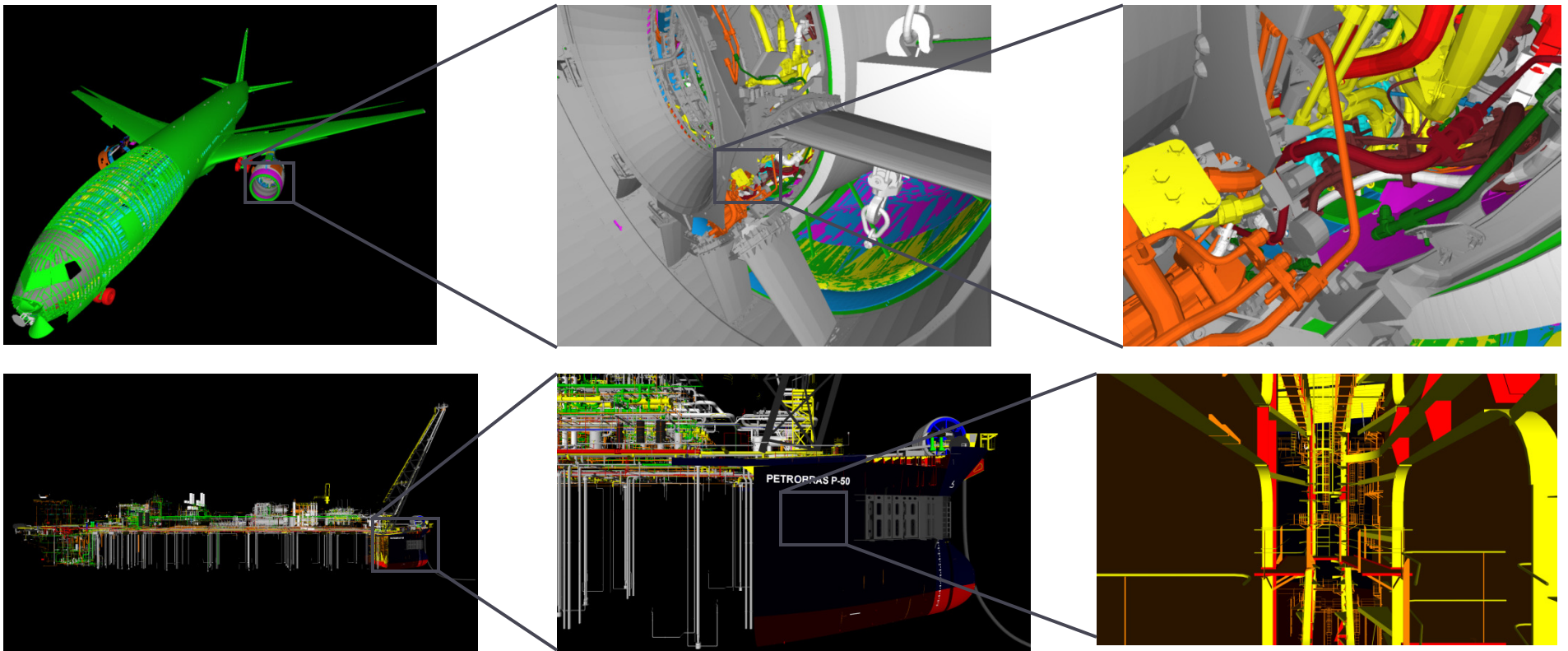
- ▶ Eixo
- ▶ Posição



From ERICSON C.: Real-Time Collision Detection. Morgan Kaufmann, 2004

Hierarquia de volume envolventes (2)

- ▶ Critério de parada
 - ▶ Três objetos



Hierarquia de volume envolventes (3)

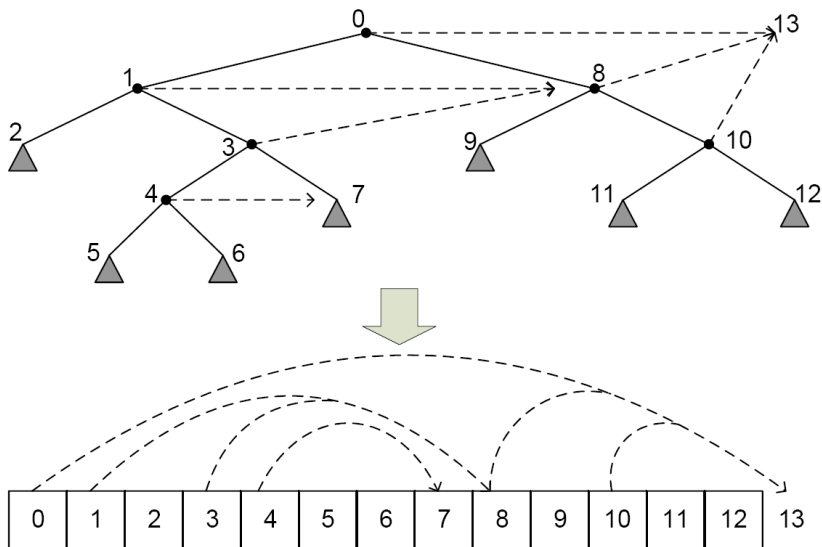
- ▶ Top-Down
 - ▶ Mediana
 - ▶ Média

Modelos	Mediana	Média
P-38	4296.68	4345.70
P-40	1625.35	1580.76
P-43	957.24	943.08
P-50	1570.51	1584.48
Boeing	1727.70	1766.86

Maior = Melhor

Otimizações (3)

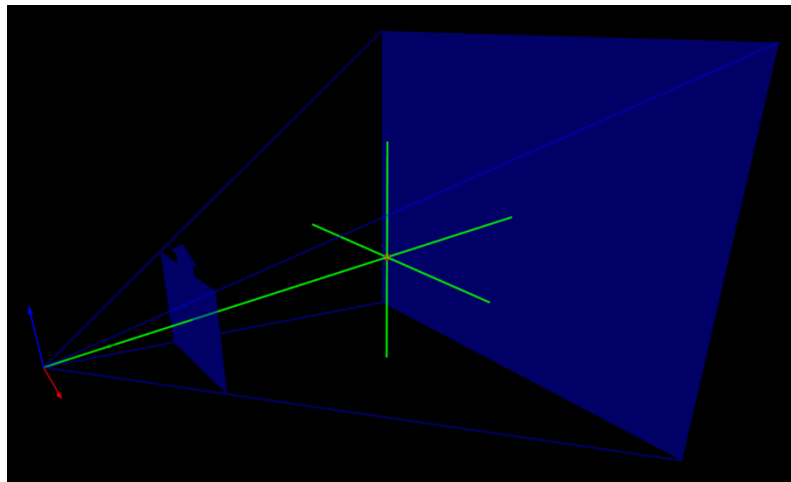
▶ Traversal sem pilha



Modelos	Sem Pilha
P-38	8.94 %
P-40	4.02 %
P-43	7.55 %
P-50	1.59 %
Boeing	11.69 %

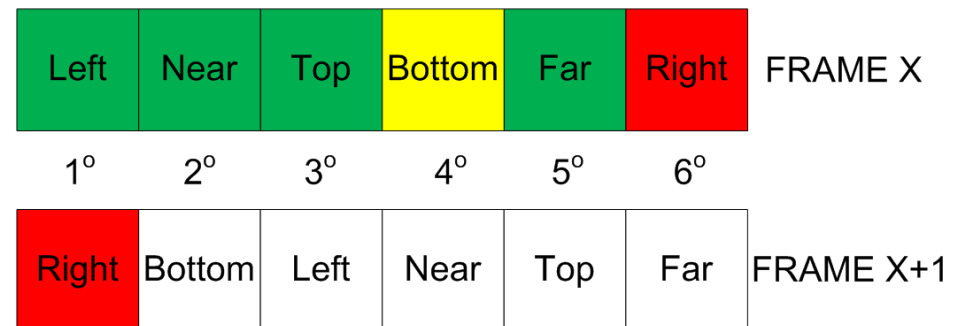
Otimizações (4)

▶ Octant test



▶ Plane-Coherency

■ Dentro
■ Interceptando
■ Fora



▶ Masking

▶ TR-Coherency

Otimizações (5)

► Octant test + Plane-coherency

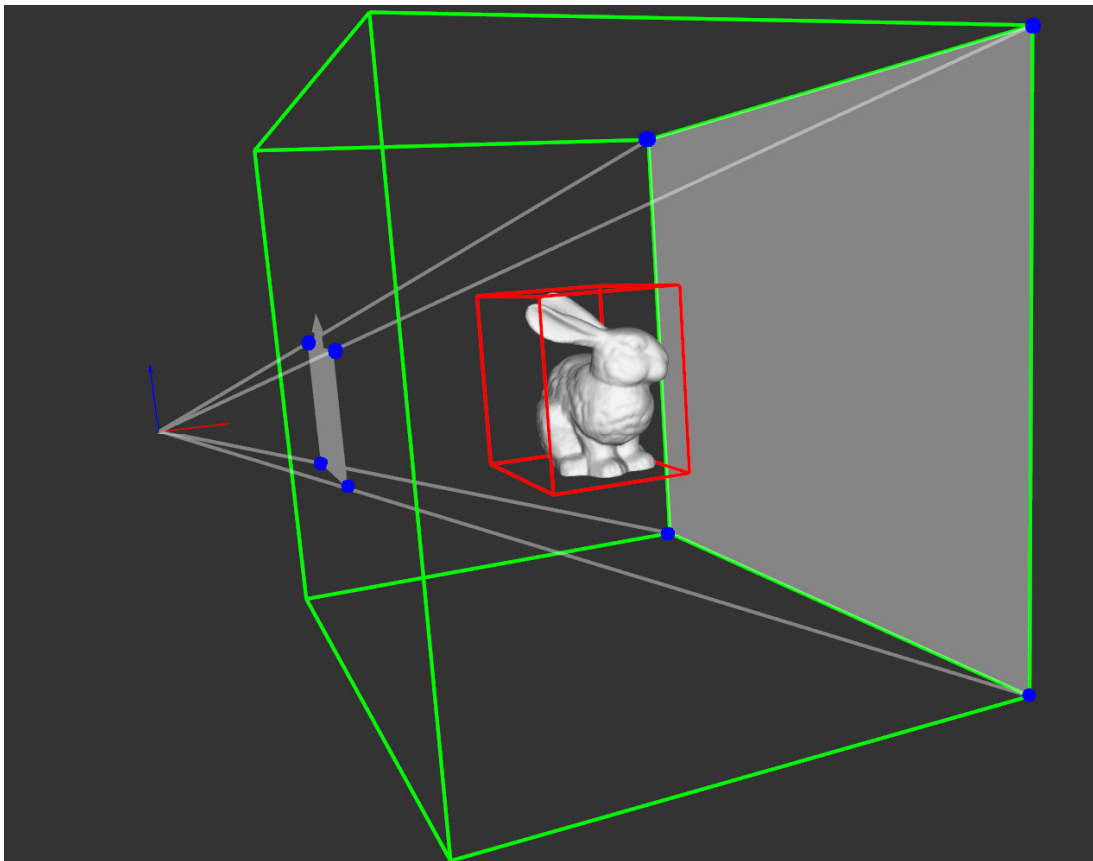
<i>path</i>	1	1	1	2	2	2	3	3	3	4	4	4
<i>model</i>	1	2	3	1	2	3	1	2	3	1	2	3
only Basic intersection test	2.8	1.9	3.9	2.2	2.0	3.1	3.9	2.5	4.3	3.1	2.2	3.7
Plane-coherency + octant test	4.0	2.4	5.1	2.8	2.6	3.9	4.8	3.5	5.6	3.3	3.0	5.1
Plane-coherency + TR coherency	3.8	2.0	4.0	2.5	2.2	3.0	5.0	2.8	4.4	8.3	3.1	11.0
Plane-coherency + octant test + TR coherency	3.7	2.2	4.5	2.6	2.4	3.6	5.1	3.0	4.8	8.0	3.3	9.0

From U. Assarson and T. Möller. Optimized View Frustum Culling Algorithms for Bounding Boxes. *Journal of Graphics Tools*, 2000

Modelos	Plane-coherency + Octant test
P-38	8.93 %
P-40	4.02 %
P-43	215.19 %
P-50	78.06 %
Boeing	11.69 %

Otimizações (6)

► Teste entre AABBs



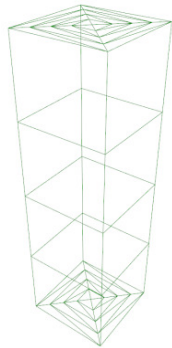
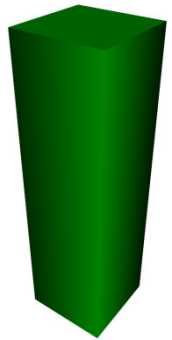
Modelos	Entre AABBs
P-38	1.58 %
P-40	-1.41 %
P-43	-5.36 %
P-50	1.28 %
Boeing	-3.60 %

Agenda

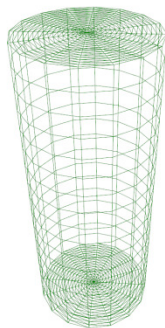
- ▶ Objetivos
- ▶ Motivação
- ▶ Algoritmos de visibilidade
- ▶ Frustum Culling na CPU
- ▶ **Frustum Culling na GPU**
 - ▶ **GPU primitives / Modelos genéricos**
 - ▶ **Frustum culling nas GPU primitives / Resultados**
 - ▶ **Frustum culling nos modelos genéricos / Resultados**
 - ▶ **Hierarquia**
- ▶ Frustum Culling híbrido
- ▶ Conclusão
- ▶ Trabalhos Futuros

GPU Primitives (1)

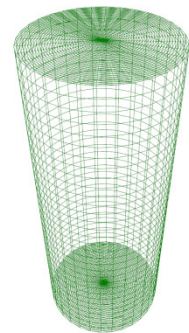
▶ Modelos genéricos X GPU primitives



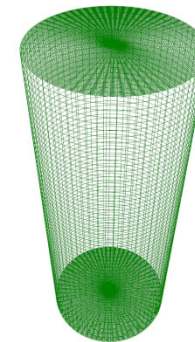
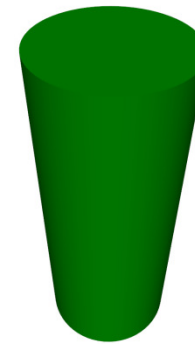
60 vértices



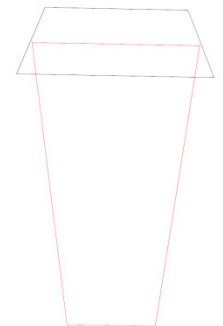
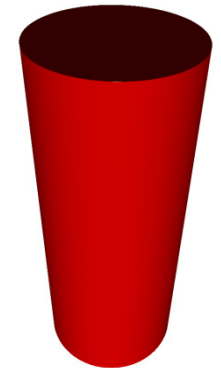
408 vértices



