## ARTICLE

# Comparison of TITAN Hybrid Deterministic Transport Code and MCNP5 for Simulation of SPECT

Katherine ROYSTON[1,*], Alireza HAGHIGHAT[1] and Ce YI[2]

[1] *Nuclear Engineering Program, Mechanical Engineering Department, Virginia Tech, Blacksburg, VA, 24061, USA*
[2] *Nuclear and Radiological Engineering Program, Georgia Tech, Atlanta, GA 30332, USA*

Traditionally, Single Photon Emission Computed Tomography (SPECT) simulations use Monte Carlo methods. The hybrid deterministic transport code TITAN has recently been applied to the simulation of a SPECT myocardial perfusion scan. The TITAN SPECT simulation uses the discrete ordinates formulation in the phantom region and a simplified ray-tracing formulation outside of the phantom. This paper seeks to validate the TITAN code's methodology for SPECT simulation by comparing projection images with the MCNP5 Monte Carlo code. In MCNP5 the collimator is directly modeled, but TITAN instead simulates the effect of collimator blur using a circular ordinate splitting technique. Projection images created using the TITAN code are compared to results using MCNP5 for three collimator acceptance angles. Visually the images are in good agreement, but maximum relative differences of up to 21.3% are observed. Profiles through the projection images revealed that the TITAN results followed the shape of the MCNP5 results with some differences in magnitude. A timing comparison on 16 processors found that the TITAN code completed the calculation two to three orders of magnitude faster than MCNP5, depending on the collimator acceptance angle being simulated. Both codes exhibit good parallel performance.

*KEYWORDS: SPECT, deterministic, hybrid, Monte Carlo, parallel computing, imaging*

## I. Introduction

Traditionally, Single Photon Emission Computed Tomography (SPECT) simulations have been done using Monte Carlo methods. However, the hybrid deterministic transport code TITAN[1] has been applied to SPECT simulation[2] with the goal of developing methods to improve image quality and reduce patient dose. This paper seeks to validate the TITAN code's methodology for SPECT simulation by comparison with the MCNP5 Monte Carlo code.[3]

In SPECT, a radiopharmaceutical is injected into a patient and preferentially absorbed by the tissue of interest. A gamma camera then rotates around the patient and collects projection images at different angles. A collimator is placed on top of the detector to provide spatial resolution by restricting incoming photons reaching the detector to those within a small acceptance angle. If enough projection images are obtained, the data can be reconstructed into a three dimensional image. This research simulates a myocardial perfusion scan in which Technecium-99m (Tc-99m) is preferentially absorbed by the heart wall, or myocardium. Tc-99m emits 140.5 keV gamma rays, which are detected at the gamma camera to create projection images of the heart. Myocardial perfusion scans are used to evaluate heart conditions such as coronary artery disease.

Previous work on the TITAN code's methodology for SPECT simulation has focused on initial testing[2] and verification of a hybrid adjoint methodology.[4] In this paper, we compare TITAN projection images with those generated by the MCNP5 code for three different collimator acceptance angles. Section II will describe the methods used in the simulations. Section III will discuss the results of comparing projection images and an analysis of parallel timing for MCNP5 and TITAN. Conclusions are presented in Section IV.

## II. Description of Methods

### 1. Simulation Setup

The NURBS-based cardiac-torso (NCAT) code[5] was used to generate a $64 \times 64 \times 64$ voxel phantom with dimensions of $40 \times 40 \times 40 \ cm^3$ and containing thirteen materials. The NCAT code creates material density and source distributions that are used in both the TITAN and MCNP5 codes. The source distribution has a high intensity source with a value of 75 in the myocardium and a low intensity source with a value of 2 in the blood and soft tissues. The low intensity source is present due to the fact that not all of the Tc-99m will be taken up by the myocardium. Projection images are $64 \times 64$ pixels with dimensions of $40 \times 40 \ cm^2$.

