STYX 3D IMPLEMENTATION ON SMP ARCHITECTURES

Christian Aussourd
Commissariat à l'Energie Atomique
BP 12
91680 Bruyères-Le-Châtel
FRANCE

christian.aussourd@bruyeres.cea.fr

ABSTRACT

In this paper we describe a new 3D version of the AMR S_N Code STYX. Its development was hastened to comply with the increasingly pressing demand for tridimensional predictive tools. Built on the older 2D version [1,2] this new release inherits its whole feature set plus an embryonic parallel bonus.

1. INTRODUCTION

Real world transport problems are naturally 3D. However, their S_N simulations not only take hours to be computed but also require very large amounts of central memory. Until the advent of parallel phase space decompositions, physicists used to get rid of such limitations by using 2D and even 1D representations under specific assumptions.
With a close rendering of local physics using adaptive spatial meshing, STYX is well adapted for mid-size problems where mesh collapse and multithreading advantageously replace spatial decomposition.
This paper will proceed as follows. First, we compare two possible solutions and describe the associated methods. Sequential and parallel implementations of the strategy we opted for are also detailed for the current and subsequent versions.
Second, some numerical results, featuring source and eigenvalue benchmarks, are presented that show STYX ability to save Mbytes of memory while preserving accuracy.
Finally, we conclude by discussing current status of the software and suggest future fields of research.
We deliberately stripped this paper of any equations in order to focus on architecture. Theory has already been detailed in previous articles [1,2]. A thorough discussion of basic S_N equations and the related physics may also be found in literature [3,4].

2. SEEKING THE BEST STRATEGY

2.1 TOP-TWO SOLUTIONS

Among many viable strategies, we retained two candidates.
Each approach is presented for the particular case of parallelism; the same strategy applies in sequential mode.
Their respective advantages and disadvantages are listed below and justify our final choice.

2.1.1 Z-SLICING APPROACH

In order to have, as quickly as possible, a preliminary version of Styx 3D, the following strategy was envisioned.

a) Methodology

Presented on Figure 1 and in the next few lines, are the main characteristics of this approach :

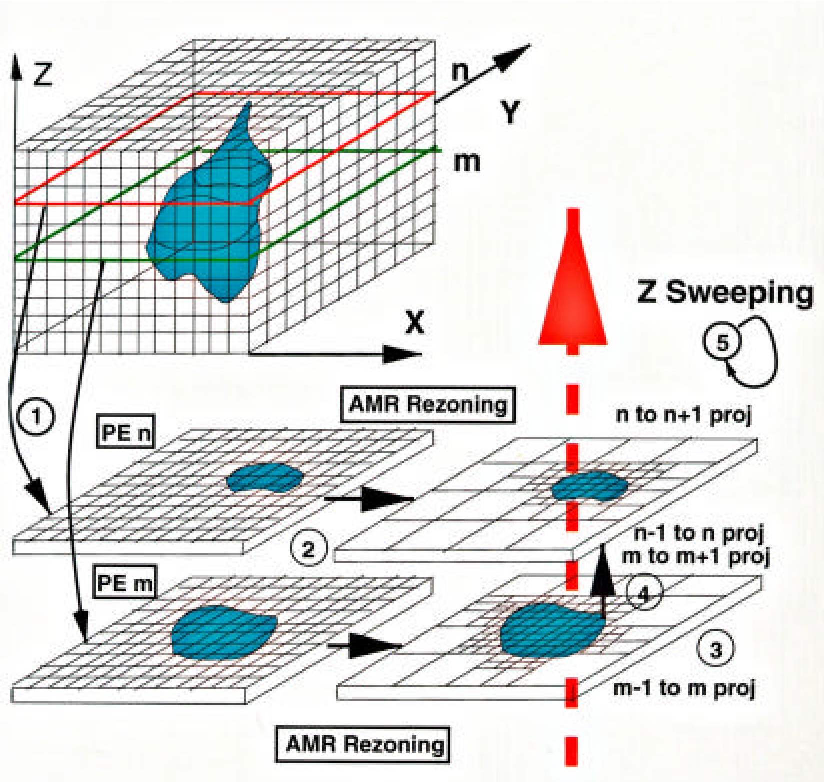

Figure 1. Z-Slicing Strategy

1) Slice the Cartesian grid along Z axis into nz layers of nx.ny cells,
2) Load balancing in mind, distribute layers on a cluster of processors (one or more each),
3) Rezone each sub-domain into a 2D dz thick AMR grid ,
4) Apply a simplified 3D transport processing to each layer ,
5) Remap layer $(m-1)$ outgoing fluxes to initialize layer m incoming fluxes,
6) Repeat 4 & 5 for all directions,
7) Iterate until global fluxes converge.

b) Pros and Cons

➚ Because it recycles most of the 2D mature components this method provides the smallest development and tuning times and is attractive in that respect.