1320 vértices



4680 vértices

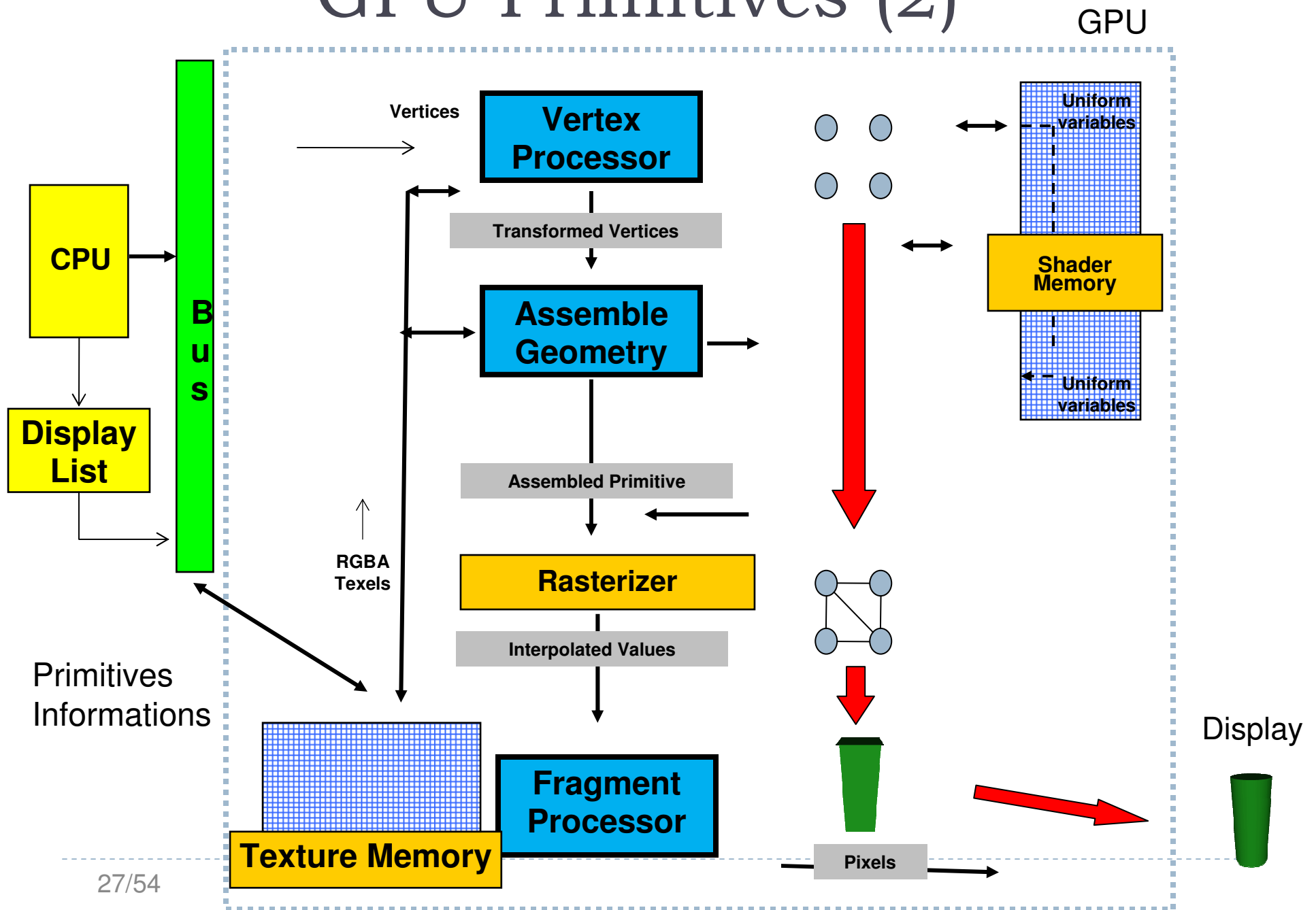


8 vértices



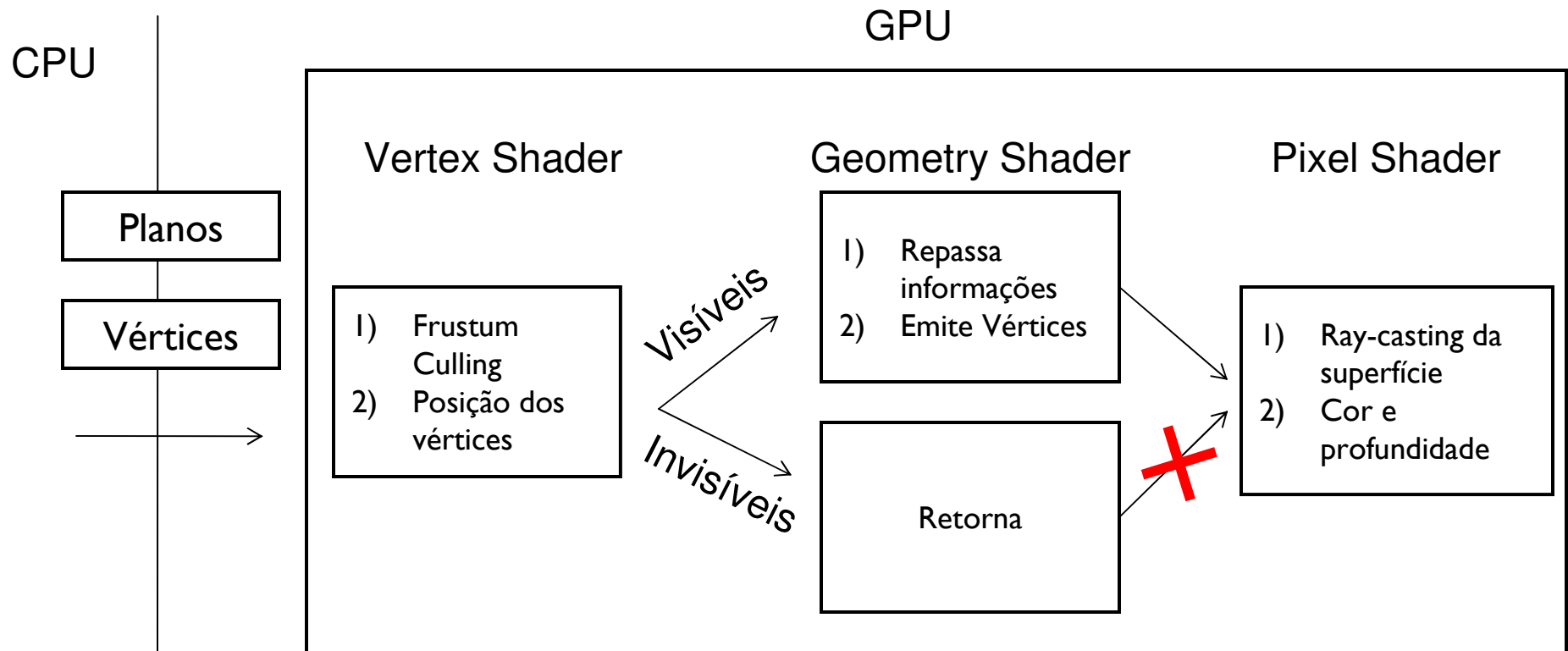
Melhor Qualidade

GPU Primitives (2)

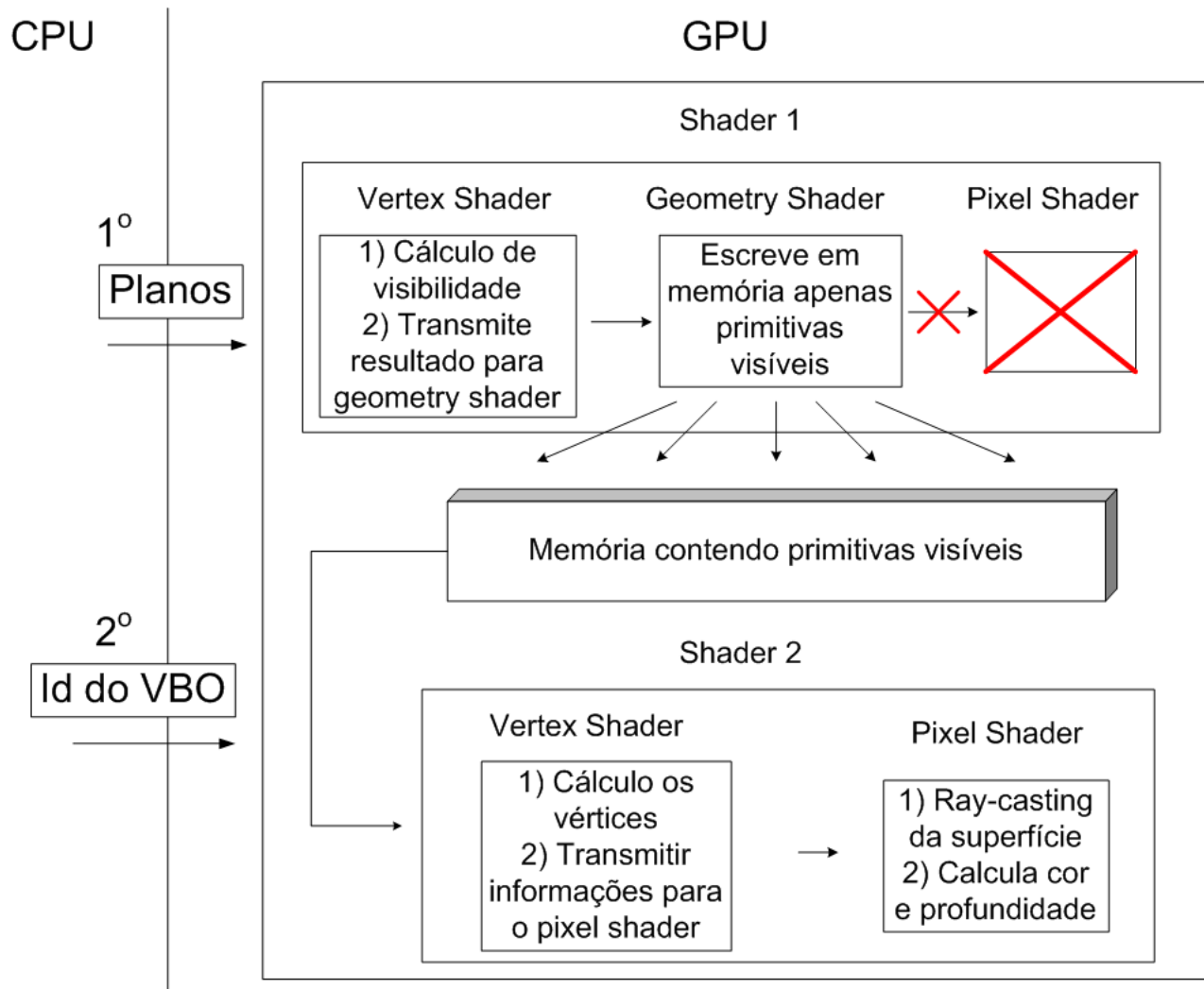


GPU FC nas GPU primitives (1)

- ▶ Planos NP
- ▶ Algoritmo de frustum culling nos shaders das primitivas

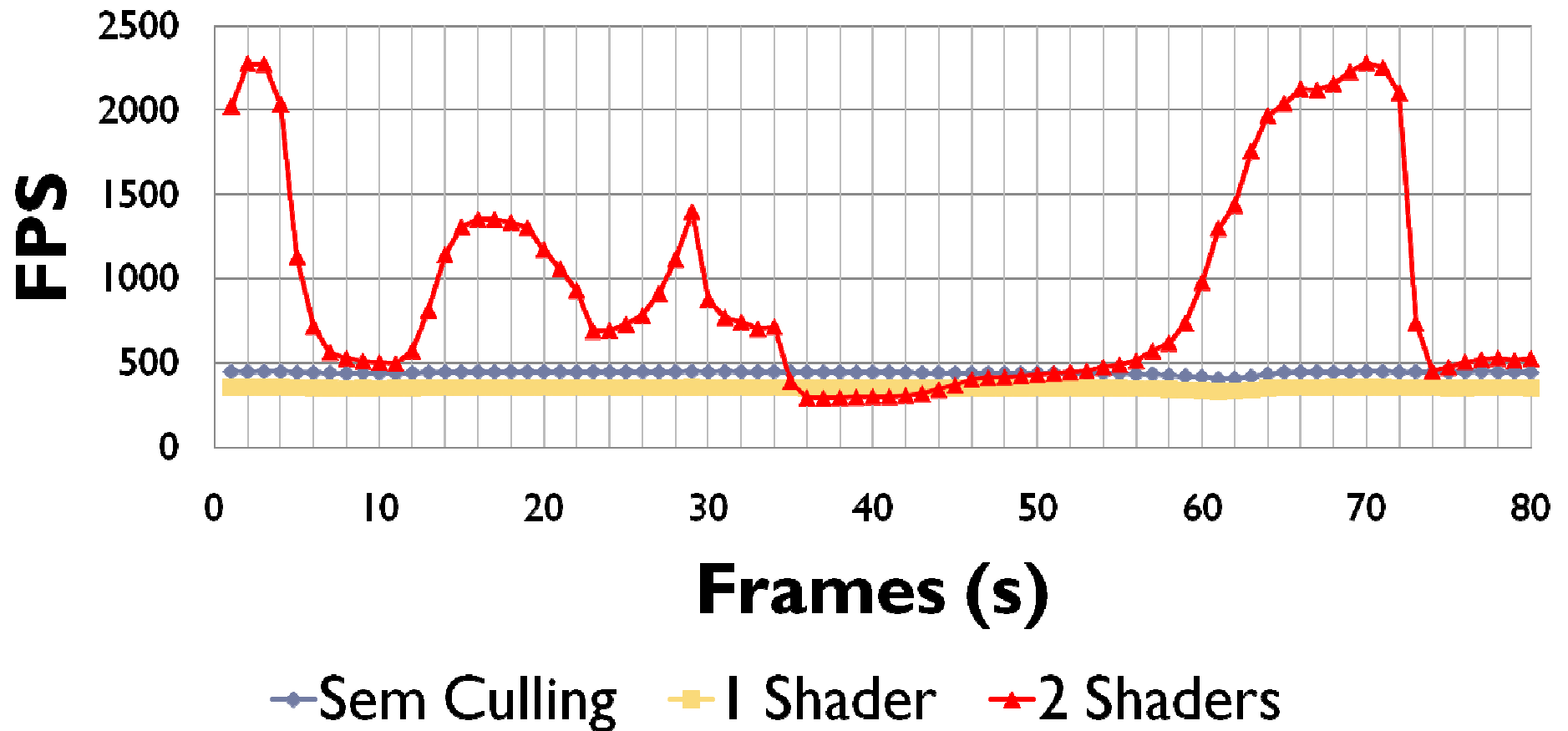


GPU FC nas GPU primitives (2)