The CEPXS code[6] was used to generate three-group multigroup cross sections. The first energy group is 126.45-154.55 keV to account for a typical energy window of 20% and is the only source containing group. The second energy group is 98.35-126.45 keV and the third is 10-98.35 keV. Both the TITAN and MCNP5 codes were run using these multigroup cross sections. In a SPECT study, the source containing group is what is of interest so the first energy group will be the focus of this paper.
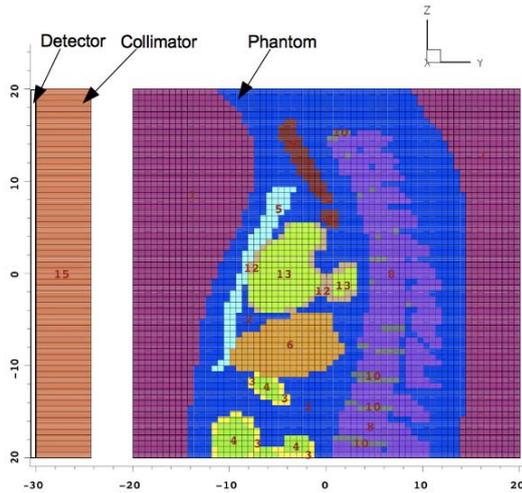
**Fig. 1** Sagittal slice through center of the MCNP5 geometry showing the phantom, collimator and detector position
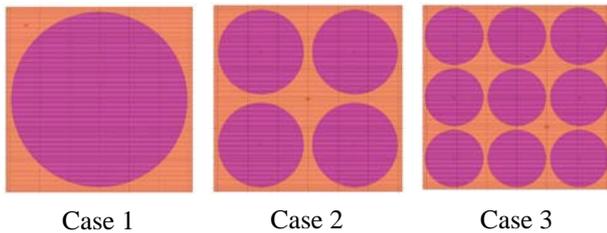


Case 1                    Case 2                    Case 3

**Fig. 2** Surface of collimator hole(s) covering a single detector pixel area (0.625 by 0.625 cm$^2$) for Case 1, Case 2, and Case 3

Three collimator sizes with varying acceptance angles were simulated. The acceptance angle is defined as the angle about the normal to the collimator within which incoming photons will reach the detector. The collimator acceptance angles for Cases 1, 2 and 3 were 2.97°, 1.42°, and 0.98°, respectively.

**2. MCNP5 Model**

A 2D view of the geometry through the center of the MCNP5 model is given in **Fig. 1**.

Figure 1 shows that the MCNP5 simulation models the collimator in front of the detector. Because the TITAN code does not actually model a detector, a simple flux tally mesh in the air behind the collimator is used to create the projection images. Each MCNP5 case has a 1-σ uncertainty of ≤3.0% in the heart flux tallies.

The collimator is a 5.7 cm deep block of lead filled with parallel holes. The diameter of these holes is then modified to achieve a different number of collimators per detector pixel (0.625 × 0.625 cm$^2$). The three collimator acceptance angles (2.97°, 1.42°, and 0.98°,) are simulated by different collimator sizes (diameters of 0.591, 0.283, and 0.194 cm) as displayed in **Fig. 2**.

Figure 2 shows only one detector pixel for each collimator case and the pattern is then repeated over the entire detector area.

**3. TITAN Model**

The TITAN code is a hybrid code because it allows the user to specify either a discrete ordinates[7] ($S_N$) or
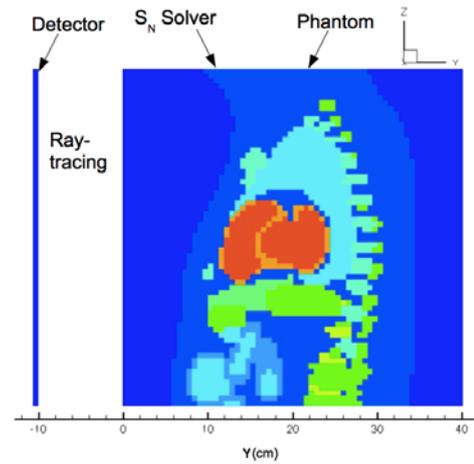


**Fig. 3** Sagittal slice through TITAN geometry with regions using the $S_N$ and CM solvers indicated

characteristics method (CM) solver in different regions of a problem. To simulate SPECT, TITAN uses the $S_N$ solver in the phantom and a simplified ray-tracing[8] formulation around the phantom as depicted in **Fig. 3**.

