Application Paper

Axial Tomographic Reconstruction

Computed axial tomographic reconstruction using back-projection is presented as a challenging real-world application illustrating the performance obtainable with Alacron's DPLM architecture, compared to standard cluster architectures. Implementation on a mesh architecture computer is impractical because the large amounts of memory that are required.

Description of the Application

The reconstruction region is a square array of 1024 x 1024 pixels centered on the axis of rotation of the x-ray head (Figure 1). Data is collected from the x-ray head as radial scans, with 4 channels of 4096 values for each of 360 angles. The data collected occupies 4*4096*360*2 = 11,796,480 bytes. The data is moved into memory at an average rate of 5 megabytes per second (Figure 2).

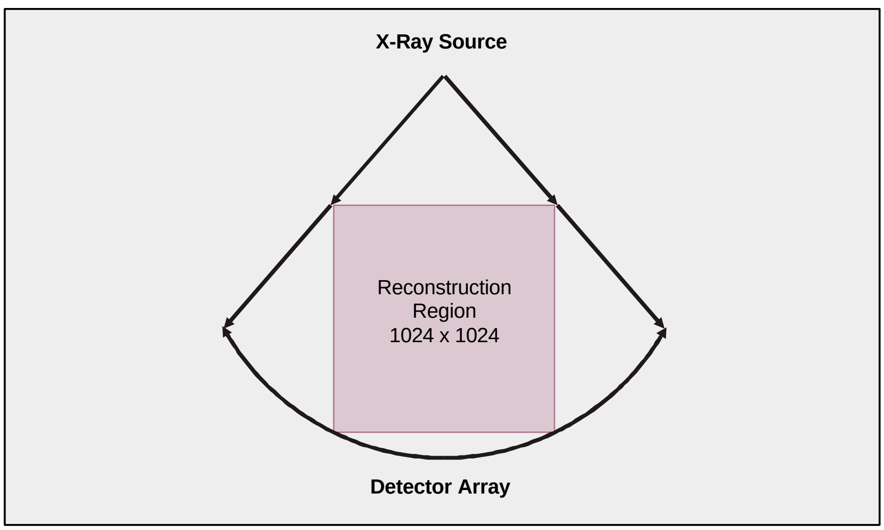

Figure 1: Relation of the region of reconstruction to the x-ray source and detector array.

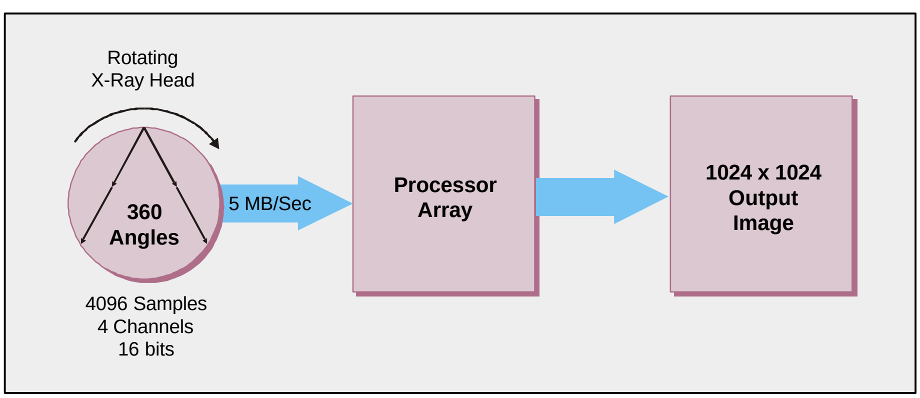

Figure 2. Data collection.

The following steps reconstruct the pixel densities:

1. Calculation of the density estimate from the output of the four detectors at the end of each beam path.
2. Back projection. Conversion of the fan-shaped polar form of the data to the square array of pixels in the reconstruction region is accomplished by summing the density along each ray into each pixel the ray encounters. Linear interpolation is used when the ray passes between two pixels (Figure 3).

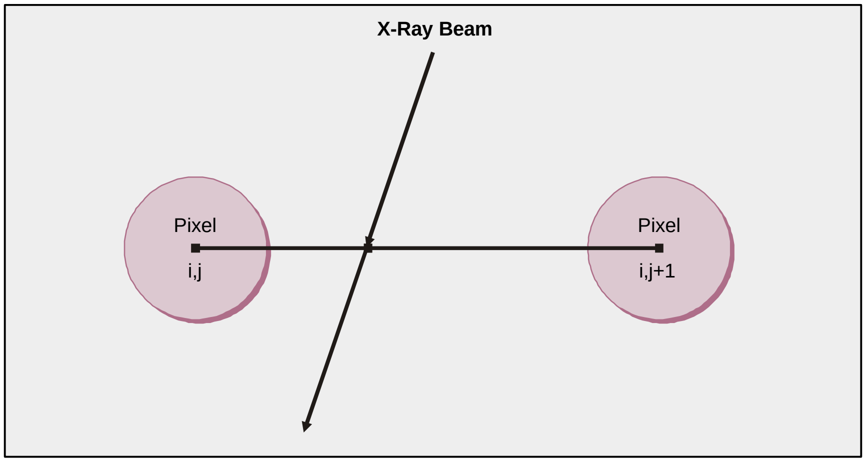

Figure 3. Back Projection.