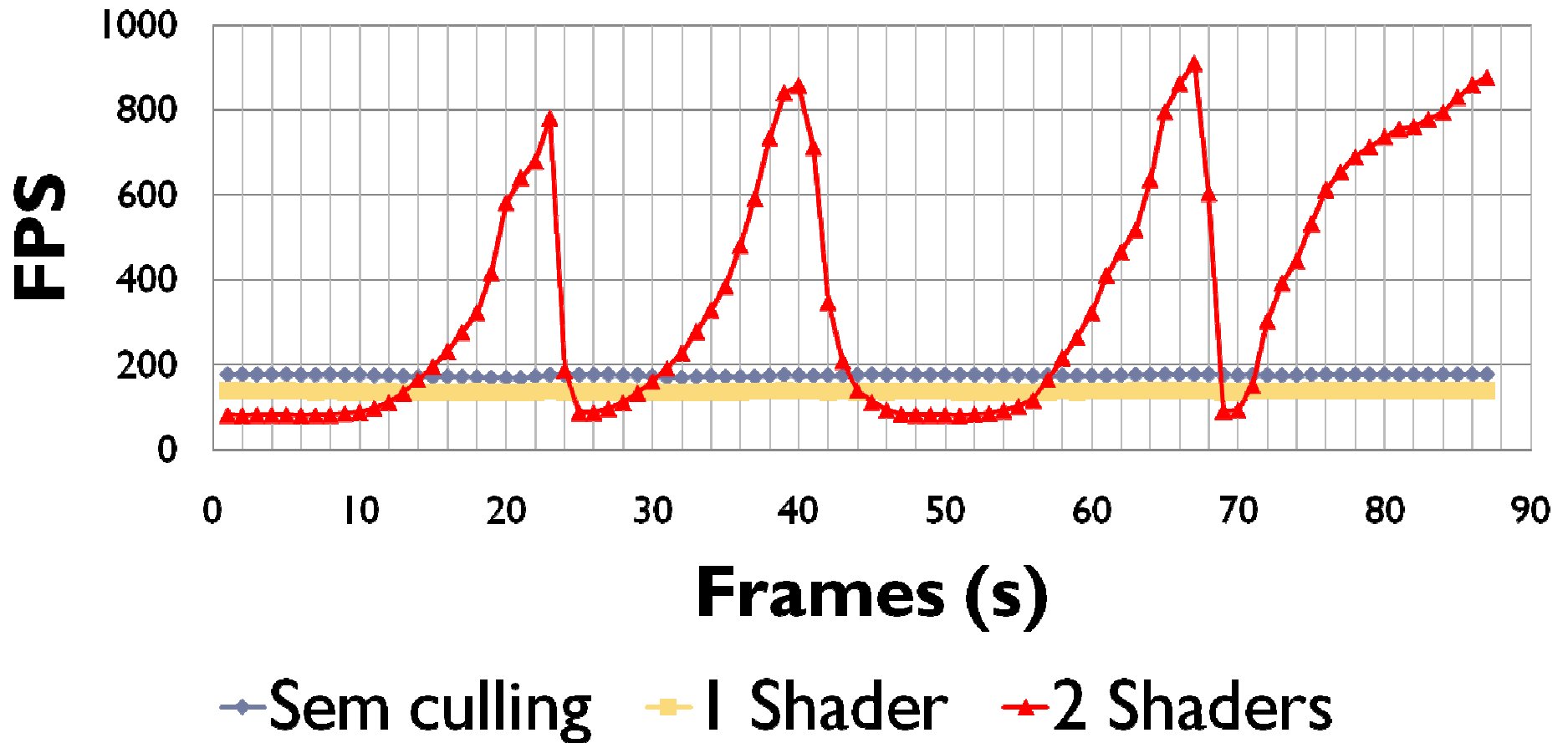
GPU FC nas GPU primitives (3)

P-38



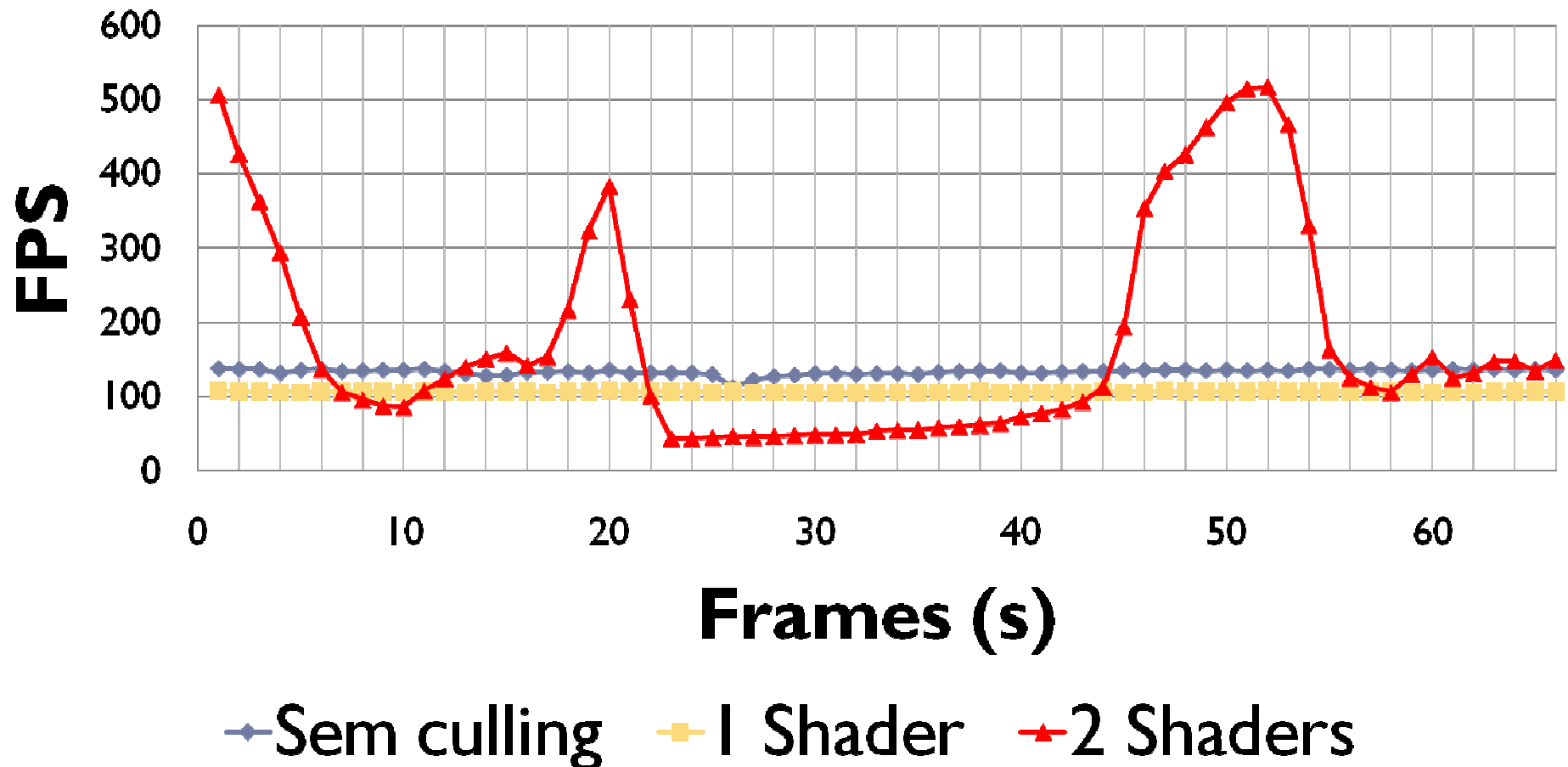
GPU FC nas GPU primitives (3)

P-40



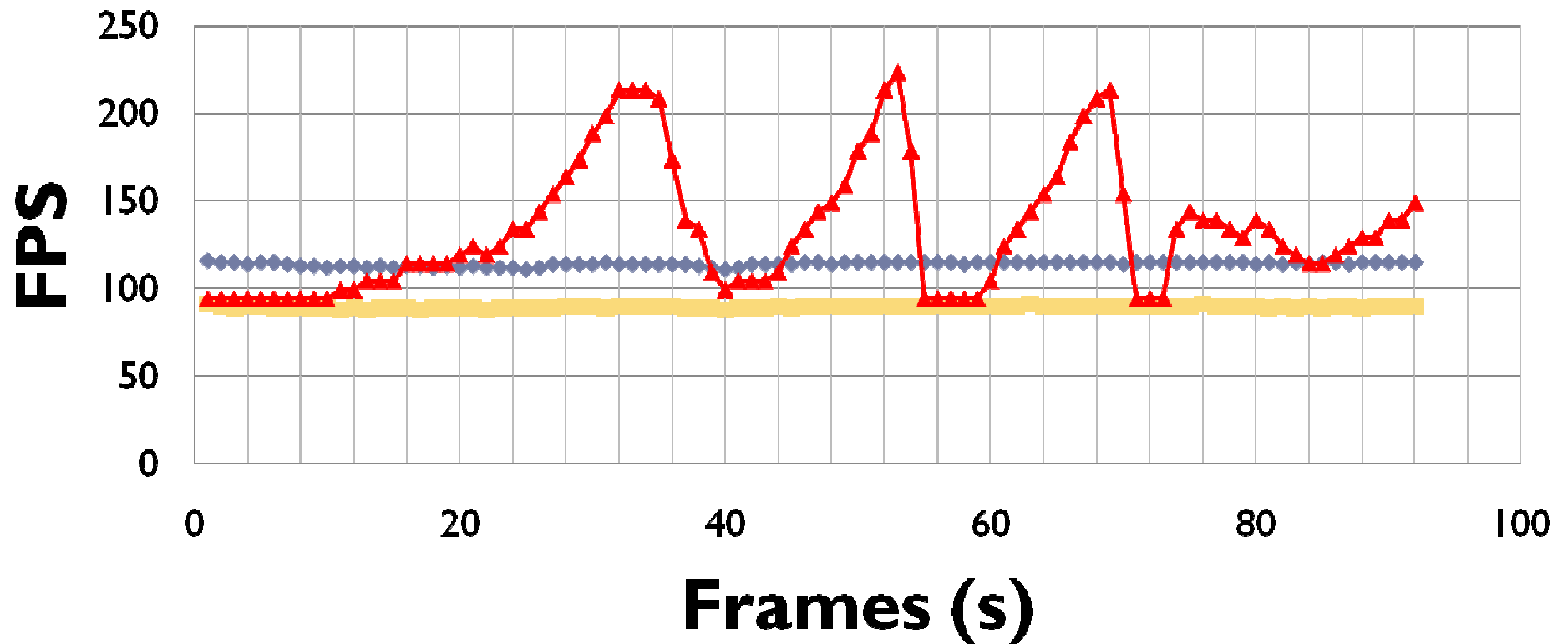
GPU FC nas GPU primitives (3)

P-43



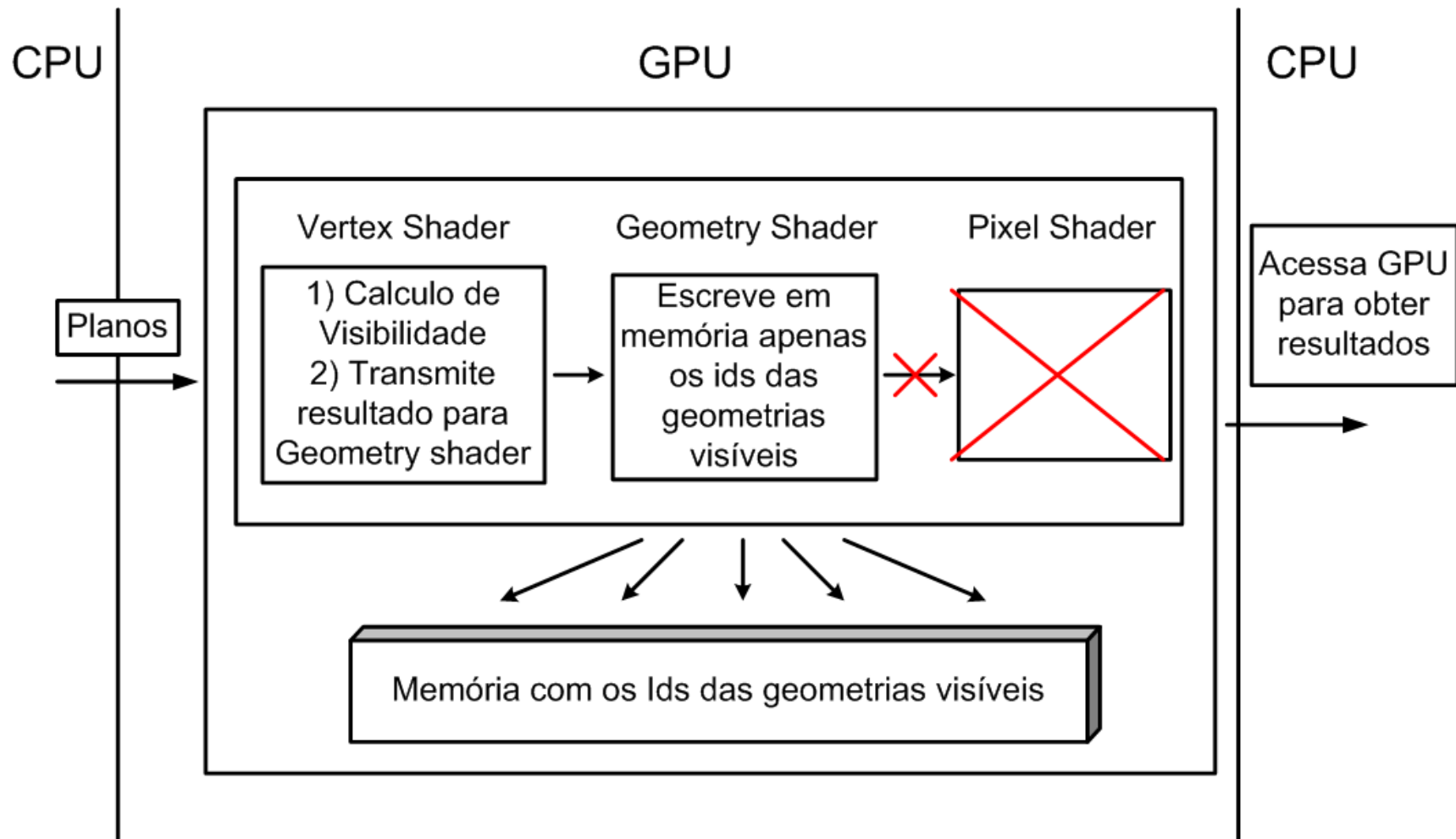
GPU FC nas GPU primitives (3)