The detector on the left in Fig. 3 is not modeled in TITAN. The TITAN code also does not model the collimators but has a methodology for simulating collimator blur using a circular ordinate splitting (COS) technique.[2] The COS technique allows the user to specify a collimator acceptance angle. The flux is then solved for along directions within the acceptance angle around the projection direction. The flux is then averaged over this angle to "blur" the projection image.

**III. Results and Analysis**

**1. Projection Images**

First energy group projection images from the TITAN and MCNP5 codes are found in **Figs. 4-6**. Each image is normalized to the highest pixel value.

Figures 4-6 show that the TITAN projection images have good visual agreement with the MCNP5 predictions. In Fig. 4, the TITAN image appears less blurred than the MCNP5 image. It can also be seen that the images become sharper as the acceptance angle decreases, i.e., the case number increases.



(a)                              (b)
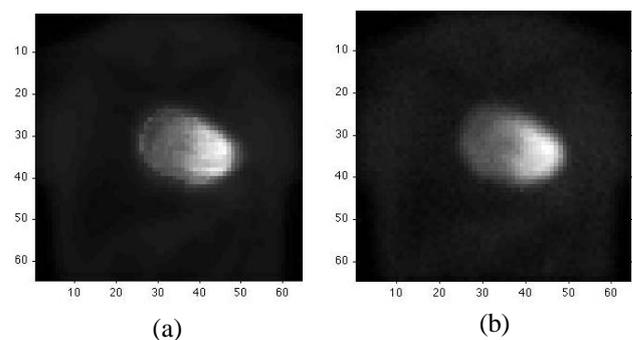
**Fig. 4** Collimator Case 1 projection image using (a) TITAN and (b) MCNP5
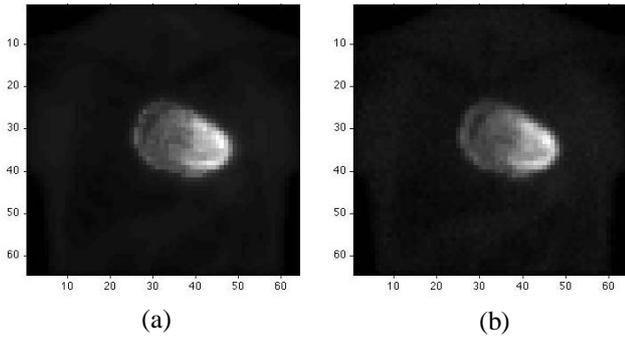
**Fig. 5**  Collimator Case 2 projection image using (a) TITAN and (b) MCNP5
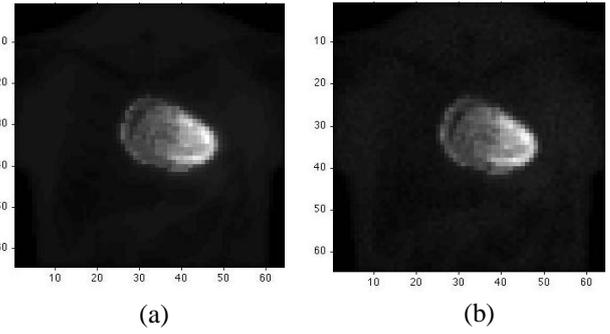


**Fig. 6**  Collimator Case 3 projection image using (a) TITAN and (b) MCNP5

The maximum relative difference between the high count myocardium pixels in the MCNP5 and TITAN projection images is given in **Table 1** for each collimator case.

The results in Table 1 indicate that the differences in the two codes are reduced as smaller acceptance angles are modeled. Recall that the collimator representation is the main difference between the models in the two codes.

To further compare the two codes, profiles through the normalized projection images were plotted for each collimator case along with the relative difference. Profile plots correspond to the rows and columns labeled in the projection images of Figs. 4, 5, and 6. **Figures 7**(a) and (b) are for Case 1, **Figs. 8**(a) and (b) are for Case 2, and **Figs. 9**(a) and (b) are for Case 3.

