

Multi-streaming and multi-GPU optimization for a matched pair of Projector and Backprojector

Nicolas Georgin^{1,2}, Camille Chapdelaine^{1,2}, Nicolas Gac¹, Ali Mohammad-Djafari¹, Estelle Parra²

¹Groupe Problèmes Inverses, L2S (CentraleSupélec - CNRS - Univ Paris Sud), 3 rue Joliot Curie, 91192 Gif-sur-Yvette, FRANCE.

²SAFRAN SA, Safran Tech, Rue des Jeunes Bois-Châteaufort CS 80112, 78772 Magny-Les-Hameaux, FRANCE.

camille.chapdelaine@safrangroup.com, nicolas.gac@l2s.centralesupelec.fr

Fully3D 2019, June 2-5, Philadelphia

Summary: Iterative reconstruction methods are used in X-ray Computed Tomography in order to improve the quality of reconstruction compared to filtered backprojection methods. However, these methods are computationally expensive due to repeated projection and backprojection operations. Among the possible pairs of projector and backprojector, the Separable Footprint (SF) pair [1] has the advantage to be matched in order to ensure the convergence of the reconstruction algorithm. Nevertheless, this pair implies more computations compared to unmatched pairs commonly used in order to reduce the computation time. In order to speed up this pair, the projector and the backprojector can be parallelized on GPU. Following one of our previous work [2], in this work, we propose a new method which takes benefits from the factorized calculations of the SF pair in order to increase the number of data handled by each thread. We also describe the adaptation of this implementation for multi-streaming computations. The method is tested on large volumes of size 1024^3 and 2048^3 voxels.

Separable Footprint Projector/Backprojector pair [1]

The SF projector computes on each detector cell (u_e, v_e, ϕ) :

$$g(u_e, v_e, \phi) = l_{\theta_c}(u_e, v_e) \sum_{x_e, y_e} l_{\phi_v}(\phi; x_e, y_e) F_{trans}(u_e, \phi; x_e, y_e) \sum_{z_{min}}^{z_{max}} F_{ax}(v_e, \phi; x_e, y_e, z_e) f(x_e, y_e, z_e) \quad (1)$$

The SF backprojector computes on a each voxel (x_e, y_e, z_e) :

$$b(x_e, y_e, z_e) = \sum_{\phi} l_{\phi_v}(\phi; x_e, y_e) \sum_{v_{min}}^{v_{max}} F_{ax}(v_e, \phi; x_e, y_e, z_e) \times \sum_{u_{min}}^{u_{max}} l_{\theta_c}(u_e, v_e) F_{trans}(u_e, \phi; x_e, y_e) g(u_e, v_e, \phi) \quad (2)$$

$F_{trans}(u_e, \phi; x_e, y_e)$: transaxial footprint of the voxel

$F_{ax}(v_e, \phi; x_e, y_e)$: axial footprint of the voxel

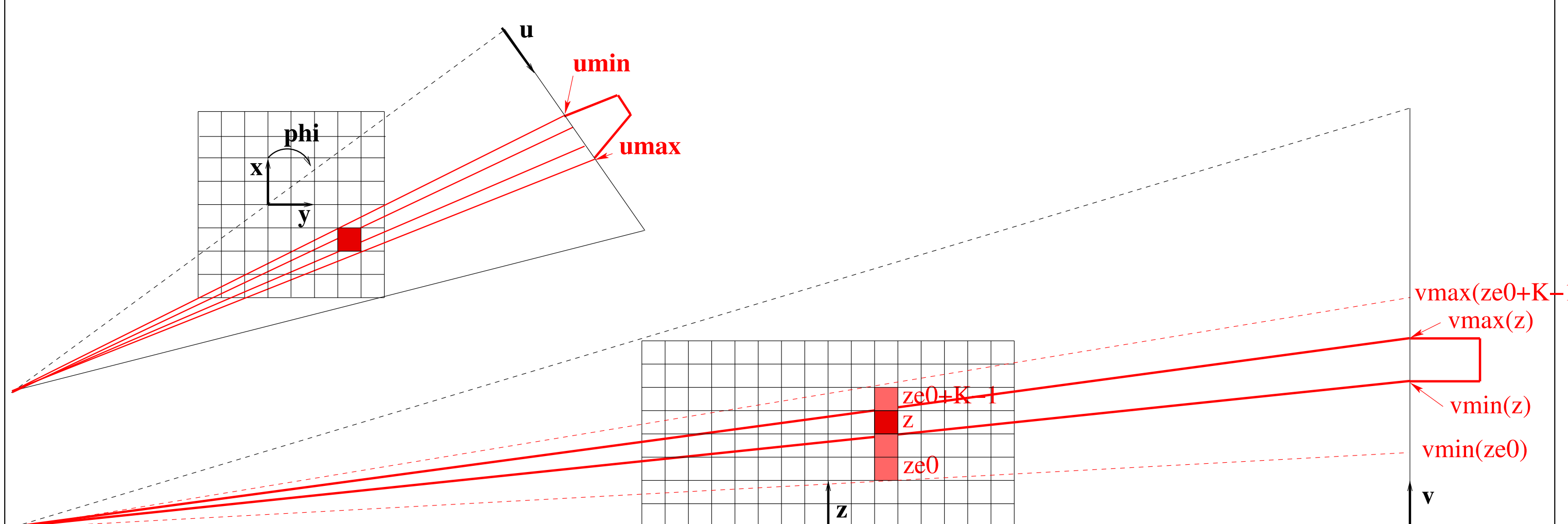
$l_{\theta_c}(u_e, v_e)$ and $l_{\phi_v}(\phi; x_e, y_e)$: amplitude functions (A2 method [2])

