# Parallel Load Balancing Heuristics for Radiative Heat Transfer Calculations*

Kamal Viswanath‡§, Ivana Veljkovic†, and Paul E. Plassmann‡

§Presenting Author
‡Department of Electrical Engineering
Virginia Tech, Blacksburg, VA 24060, USA
†Department of Computer Science and Engineering
The Pennsylvania State University, University Park, PA 16802, USA,
kamalv@vt.edu, veljkovi@cse.psu.edu, plassmann@vt.edu
Keywords : Radiative Heat Transfer, Parallel Computation, Load Balancing, Monte Carlo

**Abstract.** *The computation of radiative effects by the Photon Monte Carlo method is computationally demanding, especially when complex, non-gray absorption models are employed. To solve such computationally expensive problems we have developed a parallel software framework for the photon Monte Carlo method based on ray tracing to compute radiative heat transfer effects. The central problem with obtaining scalable performance for this method is that widely varying physical properties over the computational domain result in highly skewed processor work assignment. In this paper we present computational results that demonstrate the effectiveness of a geometry based, domain partitioning heuristic with element weights for solving this load balancing problem. We present computational results that compare this heuristic to competing schemes for a representative combustion problem.*

## 1 Introduction

Radiative Heat Transfer (RHT) plays a central role in many combustion and engineering applications. Quite often though, the computational cost of modeling radiative heat transfer effects accurately can be extremely high due to its highly nonlinear and nonlocal nature. Simulation of combustion processes, nuclear reactions, and many other high-temperature physical phenomena are influenced by the accuracy of the models used for radiative effects since the RHT rates are generally proportional to the fourth power of the temperature. Omitting radiative heat transfer from the physical model in such applications can lead to mispredictions of the computed temperature profile which in turn affect the stability and accuracy of the calculation of other variables [1] such as species concentrations in a combustion etc.

The nonlocal nature of RHT comes from the fact that the photons that carry radiation have variable mean free path (the path from the point where photon is released to the point where it is absorbed). Because of these nonlocal effects, conservation laws cannot be applied over an infinitesimal volume (as is common for many fluid mechanics problems), but must be applied over the entire computational domain. The Photon Monte Carlo (PMC) [1] sampling technique can be effectively used to solve to general thermal radiation problems. This method is based on a model of radiative energy traveling in discrete packets, like photons, and the computation of their effect while traveling

* This work was partially supported by NSF grants DGE-9987589, CTS-0121573, EIA-0202007, and ACI-0305743.

as rays, scattering, and interacting with matter within the computational domain. Advantages of this method include, ease in dealing with complex geometries, non-uniform temperature fields, and scattering as well as being able to employ a great variety of methods to calculate radiative properties of the enclosure.

In the next section, a brief overview of radiative heat transfer properties, formulation of the equation of transfer, complexities of the radiative transfer problem, and the algorithm for the PMC method is presented. Section 3 describes the parallel extension of the PMC method and discusses the partitioning, load balancing issues, and parallel algorithm. In the final section, results for a representative test problem is presented for the load balancing strategy adopted.

## 2 The Photon Monte Carlo Method

The PMC method at its core is essentially a sampling method. The complexity of the implementation grows very slowly as the problems become more difficult to solve. Hence, its quite suited for radiation calculations where the challenges include complex geometries, non-uniform temperature fields, scattering and nonuniform, nonlinear, and non-gray radiative properties. We discuss the properties of heat transfer, complexity, and the Photon Monte Carlo method in this section.

### 2.1 Properties of Radiative Heat Transfer

A "radiatively participating" medium can absorb, emit and scatter thermal radiation. Each of these interactions need to be accounted for in a simulation.

