

OpenPBR: Novel Features and Implementation Details

JAMIE PORTSMOUTH, Autodesk

PETER KUTZ, Adobe

STEPHEN HILL, Lucasfilm

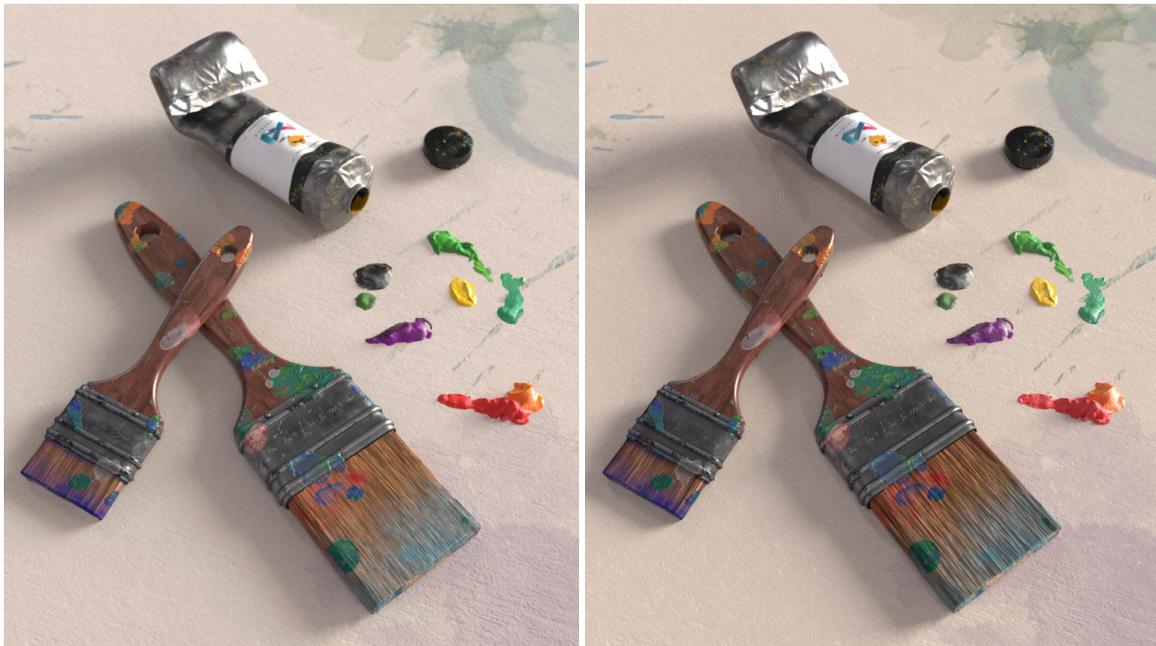


Figure 1: OpenPBR materials rendered in two separate systems (Arnold left, and Adobe's proprietary renderer right).

Abstract

OpenPBR is a physically based, standardized uber-shader developed for interoperable material authoring and rendering across VFX, animation, and design visualization workflows. This document serves as a companion to the official specification, offering deeper insight into the model's development and more detailed implementation guidance, including code examples and mathematical derivations.

We begin with a description of the model's formal structure and theoretical foundations – covering slab-based layering, statistical mixing, and microfacet theory – before turning to its physical components. These include metallic, dielectric, subsurface, and glossy-diffuse base substrates, followed by thin-film iridescence, coat, and fuzz layers. A special-case mode for rendering thin-walled objects is also described.

Additional sections explore technical topics in greater depth, such as the decoupling of specular reflectivity from transmission, the choice of parameterization for subsurface scattering, and the detailed physics of coat darkening and thin-film interference. We also discuss planned extensions, including hazy specular reflection and retroreflection.

CONTENTS

1	Introduction	3
2	An overview of the OpenPBR model	5
2.1	Model formalism	6
2.2	Microfacet models	12
2.3	Model structure	15
2.4	Geometry	17
2.5	Parameterization	18
3	Base substrate	19
3.1	Dielectric base	19
3.1.1	Glossy-diffuse model: EON	24
3.1.2	Subsurface	28
3.1.3	Translucent base	33
3.2	Metallic base: the F82-tint model	36
3.3	Emission	39
3.4	Future work	40
3.4.1	Hazy specular	40
3.4.2	Retro-reflection	41
4	Thin-film iridescence	42
5	Coat	44
5.1	Darkening	45
5.2	View-dependent absorption	48
5.3	Roughening	48
5.4	Total internal reflection	49
6	Fuzz	51
7	Thin-walled mode	54
7.1	Thin-wall dielectric	54
7.2	Thin-wall subsurface	55
8	Conclusion and Future Work	57
9	Acknowledgements	57
A	Parameterization	62
B	A mixture model example	63
C	Similarity theory of subsurface anisotropy	66
D	F82-tint model fits for a variety of real metals	67
E	Implementing thin-film iridescence from first principles	68
F	The physics of coat darkening	73

1 INTRODUCTION

The OpenPBR project originated in 2023 as a collaboration between Autodesk and Adobe to consolidate and refine their existing shading models, the Autodesk Standard Surface [Geo+19] and Adobe Standard Material [KHE21], respectively. The goal was to create a single, open, and standardized physically based shading model that would be suitable for the most common use cases in media and entertainment applications, while also being practical for real-time rendering.

OpenPBR is not intended to be a general framework for building material models (which exists in various forms [Gri+10; Ket+15; SS16; HD24]), but a reasonably unambiguous description of a particular form of pre-built “uber-shader”, which experience has shown is practically very useful.

There is a long history of previous proposals for such a physically based uber-shader. Figure 2 shows an approximate genealogy, including Disney’s “Principled” shader [Bur12], Allegorithmic’s PBR shading model [McD18], and most recently Autodesk Standard Surface [Geo+19] and Adobe Standard Material [KHE21]. These most recent models have been found to be generally useful and practical for media and entertainment, so we based OpenPBR on them.

There are several advantages to defining such a general-purpose uber-shader model, including:

- *Standardization*: If the model is widely used to define material properties, it facilitates the exchange of assets between different facilities and applications. It is also useful for artists to be presented with a familiar common interface and parameters, which helps with user onboarding and education. Building such a complicated model from scratch can be quite challenging and it is obviously convenient to provide a standardized model that can be used as a starting point.
- *Simplicity*: The model provides a fairly simple interface (of sliders and color pickers) that is sufficient for most typical use cases. This can be more convenient than a fully general material system. For the more specialized use cases it does not cover (for example very high-end skin, hair, cloth or volume shading), one may need to use a renderer-specific shader, or build a bespoke shading network.
- *Predictability*: The user-specified values and colors are designed to produce a reasonably intuitive resulting appearance, with linear behavior, while avoiding unexpected color shifts.
- *Plausibility*: The model is defined in terms of physical light transport, so that its appearance is plausibly close to the reality of a physical material and energy conserving. This physically plausible appearance should be the default, so the user does not have to work hard to achieve it. However, in some well-defined cases the user may want to explicitly “break” physics, to achieve some useful visual effects.

Another positive outcome of defining a practical general-purpose model is the assurance that the existing standard general material frameworks [Gri+10; Ket+15; SS16; HD24] have the required capabilities to implement it, and if not, arguing for and ensuring their adoption. In several cases, the effort has stimulated research into refinements of existing models, and generally helped to identify areas where further research on the underlying BSDF models is warranted.

A key difference between OpenPBR and most previous models mentioned is that in OpenPBR the material is specified as a particular physical structure consisting of layered slabs of material with specified BSDFs. The ground-truth appearance of the model is then defined as the result of light transport through this structure, apart from some well-defined edge cases where physics is broken, as noted. For implementers, this then makes

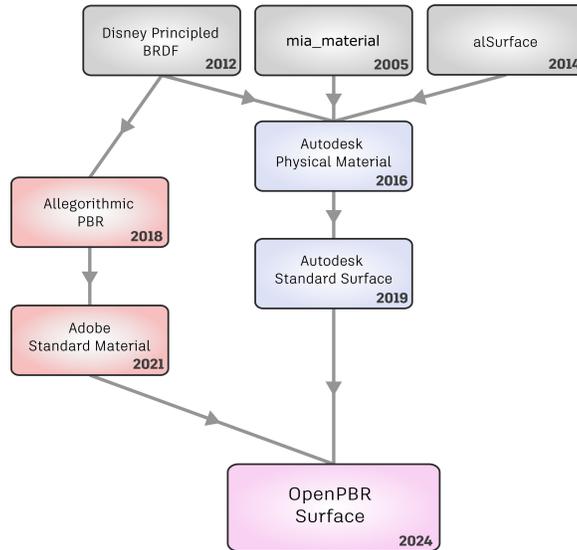


Figure 2: The lineage of shading models leading to OpenPBR Surface.

the target appearance unambiguous so that there is clarity about how it should look, even if it may not always be practical to achieve this except approximately. Each implementation is free to make approximations (e.g., for modeling the layering) as needed based on standard theory, according to the constraints imposed by their use case (e.g., real-time rendering).

This document is a technical companion to the OpenPBR specification. It provides an in-depth explanation of the physical and mathematical foundations, implementation guidance, and rationale behind the OpenPBR layered material model. Unlike the official OpenPBR specification, which is a concise reference for the model’s structure and parameters, this document covers the underlying theory, practical approximations, parameterization pitfalls, and implementation tips for production renderers. It also discusses edge cases, future improvements, and gives more detailed derivations and example code.

In Section 2, we give a brief overview of the model structure. We describe the [layering](#) and [microfacet](#) formalism used to define the model, the [model structure](#) specified via this formalism, and the user-facing [parameterization](#). In Section 3, we proceed to describe the model structure in detail, from the bottom up, starting with the base substrate (consisting of four slabs: [metal](#), [translucent base](#), [subsurface](#), and [glossy-diffuse](#)). We also discuss the [emission](#) properties of the base substrate, and two upcoming improvements still being actively worked on: [hazy-specular](#) and [retro-reflection](#). In Section 4, we discuss the [thin-film iridescence](#) model, and in Sections 5, 6 discuss the [coat](#) and [fuzz](#) layers, respectively. Finally, the special-case [thin-walled](#) mode is covered in Section 7.

The current specification is available at <https://github.com/AcademySoftwareFoundation/OpenPBR>, including the reference MaterialX graph. These notes pertain specifically to the OpenPBR 1.1 model, which is the current version as of August 2025. The model is still evolving, and we expect to add new features in future releases, so please check the repository for the latest version. We also encourage those interested in contributing to the development of the model to join the discussion on the OpenPBR GitHub repository and in our ASWF Slack channel.