P-50

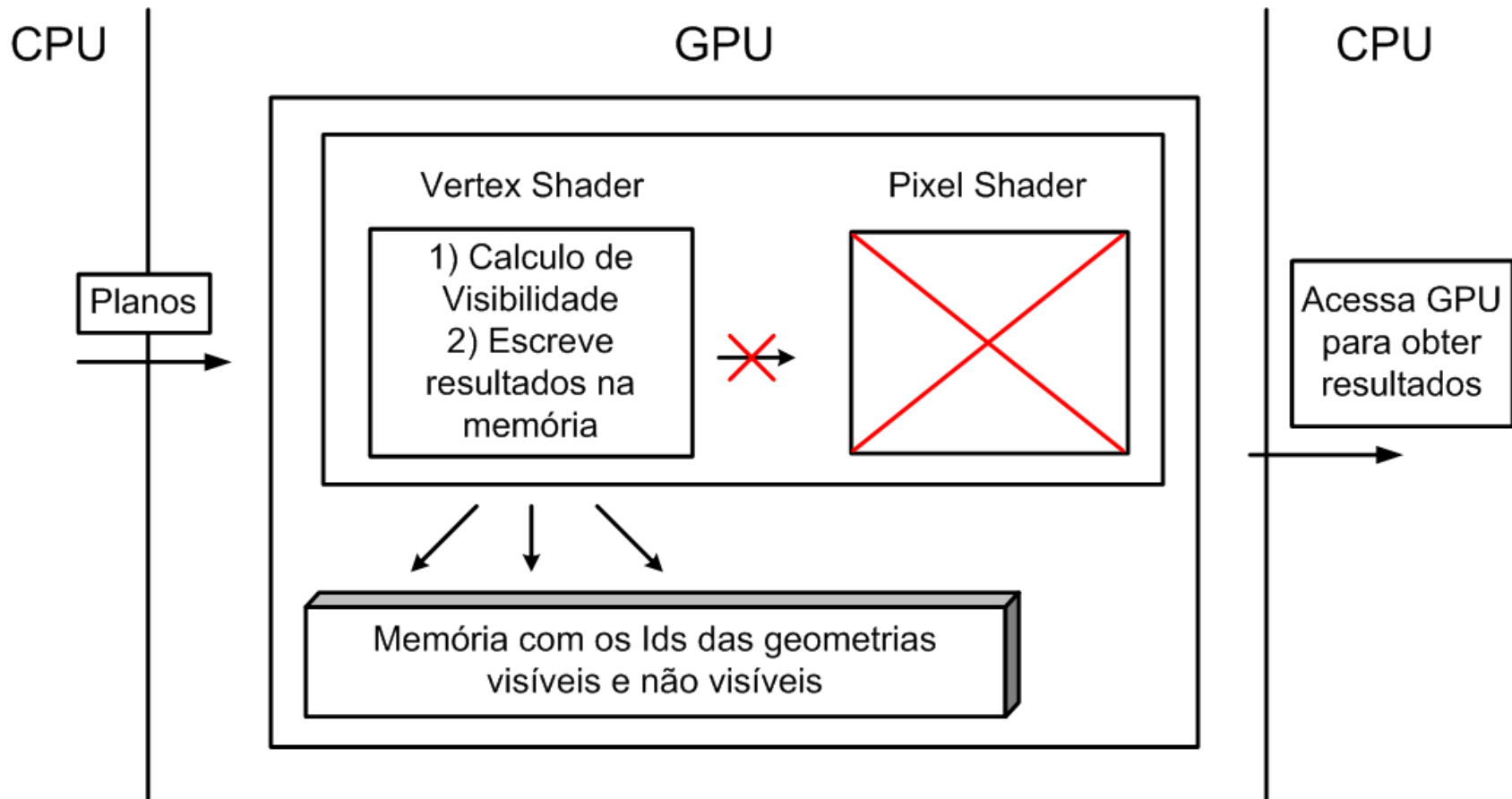


—◆— Sem Culling —■— 1 Shader —▲— 2 Shaders

GPU FC em modelos genéricos (1)

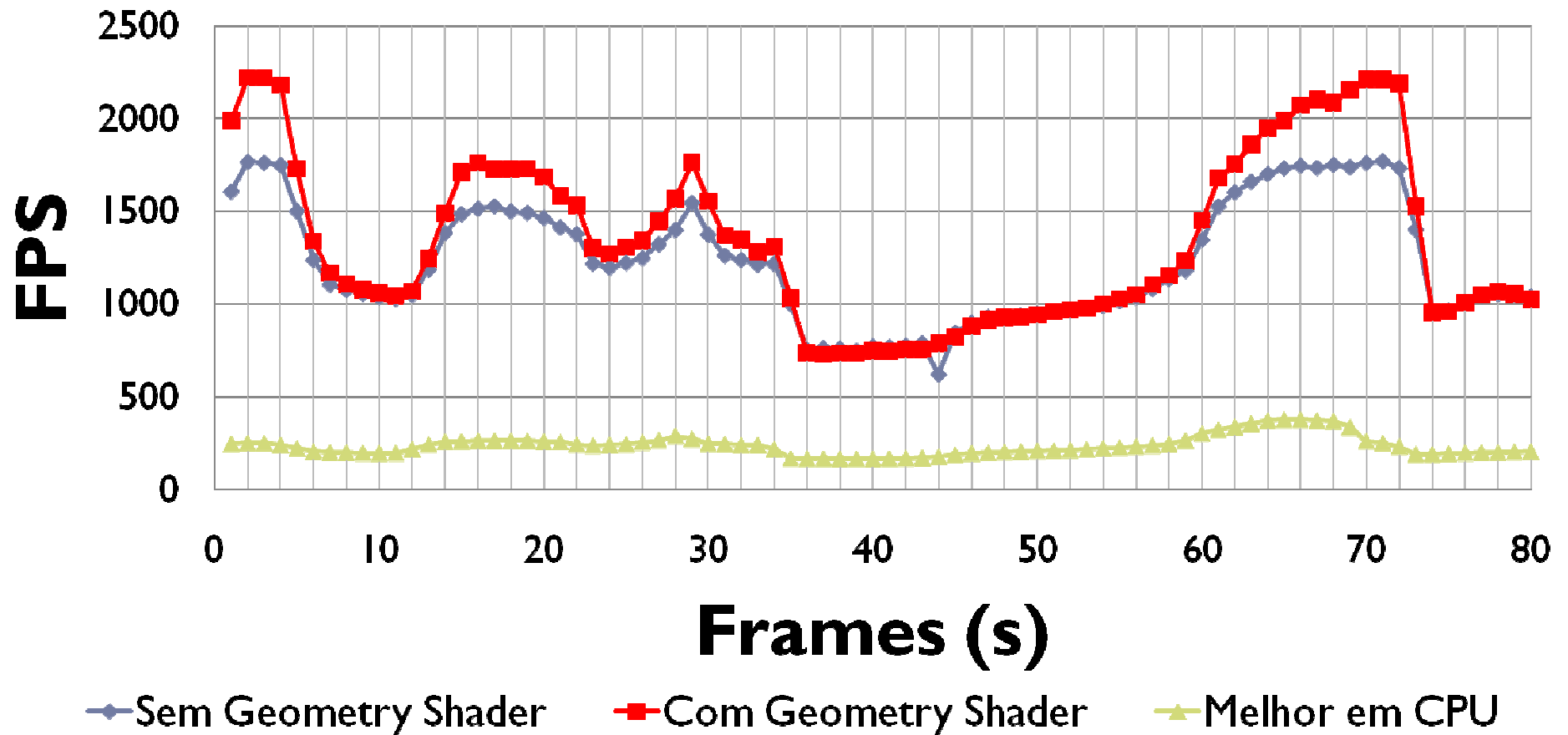


GPU FC em modelos genéricos (2)



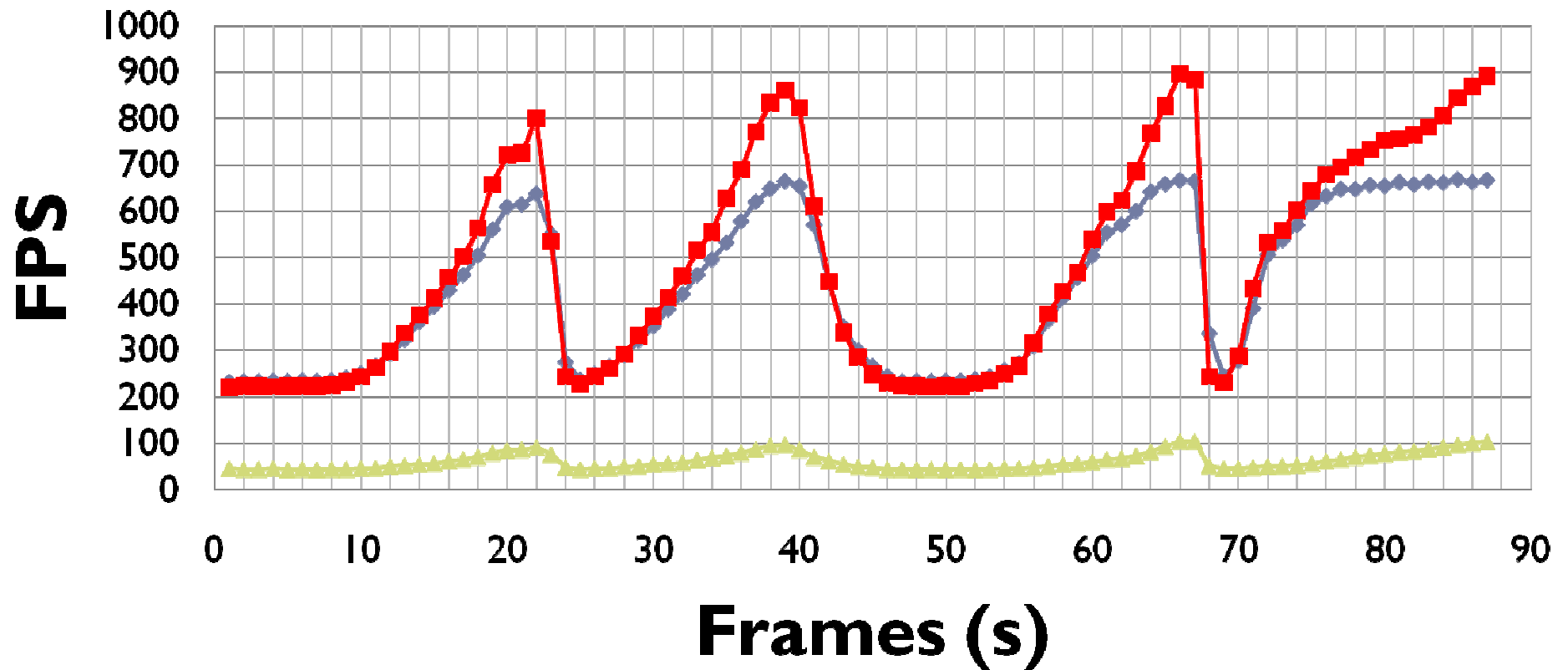
GPU FC em modelos genéricos (3)

P-38



GPU FC em modelos genéricos (3)