**Table 1**  Maximum difference of TITAN results relative to MCNP5 results for each collimator case

| Case Number | Acceptance Angle (degrees) | Maximum Relative Difference (%) |
|---|---|---|
| 1 | 2.97 | 21.3 |
| 2 | 1.42 | 11.9 |
| 3 | 0.98 | 8.3 |

The profile plots demonstrate how well the TITAN code follows the shape of the MCNP5 results. Each profile peaks in the heart with tails on either side. In all six profile plots, the tails match well, but the peaks are what is of interest. In the column profile in Fig. 7(a), there are some discrepancies in



(a)



(b)
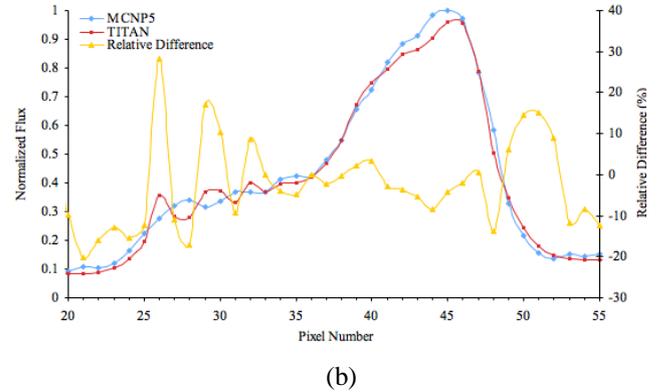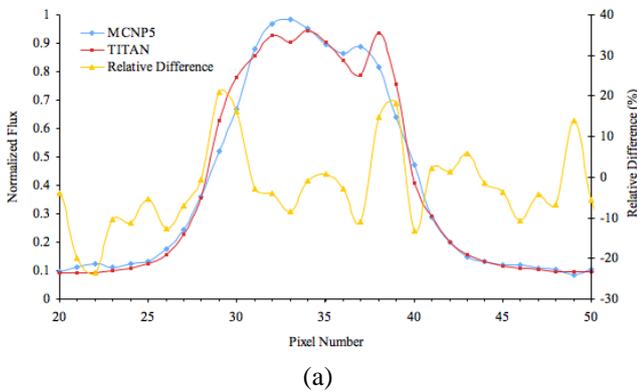
**Fig. 7**  Case 1 profiles through MCNP5 and TITAN projection images and relative difference for (a) column 44 and (a) row 33



(a)



(b)
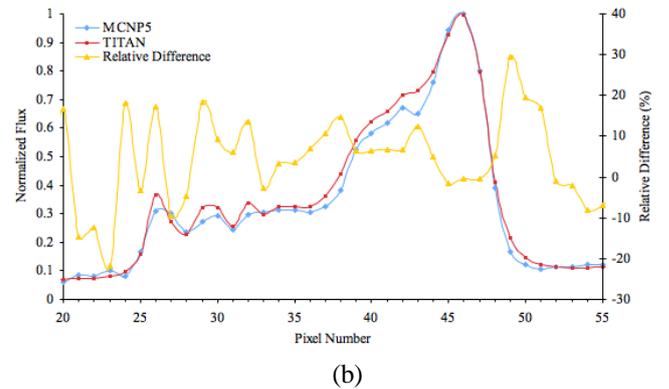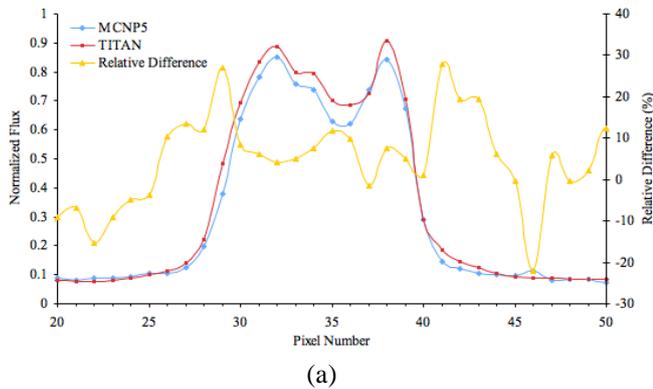
**Fig. 8**  Case 2 profiles through MCNP5 and TITAN projection images and relative difference for (a) column 44 and (b) row 33

(a)