2 AN OVERVIEW OF THE OPENPBR MODEL

The specific form of material structure in OpenPBR is a distillation of the aforementioned models. See Figure 3 for a schematic of the layered structure:

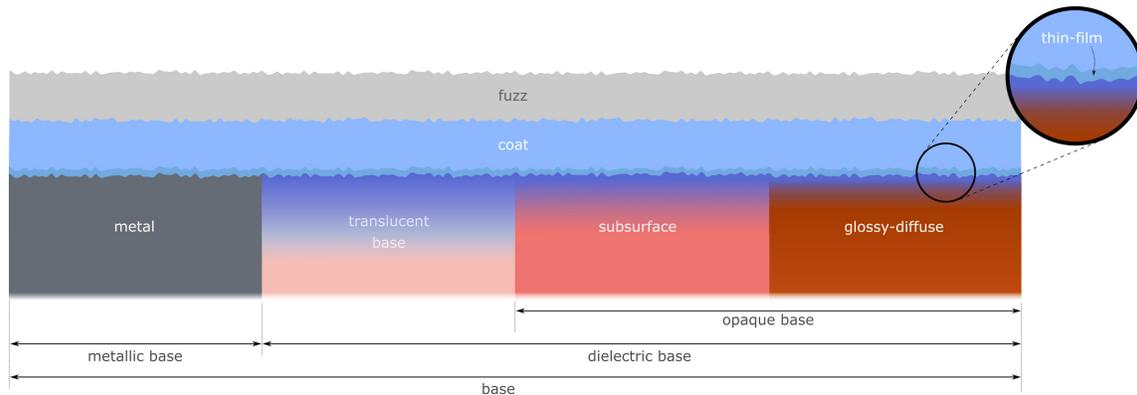


Figure 3: Schematic of the layer structure.

In outline the material consists of:

- A **base substrate** made of a mixture of **metal** or **dielectric**. The interface (dielectric or metal) of this base layer produces the primary specular reflection lobe. The dielectric base represents either of three components, which can be statistically mixed:
 - **Glossy-diffuse**: Dielectric with opaque internal media (e.g., wood, granite, concrete, cardboard, and wall paint).
 - **Subsurface**: Dielectric with dense highly scattering internal media (e.g., plastic, marble, skin, vegetation, and food).
 - **Translucent base**: Dielectric with translucent internal media (e.g., glass, crystals, and liquids).
- **Thin-film**: A layer of optically thin dielectric applied to the base microfacet surface, may be applied optionally to generate iridescence via wave interference (for soap bubbles, oily surfaces, tempered metals, etc.).
- **Coat**: On top of the base substrate sits a further optional layer of dielectric acting as a coating, which may have an absorbing medium. The dielectric interface of this coat layer provides a secondary specular lobe.
- **Fuzz**: An optional layer representing the reflection from micro-fibers (such as fine hair, peach fuzz, textile strands, and dust grains) on top of everything else.

We now describe in more detail the formalism we used to define the structure technically.

2.1 Model formalism

To define the aforementioned layered material structure in detail, we introduced a rather simple formalism that describes an abstract physical material built out of slabs, which while simple is powerful enough for our purposes. As noted, OpenPBR is not a general framework for building material models, but we found it helpful to write down the detailed assumptions in the formalism, which we will now describe.

While writing down this formalism is arguably not strictly necessary, it does help to clarify the structure of the model and how it is intended to be implemented in general material frameworks, and it also provides a common language for implementers to discuss the model. It was helpful to resolve potential ambiguities in the intended physical structure and meaning of the parameters, and to ensure that the model is well-defined in terms of light transport through the structure. Our model was designed explicitly to be compatible and consistent with the MaterialX framework [SS16] (indeed, we contributed various changes to MaterialX to ensure compatibility), so the formalism we describe is therefore rather similar to what is provided in MaterialX.

Slabs We conceptualize the material as consisting of slabs of material which are composed by layering. In describing the model structure, we are working at a “mesoscopic” scale where the layers are considered to be thick enough that geometrical optics is valid, but thin enough they are not explicitly visible in the render. This scale is also considered to be small relative to the scale of variations of the material properties, so we can consider the slabs to be locally homogeneous.

Each slab consists of a homogeneous dielectric or conducting medium V with conceptually unbounded extent horizontally and a finite vertical extent, bounded above by a surface with a given interface BSDF f . The BSDF may not necessarily correspond exactly to that of a physical dielectric or conductor (for example, to be able to say that the BSDF is simply the Lambert or Oren–Nayar model, for example). In a physical slab, the interface BSDF is just what happens where the slab of dielectric or conductor ends, so should be self-consistent with it (although we don’t enforce that in our formalism). We denote a general slab, therefore, as follows:

$$S = \text{Slab}(f, V), \tag{1}$$

where f is the interface BSDF and V is the medium VDF (as defined below).

A slab does not itself specify anything about other slabs in relation to itself (e.g., its substrate slab or overlying slab). The adjacent medium above and below the slab will depend on where it sits in the eventual layer structure. The ambient dielectric medium at the very top of the entire structure (and bottom if thin-walled) is also assumed to be given and unspecified by the model. If the renderer keeps track of the dielectric medium in which the surface is embedded (via a scheme such as “nested dielectrics” [SB02]) – which may be the interior dielectric bulk of some transparent object in the scene, such as a piece of glass or body of water – then the surrounding ambient medium is a dielectric whose IOR we denote n_{ambient} . Alternatively, if dielectric medium tracking is not performed, then n_{ambient} can be assumed to be 1 corresponding to air or vacuum.

BSDFs Each slab has a top interface whose BSDF is a function of input and output directions $f(\omega_i, \omega_o)$.¹ Physically, the BSDF should be energy conserving and reciprocal.

Energy conservation of a BRDF $f(\omega_i, \omega_o)$ amounts to the requirement that the *directional reflectance* (or *directional albedo*) $E(\omega_o) \leq 1$ for all output ray directions, where $E(\omega_o)$ is defined as the integral of the BRDF over

¹Where, following the usual convention, these both point away from the surface vertex, with ω_o in the direction of the outgoing light and ω_i opposite to the direction of the incident light.

directions in the same hemisphere \mathcal{H}_+ as ω_o with projected solid angle measure

$$E(\omega_o) \equiv \int_{\mathcal{H}_+} f(\omega_i, \omega_o) d\omega_i^\perp . \quad (2)$$

More generally, for transmissive BSDFs there is a similar constraint $E(\omega_o) + T(\omega_o) \leq 1$, where the *directional transmittance* $T(\omega_o)$ is defined as the integral of the BSDF over the opposite hemisphere \mathcal{H}_- to ω_o . All BSDFs used are energy conserving in this sense. Additionally, a BSDF that does not *dissipate* any energy, such as a dielectric interface or a diffuse surface with a white albedo, is said to be *energy preserving* and satisfies $E(\omega_o) + T(\omega_o) = 1$. (A technicality to be noted is that in order for the stated energy preservation to hold, the light transport has to be done in terms of the so-called “basic radiance” L/n^2 , where n is the local index of refraction. This accounts for the squeezing of light rays as they enter a medium with a higher index of refraction. This is accounted for in standard derivations of BSDFs and implementation of light transport [Wal+07; PJH16].)

Reciprocity is the requirement that the BSDF is symmetric under interchange of the arguments, i.e., $\omega_i \leftrightarrow \omega_o$.² In some cases this is relaxed and a non-reciprocal BSDF or formula is used for simplicity or efficiency, as in practice it does not lead to serious issues (particularly if only doing unidirectional path tracing).

If the bounding BSDFs of a slab are completely non-transmissive and opaque (e.g., metallic or diffuse), the internal medium is irrelevant for light transport purposes. For convenience, it can be omitted:

$$S_{\text{opaque}} = \text{Slab}(f) . \quad (3)$$

VDFs If the slab is non-opaque (i.e., translucent) then the interior medium of the slab is taken to be a homogeneous dielectric that optionally contains a specified homogeneous volumetric medium (or “VDF”) V . By VDF we mean the set of quantities (which, in general, are spatially varying fields) that define a volumetric optical medium, i.e., the absorption and scattering coefficients (alternatively, extinction coefficient and scattering albedo), phase function, and the index of refraction (IOR) and dispersion of the embedding dielectric medium (grouped with the VDF as “bulk” properties of the slab). A semi-infinite slab at the bottom of the material structure is denoted with a bulk medium V^∞ , which has no bottom interface:

$$S_{\text{bulk}} = \text{Slab}(f, V^\infty) . \quad (4)$$

We assume that the renderer is aware that the volumetric properties may vary from point to point, inherited from the surface parameters, and how the fields are “filled in” in the space surrounding the surfaces is a matter for implementations to deal with.

Given constituent slabs, we then build a more complex composite material by “vertically” layering and “horizontally” mixing slabs. Figure 4 shows a schematic of the slab formalism, with the basic operations on slabs.

Vertical layering The layer operation generates a composite material by depositing a slab $S_{\text{coat}} = \text{Slab}(f_{\text{coat}}, V_{\text{coat}})$ on top of another substrate slab $S_{\text{sub}} = \text{Slab}(f_{\text{sub}}, V_{\text{sub}})$ or composite material, conceptually bonding the base of S_{coat} ’s dielectric medium to the surface of S_{sub} , so that V_{coat} becomes the adjacent medium to the top interface of S_{sub} . The physical act of bringing the two independent slabs S_{sub} and S_{coat} together and bonding them produces a new composite material L that is “vertically” heterogeneous on the mesoscopic scale, denoted by

$$L = \text{layer}(S_{\text{sub}}, S_{\text{coat}}) . \quad (5)$$

²Technically, the reciprocity condition also involves the index of refraction [Vea98].

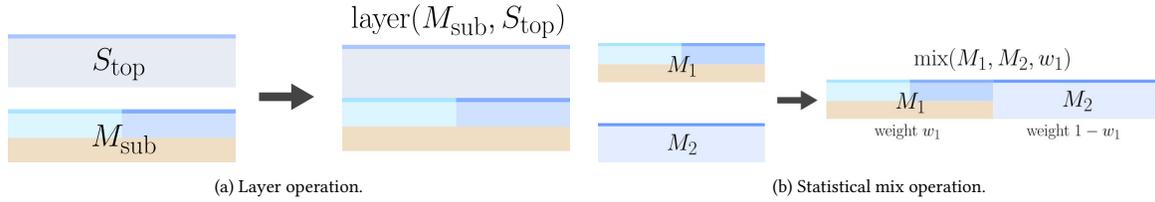


Figure 4: Basic operations on slabs in the OpenPBR slab formalism.

Figure 5 shows an example of how a **coat** layer could be added to a skin material to model effects such as water or blood stains. The partially present coat adds strong Fresnel reflection at grazing angles, and modifies the look of the underlying skin due to absorption, darkening, and roughening effects.

In OpenPBR, we are mostly concerned with defining the layer structure rather than requiring any particular method be used to solve for the light transport, the exact solution of which is supposed to correspond to the desired ground truth. However, for practical purposes, it is important to give some suggested approximate schemes, at least as a starting point. In practice, even rather crude approximations can be acceptable if they yield plausible results. There is a large body of work in the graphics literature on modeling and rendering of layered material structures [HK93; APS00; KS01; WW07; Jak+14a; GHZ18; ZJ18; Bel18]. Accurately solving for the light transport through such layered structures is a complex problem, which in general requires solving the radiative transfer equation (RTE) accounting for the scattering at the interfaces as well as inside the internal volumetric media. This is a non-trivial task that in general requires numerical methods such as Monte Carlo integration, or solving the RTE via various approximations such as diffusion theory or discrete ordinates methods [PJH16].