The equation that describes these interactions for radiative heat transfer in a participating medium is given by equation(1), known as the Radiative Transfer Equation(RTE). This is an integro-differential equation in a five-dimensional space (three spatial variables and the two angles that determine the direction of $\hat{s}$). Here the change in intensity $I_\eta$ of incoming radiation from point $s$ to point $s + ds$ is found by summing the contributions from emission, absorption, scattering away from direction $\hat{s}$ and scattering into the same direction $\hat{s}$ at wavenumber $\eta$ [1]. We express this infinitesimal change in intensity as

$$\begin{aligned} \frac{dI_\eta(\hat{\boldsymbol{s}})}{ds} &= \hat{\boldsymbol{s}}\nabla I_\eta(\hat{\boldsymbol{s}}) \\ &= k_\eta I_{b\eta} - \beta_\eta I_\eta(\hat{\boldsymbol{s}}) \\ &\quad + \frac{\sigma_{s\eta}}{4\pi}\int_{4\pi} I_\eta(\hat{\boldsymbol{s}})\Phi_\eta(\hat{\boldsymbol{s}}_i, \hat{\boldsymbol{s}})d\Omega_i \end{aligned} \tag{1}$$

where $k_\eta$ is the absorption coefficient, $\sigma_{s\eta}$ is the scattering coefficient (dependent on the incoming direction $s$ as well as the wave number), $\beta_\eta$ is the extinction coefficient that describes total attenuation, and $\Phi_\eta(\hat{\boldsymbol{s}}_i, \hat{\boldsymbol{s}})$ is the phase function that describes the probability that a ray coming from direction $\hat{\boldsymbol{s}}_i$ will scatter into the direction $\hat{\boldsymbol{s}}$. Integration is performed over the entire solid angle $\Omega_i$ .

An exact analytical solution to this equation is typically impossible to obtain. Exact solutions are possible only for a few very simple situations, usually omitting scattering, assuming constant properties and gray, diffuse radiation.

### 2.2 Complexities

As alluded to before, there are a number of phenomena that make a radiative heat transfer problem difficult to solve.

- **Geometry** The problem maybe one-dimensional, two-dimensional or three-dimensional. As the complexity of the enclosure geometry grows, the complexity of mathematical formulations become more involved.
- **Temperature Field** One of the most difficult case is when radiative heat transfer is combined with conduction and/or convection, resulting in a highly nonlinear integro-differential equation. On the other hand, an easier situation is when the temperature profile within the medium is known, such as the case of a basic isothermal medium which have been studied extensively.
- **Scattering** The simple case is when the medium does not scatter. In such a case the equation of transfer reduces to a first-order differential equation if the temperature field is known, and a relatively simpler integral equation if radiative equilibrium(conduction and convection are negligible) prevails. On the next level, isotropic scattering is often assumed and investigated. The most difficult case would be where anisotropic scattering is prevalent.
- **Properties** The participating media will have gray or nongray character. A gray media is one whose radiative properties(absorption coefficient $k$, scattering coefficient $\sigma_s$, and phase function $\Omega$, as well as emittance of boundary surfaces) does not vary across the electromagnetic spectrum. Most "exact" solutions are limited to gray media. Most participating media display strong nongray character.

### 2.3 The PMC Algorithm

Problems in thermal radiation are well suited to solution by a Monte Carlo technique. Energy travels in discrete parcels(photons) over (usually) relatively long distances along a straight path before interaction with matter. Classical methods, such as the method of spherical harmonics[2] and the method of discrete ordinates[3], have been successfully implemented in numerous applications, but they have failed to address various important issues that can arise in the applications where really complex radiative properties have to be modeled. Hence beyond a certain problem complexity, the Monte Carlo solution is preferable.

The main issue for the PMC algorithm is how to coordinate randomness of the sample with physical properties of the enclosure. In the PMC method, problems such as determining the number of photon rays, points of emission within a surface or a volume, wavelength, points of absorption and many others are solved based on the enclosure properties. If in a given problem conduction and/or convection are important, the RHT equation must be solved simultaneously with the overall conservation of energy equation:

$$\rho c_v(\frac{DT}{Dt}) = -\nabla(k\nabla T) - p\nabla.v + \mu\Phi + \dot{Q}''' - \nabla.q_r \quad (2)$$

where $c_v$ is the specific heat, $T$ is the temperature, $p$ is the pressure, $k$ is the thermal conductivity, $v$ is the velocity vector, $\Phi$ is the dissipation function, $\dot{Q}'''$ is the heat generated within the medium (for example, the energy release due to chemical reactions), and $\nabla.q_r$ is the net radiative source.