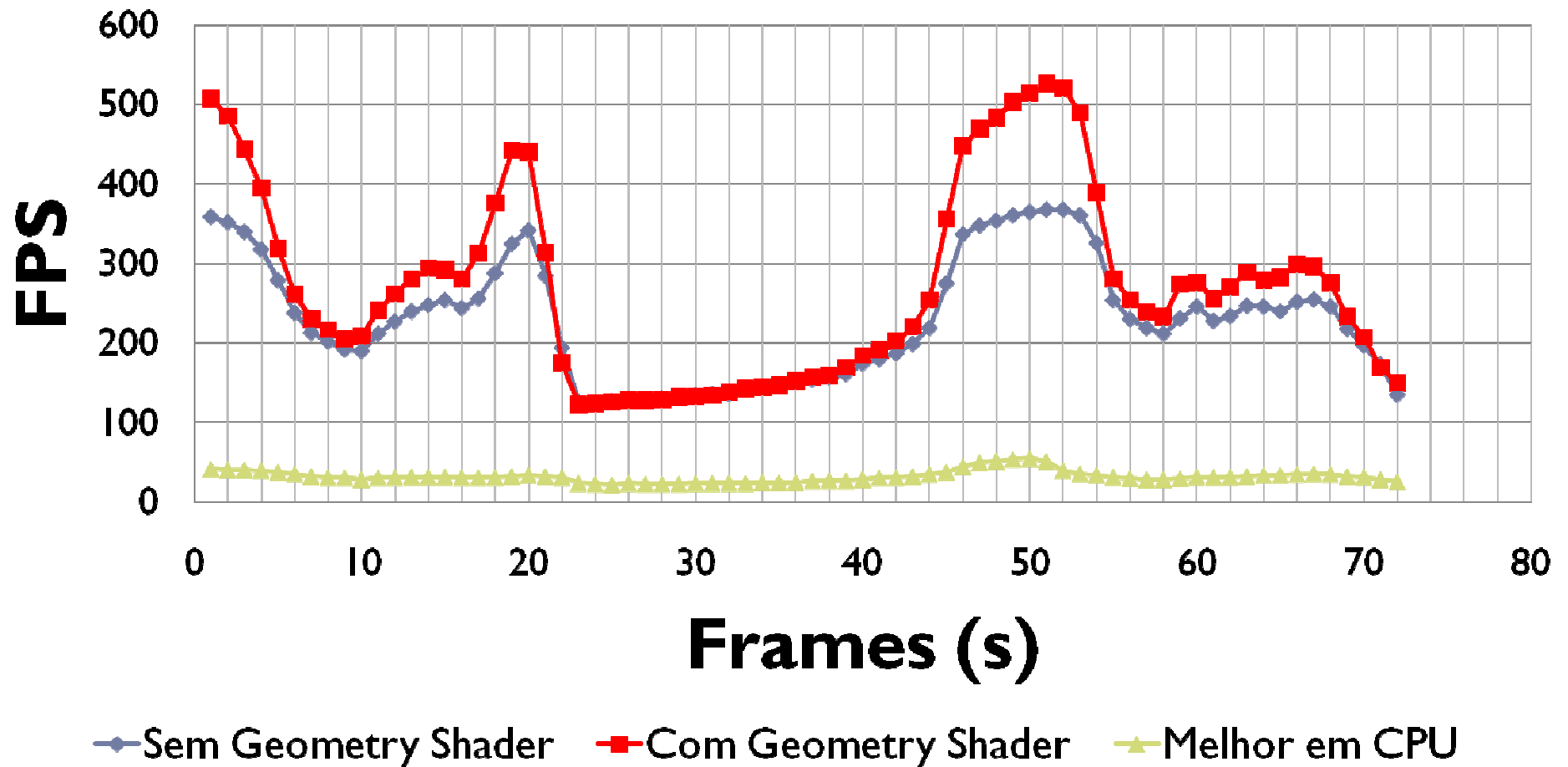
P-40



—◆— Sem Geometry Shader —■— Com Geometry Shader —▲— Melhor em CPU

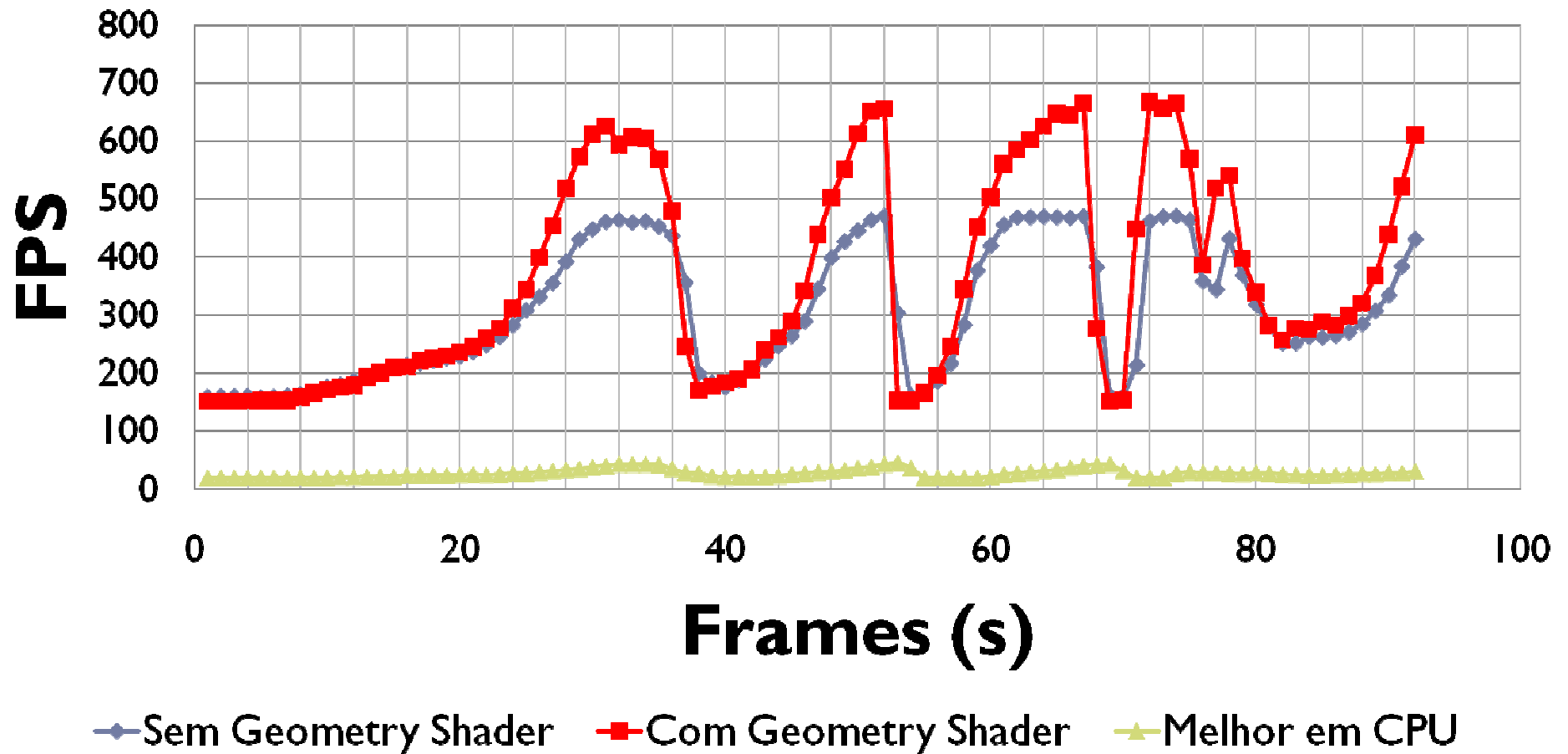
GPU FC em modelos genéricos (3)

P-43



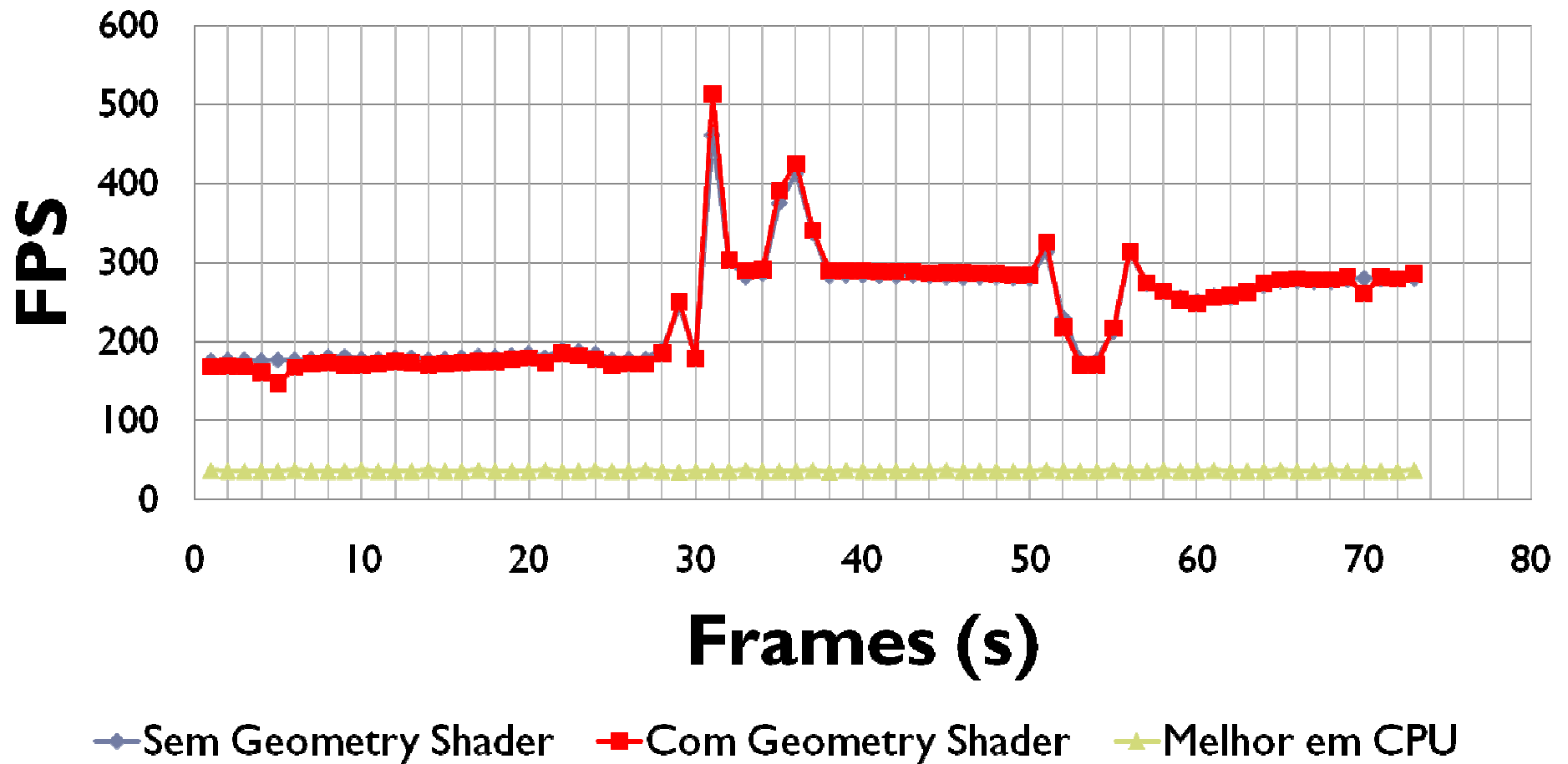
GPU FC em modelos genéricos (3)

P-50



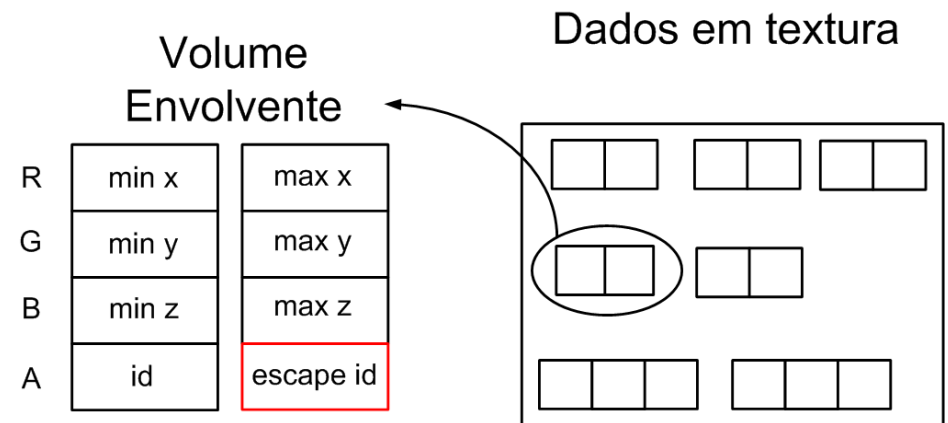
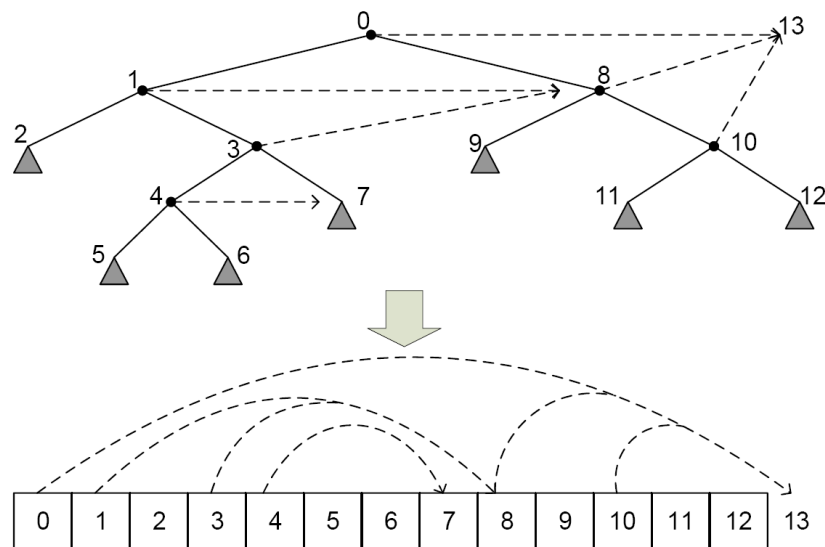
GPU FC em modelos genéricos (3)

Boeing



GPU FC nas GPU primitives e modelos genéricos (1)

- **Hierarquia** [Simonsen, L. O. and Thrane, N., (2005). A Comparison of Acceleration Structures for GPU Assisted Ray Tracing. Master's thesis]



GPU FC nas GPU primitives e modelos genéricos (2)

- ▶ Problemas do traversal na GPU
 - ▶ Ordem fixa
 - ▶ Limitações do geometry shader

Agenda

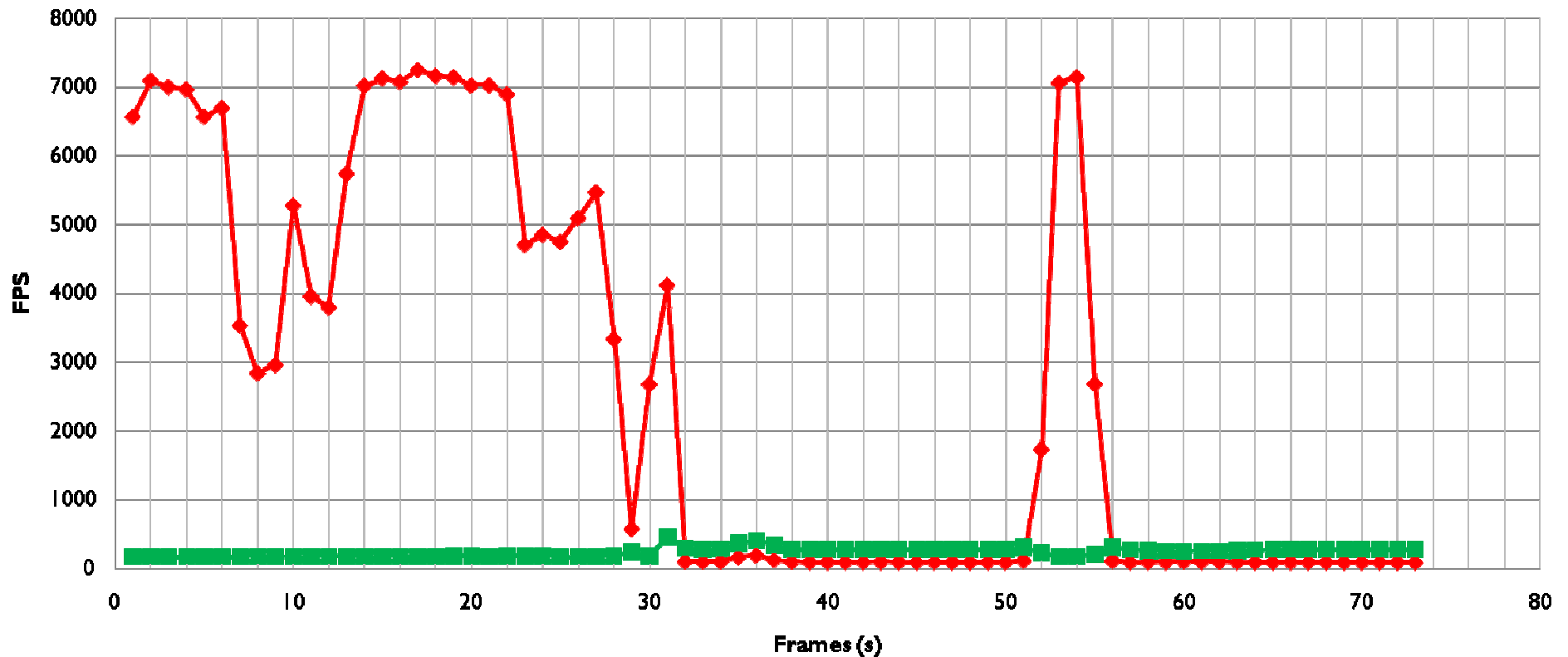
- ▶ Objetivo
- ▶ Motivação
- ▶ Algoritmo de visibilidade
- ▶ Frustum Culling na CPU
- ▶ Frustum Culling na GPU