Four channels of data are used to estimate the actual density along the beam path to compensate for the normal beam hardening that occurs as x-rays pass through an object.

The value summed into each pixel is:

$$p[i, j] += (1 - \alpha) D_{beam}$$

$$p[i, j+1] += \alpha D_{beam}$$

$$\alpha = \frac{x_{pixel[i,j+1]} - x_{beam}}{\Delta x}$$

where D_{beam} is the density measured for the current beam being processed and a is determined from the geometry of the beam. Application of a 2D FFT to the image array.

3. Application of a weighting factor of the distance of each pixel from the axis of rotation. This compensates for the greater number of rays passing through pixels closer to the axis of rotation.
4. Application of an inverse 2D FFT to the whole array.
5. Steps 2 through 5 effectively apply a convolution to the data to correct for the weighting error generated by the beam geometry, giving a good approximation to the density distribution of the object positioned in the reconstruction area.

Algorithmic Complexity

The number of operations performed in each step can be counted as follows. These are summarized in Table 1.

1. Density estimation from detector data

   A fifth-order polynomial is applied to the output of each of the four detectors at the end of every beam path, and the results of these four polynomials are summed to give the density estimate. This step requires 4x(5 multiplies + 5 adds) plus three adds per beam path, or 81,920 MACs (multiply accumulates) and 12,288 adds.

For 360 angles, the total number of MACs is 29,491,200, and the total number of adds is 4,423,680. Additionally, each detector has an array of 24 coefficients and 4 values that must be read from memory, followed by a write of the density estimate. This results in 41,287,680 reads from memory, and 1,474,560 writes to memory to process all 360 angles.

2. Back projection.

   For each angle, one third of the beams path pass into the top and out from the bottom of the reconstruction region, encountering 2 pixels for each of the 1024 lines it crosses. The weights for each beam path is precomputed, and so each of the 2048 pixels on these 1366 beams requires one MAC, or a total of 2,797,568 MACs. The other two thirds of the beam paths access from 3 to 2048 pixels depending on how far up the left or right side of the reconstruction region the ray path lies. In total, these beam paths access 2,300,323 pixels, requiring 2,300,323 MACs. For all beams, then, 5,097,891 MACs are needed. Each MAC requires a pixel value to be read from memory, a table value to be read from memory, and a final write of the new pixel value back to memory. For all of the 360 angles, the back-projection step required 1,835,240,760 MACs, 3,670,481,520 reads from memory, and 1,835,240,760 writes to memory.

3. 2D FFT

   A 1024 x 1024 2D real FFT is composed of 2048 real 1024 point FFTs, 1024 along the rows and 1024 along the columns. Two real signals can be processed concurrently with one complex 1024-point FFT plus 2,044 additional adds. Two real FFTs can be performed by this method in 460 µs on a single 40 MHz SHARC 2106x processor, averaging to 230 µs for a 1024 point real FFT. The whole 2D FFT thus requires 0.470 seconds plus, plus 0.013 seconds required to read and write the whole array twice from memory, or 0.483 seconds on a single processor.

4. Pixel weighting by radial position

   A precomputed table of radius values is read with the image data, multiplied, and written back to memory. This step requires 2,097,152 reads from memory, 1,048,576 multiplies, and 1,048,576 writes to memory.

5. Inverse 2D FFT

   The 2D inverse FFT requires the same number of operations as required by the 2D FFT in step 3.

Step	MACs	Adds	Reads	Writes
1	29,491,000	4,423,000	41,287,000	1,474,000
2	1,835,240,000	0	3,670,481,000	1,835,240,000
3	28,800,000*		2,097,000	2,097,000
4	1,048,000		2,097,000	1,048,000
5	28,800,000*		2,097,000	2,097,000
Total	1,923,380,000	4,423,000	3,718,060,000	1,841,958,000

*Based on averaged 230 µs FFT time.

Table 1. Operations required

Parallelization of the Algorithm

In order to improve performance, the computation will be divided between processors by splitting the reconstruction region into horizontal stripes. The first processor will get the first N rows (N = 1024 / number of processors), and each processor will get succeeding rows. The algorithm is inherently linear allowing this natural division between processors.

Memory Requirements

This algorithm uses several pre-computed tables. Each detector has a table of 24 calibration coefficients for a total of 98,304 four-byte floating point numbers. The pixel radius value table (distance from the rotation axis) requires 131,072 floats, taking advantage of symmetries. Again taking advantage of symmetries the ray table requires 57,351,273 floats. The reconstructed image requires 1,048,576 floats, and the input data requires 11,796,480 bytes. The total memory requirement is therefore under 256 megabytes.