For media and entertainment applications, however, we are often interested in a practical approximation which need not be very accurate, but must be fast to compute while still producing visually plausible results. In this context, the most common approach is to use a simplified model that approximates the light transport through the layered structure by combining the BSDFs of the individual layers in a linear combination, with various heuristics to account for the effects of the layering structure on the weights and parameters of the BSDFs in this combination.

In this spirit, the most simple representation of a layered configuration amounts to simply making a linear combination of the interface BSDFs f_{sub} and f_{coat} . A common approach is the so-called *albedo-scaling* approximation

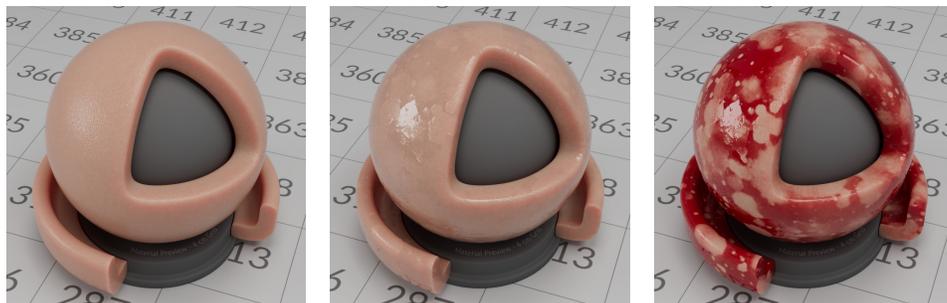


Figure 5: A simple example of layering: adding a coat layer to skin in order to model wet or bloody skin.

[SS16], where the total BSDF of this layered configuration is given by summing f_{coat} and f_{sub} , with the substrate lobe weighted by a factor depending on the directional reflectance of the coat, which is designed to ensure that the resulting BSDF is energy conserving:

$$f_{\text{layer}}(\omega_i, \omega_o) = f_{\text{coat}}(\omega_i, \omega_o) + (1 - E_{\text{coat}}(\omega_o)) f_{\text{sub}}(\omega_i, \omega_o), \quad (6)$$

where the directional albedo integral $E_{\text{coat}}(\omega_o)$ can be either precomputed and tabulated, or computed on the fly via [Monte Carlo](#). This form ensures that if the directional albedos of the coat and substrate BSDFs are energy conserving (i.e., $E_{\text{coat}}(\omega_o) \leq 1$, $E_{\text{sub}}(\omega_o) \leq 1$), then the combined BSDF is also energy conserving, since

$$E_{\text{layer}}(\omega_o) = E_{\text{coat}}(\omega_o) + (1 - E_{\text{coat}}(\omega_o)) E_{\text{sub}}(\omega_o) \leq 1. \quad (7)$$

It also ensures that if the substrate BSDF perfectly preserves energy (i.e., $E_{\text{sub}}(\omega_o) = 1$) then the layer BSDF does also, ensuring that a “white furnace” test would pass. It ignores the physical requirement of reciprocity but captures the essential view dependence of the coat and substrate BSDFs, and is a good approximation for many practical cases. The Autodesk Standard Surface model [Geo+19], for example, uses this formulation for its layering.

This albedo-scaling approximation is somewhat limited as it does not explicitly take into account the effect of multiple light bounces back and forth between the interfaces, or absorption and scattering in the volumetric medium of the coat V_{coat} . In general, of course, the resulting BSDF lobe shape will not be a simple linear combination of the interface BSDFs. However, to some extent, these effects can be adequately accounted for within the albedo-scaling model by making various approximations. For example, the effect of the volumetric transmittance through the coat in the incident and output directions could be approximately modeled as

$$f_{\text{layer}}(\omega_i, \omega_o) = f_{\text{coat}}(\omega_i, \omega_o) + T_{\text{coat}}^2(\omega_i, \omega_o) (1 - E_{\text{coat}}(\omega_o)) f_{\text{sub}}(\omega_i, \omega_o), \quad (8)$$

where $T_{\text{coat}}^2(\omega_i, \omega_o)$ accounts for the total volumetric absorption of the coat along the input and output rays. Similarly, if the coat is rough, it will effectively also roughen the substrate BSDF lobe, which can be approximately accounted for via various heuristics.

The layering system also must handle the different physical effects of the layering for rays exiting and entering the surface. For example, consider the case of a glass object, with a coat and fuzz (see Figure 6). Rays entering

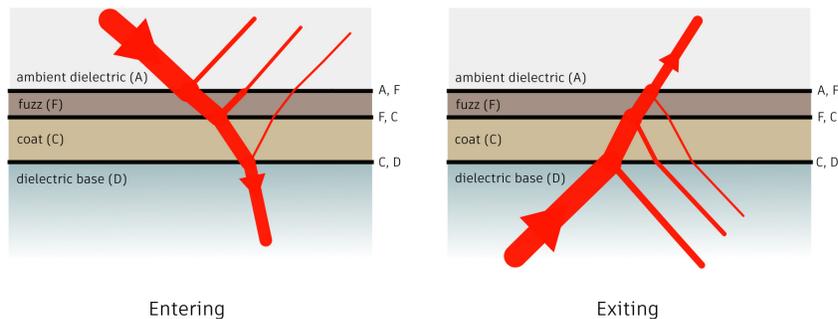


Figure 6: Difference between entering and exiting rays.



Figure 7: Varying coat mix (or presence) weight w_{coat} from 0 to 1, the surface transitions smoothly from an uncoated state to fully coated, with a partially present coat in between. Note that the [darkening](#) effect of the coat becomes more prominent as the presence weight increases.

from the exterior (i.e., the ambient dielectric medium) will enter through the fuzz, then the coat, then transmit into the base glass. While rays that hit the surface from the interior of the glass (having refracted into the glass at some earlier point in the path) instead hit the bottom side of the coat, then the fuzz, then transmit into the ambient medium.

For rays entering from the top (ambient medium) and exiting through the dielectric base, the fuzz reflection is un-tinted by the coat absorption, the coat reflection is dimmed and roughened by the fuzz, and the dielectric reflection and transmission are both dimmed and roughened by both the coat and the fuzz. While for rays entering from the internal dielectric and exiting from the top, the fuzz reflection (viewed from inside) will now be tinted by the coat absorption, and the reflection from the coat “top” interface (with the fuzz) is not dimmed or roughened by the fuzz. This reflection is also an internal one, so it has a different Fresnel factor. The dielectric transmission is dimmed and roughened by the coat and fuzz, while the reflection is unaffected (except for being an internal reflection as well).

A related issue is how we deal with rays that are exiting from the surface (having entered the base dielectric elsewhere) when the surface is locally opaque (e.g., metallic). Arguably one can make sense of this by assuming that the metal must be a thin sheet of foil covering the interior dielectric, so we would reflect from this foil back into the interior. This would allow, for instance, the rendering of bottles with metallic labels, with the label visible through the glass (without the need for modeling a separate label). In general, however, the interior properties are (currently) ill-defined if different parts of the surface specify different base interiors (i.e., metal, diffuse, subsurface, volume, or a mixture thereof). For the moment it is implementation-dependent as to how this is handled.

Horizontal mixing The surface properties and layer structure of a material obviously vary depending on the position, often with large patches of one type of material abruptly changing to another type. In the transition regions, it is often useful to be able to smoothly interpolate between the two materials, so that the transition is not abrupt and does not produce aliasing artifacts.

This is achieved with a *horizontal mixing* operation which models a statistical blend between two materials at the mesoscopic level. The physical picture is that the mesosurface consists of randomly distributed patches of each material in proportion to the blend weight, which thus exhibits heterogeneity in the “horizontal” direction. The mix operation of two slabs S_0 and S_1 with weight w_1 , generating a new “horizontally” heterogeneous composite material M , is denoted:

$$M = \text{mix}(S_0, S_1, w_1) \quad (9)$$

where $M \rightarrow S_0$ as $w_1 \rightarrow 0$, and $M \rightarrow S_1$ as $w_1 \rightarrow 1$.

The mix operation is also used to describe the probability that a particular layer structure exists in the material at a particular point. For example, a coating S_{coat} may cover the substrate layers S_{sub} but be intermittent and applied only to some fraction w_{coat} (the coat *presence* or *coverage* weight) of the substrate (Figure 7). This would be specified as

$$\mathbf{mix}(S_{\text{sub}}, \mathbf{layer}(S_{\text{sub}}, S_{\text{coat}}, w_{\text{coat}})) . \quad (10)$$

For convenience, this may also be written more concisely as a weighted layer operator which covers a given fraction of the substrate with the coat layer:

$$\mathbf{layer}(S_{\text{sub}}, S_{\text{coat}}, w_{\text{coat}}) . \quad (11)$$

This physical picture of the mix operation becomes somewhat unrealistic in some cases where the bottom-most bulk materials being blended are not obviously consistent (e.g., blending dielectric and metallic bulks), but in such cases it is understood that the implementation should do the best it can to make sense of the physics. For example, the metal bulk could be considered to actually be surface metallic flakes on top of a single, consistent dielectric.

The implementation of the mix operation in a renderer depends on how the material model is approximated. Typically the model will be reduced to an effective BSDF per slab being mixed, in which case the mix can be implemented simply as a linear blend of those BSDFs. That is, if slabs S_0, S_1 have BSDFs f_0, f_1 respectively, then the BSDF of $\mathbf{mix}(S_0, S_1, w_1)$ can be mapped to³

$$f_{\text{mix}} = (1 - w_1)f_0 + w_1f_1 = \text{lerp}(f_0, f_1, w_1) . \quad (12)$$

In the case of the weighted layer operation, if the BSDF of the substrate is f_{sub} then $\mathbf{layer}(S_{\text{sub}}, S_{\text{coat}}, w_{\text{coat}})$ maps to

$$f_{\text{weighted-layer}} = (1 - w_{\text{coat}})f_{\text{sub}} + w_{\text{coat}}f_{\text{layer}} , \quad (13)$$

where f_{layer} is the BSDF corresponding to $\mathbf{layer}(S_{\text{sub}}, S_{\text{coat}})$. If we use the non-reciprocal albedo-scaling approach described in the Layering section (Equation 8), then f_{layer} can be expressed as

$$f_{\text{layer}} = f_{\text{coat}} + T_{\text{coat}}^2 (1 - E_{\text{coat}}) f_{\text{sub}} , \quad (14)$$

where f_{coat} is the BSDF of the coat and T_{coat}^2 is its transmittance (accounting for the entry and exit of the path, assuming normal incidence for simplicity). Thus combining these gives

$$f_{\text{weighted-layer}} = w_{\text{coat}}f_{\text{coat}} + \text{lerp}(1, T_{\text{coat}}^2(1 - E_{\text{coat}}), w_{\text{coat}})f_{\text{sub}} , \quad (15)$$

which, for example, is the formula used in the Autodesk Standard Surface [Geo+19] coat layer.