- ▶ **Frustum Culling híbrido**
 - ▶ **Identificação do momento de transição**
 - ▶ **Paralelização do algoritmo**
 - ▶ **Possíveis estados do algoritmo**
 - ▶ **Resultados**

- ▶ Conclusão
 - ▶ Trabalhos Futuros
-

Frustum culling Híbrido (1)

▶ CPU X GPU **Boeing**



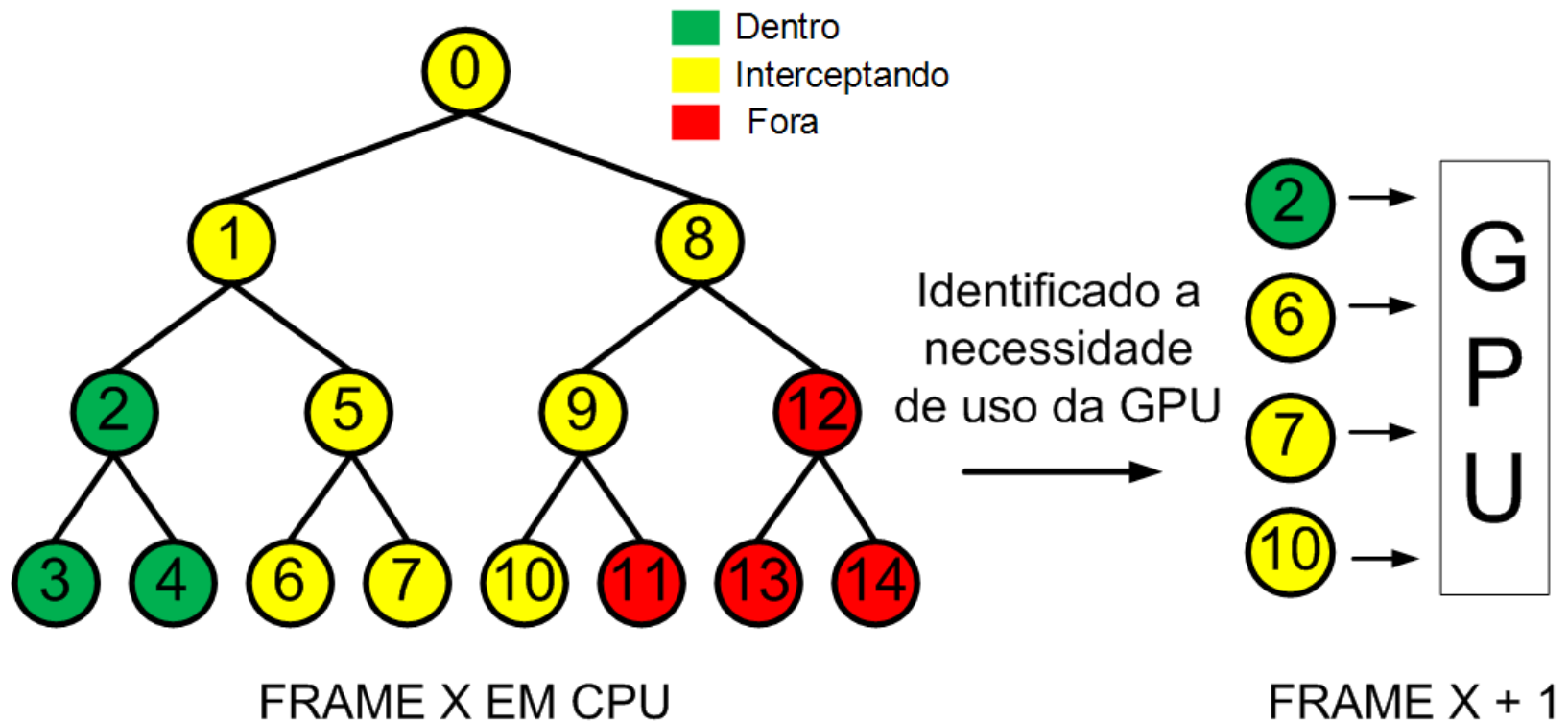
—◆— Melhor em CPU —■— Melhor em GPU

Frustum culling Híbrido (2)

- ▶ Identificação do momento de transição
 - ▶ Altura da hierarquia
 - ▶ Número de interseções
 - ▶ Porcentagem de nós processados
 - ▶ Tempo de processamento

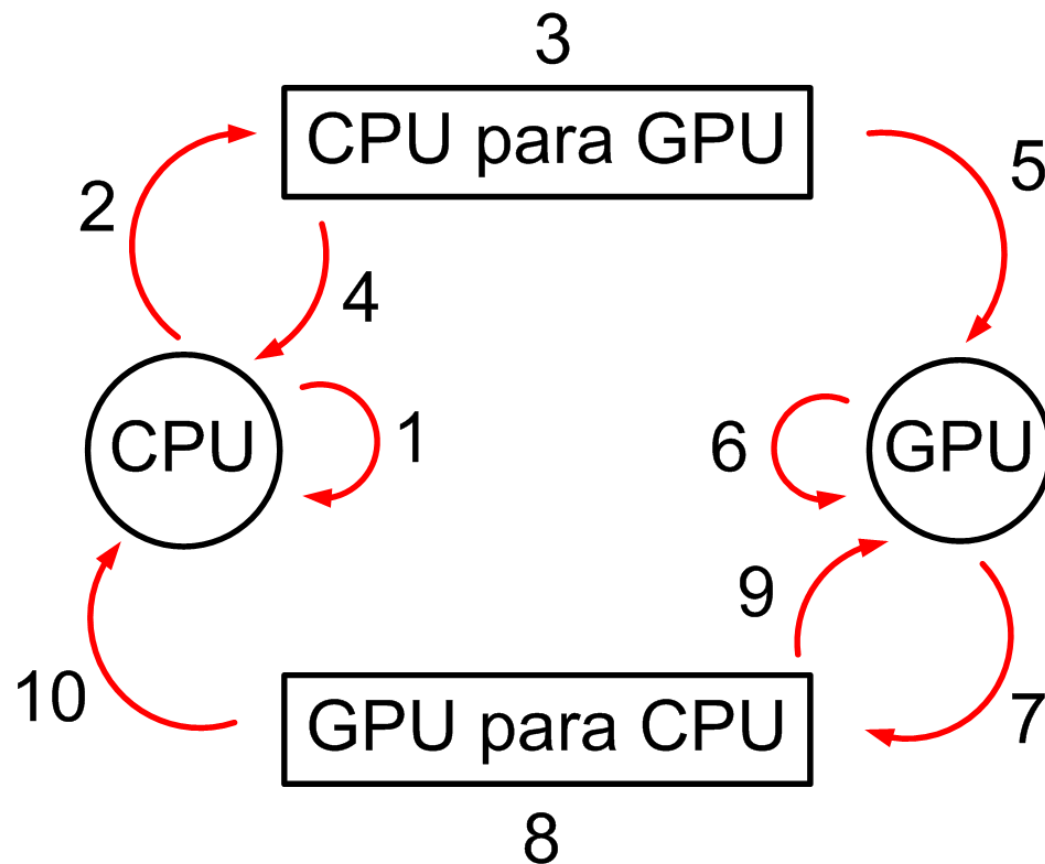
Frustum culling Híbrido (3)

▶ Paralelização do algoritmo



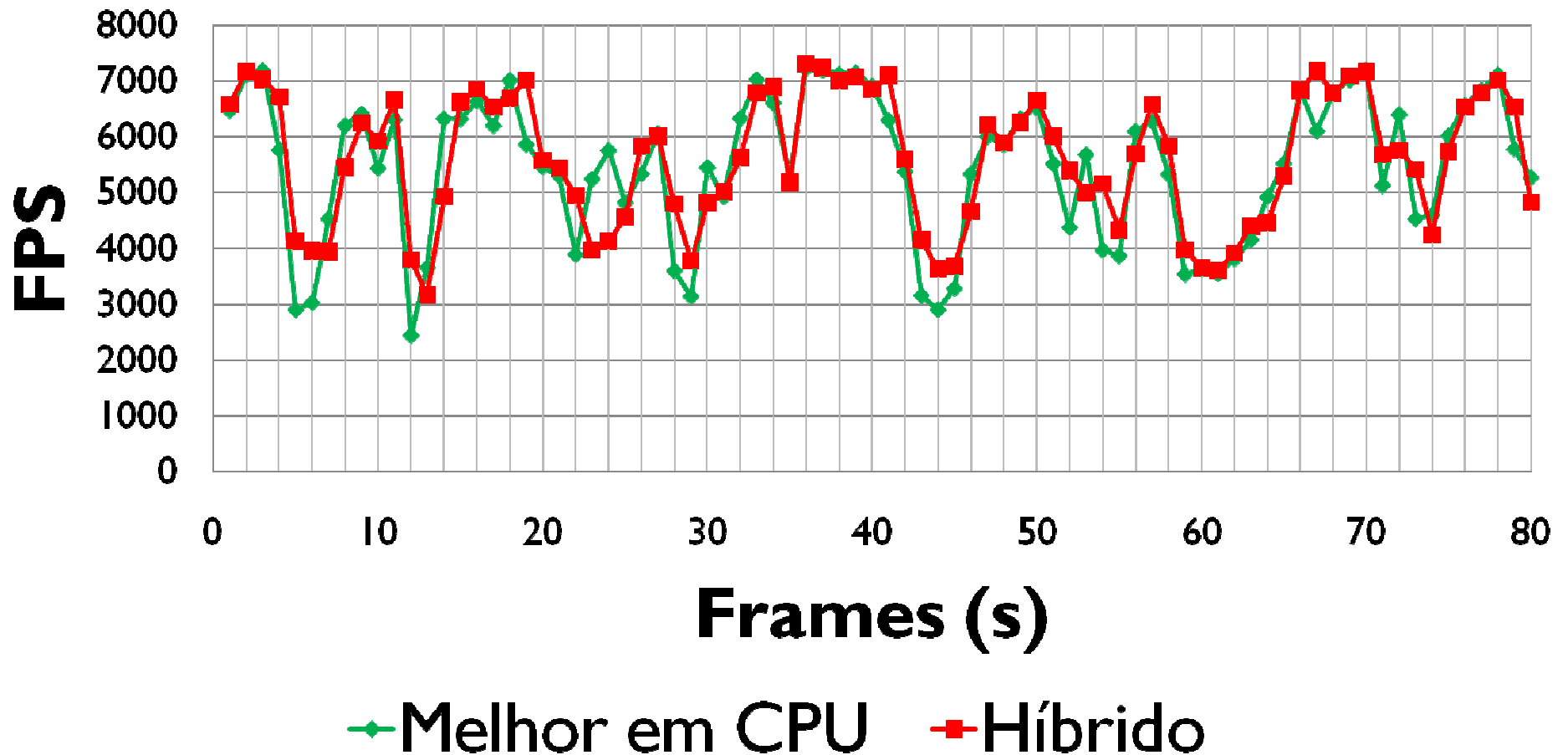
Frustum culling Híbrido (4)

► Possíveis estados do algoritmo



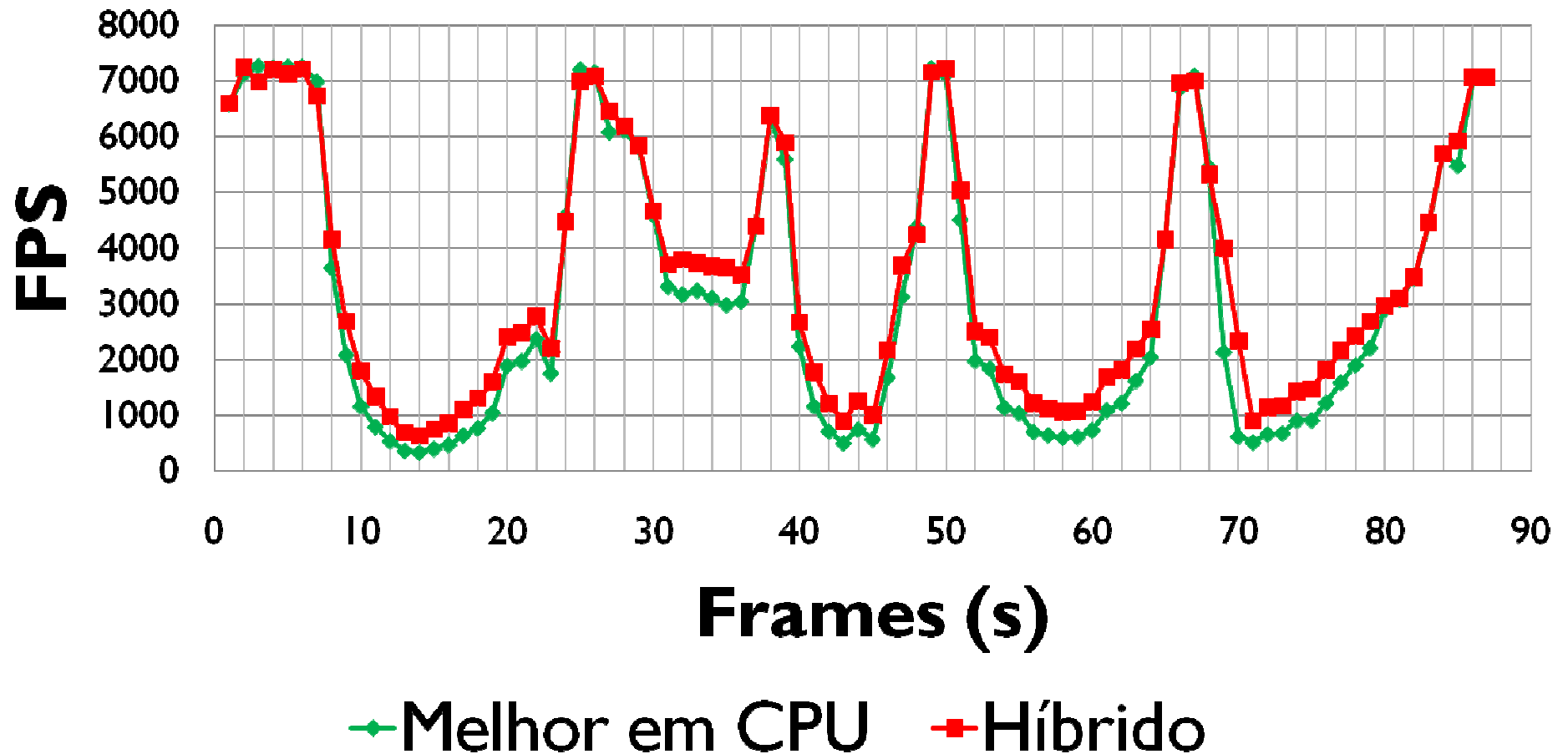
Frustum culling Híbrido (5)