➘ However, it seems difficult to ensure a good load balancing especially for head and tail slices.

➘ Two reasons may induce a slower convergence process :

- Incoming fluxes arise from the previous iteration,
- Z perturbations are advected one layer per iteration.

2.1.2 FULL 3D APPROACH

Here, we consider the 3D problem in its entire generality.

a) Methodology

This 3D architecture directly derive from the older 2D version except a complete recasting of all space dependent routines is required :

- rezoning,
- neigborhood management,
- cell sorting,
- remapping of neutron populations for conservation from time t to time $t+dt$,
- . . .

.

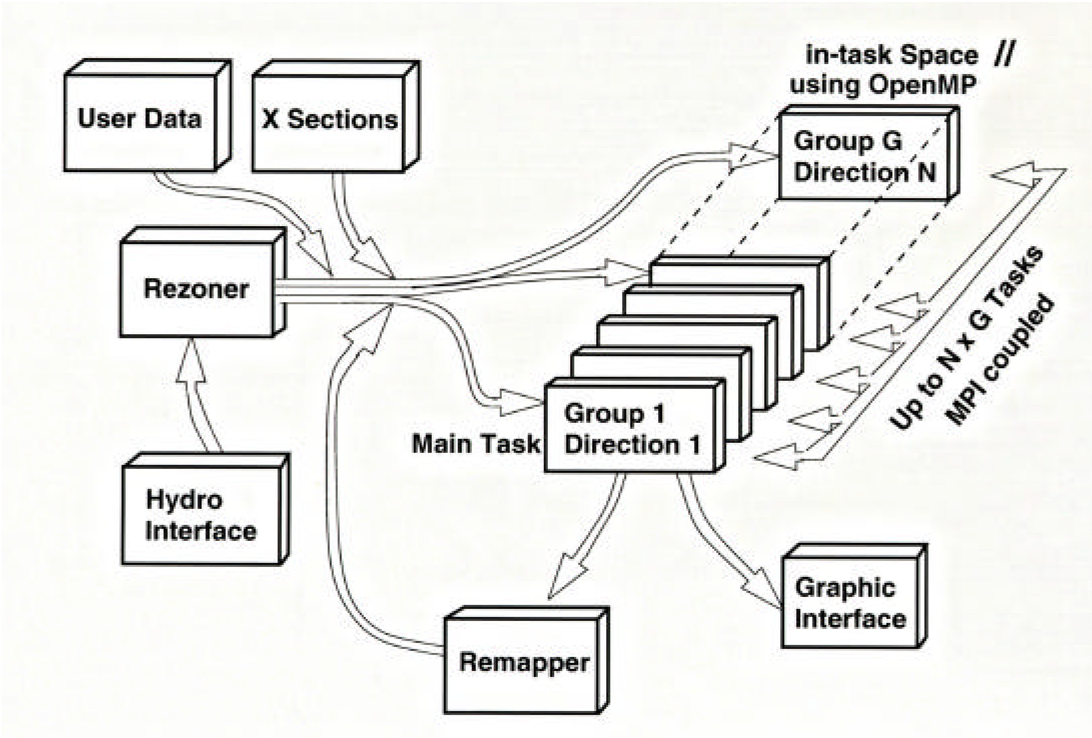

Figure 2 . Full 3D Strategy

b) Pros and Cons

➚ Performances and accuracy are expected to be far better than those accessible to the previous solution.

➚ Reuse of 2D architecture,

➘ Because AMR cells are unevenly distributed over space, it is necessary (at least for the moment) to load the entire geometry on each processor,

➘ A long and cumbersome process of development, debugging and tuning is necessary.

2.2 DECISION MADE

The slicing strategy was always considered as a disposable solution or in other words the first step toward a 3D neutron transport code.

A disadvantage of the second strategy is the need to load each PE with the entire AMR grid. However, with up to 90 % of the original cells discarded by the rezoning process, this is not really a major problem.

The emergency led us to choose the full 3D version and so to skip the transient first solution.

3. IMPLEMENTATION

This paper essentially differs from previous work in that :

- upgrade from 2D axisymmetric AMR to full 3D AMR,
- embryonic parallel version.

For all unchanged algorithms, we refer to other papers [1,2].

3.1 SEQUENTIAL IMPLEMENTATION

We now focus on solving the transport problem in sequential mode (i.e. on a single processor).
The main differences between the older 2D version and the current 3D one are essentially a complete recasting of :

- the rezoner (translation from Cartesian to AMR),
- the remapper (reuse time t converged flux map to initialize t+dt in order to gain iterations),
- all space and neighborhood related routines in the transport processing unit,
- all graphic output routines.