In this fashion, the abstract mix and layer operations can be mapped to a more computationally convenient approximate representation in terms of a weighted sum of BSDF lobes. In Appendix B, a brief example derivation of the complete form of the OpenPBR model as such a linear mixture model of BSDFs is presented.

³Where $\text{lerp}(a, b, t) \equiv (1 - t)a + tb$.

2.2 Microfacet models

Here we provide general assumptions about the form and parameterization of the BSDFs which describe the interfaces in the model formalism outlined in the previous sections.

The BSDFs $f_{\text{conductor}}$, $f_{\text{dielectric}}$, f_{coat} and f_{diffuse} of the [metal](#), [dielectric](#), [coat](#) and [glossy-diffuse](#) slabs, respectively, are each assumed to be described by a standard *microfacet model*. This is a widely used physical picture [[Hei14](#); [PJH16](#)] in which the surface is assumed to be composed of a heightfield consisting of an ensemble of smooth microfacets of either metal, dielectric or Lambertian material, where the statistical distribution of the normal of these facets, termed the *micronormal*, determines the surface roughness characteristics at the macroscopic scale. (The [fuzz](#) model is distinct and based on a volumetric “microflake” model [[Hei+15](#)].) At the present time, microfacet models are effectively the lowest-level physical description in the CG literature of the surface properties of materials. (There is interesting work on more sophisticated proposed formalisms that, for example, account for wave optical effects [[SY22](#); [Ste+24](#)], but these are not yet widely used in practice.)

A microfacet BRDF has the standard single-scattering form [[Wal+07](#); [PJH16](#)] consisting of a product of the Fresnel factor, the masking-shadowing function, and the Normal Distribution Function (NDF):⁴

$$f(\omega_i, \omega_o) \propto F(\omega_i, h) D(h) G(\omega_i, \omega_o), \quad (16)$$

where $h = (\omega_i + \omega_o) / |\omega_i + \omega_o|$ is the half-vector (i.e., the micronormal), which mirror reflects ω_i into ω_o . For dielectrics there is also a BTDF, i.e., the portion of the BSDF where the input and output directions lie in opposite rather than the same hemispheres, which has a similar form to the BRDF except with a modified half-vector, Fresnel factor, and Jacobian [[Wal+07](#)].

The *Fresnel factor* $F(\omega_i, h)$ is determined by the complex index of refraction (IOR) of the reflecting material of each microfacet (or, more technically, the ratio of this to the exterior IOR), and its form differs depending on whether the material is a dielectric or conductor [[Wal+07](#)].

The *masking-shadowing function* $G(\omega_i, \omega_o)$ accounts for the probability that the input and output directions are occluded by the microsurface. It is usually derived using the Smith model, which determines G given the NDF, and for the GGX NDF Equation 17 the masking-shadowing function then has a well-known form [[Hei14](#)].

The *Normal Distribution Function* (NDF) $D(m)$ describes the relative probability of occurrence of micronormal m on the surface, and thus the roughness characteristics. We assume that the NDF is the so-called GGX distribution, which closely approximates the roughness of real materials (the name GGX derives from “ground glass”, but the formula was originally due to Trowbridge and Reitz [[TR75](#); [Wal+07](#); [Bur12](#); [Hei14](#); [PJH16](#)]), which has the basic form

$$D_{\text{GGX}}(m) \propto \left(1 + \frac{\tan^2 \theta_m}{\alpha^2}\right)^{-2}, \quad (17)$$

where θ_m is the angle between m and the (macroscopic) surface normal, and the parameter α controls the apparent roughness of the microsurface. As $\alpha \rightarrow 0$, the distribution of normals becomes highly peaked around $\theta_m = 0$ so the microsurface is mostly flat, leading to a smooth appearance, while as α grows the microsurface becomes increasingly jagged, leading to a rough appearance.

⁴Also a Jacobian factor not shown here.



Figure 8: Comparison of carbon fiber materials with varying specular roughness anisotropy a . The left image shows the isotropic case ($a = 0.0$), the middle image shows moderate anisotropy ($a = 0.5$), and the right image shows high anisotropy ($a = 1.0$).

In practice, we restrict to the range $\alpha \in [0, 1]$, as $\alpha > 1$ does not produce a plausible rough appearance. Following Disney’s “Principled” shader [Bur12], we set (in the isotropic case)

$$\alpha = r^2, \quad (18)$$

where $r \in [0, 1]$ is the user-facing *roughness*, as this produces a more perceptually linear resulting change in apparent roughness as r is varied.

The single-scattering microfacet BRDF of Equation 16 does not conserve energy, as it neglects to account for multiple scattering between the microfacets. In OpenPBR, we generally consider it important that a material which is supposed to be non-absorbing (e.g., a metal with Fresnel factor close to 1) should conserve energy and pass a white furnace test. Thus an implementation should account for multiple scattering via one of a number of schemes, otherwise the reflection from rough metals and dielectrics will be dimmer and less saturated than it should be. An accurate approach based on the Smith microsurface model is described in Heitz et al. [Hei+16b]. Simpler approximate models are presented in Kulla and Conty [KC17] (which functions by adding compensation lobes to account for the missing energy), and Turquin [Tur19] (which scales the albedo of the lobe to maintain energy preservation at the expense of reciprocity).

Roughness anisotropy In the general case, the roughness is anisotropic, that is the NDF is not circularly symmetric but stretched along some direction in the surface plane, producing an elongation of the specular highlight along that direction. This simulates coherent microscale groove geometry due to processes such as brushing, scratches, or materials with inherent directionality such as carbon fiber (Figure 8). It is assumed that a reference tangent vector field is defined (via `geometry_tangent` and `geometry_coat_tangent`). The reference tangent vector indicates the direction along which the NDF is stretched, meaning the microscale grooves tend to be aligned with the orthogonal bitangent. Note that usually the normal map will be generated assuming some fixed per-vertex reference tangents (computed often from the UVs), but those tangents will not necessarily correspond to the desired anisotropy flow.

In practice, the renderer will have the per-vertex reference normals N and tangents T (from UVs), and normal maps are generated relative to these reference N , T (and orthonormal bitangent B). The anisotropy tangents can then be defined via a “flow map” that specifies a 2d vector relative to the reference B , T . From this, one can

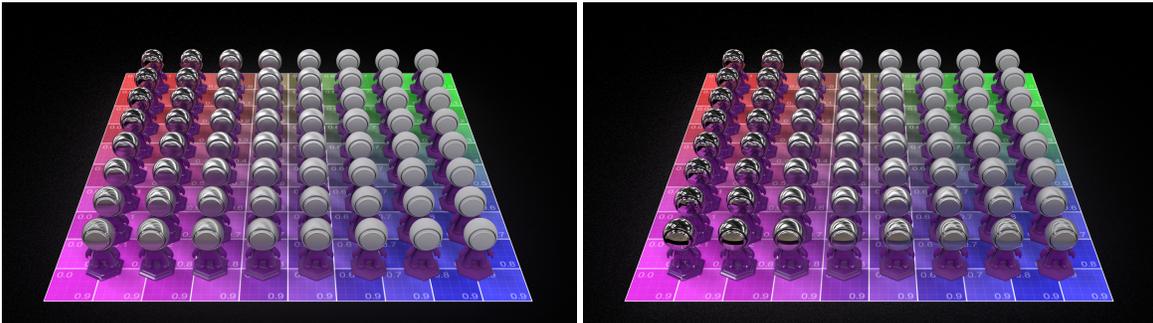


Figure 9: Wedges of roughness and anisotropy, for the Adobe Standard Material model (left), and OpenPBR (right). The roughness goes from 0 on the left to 1 on the right, and the anisotropy goes from 0 at the top, to 1 at the bottom.

then construct a consistent vertex frame with an orthonormal (N', T') shading normal and anisotropy direction. The GGX distribution in the anisotropic case is then parameterized by two separate α -roughnesses α_t and α_b along the tangent and bitangent vectors as follows, given the polar angle ϕ_m of the micronormal m relative to the tangent (counterclockwise about the normal):

$$D_{\text{GGX}}(m) \propto \left(1 + \tan^2 \theta_m \left(\frac{\cos^2 \phi_m}{\alpha_t^2} + \frac{\sin^2 \phi_m}{\alpha_b^2} \right) \right)^{-2}, \quad (19)$$

which reduces to the isotropic form when $\alpha_t = \alpha_b = \alpha$. Efficient techniques for sampling BSDFs employing the anisotropic GGX microfacet model are presented in Heitz [Hei18] and Dupuy and Benyoub [DB23].

The NDF terms α_t and α_b are more conveniently parameterized as the total roughness r (`specular_roughness`) and an anisotropy $a \in [0, 1]$ (`specular_roughness_anisotropy`). In OpenPBR we require the following mapping from r, a to α_t, α_b :

$$\begin{aligned} \alpha_t &= r^2 \sqrt{\frac{2}{1 + (1 - a)^2}} \\ \alpha_b &= (1 - a) \alpha_t. \end{aligned} \quad (20)$$

This formulation satisfies $\alpha_t^2 + \alpha_b^2 = 2\alpha^2$, to preserve the average roughness regardless of the anisotropy. A rationale is that if a renderer doesn't support anisotropy (or if the feature is turned off for performance considerations, such as level of detail), using only the roughness parameter should result in an isotropic specular highlight perceptually close to the original anisotropic one. Figure 9 shows the resulting behavior for various roughness and anisotropy values according to this formulation (right panel), which was considered to behave more intuitively than the parameterization used in Adobe Standard Material (left panel).

Summarizing the NDF parameterization, the **dielectric-base** BSDF $f_{\text{dielectric}}$ and **metallic-base** BRDF $f_{\text{conductor}}$ share the same roughness parameters (`specular_roughness` and `specular_roughness_anisotropy`), while the **coat** BSDF f_{coat} uses an independent set (`coat_roughness` and `coat_roughness_anisotropy`).

