# Artist Friendly Metallic Fresnel

Ole Gulbrandsen
Framestore

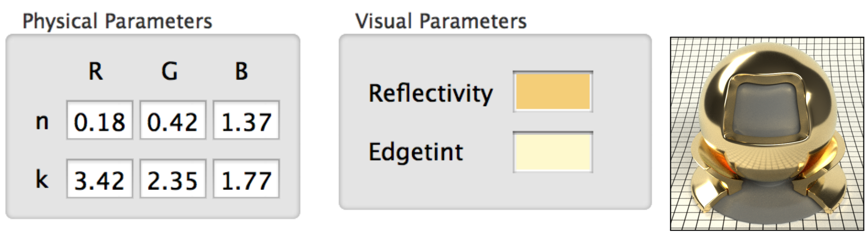



**Figure 1**. Physical versus visual parameters (these are approximate values for gold).

## Abstract

Light reflecting off metallic surfaces is described by the Fresnel equations [Born and Wolf 1999], which are controlled by the complex index of refraction $\eta = n + ik$.

Together, $n$ and $k$ determine the two characteristics of the Fresnel curve for a material: the reflectivity at normal incidence and how quickly it fades to white at grazing angles. However, this parameterization presents some artistic challenges because both characteristics depend on both parameters. An artist would ideally manipulate each property independently and have them on a unit scale. This paper describes a remapping for the approximated unpolarized Fresnel equations of $n$ and $k$ to the more intuitive *reflectivity*, $r$, and *edgetint*, $g$, both in the range from 0 to 1.

## 1. Introduction

Balancing the energy between reflected and internally scattered light, the Fresnel equations play a central role in physically-based shading. The light reflected from a metallic surface is dependent on the angle of incidence, and to describe it, an artist would typically need to enter one pair of $n, k$ per RGB channel:

$$\mathbf{n} = (n_R, n_G, n_B), \ \mathbf{k} = (k_R, k_G, k_B)$$

as shown left in Figure 1. In this paper, we introduce a remapping to the normalized parameters

$$\mathbf{r} = (r_R, r_G, r_B), \ \mathbf{g} = (g_R, g_G, g_B)$$

 ISSN 2331-7418

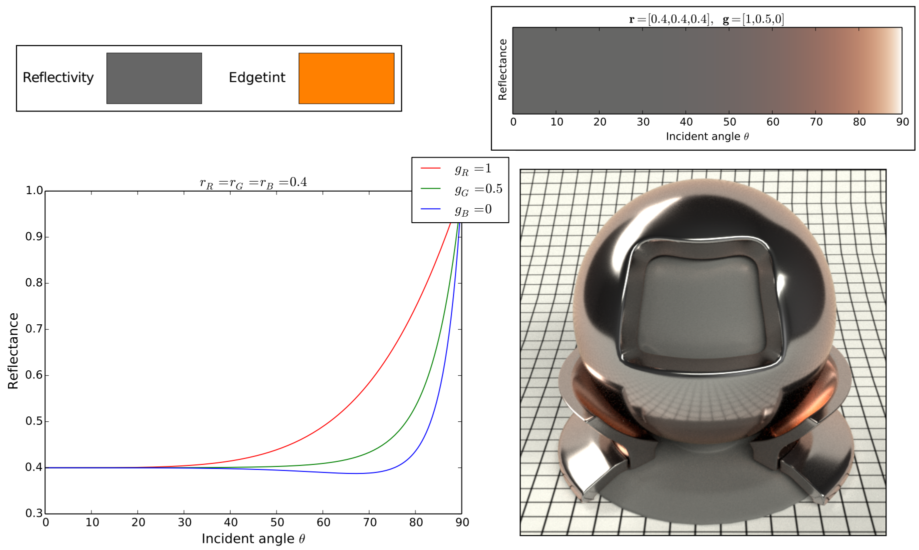



**Table 1.** Perhaps not the most plausible metal, but by choosing a monochrome $\mathbf{r} = 0.4$ we can see how the reflected color starts at $\mathbf{r}$ and is biased by $\mathbf{g}$ before it fades to white on the edge. See Equation 7.

as shown on the right of Figure 1. The first parameter, $\mathbf{r}$, determines the reflectance at normal incidence, as in Cook and Torrance [Cook and Torrance 1981] and Schlick's approximated Fresnel factor [Schlick 1994].

The second parameter, $\mathbf{g}$, the main contribution in this paper, controls the color bias as the viewing direction becomes parallel to the surface, as shown in Table 1. Combined, $\mathbf{r}$ and $\mathbf{g}$ map the same range of Fresnel reflectance as $\mathbf{n}$ and $\mathbf{k}$.