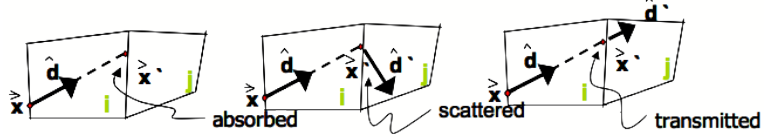


**Fig. 1.** Possible photon-volume interactions in a two-dimensional domain

In numerical simulations, the computational domain containing a participating medium is typically discretized into $N$ sub-surfaces and $M$ sub-volumes. The PMC method traces a statistically significant sample of photon bundles from their point of emission within surface/volume j to their point of absorption within surface/volume k. When the photon bundle is absorbed, its energy is added to the absorbed energy of the absorbing element. With this approach, the PMC method is able to calculate the energy gains/losses for every element (or fluid particle) in the enclosure. The emitted energy from the element can be successfully modeled with various methods (depending on the application needs). We can then calculate the divergence of the heat flux for every sub-volume in the domain (and the radiative heat flux for the surface elements):

$$(\nabla . q_r)_i = \frac{1}{V_i}(Q_{em,i} - Q_{abs,i}) \qquad (3)$$

This flux is needed as the radiation component along with convection and conduction in the energy conservation equation. The ray tracing is done by tracing a ray through one mesh element and then proceeding to the next ray. The rays are organized in a linked list—after a ray has been traced through one element, the next element it will go through is calculated and ray is pushed back into the list. Such an approach gives better data locality since rays that are close in the list have more chance to traverse the same element.

## 3 Parallel PMC Methods

There are two broad choices on how to conduct the parallelization for the above Monte Carlo algorithm. These two approaches are described below.

**Parallelization through Ray Partitioning** In the ray partitioning method, the total number of rays are divided among the processors while the simulation domain may or may not be partitioned. A processor tracks each of the assigned rays through its entire path until it is absorbed or its energy is depleted. An advantage of this method is that a good load balance between processors can be achieved with a reasonable assignment of statistically independent rays to the processors. With ray partitioning, every processor either has to have the entire mesh structure in its memory, or it has to have intensive communication with the processors that own the mesh elements that are traversed by the rays. In the case when mesh data structure can fit on the memory of one processor this method is feasible and efficient since it introduces minimal communication overhead

**Parallelization Through Domain Partitioning** In large scale simulations, the discretization of the domain is usually very fine to satisfactorily resolve the underlying phenomena. This implies very large mesh data structures which will not sit in the memory of a single processor. In such a case, the domain has to be partitioned across the processors. In domain partitioning, the underlying mesh is divided among processors and every processor emits rays from its portion of the mesh. When the ray crosses the mesh boundary, it is communicated to the neighboring processor. This is the more general approach, in case of large meshes the only one, and the one adopted here. Since this can potentially lead to extensive communication, the load balancing issues which result from such a partitioning are discussed in the next subsection.

### 3.1 Load Balancing

The issue of the distribution of the computational load among the processors is called load balancing. In every parallel application where extensive communication is present, it is very important that the computational load is well distributed. If not, the resulting imbalance in computational load will result in some processors spending time idling while waiting for work. Communication load is measured by the number of rays that are communicated to neighboring processors and the computational load is measured by the number of elements each ray traverses before it is absorbed.

With domain partitioning, if one part of the enclosure is hot, it will emit more photon bundles than regions with lower temperature, since the emitted energy depends on the fourth power of temperature. In this case, the processor that owns the hot, emitting region will be overwhelmed with work while the processor which owns the cold region will have much fewer bundles to track. However, the computational load depends not only on the number of emitted rays from a given subdomain, but also on the number of mesh elements that are traversed by the rays. If the subdomain is optically thick, rays travel short distances and the average number of elements a ray traverses is small. The opposite holds true for optically thin media. The optical thickness of the medium depends on the absorption coefficient, and the absorption coefficient decreases as the temperature increases. For an optically thin medium, the ray travels far and both communication and computational loads are large.