2.3 Model structure

In summary, the formal structure consists of the following slabs:

$$\begin{aligned}
S_{\text{ambient-medium}} &= \text{Slab}(\emptyset) \\
S_{\text{fuzz}} &= \text{Slab}(f_{\text{fuzz}}, V_{\text{fuzz}}) && \text{fuzz} \\
S_{\text{coat}} &= \text{Slab}(f_{\text{coat}}, V_{\text{coat}}) && \text{coat} \\
S_{\text{metal}} &= \text{Slab}(f_{\text{conductor}}) && \text{metal} \\
S_{\text{translucent-base}} &= \text{Slab}(f_{\text{dielectric}}, V_{\text{dielectric}}^{\infty}) && \text{translucent-base} \\
S_{\text{subsurface}} &= \text{Slab}(f_{\text{dielectric}}, V_{\text{subsurface}}^{\infty}) && \text{subsurface} \\
S_{\text{gloss}} &= \text{Slab}(f_{\text{dielectric}}, V_{\text{dielectric}}) && \text{glossy-diffuse} \\
S_{\text{diffuse}} &= \text{Slab}(f_{\text{diffuse}}) && \text{glossy-diffuse}
\end{aligned} \tag{21}$$

These are composed to build the material structure, denoted M_{PBR} below, as follows:

$$\begin{aligned}
M_{\text{PBR}} &= \mathbf{mix}(S_{\text{ambient-medium}}, M_{\text{surface}}, \alpha) && \text{where } \alpha = \text{geometry_opacity} \\
M_{\text{surface}} &= \mathbf{layer}(M_{\text{coated-base}}, S_{\text{fuzz}}, F) && \text{where } F = \text{fuzz_weight} \\
M_{\text{coated-base}} &= \mathbf{layer}(M_{\text{base-substrate}}, S_{\text{coat}}, C) && \text{where } C = \text{coat_weight} \\
M_{\text{base-substrate}} &= \mathbf{mix}(M_{\text{dielectric-base}}, S_{\text{metal}}, M) && \text{where } M = \text{base_metalness} \\
M_{\text{dielectric-base}} &= \mathbf{mix}(M_{\text{opaque-base}}, S_{\text{translucent-base}}, T) && \text{where } T = \text{transmission_weight} \\
M_{\text{opaque-base}} &= \mathbf{mix}(M_{\text{glossy-diffuse}}, S_{\text{subsurface}}, S) && \text{where } S = \text{subsurface_weight} \\
M_{\text{glossy-diffuse}} &= \mathbf{layer}(S_{\text{diffuse}}, S_{\text{gloss}})
\end{aligned} \tag{22}$$

While this compactly describes things mathematically, it can be helpful to view the composition as a tree formed from the layer and mix operations, as shown in Figure 10, since it visually connects the equations with the physical layer structure (Figure 3).

The task of the implementation is to compute the BSDF of the material M_{PBR} given this stated structure. Complete conformance to the specification is defined as reproducing all the physical inter-layer light transport effects in the structure thus described. However, enforcing a particular implementation would make the use of the material model impractical for certain classes of renderers and ultimately make it less useful. For this reason, we consider the choice of a specific implementation of the final BSDF to be outside the scope of the specification. In practice, each implementation must decide what level of approximation to use for the light transport within layers, trading off accuracy for efficiency according to its own particular use case.

For convenience and efficiency, at present it is most likely to be mapped to a model consisting of a mixture of BSDF lobes similar to the Autodesk Standard Surface shader [Geo+19] and its representation in MaterialX. An example derivation of such a model is provided in Appendix B. In the official repository we provide a [reference implementation](#) in MaterialX, which has a structure similar to this derivation.

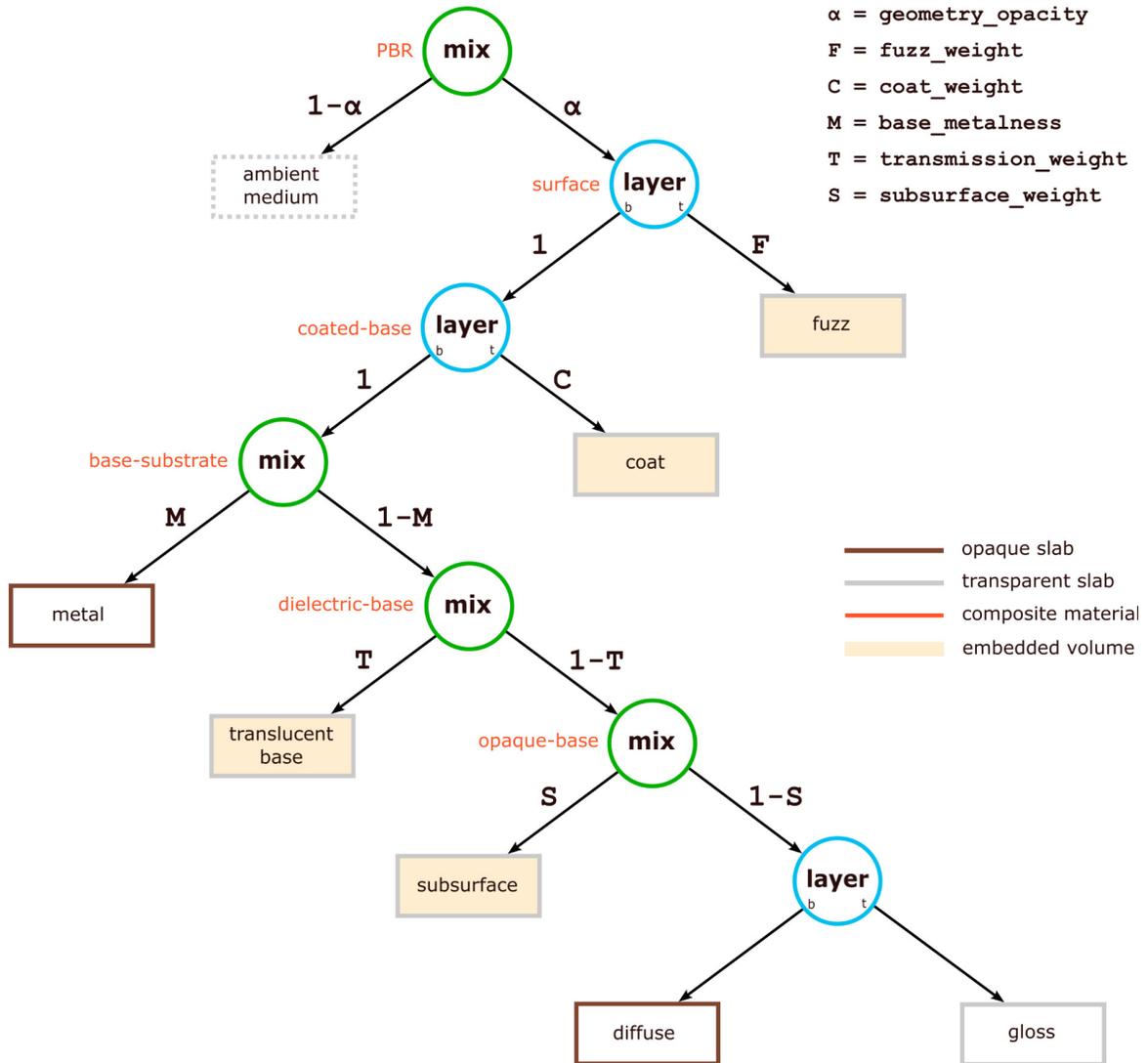


Figure 10: Tree structure of the OpenPBR shading model.

2.4 Geometry

We define a number of parameters which are intended to control geometrical aspects of the surface:

- `geometry_opacity`: the presence weight of the entire surface, thus functioning effectively as a linear transparency “alpha blend”.
- `geometry_normal`, `geometry_coat_normal`: these normal inputs define the perturbation of the shading normal for the base and coat BSDF models, respectively. Separately perturbing the coat normal allows for the appearance of a finite thickness coat on top of the base.
- `geometry_tangent`, `geometry_coat_tangent`: these define the direction of the microfacet anisotropy on the base layer and coat layer, for effects such as brushed metal. Similar to normal mapping, the tangent may be transformed with a 2D vector map to alter the direction of the anisotropy.
- `geometry_thin_walled`: a boolean parameter which indicates whether the surface is thin-walled, i.e., the thickness of the material is negligible compared to the size of the surface. This is described in more detail in [Section 7](#).

The perturbed normals and tangents will typically be specified by input textures (or possibly procedurally). The particular parameterization which maps the contents of the texture to the perturbation of the normal or tangent in the shading frame is not defined in the model itself, as this can be done in multiple ways. It is instead assumed that the material is packaged with metadata that makes this mapping unambiguous.

Opacity Opacity is implemented by defining the surface as a mix of the material bulk with the ambient medium:

$$M_{\text{PBR}} = \text{mix}(S_{\text{ambient-medium}}, M_{\text{surface}}, \alpha) \quad \text{where } \alpha = \text{geometry_opacity}. \quad (23)$$

Note that in the case of a non-thin-walled material, $\alpha < 1$ doesn’t make strict physical sense unless the entire surface is removed, whereas in the [thin-walled](#) mode the opacity has a clear physical interpretation as the presence weight of the wall (or “cutout” areas where $\alpha = 0$) like in the leaf render below.

We generally leave it as an implementation detail for a renderer to determine how connections to light sources be made through the surface. However a very common approximation used by many renderers is “transparent shadows”, where a straight-line connection is made to lights and the contribution of the light determined by the total transmittance along the ray, ignoring any refraction events. We give here a suggested form for this shadow ray transmittance. The computed transmittance should take into account the presence weight of the entire geometry (`geometry_opacity`), and the transmittance through the geometry if present. The latter is the transmittance through the base dielectric only (as the metal is opaque), denoted $T_{\text{dielectric}}$, which should take into account the Fresnel factor of the dielectric interface and the extinction in the volumetric media (in general a statistical mixture of the subsurface medium $V_{\text{subsurface}}^{\infty}$ and translucent-base medium $V_{\text{dielectric}}^{\infty}$). The mix weight of the base dielectric is $w_{\text{dielectric}} = 1 - M$, where $M = \text{base_metalness}$. The total transmittance can thus be approximated as

$$\begin{aligned} T_{\text{pbr}} &= \text{lerp}(1, w_{\text{dielectric}} T_{\text{dielectric}}, \alpha) \\ &= 1 - \alpha(1 - (1 - M)T_{\text{dielectric}}). \end{aligned} \quad (24)$$

2.5 Parameterization

There is a suite of parameters exposed to the artist which effectively define the properties of all the slab BSDFs and volumes, as well as the mixture weights of the tree structure defined in Figure 10. The full parameter set is detailed in Appendix A.

The parameterization follows several guiding principles to ensure ease of use and consistency.

- Each parameter has a unique identifying name. Consistent with Autodesk Standard Surface and Adobe Standard Material, the parameter names are grouped intuitively according to the BSDF lobe whose appearance they most influence, with a corresponding prefix such as `specular` or `subsurface`. It is suggested to use this prefix to group parameters in the user interface. There is an associated, simpler and capitalized label to be used in the user interface.
- Most parameters range preferably between 0 and 1 for ease of texturing and manipulation. In the same spirit, each color is systematically broken down into an albedo and an intensity, with the suffix `color` and `weight`, respectively. This also avoids having to insert a shader between a color texture and its input to modulate its intensity, thus it doesn't require adopting renderers to implement shader networks. Both absolute and suggested useful (i.e., soft) ranges are specified to help ensure a consistent user experience.
- The default value is chosen so that the model behaves physically by default, with the possibility for the user to drift away from a realistic look, as is sometimes needed in production. Units are specified for each parameter to facilitate easy conversions between different formats.
- The specification also tries to avoid parameters that require setting another parameter to see an effect. As the parameterization grew more complex, it was considered important to resist the urge to add modal parameters enabled or disabled by dropdowns, which would unnecessarily complexify user experience.
- We also provided the logic to determine which parameters can currently be disabled (e.g., presented as "grayed out" in the user interface) based on the current state of the model. For example, if the coat weight is set to zero, then the rest of the coat parameters are disabled as they have no effect on the model. This automatic hiding of irrelevant parameters is important to avoid overwhelming the user with too many options, and to ensure that the user can focus on the parameters that are relevant to the current configuration of the material.