P-38



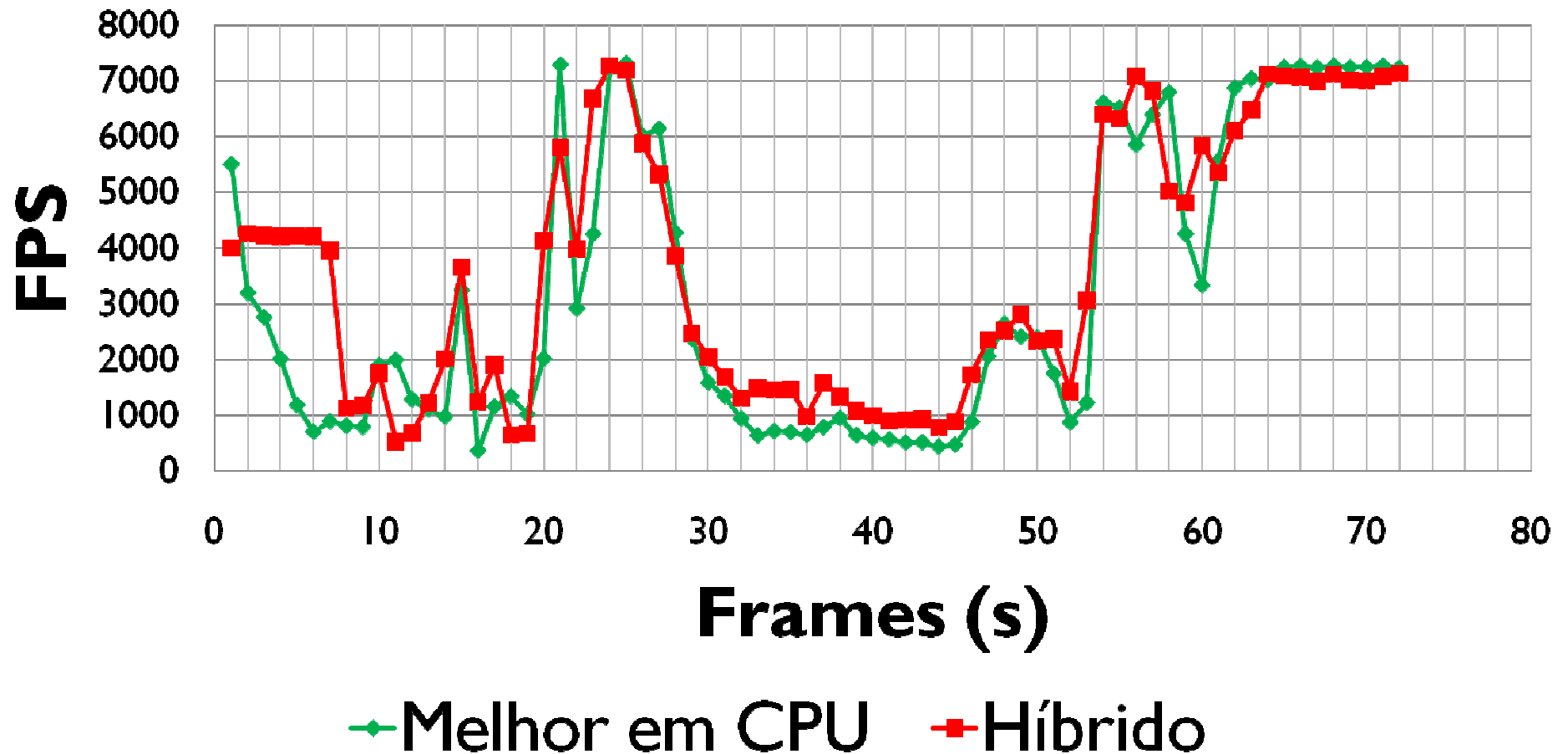
Frustum culling Híbrido (5)

P-40



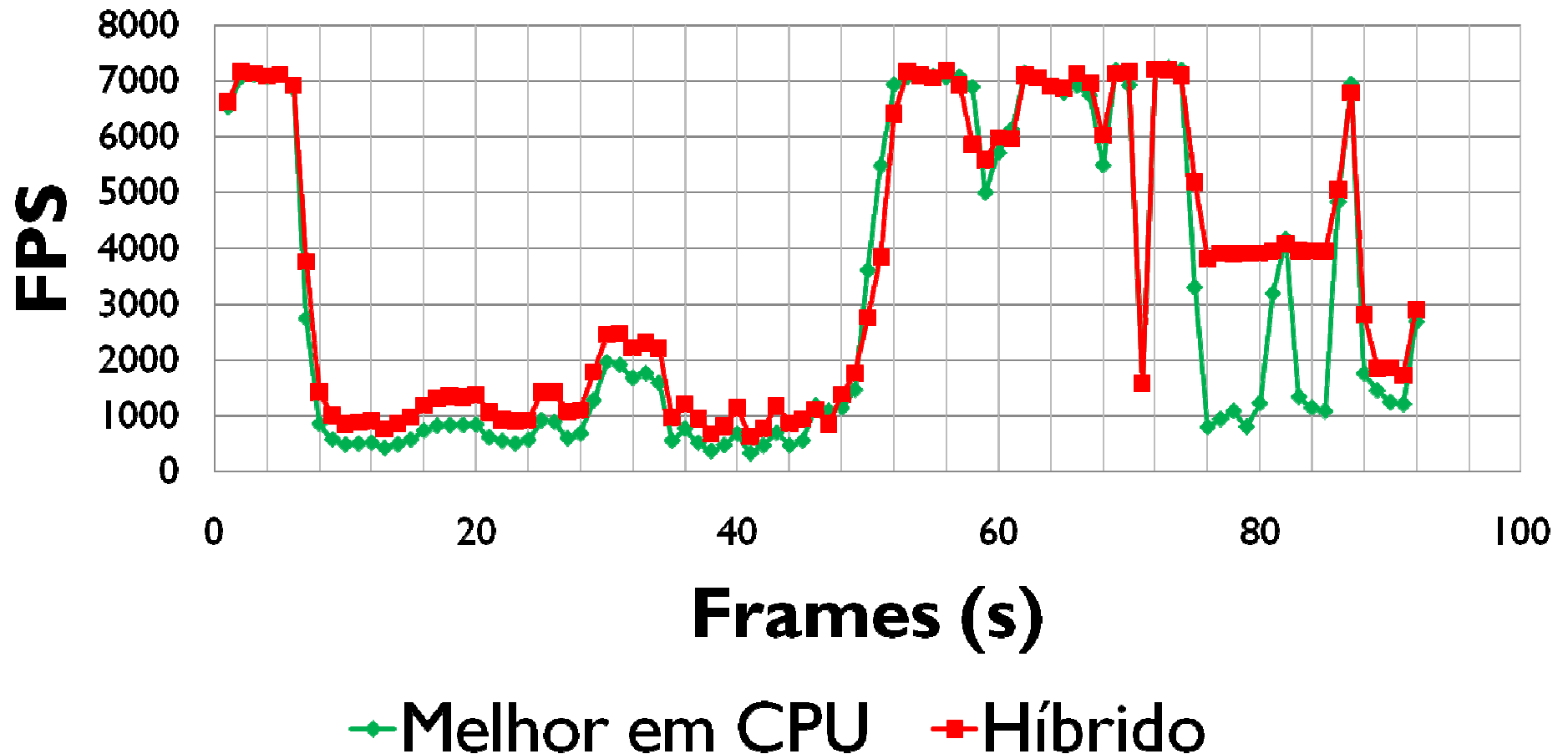
Frustum culling Híbrido (5)

P-43



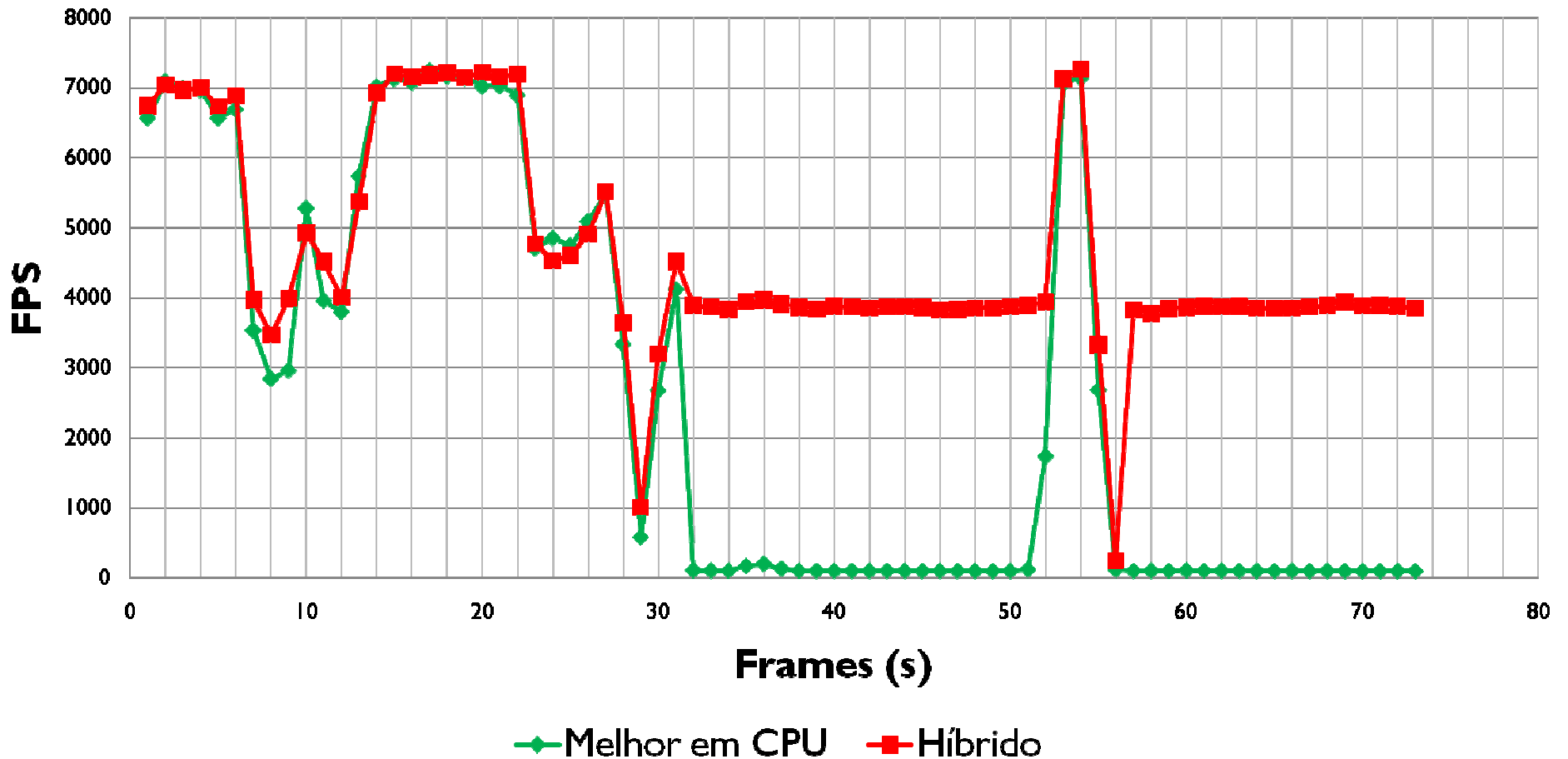
Frustum culling Híbrido (5)

P-50



Frustum culling Híbrido (5)

Boeing



Conclusão

- ▶ Bom desempenho da CPU
- ▶ Frustum culling em GPU
- ▶ Heurística híbrida

Trabalhos Futuros

- ▶ Melhor investigação do multiprocessamento
- ▶ Minimizar memória utilizada na GPU_[Mah05]
- ▶ Melhoria no algoritmo de Reshetov
- ▶ Técnicas de aceleração em GPU
- ▶ Determinação dos momentos ideais
- ▶ Estrutura Híbrida

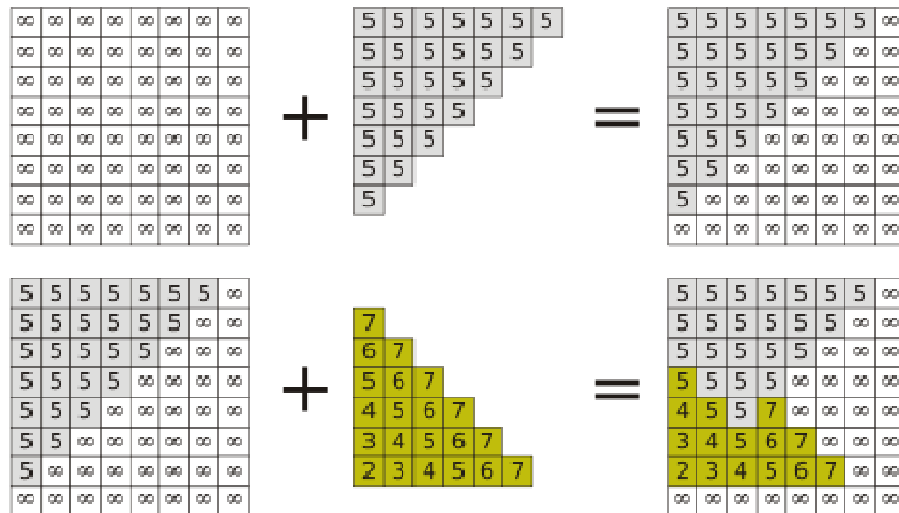
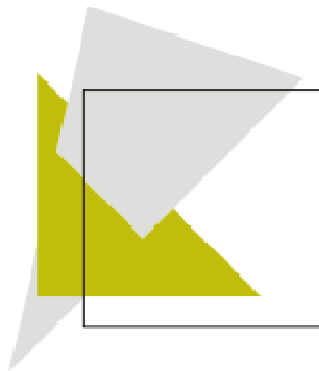
Perguntas ?