Cluster Mode Architecture Analysis

The size of global memory and the pre-computed tables make the use of SRAM prohibitive, in terms of cost, physical size, and power requirements. This analysis assumes that global memory is DRAM, accessible in 2 clocks per cycle (i.e. one-wait-state). The reconstructed image will, however, be kept in SRAM accessible in 1 clock cycle (i.e. no-wait-states). This analysis assumes a cluster of six processors, a limitation imposed by the design of the SHARC.

Since a single SHARC can access memory at 160 MB/sec (40 megafloats/sec), six processors will compete for global memory reducing the average access time to SRAM to 6.67 megafloats per second, and the access time for DRAM to 3.33 megafloats per second. In the density estimation step, one processor broadcasts calibration data to all SHARCs in the cluster, and data sensor is accessed by each SHARC. In the back-projection step, data and table values are broadcast to each SHARC by one of the processors in the cluster. Broadcasting these data reduces the bus loading by 30%; however, the access by the processors to the image data still loads the bus significantly. Compared to a single processor, a 6-processor cluster therefore is able to improve performance by about 50%.

For the inverse 2D FFT, each processor reads rows of data from SRAM memory performing the FFT calculation in internal memory. This step is limited by both computational time, and the time required to twice read and write the data in the image array. Memory bandwidth restriction limits the improvement in performance to a factor of 2.3. Similar consideration applies to the 2D FFT. The pixel-weighting step is bus limited, and is determined by the time required for all processors to read data and table values from memory. Additional processors do not improve performance over that obtainable from a single processor. Table 2 illustrates the time required for this algorithm on a 6-processor cluster mode system, based upon the above considerations.

Step	Seconds
1	2.138
2	122.349
3	0.209
4	0.105
5	0.209
Total	125.0

Table 2. Cluster mode performance

Alacron’s DPLM Architecture Analysis

Alacron's DPLM architecture permits several important optimizations: (The DPLM architecture is not limited to 6 processors in a cluster.)

- During the back projection calculation, all processors are able to operate in parallel, without bus contention. The reconstructed image is kept in VRAM reducing the interference between processors during the back projection phase of the computation.
- When the raw data is passed to the processors it is broadcast to all the processors by the DMA controller.
- When the ray table is broadcast to the processors, it is sent along with the raw data for each angle, reducing the memory requirements at each processor node.
- Alacron's DPLM architecture can operate in SIMD reducing the 2D FFT time significantly

In the density estimation step the raw data is DMAed into the SHARC array along with the coefficient table. This step is limited by the transfer time of the DMA controller. For the back projection step, data and table values are broadcast DMAed to the processor array. Each processor operating on 1/8 of the image in private VRAM memory processes data in VRAM. This allows each processor to access memory at 20 mega-floats per second. Although this step is memory bandwidth limited, execution on DPLM architecture is much faster than using a cluster architecture, because 8 private busses operates much faster one common bus.

The 2D FFT step uses the SIMD version of the FFT. The array is processed as rows; the resulting data are passed through the link ports in parallel with computation, and then processed in columns. Including the time used to move data to and from VRAM, the entire operation requires 0.083 sec. Pixel weighting by the radial position is limited by the time required to pass the radius table data to VRAM and the time needed to access VRAM in processing the table. The inverse 2D FFT has the same characteristics the forward 2D FFT. Additional time is required to pass the final image back to global DRAM. Table 3 indicates the execution times for an 8-processor DPLM system.

Step	Computation Time (sec)	Transfer Time DRAM to VRAM (sec)	VRAM Memory Access Time (sec)	Transfer Time out (sec)	Total Time (sec)
1	0.105	0.076	0.152	0.036	0.264
2	5.735	1.434	22.940	0.026	24.400
3	0.070	0.000	0.026	0.000	0.096
4	0.003	0.007	0.013	0.000	0.013
5	0.070	0.000	0.026	0.052	0.148
Total	5.983	1.517	23.157	0.114	24.866

Table 3. DPLM performance.

Conclusions

In this example, where performance is memory limited, the Alacron's DPLM architecture is over five times faster than the standard cluster mode architecture. This is attributable to two important differences: 1) Alacron's DPLM architecture is not limited to 6 processors, and 2) each processor operates on data in private memory isolated from the other processors. The isolated private memory accounts for about 80% of that improvement. Furthermore, if additional processors were added, the Alacron's DPLM architecture would provide nearly linear improvement in performance. In contrast, even if it were possible to add additional SHARCs to the cluster design, performance would not improve because the common bus is completely saturated by data accesses.

ALACRON

71 Spit Brook Road, Suite 200
Nashua, NH 03060
Tel: 603.891.2750   Fax: 603.891.2745
Web: www.alacron.com   E-mail: sales@alacron.com