(b)
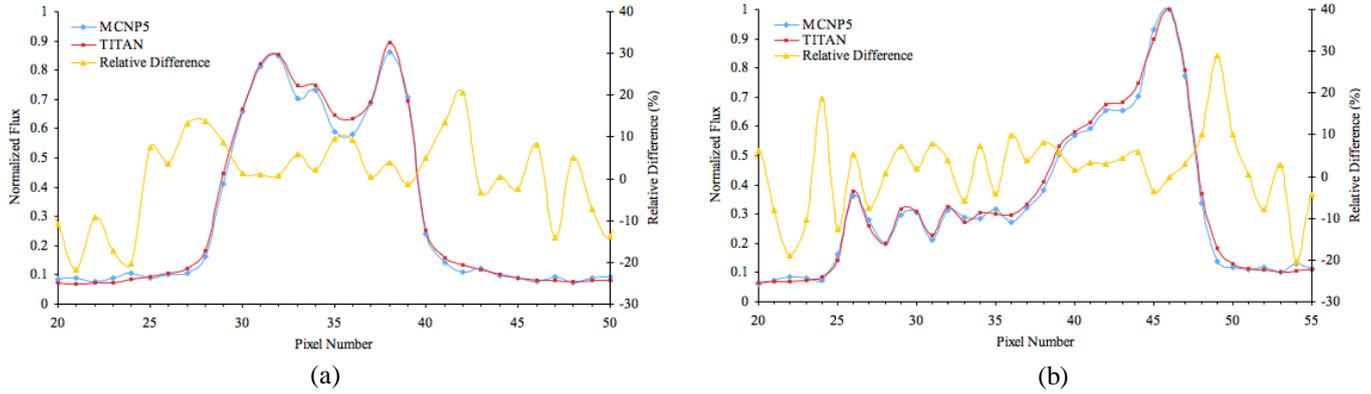
**Fig. 9** Case 3 profiles through MCNP5 and TITAN projection images and relative difference for (a) column 44 and (b) row 33
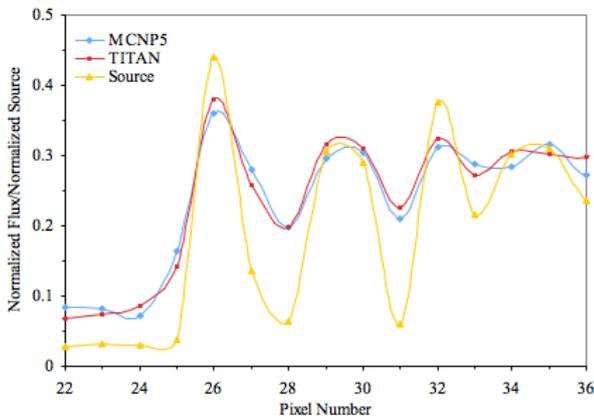


**Fig. 10** Case 3 profile plot through row 33 of MCNP5 and TITAN projection images and source projection

**Table 2** Computation time comparison with MCNP5 and TITAN on 16 processors

| Case Number | Acceptance Angle (degrees) | Code | | Speedup Factor (MCNP5/ TITAN) |
|---|---|---|---|---|
| | | MCNP5 (min)* | TITAN (min)† | |
| 1 | 2.97 | 313.8 | 0.82 | 382 |
| 2 | 1.42 | 1071.8 | 0.82 | 1304 |
| 3 | 0.98 | 2289.7 | 0.82 | 2787 |

*Time to achieve 1-σ uncertainty of ≤3.0% in the heart
†180 projection angles

the peak shape that support the significant relative difference in Table 1. These are attributed to the TITAN COS technique not creating sufficient collimator blur. Figure 8(a) has a smaller collimator acceptance angle so more detail is visible and two distinct peaks are now visible in both codes; however, the magnitude is moderately higher in TITAN. With the smallest collimator acceptance angle, Fig. 9(a), the profiles match more closely, although some magnitude differences are still seen.

The row 33 TITAN profile in Fig. 7(b) shows oscillations in pixels 25-33; however the smaller collimator acceptance angles in Figs. 8(b) and 9(b) show that these oscillations match the MCNP5 profile. The source distribution was examined to verify that these oscillations are real, as shown in **Fig. 10**.