Contato:

etc@tecgraf.puc-rio.br

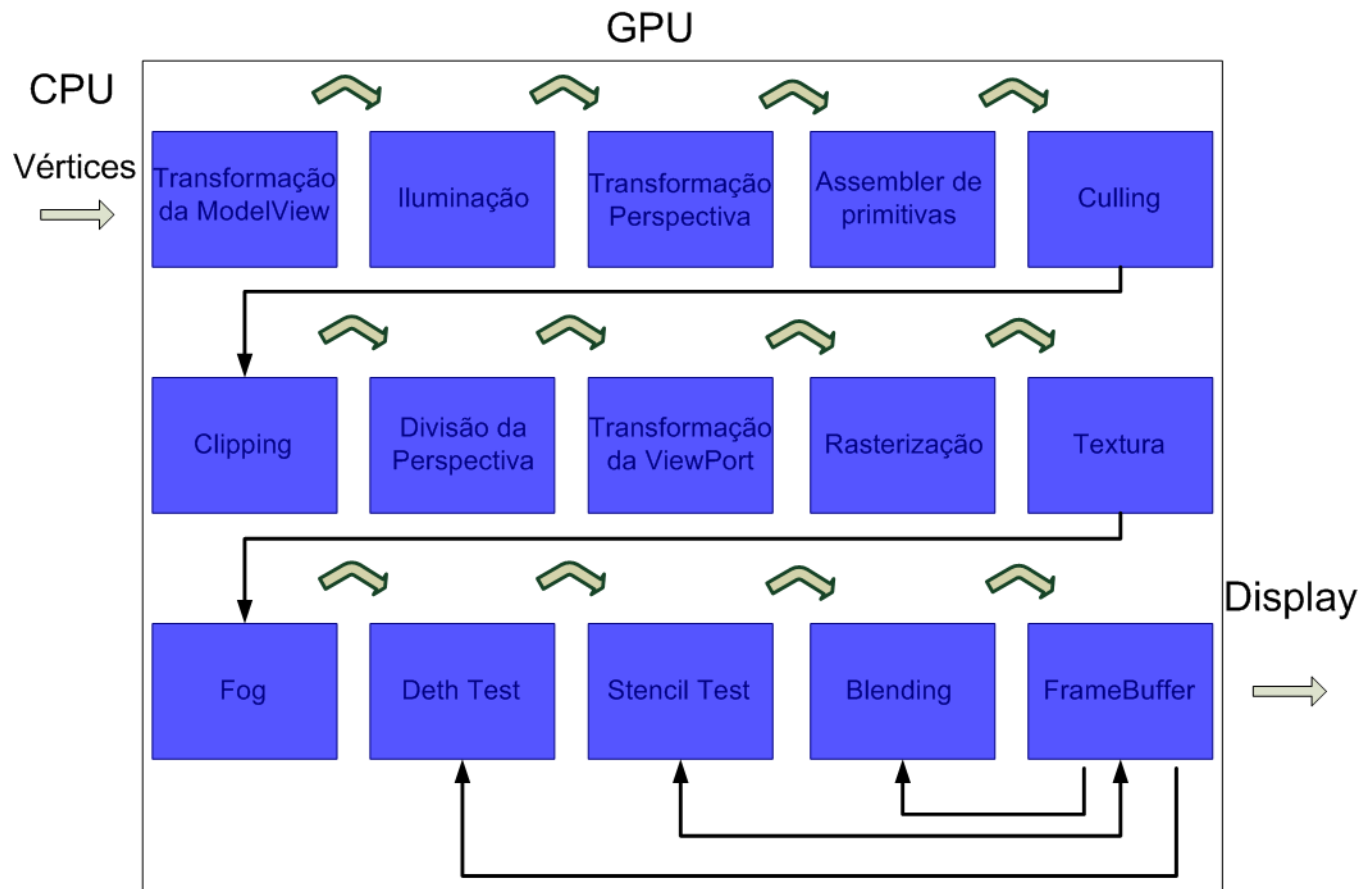
Algoritmos de visibilidade

- ▶ Hidden surface removal
 - ▶ Algoritmo do pintor
 - ▶ Z-Buffer
 - ▶ Ray tracing



Pipeline do Frustum Culling

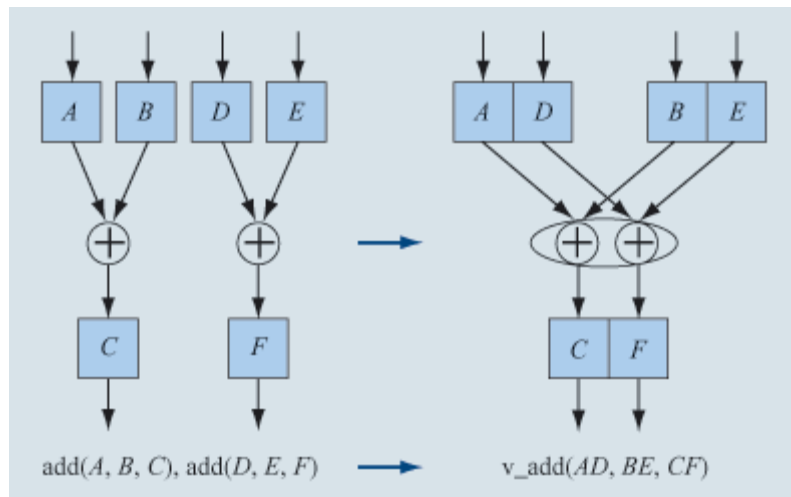
► Pipeline da GPU



Otimizações (7)

▶ SIMD

- ▶ Extração dos planos
- ▶ Volume envolvente da câmera
- ▶ Cálculo de distância

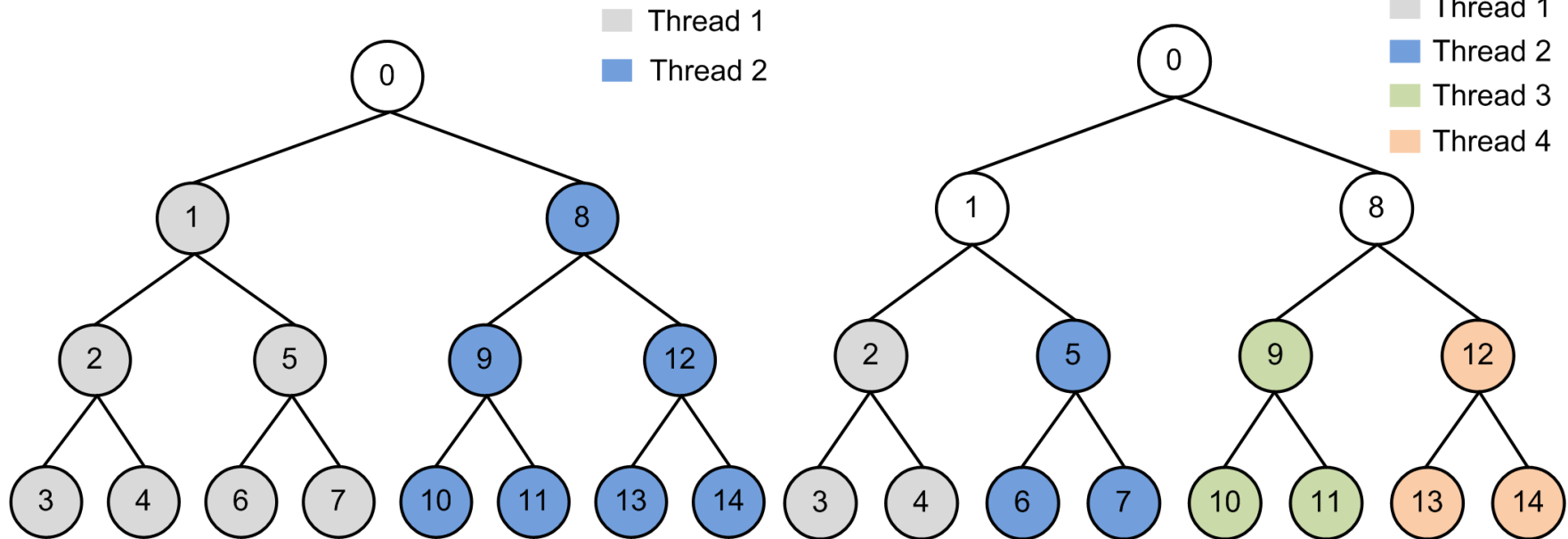


Modelos	SIMD
P-38	-35.17 %
P-40	-30.42 %
P-43	-34.06 %
P-50	-51.66 %
Boeing	-67.23 %

Multiprocessamento (1)

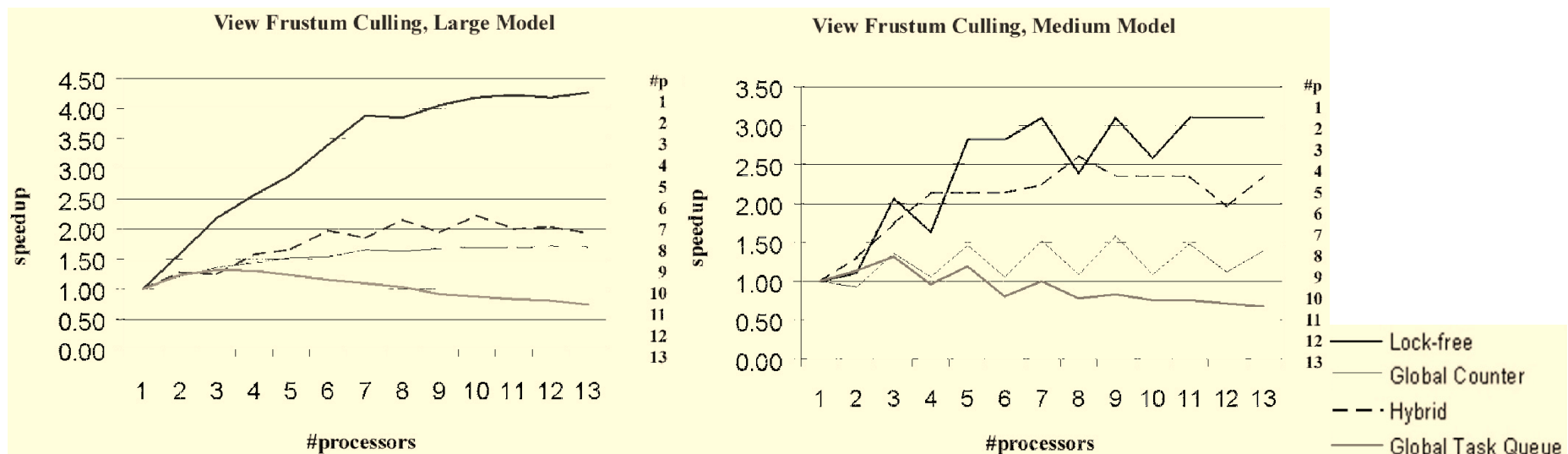
- ▶ 2 Threads
 - ▶ 4 Threads
- } OpenMP 2.5

Modelos	2 Threads	4 Threads
P-38	-5.40 %	-17.48 %
P-40	-12.15%	-22.61%
P-43	-0.35 %	-17.48 %
P-50	0.90 %	-8.44 %
Boeing	-5.36 %	-17.32 %



Multiprocessamento (2)

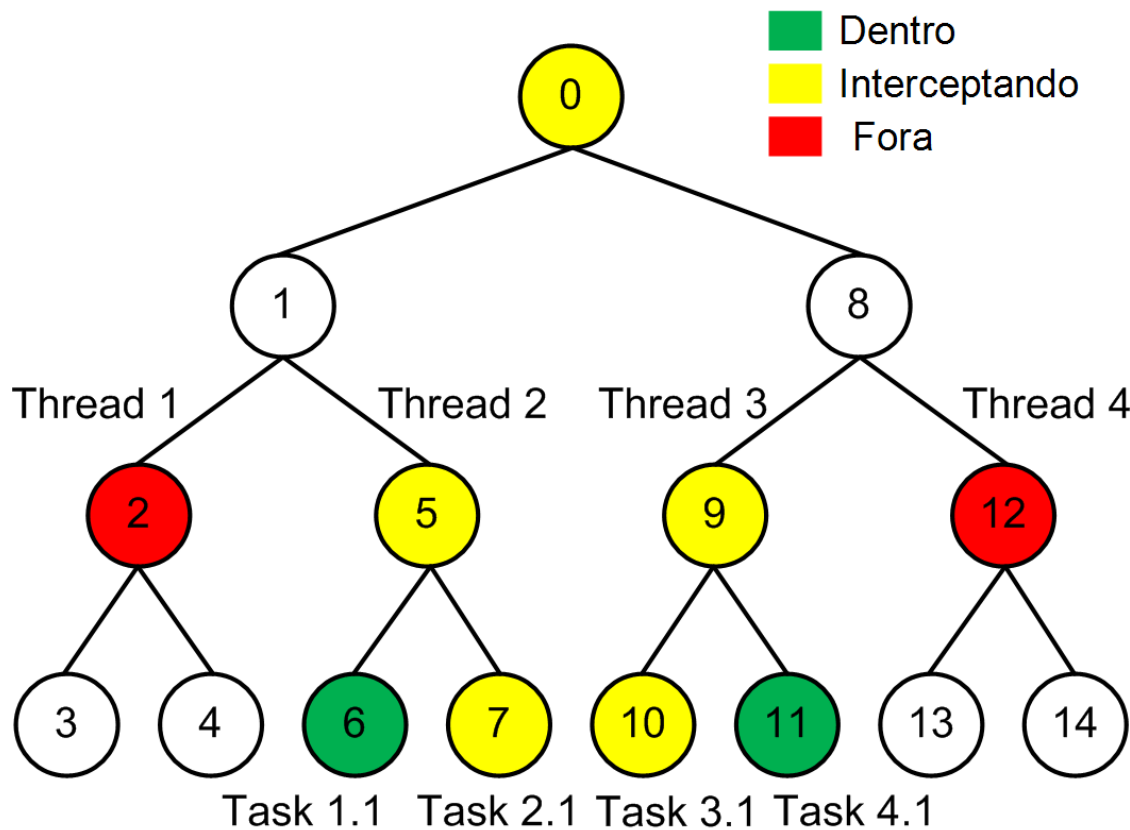
- ▶ Global Task Queue
- ▶ Global Counter Scheme
- ▶ Hybrid Scheme
- ▶ Lock-free Scheme



From U. Assarsson, P. Stenstrom, A Case Study of Load Distribution in Parallel View Frustum Culling and Collision Detection. Journal LECTURE NOTES IN COMPUTER SCIENCE, 1999

Multiprocessamento (3)

► OpenMP 3.0



Modelos	Lock-Free	OpenMP 3.0
P-38	-51.98 %	-45.35 %
P-40	-29.16 %	-32.35 %
P-43	-84.86 %	-56.86 %
P-50	-28.14 %	-24.92 %
Boeing	-37.59 %	-35.81 %