We now proceed to describe all of the component slabs and their parameterizations in detail, working from the base of the material towards the top.

3 BASE SUBSTRATE

The bulk at the bottom of the material structure, termed the base substrate, consists of a statistical mix of **metal** and **dielectric** semi-infinite slabs:

$$M_{\text{base-substrate}} = \mathbf{mix}(M_{\text{dielectric-base}}, S_{\text{metal}}, M), \quad (25)$$

where $M = \text{base_metalness}$, which controls the fraction of microsurface area that is metallic. As described in the following section, the **dielectric base** is a mixture of three different components (**translucent base**, **subsurface**, and **glossy-diffuse**). Areas of surface will normally be either fully metallic or fully dielectric, but values of base_metalness between 0 and 1 can be used to simulate smoothly blended transitions between areas of bare metal and areas of dielectric (modeling, for example, opaque rust or paint on top of the metal, as in Figure 11).

For both the metal and dielectric cases, the primary specular lobe shape is controlled by the roughness properties of the surface, parameterized by $\text{specular_roughness}$ and $\text{specular_roughness_anisotropy}$ (as described in Section 2.2).

Figure 12 shows all four component slabs of the base.

3.1 Dielectric base

The dielectric base is assumed to have a surface BSDF described by a rough GGX dielectric **microfacet** model, and a bulk volumetric medium supporting absorption and scattering (whether physically due to the inherent molecular properties of the dielectric as in water or, for instance, a dispersion of embedded particles or flakes as in paint).

However, we distinguish between a separate opaque and **translucent** dielectric bulk, which are placed in a statistical mix:

$$M_{\text{dielectric-base}} = \mathbf{mix}(M_{\text{opaque-base}}, S_{\text{translucent-base}}, T) \quad (26)$$

where $T = \text{transmission_weight}$. This mirrors the usual workflow of artists where they are typically either modeling an opaque surface potentially with some specularity and dense subsurface scattering (such as rock, plastic, skin etc.), or a translucent material with some limited amount of volumetric absorption and scattering



Figure 11: The use of textured base_metalness to achieve a rusty metal look (left), and the equivalent with base_metalness 0 and 1.

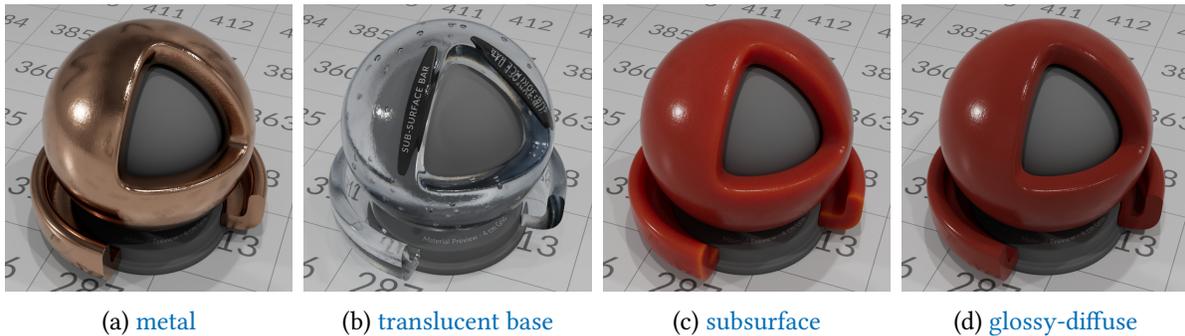


Figure 12: The component slabs of the base substrate (where (b), (c) and (d) collectively constitute the dielectric base).

(such as glass, liquids, organic matter etc.). These use cases require different parameterizations to effectively control, so it is convenient to split them into separate slabs. Both the opaque and translucent dielectric base share the same dielectric interface BSDF $f_{\text{dielectric}}$.

The opaque-base substrate is assumed to be a dielectric with dense subsurface volumetric absorption and scattering, which tends to an idealized **glossy-diffuse** BSDF in the limit of infinite density medium. In some cases a blend of subsurface and completely opaque glossy-diffuse scattering is desired, for example in skin rendering where the diffuse component provides the surface details of the skin (freckles, blemishes, makeup, etc.) and the subsurface component provides the color detail of the underlying veins and tissue. To support this, we define the opaque-base substrate as a statistical mix of **glossy-diffuse** and **subsurface** models:

$$M_{\text{opaque-base}} = \mathbf{mix}(M_{\text{glossy-diffuse}}, S_{\text{subsurface}}, S), \quad (27)$$

where $S = \text{subsurface_weight}$.

Behavior of specular_weight The `specular_ior` parameter controls the index of refraction (IOR) of the base dielectric (but is ignored for metal). The `specular_weight` parameter provides a convenient, texturable linear $[0, 1]$ multiplier of the base dielectric reflectivity at normal incidence via reduction of this IOR below the reference value set by `specular_ior`. As a convenience, we also allow the `specular_weight` to exceed 1, thus increasing the reflectivity via an increase of the IOR above the reference value. When `specular_weight` is 0, the specular reflection disappears entirely as the IOR of the dielectric is then equal to that of the surrounding medium. Equation 30 gives the formula for the applied IOR modulation. We considered multiple different possible interpretations of the `specular_weight`, but found this approach the most appealing as it modulates the Fresnel in a physically correct manner.

A complication though is that we do *not* want the refraction direction to be modified by `specular_weight`, as this control is designed to vary only the reflectivity without disturbing the refraction appearance, a behavior which was specifically requested by artists. Thus the reflection highlight and the refraction direction are effectively *decoupled*, in an unphysical but artistically convenient way (see Figure 13).

The required modification of the Fresnel factors due to `specular_weight` to achieve this decoupling works as follows (see Figure 14 for reference). The physical situation is that the interior of the dielectric has IOR $n_i = \text{specular_ior}$, while the exterior has IOR n_e , which should take into account the presence of the `coat`.



Figure 13: Varying specular_weight with a refractive material (left 0, center 0.5, right 1).

The interior to exterior IOR ratio is $\eta = n_i/n_e$. Incident rays are either entering the base dielectric from the exterior (or coat), or exiting the base dielectric from the interior. For incident rays, the physical reflection Fresnel factors R_+, R_- (for entering and exiting incident rays, respectively) are given by $F(\mu_i, \eta_{ti})$, where η_{ti} is the IOR ratio between the transmitted and incident hemispheres (i.e., $\eta_{ti} = \eta$ for rays incident from the exterior, and $\eta_{ti} = 1/\eta$ for rays incident from the interior). The corresponding transmission Fresnel factors are $T_+ = 1 - R_+$ and $T_- = 1 - R_-$. The dielectric Fresnel reflection factor at normal incidence is given by (for both entering and exiting rays)

$$F_0 = \left| \frac{1 - \eta_{ti}}{1 + \eta_{ti}} \right|^2. \quad (28)$$

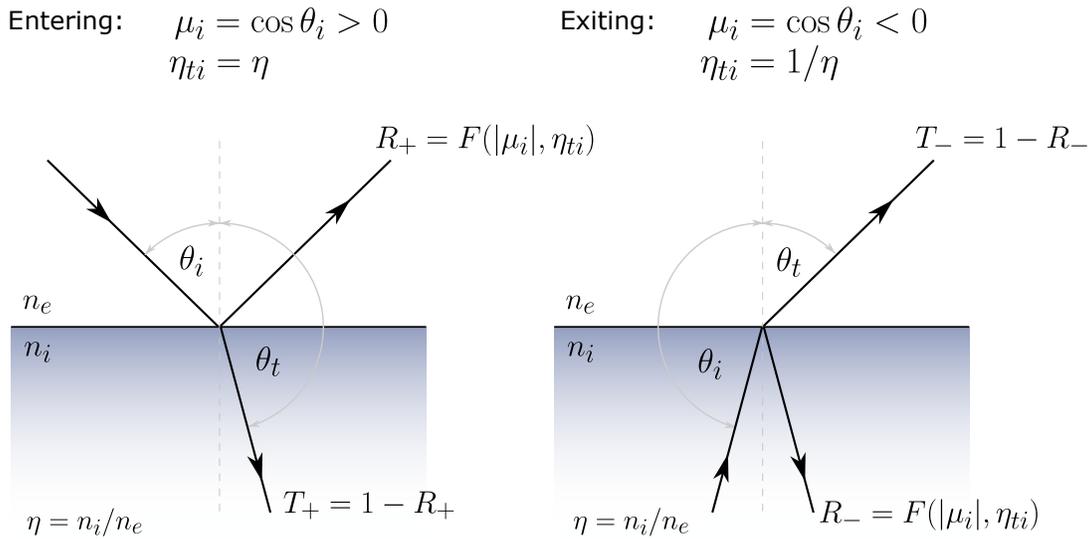


Figure 14: Fresnel factors for rays entering (left) and exiting (right) the base dielectric.

Listing 1: IOR decoupling example code, using `specular_weight`.

```

// Standard dielectric Fresnel function, depending on:
// - mu_i = magnitude of angle cosine of incident ray (mu_i >= 0)
// - eta_ti = (IOR in hemisphere of transmitted light) / (IOR in hemisphere of incident light)
float DielectricFresnel(float mu_i, float eta_ti);

// specular_weight (>= 0) modulates the Fresnel F0 linearly, without disturbing TIR/refraction
float DielectricFresnel(float mu_i, float eta_ti, float specular_weight=1.f)
{
    if (specular_weight == 1.f)
        return DielectricFresnel(mu_i, eta_ti);

    // Compute modified IOR ratio
    float F0 = sqrt((eta_ti - 1.f)/(eta_ti + 1.f));
    float eps = copysignf(min(1.f, sqrtf(specular_weight * F0)), eta_ti - 1.f);
    float eta_ti_prime = (1.f + eps) / max FLT_EPSILON, 1.f - eps);

    if (eta_ti_prime >= 1.f) // (No TIR possible)
        return DielectricFresnel(mu_i, eta_ti_prime);

    // In the possible-TIR case, check for TIR and if not, use the "un-squeezed" Fresnel curve.
    float mu2_t = 1.f - (1.f - sqrt(mu_i))/sqrt(eta_ti);
    if (mu2_t <= 0.f)
        return 1.f; // (TIR occurs)
    return DielectricFresnel(sqrtf(mu2_t), 1.f/eta_ti);
}

```

Listing 1: IOR decoupling example code, using `specular_weight`.