In Fig. 10, the source distribution has been collapsed towards the detector and normalized to create a 2D source projection from which row 33 is plotted. The flux and source values cannot be directly compared, but the profile shapes can. The locations of the peaks and valleys of the oscillating source profile agree with the peaks and valleys seen in the MCNP5 and TITAN profiles. This result further verifies that the oscillations are physical.

From the discussed results, it is clear that the TITAN projection images become closer to the MCNP5 predictions

as the collimator acceptance angle becomes smaller. It seems that the differences in how the collimator is represented in the two codes becomes a minor effect with decreasing acceptance angle. To better match larger acceptance angles, the TITAN methodology is being improved.

**2. Parallel Timing Analysis**

All simulations were performed on a PC-cluster using 2 nodes with 8 processors per node. Each processor is a 2 GHz Intel Xeon Quad Core processor with 4 GB of DDR2 SDRAM. The parallel computation times for the TITAN and MCNP5 codes are compared in **Table 2** for 16 processors. The MCNP5 times are for a 1-σ statistical uncertainty of ≤3.0% in the heart and the TITAN times are for simulating 180 projection angles. The MCNP5 code simulated 3.6 billion, 13.6 billion, and 26.7 billion histories for cases 1, 2, and 3, respectively, to achieve the aforementioned uncertainty. Even though a detector is being simulated in only one position in MCNP5, it would be possible to place detectors at other projection angles without any significant change in the calculation time. This is why 180 projection angles are being used in the TITAN simulation for comparison with MCNP5.

The results in Table 2 demonstrate an advantage of the TITAN code over MCNP5. The TITAN code first solves the $S_N$ method within the phantom, independent of the collimator acceptance angle. Photons are then transported to the detector surface along the projection angle and the acceptance angle. In this methodology a smaller collimator acceptance angle will not affect computation time. In the MCNP5 code a

**Table 3** TITAN Case 1 total computation time comparison for serial and parallel projection image computation

| Number of Processors | Number of projection images | Total time with serial projection images (sec) | Total time with parallel projection images (sec) |
|---|---|---|---|
| 1 | 1 | 201.3 | 199.8 |
| 2 | 1 | 104.4 | 104.0 |
| 2 | 4 | 109.3 | 105.6 |
| 2 | 90 | 258.9 | 181.5 |
| 2 | 180 | 419.2 | 260.9 |
| 4 | 1 | 56.7 | 56.6 |
| 4 | 4 | 61.8 | 56.6 |
| 4 | 90 | 211.3 | 95.5 |
| 4 | 180 | 368.5 | 134.0 |
| 8 | 1 | 37.8 | 37.8 |
| 8 | 4 | 42.5 | 37.7 |
| 8 | 90 | 191.9 | 57.9 |
| 8 | 180 | 348.5 | 77.7 |
| 16 | 1 | 28.5 | 29.0 |
| 16 | 4 | 34.1 | 28.3 |
| 16 | 90 | 183.1 | 38.1 |
| 16 | 180 | 340.9 | 49.3 |

**Table 4** TITAN Case 1 projection image only computation time comparison for serial and parallel

| Number of Processors | Number of projection images | Serial projection image calculation time (sec) | Parallel projection image calculation time (sec) |
|---|---|---|---|
| 1 | 1 | 2.2 | 2.3 |
| 2 | 1 | 2.2 | 2.3 |
| 2 | 4 | 7.5 | 4.1 |
| 2 | 90 | 156.6 | 80.1 |
| 2 | 180 | 316.8 | 158.4 |
| 4 | 1 | 2.2 | 2.3 |
| 4 | 4 | 7.4 | 2.4 |
| 4 | 90 | 156.9 | 41.1 |
| 4 | 180 | 314.0 | 79.9 |
| 8 | 1 | 2.3 | 2.5 |
| 8 | 4 | 7.5 | 2.5 |
| 8 | 90 | 156.9 | 22.4 |
| 8 | 180 | 313.6 | 42.4 |
| 16 | 1 | 2.3 | 2.4 |
| 16 | 4 | 7.5 | 2.5 |
| 16 | 90 | 156.5 | 11.7 |
| 16 | 180 | 315.8 | 22.7 |

significant increase in computation time is seen for smaller acceptance angles. For all cases, the TITAN code is significantly faster than MCNP5 with speedup factors ranging from 382 to 2,787.