It is worth noting that many more calculations are needed to carry out accurate 3D results :

- nz times more zones,
- twice as many angles,
- $\frac{2(n-1)}{n}$ times more real spherical harmonics (for a Pn anisotropic data set)

than for a 2D axisymmetric implementation.

The following flowchart sketches the architecture of sequential STYX 3D.

BEGIN STYX

- Read user data,
- Rezone the original Cartesian mesh in AMR,
- Sort cells and pack them into independent batches for all 8 octants,
- Define neighborhood tables,
- Compute angular (S_N quadrature and spherical harmonics) and spatial constants,
- Choose problem type : eigenvalue or fixed source

<u>Begin eigenvalue problems</u>

- Begin eingenvalue loop
    - Choose new guess
    - Begin outer iteration loop
        - Begin energy group loop
            - Begin moments of source loop
                - Down-scattering contribution (uses current iteration),
                - Up-scattering contribution (uses previous iteration),
                - self-scattering contribution (uses previous iteration),
            - End moments of source loop
            - Begin **x** loop
                - Begin **m** loop
                    - Compute angular sources,
                    - S_N calculation of outgoing angular fluxes,
                    - Update moments of flux
                - End **m** loop
            - End **x** loop
        - End energy group loop
        - Compute in group and global flux norms
        - Test outer iteration loop convergence
    - End outer iteration loop
    - Test eigenvalue convergence
- End eingenvalue loop

<u>End eigenvalue problems</u>

-OR-

<u>Fixed source problems</u>

- Begin outer iteration loop
    - Begin energy group loop
        - Begin moments of source loop (part I)
            - Down-scattering contribution (uses current iteration),
            - Up-scattering contribution (uses previous iteration),
        - End moments of source loop (part I)
        - Begin inner iteration loop
            - Begin moments of source loop (part II)
                - self-scattering contribution (uses previous iteration),
            - End moments of source loop (part II)
            - Begin **x** loop
                - Begin **m** loop
                    - Compute angular sources,
                    - S_N calculation of outgoing angular fluxes,
                    - Update moments of flux
                - End **m** loop
            - End **x** loop
            - Compute in group flux norm
            - Test inner iteration loop convergence
        - End inner iteration loop
    - End energy group loop
    - Compute global flux norm
  - Test outer iteration loop convergence
- End outer iteration loop

<u>End source problems</u>

- Graphical output

END STYX

3.2 PARALLEL IMPLEMENTATION

Since the parallel application is SIMD the aforementioned feature set applies in the same way here except when tasks need communicating.

3.2.1 PARALLELIZATION IN THE CURRENT VERSION

The current version of STYX 3D was tested on an IBM SP2 cluster constituted of NightHawk Power PC 630 processors and on an SGI-CRAY T3E.

It is fully parallelized in energy, using MPI, and partially in space, using OpenMP.

a) Methodology

➔ Execution of STYX begins with creation of N_G MPI tasks, one per energy group.

All tasks will have the same job to fulfill except task #1, referred to as main task, having extra duties such as :

- managing other processing units,
- gathering information from all slave tasks,
- . . .

➔ Tasks are now waiting for the AMR mesh to be ready (MPI_Barrier).

➔ Main task has the responsibility of launching the rezoning process that consists in reading hydrocode output and collapsing the Cartesian mesh into an AMR grid,

➔ When ready, all tasks resume working. They first read the newly created AMR mesh and then continue using the aforementioned flowchart until they arrive at the energy loop entrance.

➔ At this time, they are beginning to work on different data sets : task # n is treating Energy Group # n. Current iteration moments of source are evaluated using previous iteration moments of flux.

➔ After completion, task # n sends newly computed moments of flux to all N_G-1 other tasks (MPI_Allgather) and , in return, receives N_G-1 sets of moments,

➔ Finally, the main task is able to test convergence.

b) Performances

It is a little bit too early to speak about performances for, at least, two reasons :

- ♦ First, the newborn code STYX 3D is not enough ready for these evaluations,

- ♦ Second, exact measurements of parallel performances need a dedicated computer. Because a few sessions of this type are now available, they need to be carefully prepared and so out of reach for this preliminary version that began to work in late December 1999.

