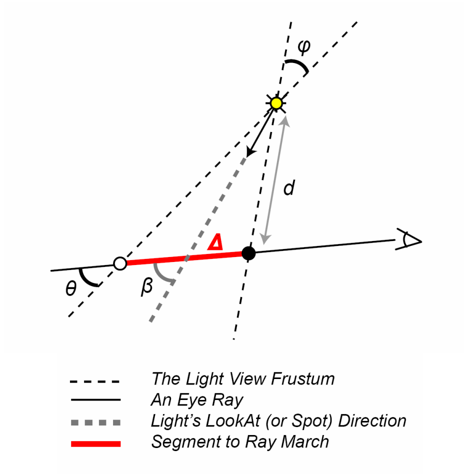
Appendix B: Determining Ray Sampling Size



We want to compute Δ to subdivide our ray into appropriate sized steps. We know the locations of the black point, the eye, and the light, as well as the light's field of view (ϕ) and the light's viewing direction.

Based on the eye and light directions, β is easily computed. The distance d is also easily computed. Given these we can compute the angle θ.

$$\theta = \pi - (\frac{\phi}{2}) - (\pi - \beta) = \beta - \frac{\phi}{2} \tag{1}$$

Then:

$$\frac{\sin\phi}{\Delta} = \frac{\sin(\beta - \frac{\phi}{2})}{d} \tag{2}$$

Or:

$$\Delta = \frac{d\sin\phi}{\sin(\beta - \frac{\phi}{2})} \tag{3}$$

$\sin(\phi)$ is constant per frame and thus easily computed. Using trig identities, the denominator becomes:

$$\sin(\beta - \frac{\phi}{2}) = \sin\beta\cos\frac{\phi}{2} - \cos\beta\sin\frac{\phi}{2} \tag{4}$$

$\cos\frac{\phi}{2}$ and $\sin\frac{\phi}{2}$ are constant per frame, while $\cos\beta$ can be computed with a dot product. $\sin\beta$ is obtained as $\sqrt{1-\cos^2\beta}$

Unfortunately, this expression for Δ is not stable as β approaches $\frac{\phi}{2}$ because the viewing direction becomes parallel with an edge of the light view frustum. Thus, the equations give very large Δ values. These values are essentially correct, but it makes no sense to step through the volume to a distance past the eye's far plane. We would like scattering samples to occur relatively near the eye, not beyond the far plane.

We have used two solutions that reduce this problem. In the first case, we compute Δ using a constant denominator. This roughly corresponds to Figure 4 in the text, where some rays will sample beyond the far side of the light frustum (or if the constant denominator is too big, sampling may end too early).

The second solution clamps the denominator to some minimal value. this can lead to visible discontinuities at the clamp point, since the sampling planes sharply change (as in Figure 4, left, in the text). To avoid this, clamping can be done with the GPU's smoothstep function (a Bernstein interpolant).