Until recently, the TITAN code calculated projection images on a single processor after calculating the flux in the phantom on a parallel machine. The projection image calculation has since been parallelized and the improvement in total computation time for varying numbers of projection images is examined in **Table 3** for Case 1.

In Table 3, the increasing of projection angles from 1 to 180 significantly increases the computation time if the projection calculation is not parallel. For 8 and 16 processors especially, it is clear that the computation of a large number of projection images will dominate the total time. For 180 projection images on 16 processors the total time is reduced by 85.5% if the projection image calculation is parallelized. Also, looking only at the single projection cases, it can be seen that doubling the number of processors initially nearly cuts the computation time in half. However, as the number of processors reaches 8 and 16, the speedup is reduced due to the increased communication time among processors.

In **Table 4**, the time to complete only the projection image calculation is displayed for different numbers of projection images and processors. The projection image computation time is the total time minus the time to calculate the flux distribution in the phantom region using the discrete ordinates method. Comparing the serial and parallel projection calculations, the times scale very closely with the number of processors for the 2- and 4-processor cases. However, as was seen in Table 3, there is some loss of speedup due to increased communication in the cases with 8 and 16 processors.

The parallel speedup, parallel efficiency and parallelizable fraction of the MCNP5 and TITAN codes were also examined for Case 1. The parallel speedup ($S_P$) is defined as the ratio of the wall-clock time for a single processor ($T_s$) to the wall-clock time in parallel. The parallel efficiency is the parallel speedup divided by the number of processors ($P$). Amdahl's Law[9] can be used to find the parallel fraction of the codes (Eq. (1)).

$$S_P = \frac{1}{(1-f_P) + f_P/P + T_c/T_s} \qquad (1)$$

In Eq. (1), $f_P$ is the parallelizable fraction of the code, $T_c$ is the parallel communication time, and the rest of the variables are as defined previously. If $T_c$ is assumed to be very small compared to $T_s$, Eq. (1) can be rewritten as Eq. (2).

$$f_P = \frac{P(1-S_P)}{S_P(1-P)} \qquad (2)$$

Using Eq. (2), the parallelizable fraction was estimated for the MCNP5 and TITAN codes. **Tables 5** and **6** display the wall-clock time, parallel speedup, parallel efficiency and parallelizable fraction found for varying numbers of processors in both codes.

It should be noted that the MCNP5 parallel algorithm designates a master processor and then distributes the number of particles to be simulated over the remaining processors. Because of this treatment of parallel problems, the efficiency in Table 5 does not start high and decline as one would expect. As the number of processors increases, the effect of having a master processor is reduced. The parallelizable fraction estimated is very high (~98%), as expected for a Monte Carlo code.

**Table 5**  MCNP5 parallel computation time analysis for 1 billion particles in Case 1 ($T_c$=0)

| Number of Processors | Wall Clock Time (min) | Parallel Speedup | Parallel Efficiency | Parallelizable Fraction |
|---|---|---|---|---|
| 1 | 1132.9 | - | - | - |
| 4 | 380.3 | 2.98 | 0.74 | 0.89 |
| 8 | 170.8 | 6.63 | 0.83 | 0.97 |
| 12 | 111.6 | 10.15 | 0.85 | 0.98 |
| 16 | 87.3 | 12.98 | 0.81 | 0.98 |

**Table 6**  TITAN parallel computation time analysis for Case 1 with 1 projection angle ($T_c$=0)

| Number of Processors | Wall Clock Time (min) | Parallel Speedup | Parallel Efficiency | Parallelizable Fraction |
|---|---|---|---|---|
| 1 | 3.33 | - | - | - |
| 4 | 0.94 | 3.53 | 0.88 | 0.96 |
| 8 | 0.63 | 5.29 | 0.66 | 0.93 |
| 12 | 0.51 | 6.49 | 0.54 | 0.92 |
| 16 | 0.48 | 6.89 | 0.43 | 0.91 |

In Table 6, TITAN simulations show that parallel efficiency is high for 4 processors and then falls as the number of processors increases due to the increased communication time. The estimated parallelizable fraction is lower than in the MCNP5 code because TITAN is a deterministic code; however it is still a high fraction (~90%).