Given the angle cosine μ_i of rays incident to the base, the angle cosine μ_t of the refracted rays follows from the IOR ratio η_{ti} :

$$\mu_t^2 = 1 - (1 - \mu_i^2)/\eta_{ti}^2. \quad (29)$$

This refraction direction remains unmodified by `specular_weight` (only the Fresnel factors change).

Now the modification of the Fresnel factors due to `specular_weight` of this standard physical setup is as follows. The transmitted-incident IOR ratio η_{ti} is modified to η'_{ti} , in such a way as to scale the Fresnel factor at normal incidence by multiplying by $\xi_s = \text{specular_weight}$ (with $\xi_s \in [0, 1]$):

$$\eta'_{ti} = \frac{1 + \epsilon}{1 - \epsilon} \quad \text{with} \quad \epsilon = \text{sgn}(\eta_{ti} - 1) \sqrt{\min(\xi_s F_0, 1)}. \quad (30)$$

For convenience, we also allow $\xi_s = \text{specular_weight}$ to exceed 1 so that the reflectivity is increased above the level set by `specular_ior`. The clamp inside the square root ensures that the scaled reflection coefficient cannot exceed 1.

To deal with transmission, there are two cases to consider:

- If no TIR is possible (i.e., $\eta_{ti}, \eta'_{ti} \geq 1$) then the modulated η'_{ti} is used in the angle-dependent dielectric Fresnel formula $F(\mu_i, \eta'_{ti})$, producing the desired reflectivity modulation at any incident angle cosine μ_i .
- However, if TIR is possible (i.e., $\eta_{ti}, \eta'_{ti} < 1$) then a different procedure is used. If the incident ray direction generates TIR then the reflection is unmodified, with Fresnel factor 1 (and no transmission occurs). However, if the incident ray does *not* undergo TIR, then we apply the physical [Stokes relations](#) (valid for

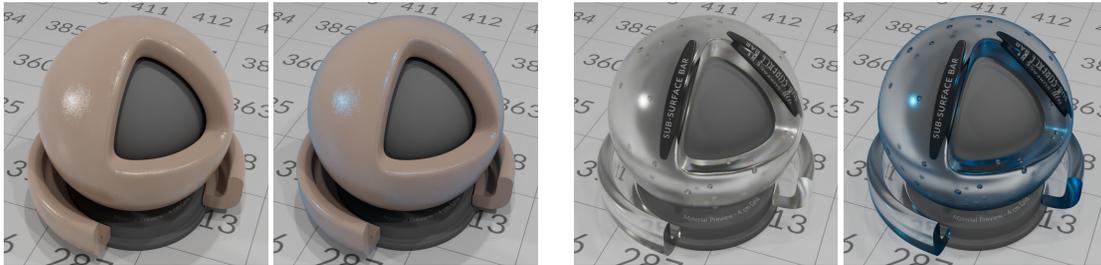


Figure 15: The non-physical behavior of `specular_color` of a dielectric, which tints only the Fresnel reflection highlight (shown for both a non-transmissive and transmissive case).

a smooth dielectric interface):

$$F(\mu_i, \eta_{ti}) = F(\mu_t, 1/\eta_{ti}), \quad (31)$$

which means the ray that enters along the refraction direction *reversed* reflects the same amount as the original reflection. We maintain this relation after the IOR modulation (ensuring reciprocity), setting the reflection Fresnel factor to $F(\mu_t, 1/\eta'_{ti})$, where (since $\eta'_{ti} < 1$) this maps to a Fresnel curve without TIR. Thus the modulated reflection curve for entering and exiting rays has the same shape, except with the curve for rays incident from the higher IOR hemisphere squeezed into a cone (with TIR occurring outside this cone).

In both cases, the associated transmission Fresnel factor for the refracted ray is given by one minus the reflection Fresnel factor. However, as noted, the refracted ray direction is computed using the *unmodified* IOR ratio η_{ti} . Listing 1 gives an example implementation.

For physical consistency, we assume that this modification of the Fresnel factor happens at the microfacet level of the base dielectric interface. Thus, in principle, the multiple scattering lobe should accordingly be calculated taking this into account. In practice, it is a good enough approximation to simply apply the usual microfacet multiple-scattering compensation schemes, with the modulated IOR ratio. As noted below, a further tint factor of `specular_color` is applied to the entire macroscopic dielectric BRDF (for both entering and exiting rays), while the BTDF is unaffected by this tint.

Behavior of `specular_color` The `specular_color` parameter tints the Fresnel factor of $f_{\text{dielectric}}$, but only in the macroscopic BRDF, i.e., the light incident from the upper or lower hemisphere which is reflected back into the same hemisphere (Figure 15). The light transmitted from above or below is assumed to be unaffected. This is technically unphysical if altered from the default white color (as real dielectrics have a Fresnel factor dependent only on the index of refraction), but it can be useful in practice to artificially tint the specular highlight in a way that is minimally disruptive to the rest of the model and harmless as it merely multiplies one scattering mode by a factor, effectively removing some energy via an unspecified mechanism.

This was deliberately chosen to differ from the behavior of the `specular_color` parameter in Autodesk Standard Surface, which had the tinting but the light transmitted through the dielectric was then compensated accordingly to preserve the energy balance, thus generating a complementary color if `specular_color` is not white. This was generally considered a bad behavior, hence we opted to fix it in OpenPBR.

3.1.1 Glossy-diffuse model: EON

The glossy-diffuse slab represents the base dielectric, embedding a semi-infinite bulk of extremely dense scattering material (like the infinitely dense limit of the subsurface). The BRDF of the slab is the combination of a “glossy” specular lobe provided by immediate reflection from the dielectric interface, and a diffuse lobe provided by scattering off the embedded substrate. This models, for example, the reflection from shiny, totally opaque surfaces such as dense plastic, rock, and concrete.

There is a long history in computer graphics of modeling such glossy-diffuse, plastic-like materials via a combination of a specular term and diffuse term, with heuristics to ensure energy conservation (Figure 16). The most well-known example is the Ashikhmin–Shirley model [APS00; PJH16], which uses a microfacet specular term and a diffuse term based on the Lambertian BRDF.

Generally speaking, however, these models did not commit to a particular physical interpretation of the underlying material, which for OpenPBR we would like to specify. Since, at the present time, there isn’t a standard model of this limit of dense scattering material embedded in a dielectric slab, we choose to model it concretely as a layer of dielectric “gloss” on top of an (index-matched) opaque slab with a specified diffuse BRDF:

$$M_{\text{glossy-diffuse}} = \text{layer}(S_{\text{diffuse}}, S_{\text{gloss}}), \quad (32)$$

where S_{gloss} is a thin slab of dielectric with the rough dielectric microfacet BRDF as parameterized in Section 3.1, and the same internal medium as the base dielectric except with zero extinction due to infinitesimal layer thickness:

$$S_{\text{gloss}} = \text{Slab}(f_{\text{dielectric}}, V_{\text{dielectric}}). \quad (33)$$

We opted to describe the glossy-diffuse slab as an explicit layer of dielectric on top of a diffuse base substrate as then it is clear what the base color and roughness mean (i.e., the color and roughness of the Oren–Nayar base).

Since the diffuse base is index-matched with the gloss, Fresnel reflection is only generated from the top interface of the gloss. The opaque substrate slab has a diffuse BRDF lobe:

$$S_{\text{diffuse}} = \text{Slab}(f_{\text{diffuse}}), \quad (34)$$

where f_{diffuse} is based on the classic Oren–Nayar model [ON94], which is a V-cavity based microfacet model in which the microfacets are assumed to be individually Lambertian. The Oren–Nayar model is parameterized by



Figure 16: Wood rendered as glossy-diffuse, composed of diffuse lobe (left), specular lobe (middle), and their normalized sum. Near the specular highlight, the diffuse lobe is automatically reduced, since energy is conserved.

roughness σ and an overall RGB scale factor ρ . The roughness models a “flattening” effect (Figure 17), which is observed in real rough diffuse materials.

During our investigation, we found that the original Oren–Nayar model loses a significant amount of energy when the roughness is high, which is not compatible with the energy-preservation requirements of the OpenPBR model. To address this, we developed a modified version of the Oren–Nayar model (termed the “energy-preserving Oren–Nayar model”, or EON model) that preserves energy at high roughness values, as described by Portsmouth, Kutz, and Hill [PKH25]. This augments the Oren–Nayar model with a simple analytical, reciprocal energy compensation term:

$$f_{\text{diffuse}}(\omega_i, \omega_o) = f_{\text{ON}}(\omega_i, \omega_o) + f_{\text{ON}}^{\text{comp}}(\omega_i, \omega_o). \quad (35)$$

The Oren–Nayar term f_{ON} is given by the form (introduced by Fujii [PKH25])

$$f_{\text{ON}}(\omega_i, \omega_o) = \frac{w_d \rho}{\pi} \left(A(\sigma) + B(\sigma) \frac{s}{t} \right). \quad (36)$$

The roughness parameter $\sigma \in [0, 1]$ is given by `base_diffuse_roughness`. The overall weight is $w_d = \text{base_weight}$. The RGB ρ parameter is determined by the specified `base_color`, as described below. The directional albedo $E_{\text{ON}}(\omega) = w_d \rho \hat{E}_{\text{ON}}(\omega)$ and corresponding *average albedo* $\langle \hat{E}_{\text{ON}} \rangle$ of the Oren–Nayar term can be determined analytically. The energy compensation term $f_{\text{ON}}^{\text{comp}}$ is given in terms of the albedo $\langle \hat{E}_{\text{ON}} \rangle$ by

$$f_{\text{ON}}^{\text{comp}}(\omega_i, \omega_o) = \frac{w_d \rho_{\text{ms}}}{\pi} (1 - \hat{E}_{\text{ON}}(\omega_i))(1 - \hat{E}_{\text{ON}}(\omega_o)), \quad (37)$$

where the factor ρ_{ms} accounts approximately for multiple scattering on the microfacet surface:

$$\rho_{\text{ms}} = \frac{\rho^2 \langle \hat{E}_{\text{ON}} \rangle / (1 - \langle \hat{E}_{\text{ON}} \rangle)}{\pi (1 - \rho(1 - \langle \hat{E}_{\text{ON}} \rangle))}. \quad (38)$$

One can verify that, as $\rho \rightarrow 1$, the total directional albedo of the BRDF of Equation 35, $E_{\text{diffuse}}(\omega) \rightarrow 1$ (in the $w_d = 1$ case), thus the compensation term ensures that the white furnace test passes. Note that in the zero roughness ($\sigma \rightarrow 0$) limit, the energy compensation term vanishes, and the parameter ρ is equal to the albedo of f_{diffuse} . As roughness increases, the albedo of f_{diffuse} becomes slightly more dark and saturated than ρ due to the multiple scattering, which is physically realistic.