3.2.2 PARALLELIZATION IN FUTURE VERSIONS

a) Parallelization over angle

It is quite easy to extend the energy group method to angle.
Reflective boundary conditions might be treated using previous iteration values of edge angular fluxes and restricted MPI communicators in order to save time during message-passing.
A one-speed/one-direction per processor strategy is envisioned.

b) Parallelization over space

The first step toward this ambitious and awkward task is to parallelize the Cocytus Rezoner.
The transport parallelization over such AMR grids is an open field of research and work is underway to find the best strategy.

4. NUMERICAL RESULTS

In order to illustrate the capabilities of this new software three benchmarks are presented below; each of them has a special purpose :

- The first one confirms that spatial symmetry is ensured,
- The second one demonstrates accuracy,
- The third one shows how Styx renders complex geometries such as pipes.

All benchmarks were modeled using the Multipurpose High Strain Rate Code Ouranos [5] and treated without reflective boundaries since they are not available in this current preliminary version. Except for the first test, that is mono-energetic, all benchmarks are computed using a 16-energy group P4 anisotropic cross-section data set.

4.1 BENCHMARK #1 : RAY EFFECT (SOURCE PROBLEM)

Ray effect is one of the few deficiencies of S_N methods, it is also a useful tool to appreciate good rendering of symmetries and detect at first glance anomalies in their treatment.
The first benchmark is a classical one-group benchmark specially designed to display this defect.

It consists in a radiating sphere within a cube. The $\Phi = 1$ sphere, referred to as point source, has a total cross-section $\sigma_t = 0.1$ and an internal source S = 4. The non-absorbent surrounding cubic medium has a null total cross-section, an ambient source S = 0.1 and a side length of 5 (arbitrary units).
The 125000 cell Cartesian mesh (50x50x50) is collapsed into a 16444 cell level 1 AMR grid (i.e. 86 % less cells; 15508 level 0 cells and 936 level 1 cells concentrated in the sphere).