The partitioning algorithm needs to partition the domain into $P$(number of processors) subdomains of approximately equal computational costs while attempting to minimize communication costs, i.e., number of rays crossing domain boundaries. While the communication information is not known apriori, it is important to avoid long, skinny subdomains, which if there is significant local communication will result in more messages than will a decomposition that generates square subdomains.

We use a recursive bisection technique which uses the geometric information to partition the domain. One of the simplest algorithms is the orthogonal recursive bisection (ORB) algorithm [4] which makes an initial cut to divide the vertices into two sets of equal size, and then orthogonal cuts are made recursively in the new subdomains until the vertices are equally distributed among the processors. Even though it achieves good load balancing, the communication may not always be optimal. Another approach is the parametric binary dissection (PBD) [5] algorithm in which each cut is chosen to minimize $load + \lambda * shape$, where $load$ is the computational load in a subdomain and $shape$ is a measure of the communication overhead across the subdomain and the parameter $\lambda$ seeks to trade off load imbalance against communication costs.

The method adopted is a generalization of the ORB algorithm called Unbalanced Recursive Bisection(URB) [6], which attempts to reduce communication costs by forming subdomains that have better aspect ratios. Instead of automatically dividing a grid in half, it considers the $P - 1$ partitions obtained by forming unbalanced subgrids with $1/P$ and $(P - 1)/P$ of the load, with $2/P$ and $(P - 2)/P$ of the load, and so on, and chooses the partition that minimizes the partition aspect ratio. The resulting subsets are further subdivided in proportion to their sizes. This heuristic generally leads to an equal distribution of grid points with better partition aspect ratios.

### 3.2 The Parallel PMC Algorithm

In the domain partitioning method, when a ray crosses the mesh boundary, it is communicated to the neighboring processor. To minimize the communication cost, rays that are to be communicated to the neighboring processors are packed into a contiguous message. This message is sent after it reaches a certain predetermined size or until all rays in the domain are processed. Simultaneously, packets of rays are received from other processors. Therefore, there are two alternating phases of computation, a calculation phase and a communication phase. Both phases are executed in turn until all the rays in the global domain are absorbed. A complete description of the parallel algorithm is given in [7].

## 4 Computational Results

The example case taken is the radiative heat transfer between two infinite parallel plates. The properties inside the medium and on plates are constant and the notion of infinite plates is modeled by periodic boundary conditions (the photon bundle that exits the domain on one side, reenters it on the other side, as shown in Figure 2). As explained in section 3.1, the optically thin medium entails large communication and computational loads and we consider the example case to be an optically thin and scattering medium.