Proposed backprojector GPU parallelization

```

Initialize SHARED_Bback[z_e0 : z_e0 + K - 1] = 0
for each phi do
  Compute v_min(z_e0) and v_max(z_e0 + K - 1)
  Initialize SHARED_Ftrans[v_min(z_e0) : v_min(z_e0 + K - 1)] = 0
  for u_e = u_min to u_max do
    Compute Ftrans(u_e, phi; x_e, y_e)
    for v_e = v_min(z_e) to v_max(z_e + K - 1) do
      SHARED_Ftrans[v_e] += Ftrans(l_theta_c(u_e, v_e) g(u_e, v_e, phi))
    end for
  end for
end for
for z_e = z_e0 to z_e0 + K - 1 do
  Compute v_min(z_e) and v_max(z_e) and initialize B_phi = 0
  for v_e = v_min(z) to v_max(z) do
    Compute F_ax(v_e, phi; x_e, y_e)
    B_phi += F_ax * SHARED_Ftrans[v_e]
  end for
  Compute and store in shared memory SHARED_Bback[z_e] += l_phi_v * B_phi
end for
for z_e = z_e0 to z_e0 + K - 1 do
  B(x_e, y_e, z_e) = SHARED_Bback[z_e]
end for
end for

```



SF Parallelization on one GPU

| cells/voxels par thread | Time P (s) | Time BP (s) |
|-------------------------|-------------|-------------|
| 1 | 187,7 | 102,0 |
| 2 | 118,2 | 62,8 |
| 4 | 82,3 | 43,3 |
| 8 | 67,5 | 33,8 |
| 16 | 61,9 | 37,9 |

| Implementation | Time P (s) | Time BP (s) |
|----------------|------------|-------------|
| SF [this work] | 11,9 | 6,7 |
| SF [3] | 12 | 7,7 |
| SF [4] | 7,2 | 4,6 |

Table: Runtime of the SF pair depending on the number of cells or voxels (Volume: 1024^3 voxels; Projections: $1024 \times 1536 \times 1152$; GPU: Titan V)

Table: Runtime of the SF pair for different implementations (Volume: $512^2 \times 64$ voxels; Projections: $984 \times 64 \times 888$; GPU: K80)

Comparison with other P/BP pairs on one GPU

| Op | Time (s) | Tsamp /s | TFLOP | FLOP /samp | Eff. (%) | |
|----------|----------|----------|-------------|------------|----------|---------|
| BP VI | 1,37 | 4,40 | 3,23 | 1,72 | 0,39 | 9,22 % |
| BP SFTR | 33,8 | 13,19 | 0,29 | 145,2 | 11,0 | 31,28% |
| P ER | 7,43 | 9,48 | 1,28 | 9,65 | 1,02 | 9,90% |
| P Jtex2D | 3,42 | 3,96 | 1,16 | 2,48 | 0,63 | 5,33 % |
| P Jtex3D | 5,14 | 3,96 | 0,77 | 3,47 | 0,88 | 4,92 % |
| P SFTR | 61,9 | 6,22 | 0,1 | 102,7 | 0,1 | 12,05 % |

Table: Compute efficiency (Volume: 1024^3 voxels; Projections: $1024 \times 1536 \times 1152$; GPU: Titan V)

Multi-GPU and multi-streaming parallelization

P ER/BP VI (non dual pair)

| op | streams | 1 GPU | 10 GPUs |
|--------|---------|----------------------|---------------------|
| P w/o | | 16,2 s | 1,76 s \times 9,2 |
| w | | 15,6 s \times 1,04 | 1,76 s \times 8,9 |
| BP w/o | | 2,38 s | 442 ms \times 5,4 |
| w | | 1,67 \times 1,43 | 397 ms \times 4,2 |

P/BP SF (dual pair)

| op | streams | 1 GPU | 10 GPUs |
|--------|---------|-----------------------|----------------------|
| P w/o | | 103,9 s | 10,4 s \times 9,97 |
| w | | 103,2 s \times 1,00 | 10,4 s \times 9,92 |
| BP w/o | | 80,7 s | 8,27 s \times 9,76 |
| w | | 80,7 s \times 1,00 | 8,30 s \times 9,72 |

Table: Execution Time on a 10 GPU server (Volume: $1024^2 \times 640$; Projections: $1024 \times 1536 \times 640$; GPU: Titan X Maxwell)

- [1] Y. Long *et al.*, "3D forward and back-projection for X-ray CT using separable footprints," *IEEE Trans. Med. Imag.*, vol. 29, no. 11, pp. 1839–1850, 2010.
- [2] C. Chapdelaine *et al.*, "New GPU implementation of Separable Footprint Projector and Backprojector : first results," in *CT meeting, Salt Lake City*, 2018.
- [3] M. Wu *et al.*, "GPU acceleration of 3D forward and backward projection using separable footprints for X-ray CT image reconstruction," in *Fully 3D 2011*.
- [4] X. Xie *et al.*, "Accelerating separable footprint (SF) forward and back projection on GPU," in *SPIE Medical Imaging*, 2017.