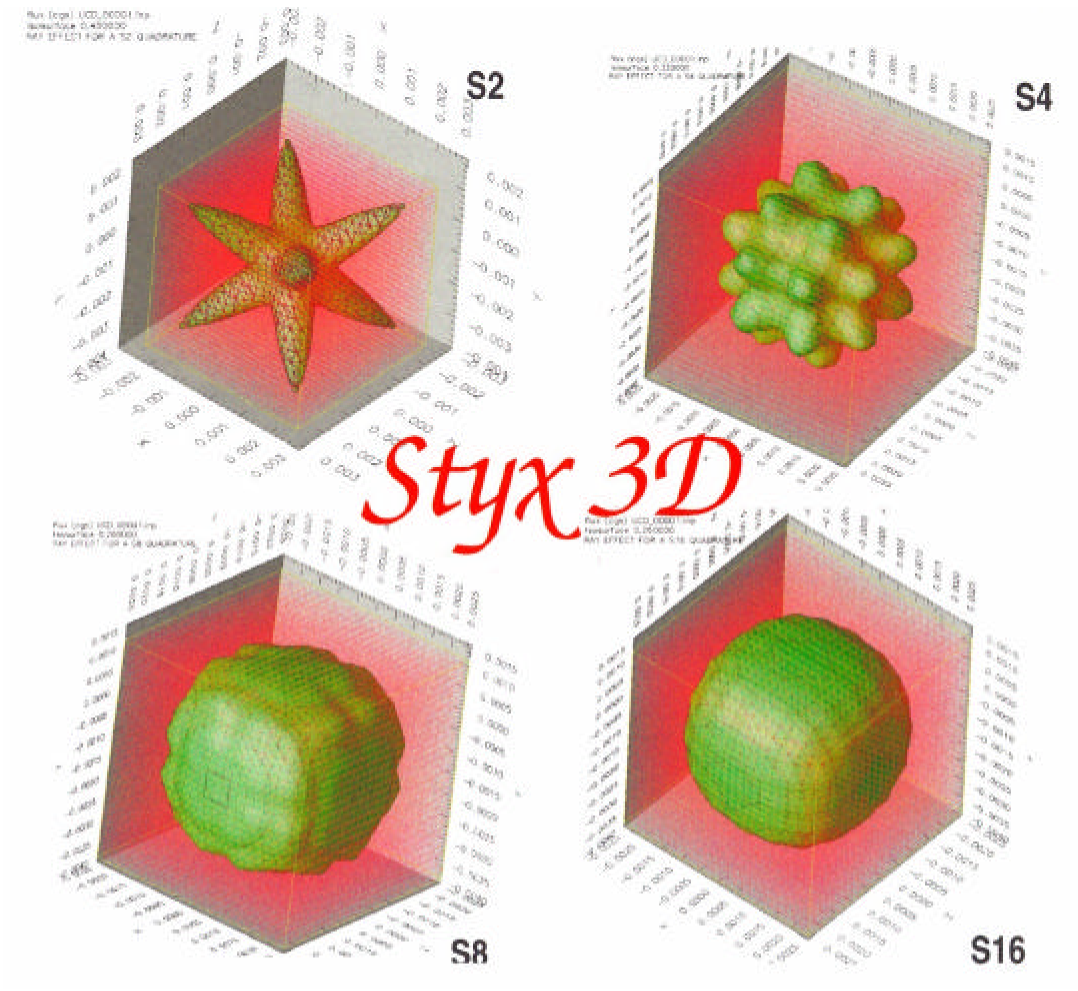

Figure 3. Ray Effect Flux Contours

The flux contours presented in Figure 3 display very good behavior of the solution in all directions for different quadrature sets. Variations (not reported here) made around the discretization confirm the fact that the finer the mesh is the more acute the ray effect is.

4.2 BENCHMARK #2 : CRITICALITY TEST KENO K14 (EIGENVALUE PROBLEM)

This problem [6] is made up of two ^{235}U objects placed in a vacuum : a cylindrical core encircled by a tangent ring (Figure 4). It is designated KENO k.14 after the well-known multi-group Monte Carlo code.

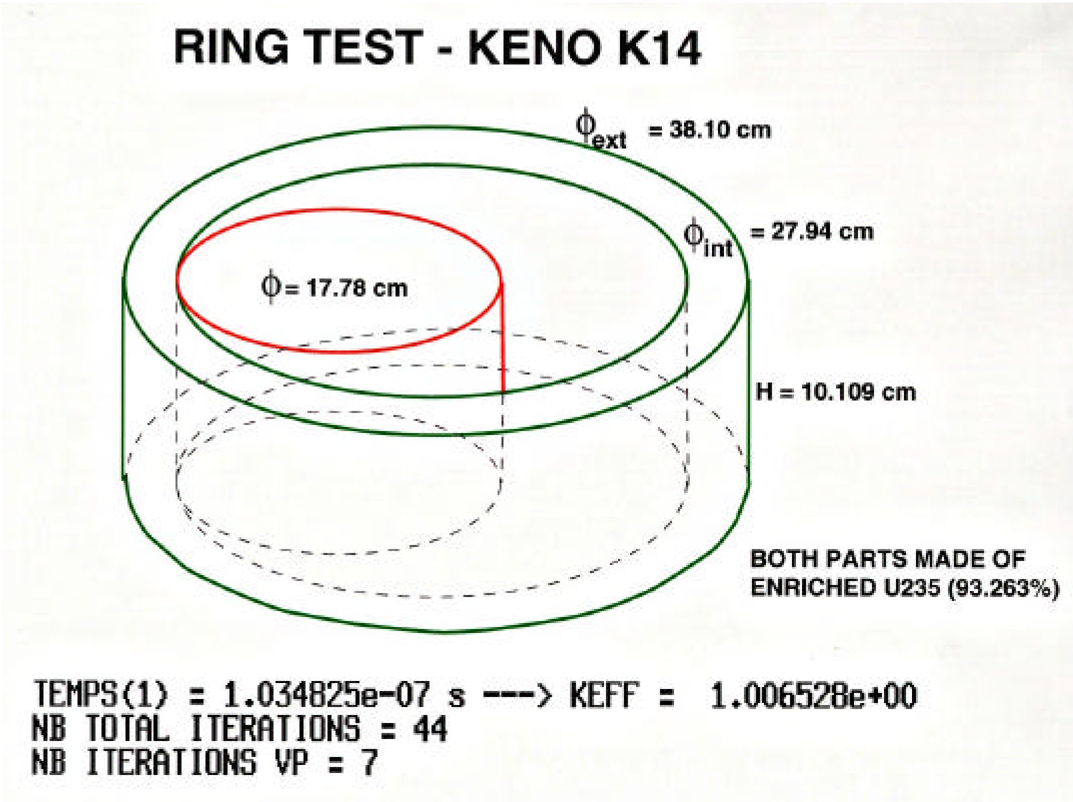

Figure 4. Ring Test-Case Outline

Both parts have an enrichment of 93.263% ^{235}U and the resulting assembly is designed to be just critical.

The original 32000 cell Cartesian mesh (40x40x20cm box with 1cm cubic cells) is collapsed into a 16460 cell level 2 AMR grid (168 level 0, 708 level 1 and 15584 level 2 cells).

The multiplication factor obtained is in good agreement (less than 100 pcm) with other codes (MCNP and an internal Monte Carlo code).