## 2. Theory

### 2.1. Visually Predictable Parameters

The motivation behind artist friendly parameters is to allow artists making visually predictable changes to physically plausible models. Ideally they should be normalized—mostly for easy texturing—and increasing a parameter should increase the result; for a function $f$ and parameter $x$ we want

$$\frac{\partial f}{\partial x} \geq 0 \text{ for all } x \in [0,1]. \tag{1}$$

### 2.2. Mathematically Plausible Values

Reflectance along normal incidence [Born and Wolf 1999] is given by

$$r_{\perp}(n,k) = \frac{(n-1)^2 + k^2}{(n+1)^2 + k^2}.$$

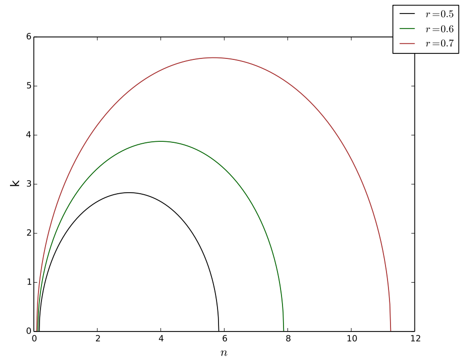



**Figure 2**. Plotting Equation 2 for three different values of $r$. By fixing $r$ we get a constraint on $n$ and $k$.

Setting $r_\perp$ equal to our input parameter, $r$, we can represent as $k$ as a function of $r$ and $n$:

$$k(r,n) = \sqrt{\frac{1}{1-r}\left(r(n+1)^2 - (n-1)^2\right)} \tag{2}$$

which is defined if the expression under the square root is non-negative, giving us the following interval for $n$:

$$n \in \left[\frac{1-\sqrt{r}}{1+\sqrt{r}}, \frac{1+\sqrt{r}}{1-\sqrt{r}}\right] \text{ for } r \in [0,1) \tag{3}$$

as shown in Figure 2. Cook and Torrance [Cook and Torrance 1981] use the upper boundary in Equation 3 to compute $n$, but we are free to choose any $n$ within this range, as long as $k$ satisfies Equation 2. We will now explore the visual effect of doing just that, for the unpolarized approximate Fresnel equations.

### 2.3. Unpolarized Approximate Fresnel Equations

In this section we will focus on a common approximation of the unpolarized Fresnel equation [Pharr and Humphreys 2010]:

$$R(r,n,\theta) = \frac{1}{2}\left(f_\perp(n,k,\theta) + f_{||}(n,k,\theta)\right) \tag{4}$$

where the perpendicular polarized reflection, $f_\perp$, and the parallel polarized reflection, $f_{||}$, are defined as

$$f_\perp(n,k,\theta) = \frac{n^2+k^2-2n\cos\theta+\cos^2\theta}{n^2+k^2+2n\cos\theta+\cos^2\theta} \tag{5}$$

$$f_{||}(n,k,\theta) = \frac{(n^2+k^2)\cos^2\theta-2n\cos\theta+1}{(n^2+k^2)\cos^2\theta+2n\cos\theta+1}. \tag{6}$$

We can replace $k$ with $k(r,n)$ from Equation 2, and write Equation 4 as a function of $r$, $n$ and the viewing angle $\theta$:

$$R(r,n,\theta) = \frac{1}{2}\left(f_\perp[n,k(r,n),\theta] + f_{||}[n,k(r,n),\theta]\right). \tag{7}$$

#### 2.3.1. *Reflectivity and Edgetint*

We want $R$ to be monotonically increasing (Equation 1) with respect to both $r$ and $g$. Starting with $r$; since we defined $r$ as the reflectivity at normal incidence, we only need to show that

$$\frac{\partial R}{\partial r} \geq 0$$

for $\theta = 0$, which is trivial, since

$$R(r,n,0) = r.$$

Hence $r$ is a valid parameter. The next challenge is finding $g$. The idea is using $g$ to interpolate between the valid values of $n$. We can show that

$$\frac{\partial R}{\partial n} = 0 \tag{8}$$

for

$$n = 0 \text{ and } n = \frac{1-r}{1+r} \tag{9}$$

and only the latter is in the valid range. It is also independent of $\theta$. This means, in order to satisfy the monotonically increasing property, we have to limit the range of $n$ either as