The parallel speedup seen for increasing number of processors is plotted in **Fig. 11** for both MCNP5 and TITAN.

In Fig. 11, the speedup for TITAN quickly levels off as the number of processors increases; however, the MCNP5 data still shows roughly linear behavior. This result is expected since the MCNP5 code does not need as much communication between processors as the deterministic TITAN code. Monte Carlo codes use independent particle histories so if the number of histories is given the processors do not need to communicate until the end of the simulation. The parallel TITAN code uses angular decomposition,[10] which requires the processors to communicate after each iteration to recalculate the scattering source.

## IV. Conclusion

The TITAN methodology for collimator simulation agrees well with the MCNP5 results both visually and quantitatively. The comparison improves as the collimator acceptance angle becomes smaller. More studies are needed to investigate ways to better represent collimator blur in the TITAN code for larger acceptance angles.

Timing comparisons showed that the TITAN code is greater than two orders of magnitude faster than MCNP5 for all three collimator cases simulated. The newer version of TITAN with parallelized projection image calculation was
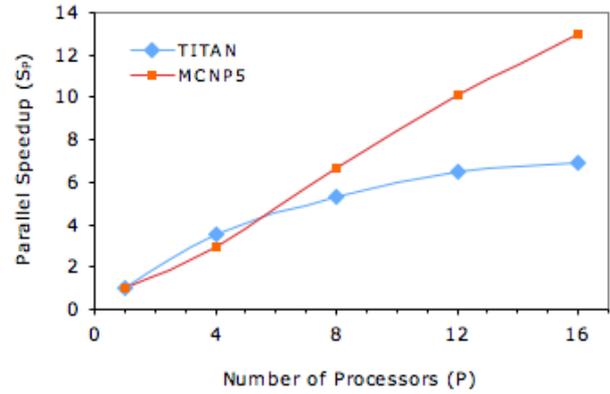


**Fig. 11**  Parallel speedup for TITAN and MCNP5 as a function of number of processors

shown to provide an 85.5% reduction in computation time for 180 projection images on 16 processors. A very high parallelizable fraction, in the range of 90-98%, has been observed for both codes.

## References

1) C. Yi, A. Haghighat, "A Three-Dimensional Block-Oriented Hybrid Discrete Ordinates and Characteristics Method," *Nucl. Sci. Eng.*, **164**, 221-247 (2010).
2) C. Yi, A. Haghighat, "Hybrid $S_N$ and Ray-Tracing with Fictitious Quadrature for Simulation of SPECT," *Proc. M&C 2009*, Saratoga Springs, NY, May 3-7, 2009 (2009), [CD-ROM].
3) X-5 Monte Carlo Team, *MCNP-A General Monte Carlo N-Particle Transport Code, Version 5*, LA-UR-03-1987, Los Alamos National Laboratory (LANL) (2003).
4) K. Royston, A. Haghighat, C. Yi, "Verification of a Hybrid Adjoint Methodology in TITAN for Single Photon Emission Computed Tomography," *Proc. PHYSOR 2010*, Pittsburgh, PA, May 9-14, 2010 (2010), [CD-ROM].
5) W. P. Segars, *Development and application of the new dynamic NURBS-based cardiac-torso (NCAT) phantom*, PhD Thesis, University of North Carolina (2001).
6) L. J. Lorence, J. E. Morel, G. D. Valdez, *User's Guide to CEPXS/ONELD: A One-Dimensional Coupled Electron-Photon Discrete Ordinates Code Package Version 1.0*, SAND89-1161, Sandia National Laboratory (1989).
7) E. E. Lewis, W. F. Miller, *Computational Methods of Neutron Transport*, John Wiley & Sons, New York (1984).
8) C. Yi, *Hybrid Discrete Ordinates and Characteristics Method for Solving the Linear Boltzmann Equation*, PhD Thesis, University of Florida (2007).
9) G. Amdahl, "The Validity of the Single Processor Approach to Achieving Large Scale Computing Capabilities," *Proc. AFIPS Conf.*, Atlantic City, NJ, April 18-20, 1967, vol. 30, 483-485 (1967).
10) C. Yi, *TITAN: A 3-D Deterministic Radiation Transport Code, TITAN User Manual Version 1.05*, University of Florida (2009).