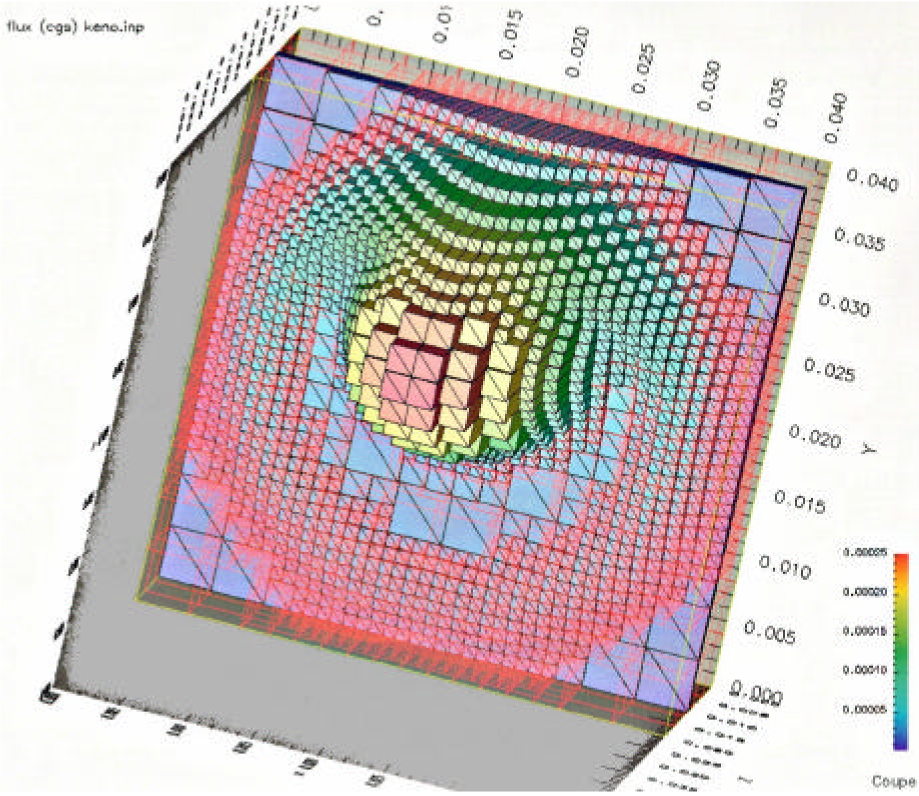

Figure 5. Ring Test-Case Flux Map

Looking at Figure 5, the reader can appreciate the graded cell size that closely renders active area shapes.

4.3 BENCHMARK #3 : WORM TEST (EIGENVALUE PROBLEM)

This benchmark (Figure 6) is composed of a sub-critical sixfold ^{239}Pu square rod radiating in a vacuum. It is specially designed to show how Styx 3D is able to model complex assemblies.

.