$$n \in \left[\frac{1-\sqrt{r}}{1+\sqrt{r}}, \frac{1-r}{1+r}\right] \tag{10}$$

or

$$n \in \left[\frac{1-r}{1+r}, \frac{1+\sqrt{r}}{1-\sqrt{r}}\right] \tag{11}$$

as shown as the two regions in Figure 3. Here we see that the right (blue) region, Equation 11, gives the most linear behavior. In this range

$$\frac{\partial R}{\partial n} \leq 0$$

for every $r$ between $[0,1)$ and every $\theta$ between $[0^\circ, 90^\circ]$. This is almost what we want for $g$ except for the negative derivative. If we use $g$ as parameter in a linear interpolation

$$n(r,g) = g\frac{1-r}{1+r} + (1-g)\frac{1+\sqrt{r}}{1-\sqrt{r}} \tag{12}$$

we get

$$\frac{\partial R}{\partial g} = \frac{\partial R}{\partial n}\frac{\partial n}{\partial g} + \frac{\partial R}{\partial r}\frac{\partial r}{\partial g} + \frac{\partial R}{\partial \theta}\frac{\partial \theta}{\partial g} = \frac{\partial R}{\partial n}\frac{\partial n}{\partial g} \geq 0 \tag{13}$$

for all $r$ in $[0,1)$ and all $\theta$ in $[0^\circ, 90^\circ]$ satisfying Equation 1 for $g$.

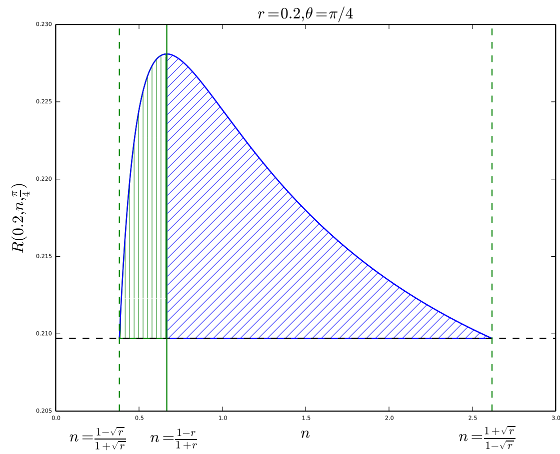



**Figure 3**. Plotting $R$ as a function of $n$. We can use either the left or the right side, since $R$ is the same at $n = \frac{1-\sqrt{r}}{1+\sqrt{r}}$ and $n = \frac{1+\sqrt{r}}{1-\sqrt{r}}$.

#### 2.3.2. *Invertibility*

Finally we want to make sure that for each $(n,k)$ there is a unique $(r,g)$ and vice versa. Since we have the inverse mapping

$$r(n,k) = \frac{(n-1)^2 + k^2}{(n+1)^2 + k^2} \tag{14}$$

and

$$g(n,k) = \frac{\frac{1+\sqrt{r(n,k)}}{1-\sqrt{r(n,k)}} - n}{\frac{1+\sqrt{r(n,k)}}{1-\sqrt{r(n,k)}} - \frac{1-r(n,k)}{1+r(n,k)}} \tag{15}$$

and the determinant of the Jacobian

$$\begin{vmatrix} \frac{\partial n}{\partial r} & \frac{\partial n}{\partial g} \\ \frac{\partial k}{\partial r} & \frac{\partial k}{\partial g} \end{vmatrix} > 0$$

for all $r, g \in (0,1)$, the map from $(n,k)$ to $(r,g)$ is a bijection.

#### 2.3.3. *Final Parameters*

For the approximated Fresnel equations, we have the following remapping

$$n(r,g) = g\frac{1-r}{1+r} + (1-g)\frac{1+\sqrt{r}}{1-\sqrt{r}}$$

$$k(r,n(r,g)) = \sqrt{\frac{1}{1-r}\left(r(n(r,g)+1)^2 - (n(r,g)-1)^2\right)}$$

where $r \in [0,1)$ and $g \in [0,1]$. Table 2 shows a list of common materials and their reflectivity/edgetint parameters.

