# Progressive volume photon tracing


Charly Collin*
University of Central Florida, USA

Mickaël Ribardière†
IRISA Rennes, France

Rémi Cozot ‡
IRISA Rennes, France

Kadi Bouatouch§
IRISA Rennes, France


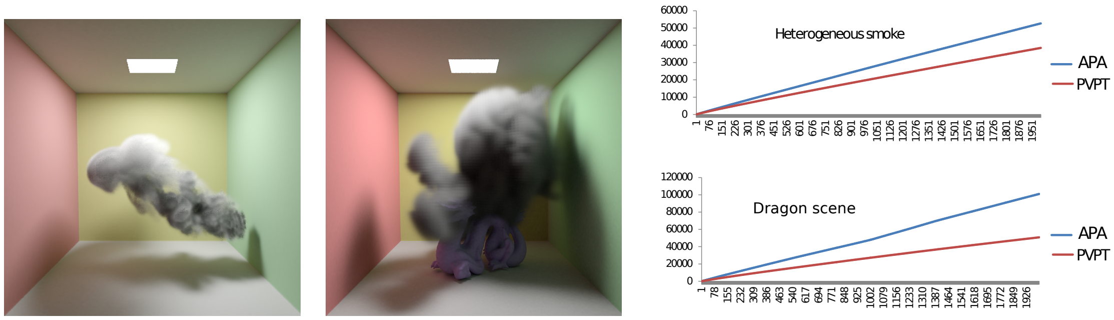


**Figure 1:** *Final results of the two scenes. The chart on the right compares the cumulated rendering time (in seconds) over the iterations using our method (red) with that of the method entitled Progressive Photon Mapping: A Probabilistic Approach.*

Over the past decade, photon mapping is being more and more used to simulate light transport in scenes containing participating media. However, photon mapping is limited practically because of its large memory requirements. Progressive photon mapping addresses this problem [Hachisuka et al. 2008; Knaus and Zwicker 2011]. This approach computes many images of the same scene from the same viewpoint using different sets of photons and different parameter values. The obtained low quality images, when combined together, generate images of higher quality. In this talk, we propose a volume photon mapping method which does not have to store the photon maps, ensures a faster convergence to the final solution, and allows the user to have a full control of the size of the used memory. We assign a beam to each primary or secondary ray crossing a participating medium, then build a beam hierarchy. As soon as the computation of an emmited photon's contribution to a beam or to a visible point is done, we throw away this photon. In this way there is no need to store any photon map, which solves the problem of memory storage.

## 1 Overview

Our algorithm consists of two steps: a preprocessing step followed by a progressive photon tracing step. In the rest of the talk, each iteration of the progressive photon tracing step is called *progressive pass*.

**Preprocessing.** Each view ray, cast from the camera through a participating medium, is assigned one beam, which is a cylinder representing the portion of the ray within the medium. The secondary rays (reflected and refracted) are also assigned beams. All these beams are used to build a Kd tree beam hierarchy similar to [Havran et al. 2005]. Besides, the visible points resulting from the intersection of the view rays with the scene's surfaces are also stored in a Kd-tree (called surface Kd-tree) [Hachisuka et al. 2008]. Finally, for each pixel, the visible point as well as the beam associated with the corresponding view ray are stored. Each photon may contribute to neighboring visible points and to the beams (leaves of the beam Kd-tree) containing the photon. These beams are determined by a top-down traversal of the beam hierarchy. For each visible point and each beam, the current radiance is stored along with a temporary variable to store the contributions of the photons cast during a pass of the progressive photon mapping. Note that only the two Kd-trees have to be stored during the progressive photon mapping process, which makes our approach less demanding in terms of memory for medium size scenes.

**Sending photons.** Photons are emitted using classical photon based methods. Once a cast photon contributes to a visible point or a beam, it is thrown away. This contribution is added to the temporary radiance of the neighboring visible points or the beam containing the photon. The cumulated radiance is updated whenever a progressive pass is completed. The radius of a visible point neighborhood must be updated as well by decreasing it after each progressive pass, making the algorithm focus on the less updated parts of the scene.

## 2 Early results

So far we have tested our method on simple scenes. The first one consists of a heterogeneous smoke in a box, and the second contains a diffuse dragon inside a heterogeneous smoke. To evaluate our method on these scenes, we compared it with [Knaus and Zwicker 2011]. These early results show that our method is faster to produce images of same quality. (see figure 1).

## 3 Conclusion

We proposed a new approach to progressive photon mapping allowing a constant memory cost and a faster rendering. The first version of the implementation provided results with quality similar to existing methods and seems very promising. In the future, we would like to test a fully functional algorithm on more complex scenes. We would also like to explore other aspects of our method such as parallelism.

## References


HACHISUKA, T., OGAKI, S., AND JENSEN, H. W. 2008. Progressive photon mapping. *ACM Trans. Graph. 27* (December), 130:1–130:8.

HAVRAN, V., BITTNER, J., HERZOG, R., AND SEIDEL, H.-P. 2005. Ray Maps for Global Illumination. 43–54.

KNAUS, C., AND ZWICKER, M. 2011. Progressive photon mapping: A probabilistic approach. *ACM Trans. Graph. 30* (May), 25:1–25:13.



*e-mail: charly.collin@bobbyblues.com
†e-mail: mickael.ribardiere@irisa.fr
‡e-mail: remi.cozot@irisa.fr
§e-mail: kadi.bouatouch@irisa.fr