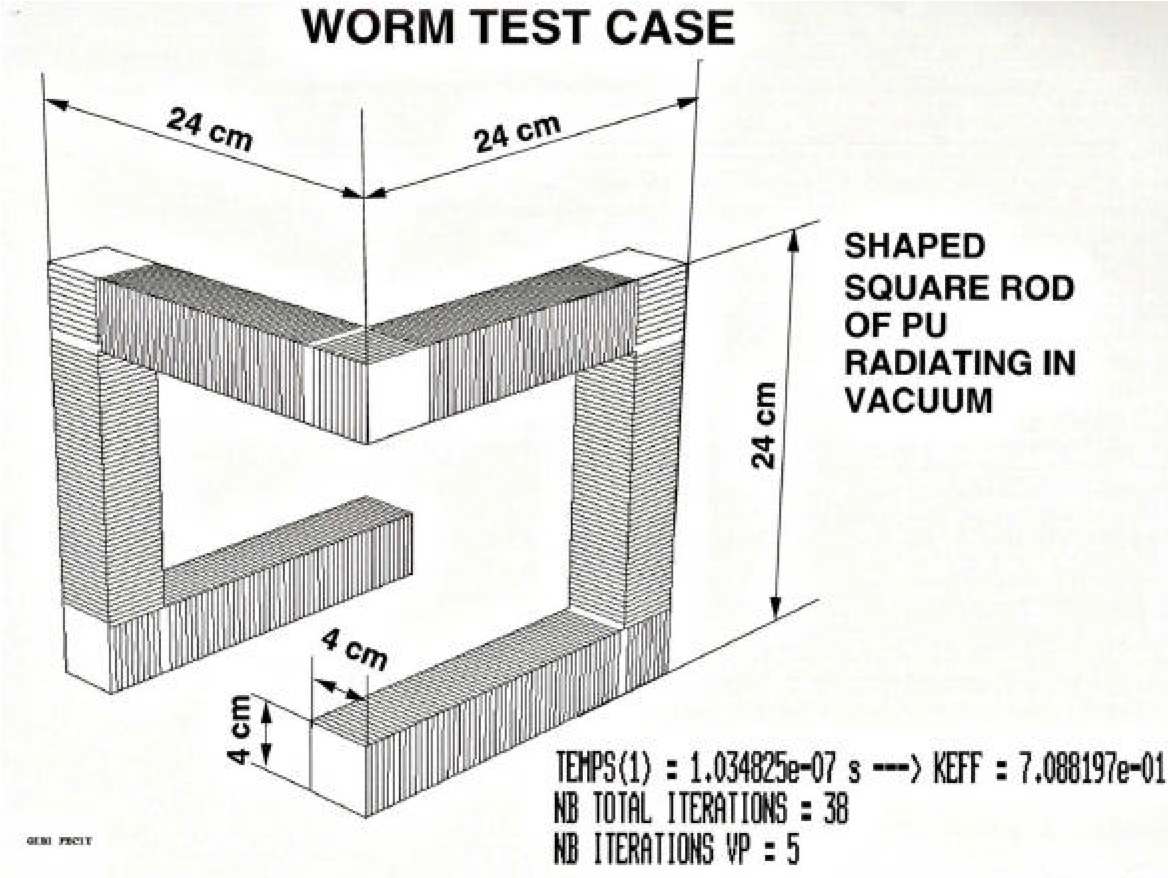

Figure 6. Worm Test Case Outline

Here 80 % of the original Cartesians cells are discarded when using a level 3 AMR rezoning (10 cells on level 0, 285 on level 1, 506 on level 2 and 5360 cells on level 3).
Figure 7 shows the flux contour around the rod and the underlying mesh.