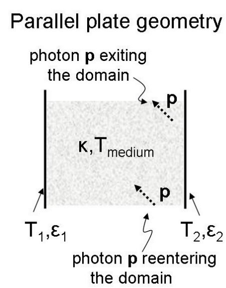


**Fig. 2.** The geometry of the parallel plate problem

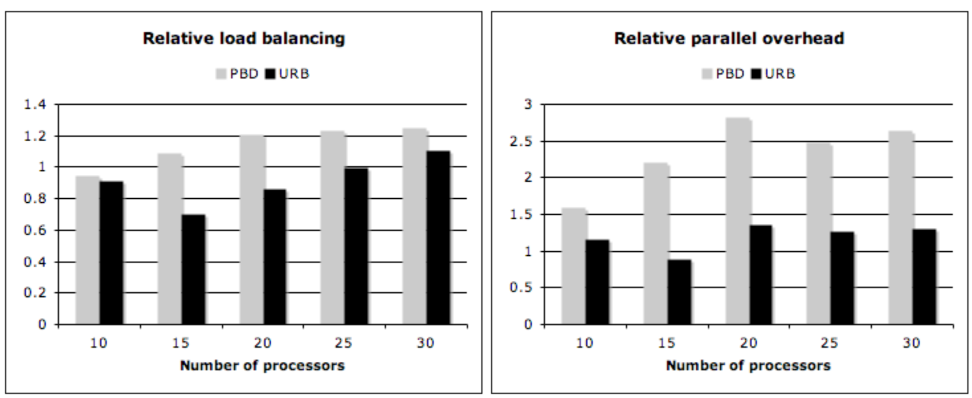


**Fig. 3.** Relative load balancing factor and relative parallel overhead with optically thin medium and weak scaling

We consider the case of weak scaling, that is the ratio between number of processors and overall computational load is kept constant as we increase the number of processors. Since the computational load depends linearly on the total number of rays emitted from the simulation domain, we use the number of rays as a measure of the computational load while the mesh size is kept constant.

Figure 3 shows the relative load balancing factor and relative parallel overhead for this case using PBD (considering only computational load) and URB heuristics. The relative load balancing factor is defined as the difference between maximum and minimum time spent in calculation across all processors, divided by the average time spent in calculation. The relative parallel overhead is defined as the difference between average overall CPU time and average time spent in calculation divided by the average time spent in calculation. It includes the idle time processors spend waiting for messages to arrive as well as communication time. From the left figure, one can see that the relative load balancing factor does not change significantly for different number of processors, which demonstrates that the partitioning algorithm is scalable and the load per processor is kept roughly the same. On the other hand, the relative parallel overhead, as seen, is significantly different when using the different heuristics. The PBD scheme consistently incurs greater overhead than the URB scheme, as is expected when the partitioning is done based only on computational load without considering aspect ratios of the subdomains.

Considering the instance using URB, the computational load can be balanced, but the parallel overhead is significant. This is reasonable since we have assumed an optically thin medium as well as scattering. As the rays get absorbed, towards the end, there will be less number of rays traveling through many subdomains, which implies large number of small messages and a lot of idle time for waiting processors until all rays are absorbed. As the optical thickness of the medium increases, these rays will travel smaller distances and get absorbed faster resulting in lower parallel overheads.

So with the URB partitioning scheme, we do arrive at reasonable load balancing albeit not so small communication overhead for this case of an optically thin medium. Only using the geometric data, we are still able to keep the load balanced across processors and the communication less.

## 5 Conclusions and Future Work

The central issue in efficiently implementing the parallel framework for the photon Monte Carlo method is how to conduct the partitioning of the work. For large-scale simulations, in most cases, parallelization using domain partitioning is the only choice. The distribution of the computational load to the processors depends heavily on how the global mesh is partitioned among the processors. Therefore, in this paper we have examined the application of the URB heuristic as a domain partitioning strategy and presented a quantitative study of its performance compared to other geometric partitioning strategies. In future work, the parallel implementation will be tested with a variety of mediums with varying properties.

## References


[1] Modest, M.F.: Radiative Heat Transfer. Academic Press (2003)

[2] Jeans, J.: The equations of radiative transfer of energy. Monthly Notices Royal Astronomical Society (1917) 28–36

[3] Chandrasekhar, S.: Radiative Transfer. Dover Publications (1960)

[4] Berger, M., Bokhari, S.: A partitioning strategy for nonuniform problems on multiprocessors. IEEE Transactions on Computers (1987) C–36:570–580

[5] Bokhari, S., Crockett, T., Nicol, D.: Binary dissection: Variants and applications. ICASE Report No. 93-39 (1993)

[6] Jones, M., Plassmann, P.: Parallel algorithms for the adaptive refinement and partitioning of unstructured meshes. In: Proceedings of the 1994 SHPCC, Knoxville, TN, IEEE (1994) 726–733

[7] Veljkovic, I., Plassmann, P.E.: Scalable photon Monte Carlo algorithms and software for the solution of radiative heat transfer problems. In Yang, L.T., Rena, O.F., Martino, B.D., Dongarra, J., eds.: High Performance Computing and Communications. Volume 3726 of Lecture Notes in Computer Science (LNCS). Springer (2005) 928–937