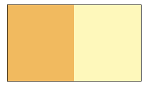
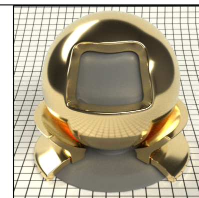
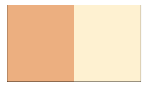
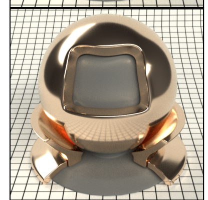
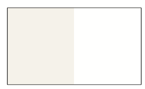
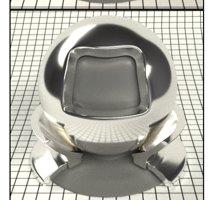
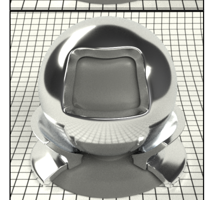
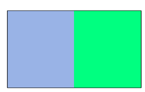
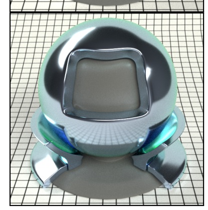
| Material | Reflectivity and Edgetint | Render |
|---|---|---|
| Gold |  |  |
| Copper |  |  |
| Silver |  |  |
| Aluminium |  |  |
| Unobtainium |  |  |

**Table 2**. Reflectivity and edgetint for common (and less common) metals. Data for the top four metals is borrowed from Refractive Index.info [Polyanskiy 2014].

## 3. Conclusion

We have show how we can start with the physical parameters **n** and **k** and remap them to **r** and **g**. While the choice of **g** as linear weight might appear somewhat arbitrary, it is a simple solution, yielding predictable behaviour for the artist.

Several scenarios have not been touched; these include polarized reflections and the full Fresnel equations [Glassner 1994]. From the authors experience these are not often used in production rendering.

## Acknowledgements

Thanks to Harry Bardak and Paul Beilby for suggesting, "there must be an easier way to do this!" Thanks to Serguei Lapine for helpful suggestions, Anders Langlands for spotting an embarrassing bug and Mikhail Polyanskiy for his excellent website [Polyanskiy 2014]. All images were rendered using Mitsuba [Jakob 2010].

## References

BORN, M., AND WOLF, E. 1999. *Principles of Optics*, 7 ed. Cambridge University Press. 64, 65

COOK, R. L., AND TORRANCE, K. E. 1981. A reflectance model for computer graphics. In *Computer Graphics (Proceedings of SIGGRAPH 81)*, 307–316. 65, 66

GLASSNER, A. S. 1994. *Principles of Digital Image Synthesis*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. 69

JAKOB, W., 2010. Mitsuba renderer. http://www.mitsuba-renderer.org. 70

PHARR, M., AND HUMPHREYS, G. 2010. *Physically Based Rendering, Second Edition: From Theory To Implementation*, 2nd ed. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. 66

POLYANSKIY, M., 2014. Refractive index.info. http://refractiveindex.info. 69, 70

SCHLICK, C. 1994. An inexpensive brdf model for physically-based rendering. *Computer Graphics Forum 13*, 233–246. 65

## A. Implementation

```
float n_min(float r){
    return (1-r)/(1+r);
}
float n_max(float r){
    return (1+sqrt(r))/(1 - sqrt(r));
}
float get_n(float r,float g){
    return n_min(r)*g + (1-g)*n_max(r);
}
float get_k2(float r, float n){
    float nr = (n+1)*(n+1)*r-(n-1)*(n-1);
    return nr/(1-r);
}
float get_r(float n, float k){
    return ((n-1)*(n-1)+k*k)/((n+1)*(n+1)+k*k);
}
float get_g(float n, float k){
    float r = get_r(n,k);
    return (n_max(r)-n)/(n_max(r)-n_min(r));
}
float fresnel(float r, float g,float theta){
    //clamp parameters
    float _r = clamp(r,0,0.99);
    //compute n and k
    float n = get_n(_r,g);
    float k2 = get_k2(_r,n);

    float c = cos(theta);
    rs_num = n*n + k2 - 2*n*c + c*c;
    rs_den = n*n + k2 + 2*n*c + c*c;
    rs = rs_num/rs_den;

    rp_num = (n*n + k2)*c*c - 2*n*c + 1;
    rp_den = (n*n + k2)*c*c + 2*n*c + 1;
    rp = rp_num/rp_den;

    return 0.5*(rs+rp);
}
```

### Author Contact Information

olegul@hotmail.com olegul.blogspot.com

---

Ole Gulbrandsen, Artist Friendly Metallic Fresnel, *Journal of Computer Graphics Techniques (JCGT)*, vol. 3, no. 4, 64–72, 2014
http://jcgt.org/published/0003/04/03/

| | | | |
|---|---|---|---|
| Received: | 2014-07-13 | | |
| Recommended: | 2014-08-20 | Corresponding Editor: | Natalya Tatarchuk |
| Published: | 2014-12-09 | Editor-in-Chief: | Morgan McGuire |