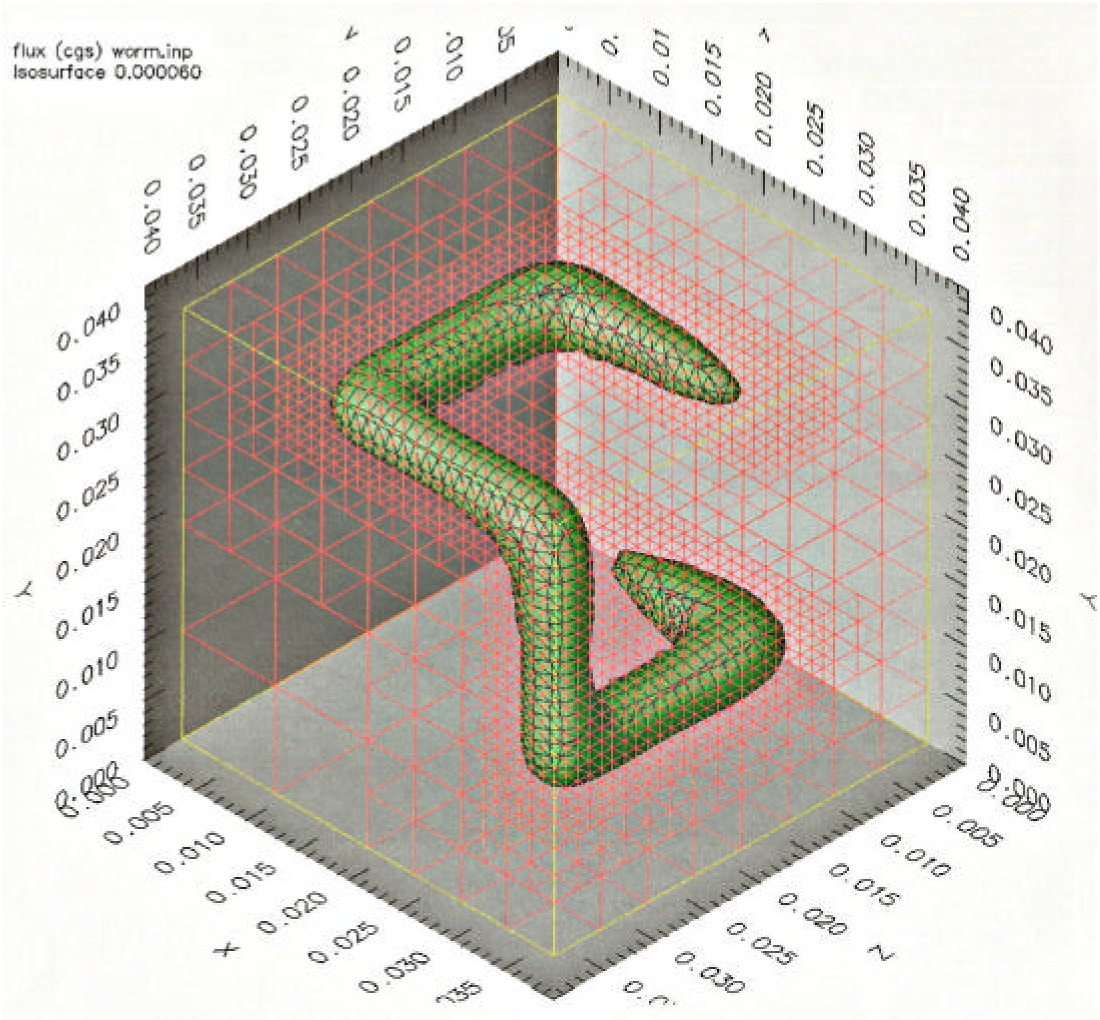

Figure 7. Worm Test Case Flux Map

CONCLUSIONS AND FUTURE WORK

Developed in under three months, this brand new version of STYX works fairly well for solving neutral particle transport on 3D AMR grids using the same modified S_N scheme derived for the 2D case. The parallelized version currently uses a combination of message passing (MPI) for energy (very soon for angles) and threading (OpenMP) for space, to draw the best out of SMP architectures.

The benchmarks presented confirm that this method, not only works in 3D, but also gives low cost, useful and good results for a wide range of radiation transport problems.

This method has many desirable qualities. It is robust, accurate and, thanks to its adaptive meshing, remains non-negative, without fix-up, for most of the problems encountered.

Our strategy is not without drawbacks. Parallelizing over space is not yet at its best.
Work is underway to find an efficient partitioning technique that would enable our code not to be confined in solving mid-size problem.

A lot of options will be added to future versions :

- Treating simple and periodic reflective boundaries ,
- Extending parallelism to angle and space,
- Implementing diffusion synthetic acceleration [7] (DSA),
- Adding more physics,
- Parallelizing satellite processes,
- . . .

As a reminder, here is a summary of the main feature set of the current STYX 3D version :

- Multigroup anisotropic AMR S_N solver,
- Fixed Source and Eigenvalue problems,
- Efficient eigenvalue tracking algorithm,
- Inner and outer iteration scheme,
- Fully parallel in energy (MPI based) and partially in space (OpenMP based),
- Automatic AMR rezoning of Cartesian grids (up to 90 % cells saved),
- Pseudo-diagonal cell sweeping,
- Flux remapping to reduce convergence overhead of serial calculations,
- Unlimited anisotropic scattering Legendre development ,
- Unique coding for all octants,
- Enhanced memory cache management,
- Mixed ANSI C-F90 programming (over than 25000 source lines),
- UCD graphical output,
- . . .

ACKNOWLEDGEMENTS

I would like to thank :

- Antoine Tabary (IBM France) & Jacques Noé (SGI-CRAY Research) for their useful advice about MPI,
- Bruno Frogé, Christophe Lemaire & Jean-Philippe Nominé for the brilliant graphical applications they developed,
- Alireza Haghighat, Robert Roy, Jasmina Vujic & Daniel Verwaerde who triggered this work.

REFERENCES

1. C. Aussourd, "An Adapted DSN Scheme for Solving the Two-Dimensional Neutron Transport on a Structured AMR Grid", Proceedings of Mathematics and Computation, M&C'+SNA'97, Vol. 1, pp. 41-50, American Nuclear Society, Saratoga Springs, NY (October 1997).

2. C. Aussourd, "Latest Developments in the 2D-AMR S_N Code Styx", Proceedings of Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications, M&C'99, Madrid Spain, Vol. 2, pp. 1205-1214, (September 1999).

3. G. Bell and S. Glasstone, *Nuclear Reactor Theory*, Krieger Publishing Company, Malabar, Florida (1970).

4. E. E. Lewis and J. W. Miller, *Computational Methods of Neutron Transport*, American Nuclear Society, (1993).

5. H. Jourdren, J.M. Sibeaud and M. Adamczewski-Cauret, "Logiciel OURANOS : présentation et utilisation en détonique," *Revue Scientifique et Technique de la Défense*, (1995).

6. G. E. Sjoden & A. Haghighat , "3-D Parallel Discrete Ordinates Methods for Criticality Safety Calculations", Proceedings of Mathematics and Computation, Reactor Physics and Environmental Analysis in Nuclear Applications, M&C'99, Madrid Spain, Vol. 2, pp. 1403-1412, (September 1999).

7. R.E. Alcouffe, "Diffusion Synthetic Acceleration for the Diamond Differenced Discrete Ordinate Equations," *Nuclear Science Engineering*, 66, 344 (1977).