Although physically the glossy-diffuse slab as described should exhibit darkening and saturation due to the same physics as the coat darkening (i.e., multiple bounces inside the gloss layer losing energy as they hit the colored base), we wanted to prevent this so that the user sees – to the extent possible – the base color they specified. To define the meaning of the specified color in terms of the underlying base albedo, we write the normal-direction reflectance of the glossy-diffuse slab, $E_{\text{glossy-diffuse}}$ in the general form

$$E_{\text{glossy-diffuse}} = E_{\text{spec}} + E_{\text{diffuse}}, \quad (39)$$

where E_{spec} is the normal-direction reflectance of all energy reflected from the dielectric interface without transmission, and E_{diffuse} is the normal-direction reflectance of all energy transmitted through the interface, scattered off the diffuse medium, and transmitted back out.

We then *define* $C = \text{base_weight} \times \text{base_color}$ to be such that the reflectance of the remaining energy transmitted into the slab, E_{diffuse} in the *zero* `base_roughness` case (i.e., a Lambertian base) and assuming vacuum exterior, is given by

$$E_{\text{diffuse}} = (1 - E_{\text{spec}})C. \quad (40)$$



Figure 17: Comparison of diffuse models at maximum roughness: (left) Lambert, (center) QON, (right) EON. The EON model preserves energy at high roughness, while the QON model loses energy.

In other words, the selected color C parameterizes the fraction of the energy transmitted into the dielectric layer that is subsequently transmitted back out due to reflection from a Lambertian interface. Thus, since $0 \leq C \leq 1$, $E_{\text{glossy-diffuse}} \leq 1$ so energy is always conserved, and if $C = 1$ then $E_{\text{glossy-diffuse}} = 1$, which guarantees that a white C passes the furnace test. According to this definition, C is the observed reflection color (viewed at normal incidence under uniform illumination) in areas where the Fresnel reflection is negligible, and otherwise the observed color is a blend of C with the gray Fresnel reflection that conserves total reflected energy.

Given the required diffuse albedo E_{diffuse} according to the formula above, then in principle the albedo of the diffuse Oren–Nayar lobe $f_{\text{diffuse}}(\omega_i, \omega_o)$ that generates the required E_{diffuse} can be determined. A reasonable, practical (albeit non-reciprocal) approximation to the resulting BRDF of the glossy-diffuse slab that satisfies this requirement is obtained via the non-reciprocal albedo-scaling approximation of Equation 6:

$$f_{\text{glossy-diffuse}}(\omega_i, \omega_o) \approx f_{\text{dielectric}}(\omega_i, \omega_o) + (1 - E_{\text{dielectric}}(\omega_o)) f_{\text{diffuse}}(\omega_i, \omega_o), \quad (41)$$

where the albedo of the Oren–Nayar lobe $f_{\text{diffuse}}(\omega_i, \omega_o)$ must be taken to be $C = \text{base_weight} \times \text{base_color}$ in order to satisfy Equation 40. If a reciprocal formulation is desired, the classic Ashikhmin–Shirley or Kelemen model [APS00; KS01; KC17; KHE21] form is

$$f_{\text{glossy-diffuse}}(\omega_i, \omega_o) \approx f_{\text{dielectric}}(\omega_i, \omega_o) + \mathcal{N}(1 - E_{\text{dielectric}}(\omega_i))(1 - E_{\text{dielectric}}(\omega_o)) f_{\text{diffuse}}(\omega_i, \omega_o), \quad (42)$$

where \mathcal{N} is a normalization factor such that Equation 40 is satisfied. In the case of a zero roughness (i.e., Lambertian) Oren–Nayar base \mathcal{N} can be tabulated in terms of the dielectric IOR and roughness [KHE21], and the required albedo of the Lambertian base is equal to C as in the non-reciprocal albedo-scaling approximation.

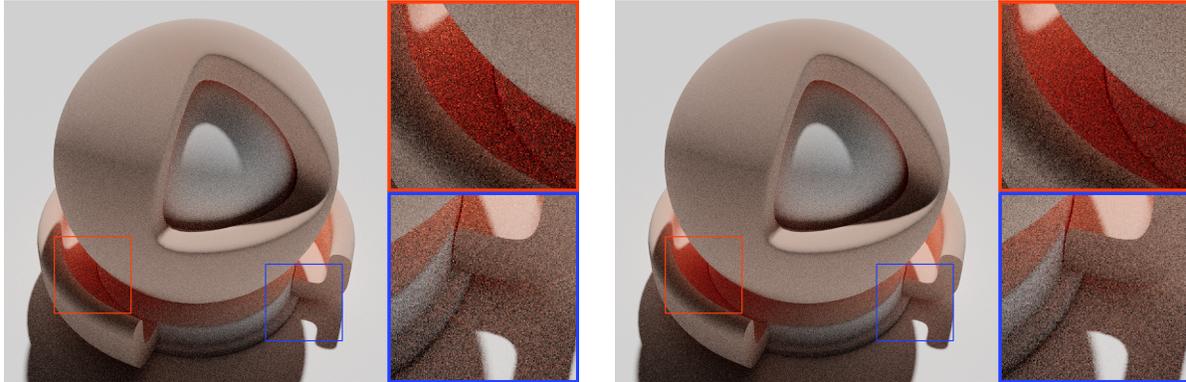


Figure 18: Variance reduction achieved in renders via CLTC importance sampling of the EON model (left panel is cosine sampling, right panel is CLTC sampling).

Extending this to the more general case of a non-Lambertian rough Oren–Nayar base would require adding the roughness dimension to the tabulation, and the required Oren–Nayar albedo will not simply equal C . We leave the specific choice of model and these details to the implementation.

The EON diffuse model also provides more sophisticated importance sampling than standard cosine-weighted sampling typically used for diffuse materials [PKH25]. The sampling is based on *Linearly Transformed Cosine* (LTC) sampling [Hei+16a], with a modification to avoid generating redundant samples under the hemisphere (so-called CLTC sampling). This results in a significant reduction in noise (see Figures 18 and 19).

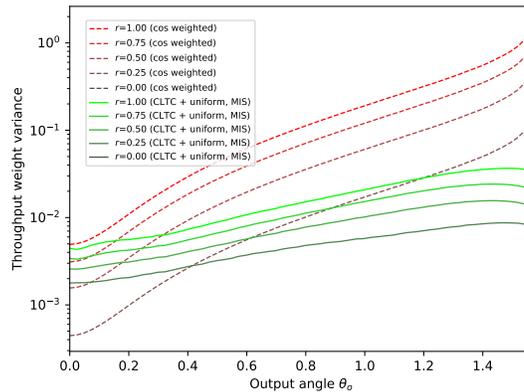


Figure 19: A plot of the variance reduction achieved via CLTC importance sampling of the EON model (compared to traditional cosine-weighted sampling), as a function of the incident angle and roughness. Note that the $r = 0$ curves (for both uniform and CLTC sampling) are simply zero, so don't show up on the log plot (since in this case, both uniform and CLTC sampling degenerate to perfect importance sampling, with constant throughput weight).

3.1.2 Subsurface

In OpenPBR, we define two separate but related volumetric components: subsurface scattering and the transmission volume (or [translucent base](#)). While the underlying physics is the same for both, they are parameterized differently and are intended for different use cases. Essentially, subsurface scattering is designed to make an opaque diffuse material more translucent, while the transmission volume is designed to make a clear transparent object less translucent.

While developing OpenPBR, we originally considered a unified volumetric approach, with a continuum from diffuse to subsurface to transmission, since all of these effects are conceptually different scales of the same underlying physics. However, we abandoned that approach in favor of the multi-component model due to its familiarity for artists and the ability to use a different appropriate parameterization for each component.

As in the cases of the [glossy-diffuse slab](#) and the [translucent base](#), the subsurface is bounded by a dielectric interface with BSDF $f_{\text{dielectric}}$, which generates the primary specular reflection lobe parameterized via the “specular” parameters. Combined with this is the reflection generated by light that is transmitted through the dielectric interface into the underlying embedded subsurface medium, where it scatters around and eventually transmits back out. In this case, the subsurface medium $V_{\text{subsurface}}^{\infty}$ is given a parameterization which is particularly convenient for controlling the volumetric effect of dense subsurface scattering.

Subsurface radius One of the biggest challenges of parameterizing subsurface scattering is specifying the length scale of the effect. An ideal scale parameter would be intuitive, expressive, and physically meaningful, exhibit consistent behavior when varied, and not be prone to artifacts. In practice, however, there are many tradeoffs between these properties, and the optimal choice depends on the use case.

For example, for thin or low-albedo volumes, it might be useful to specify the distance traveled by light along a single ray before it first interacts with the volume, but this is not as straightforward as it might sound, since the distance is probabilistic, and since there are two different types of interactions that can occur: scattering and absorption. For dense or high-albedo volumes, it might be useful to specify the distance traveled along the surface before light re-emerges, however this is even less straightforward, since that distance is also probabilistic and varies strongly with the single-scattering albedo of the volume.

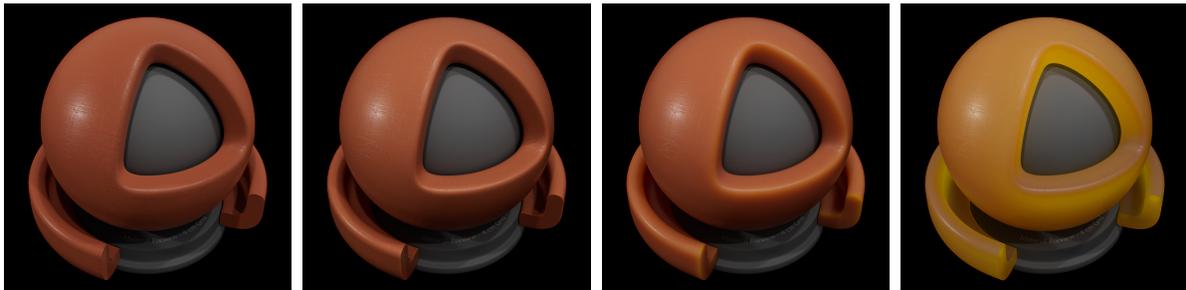


Figure 20: Comparison of diffuse and subsurface scattering with varying radius. The left image shows a diffuse material with orange `base_color`, while the other three images show subsurface scattering with the same orange color `subsurface_color` and a yellow `subsurface_radius_scale`, with `subsurface_radius` of 0.01, 0.1, and 1.0 respectively. At low radius, the reflection color is a good match to the diffuse case, while at higher radii the yellow color becomes more visible.

While designing OpenPBR, we considered and experimented with a number of different options. One of the options we tentatively finalized involved setting the average distance that light visibly travels along the surface for each color channel after all scattering and absorption has occurred. We based this on the theory presented in Christensen [Chr15]. Effectively the user would be controlling the resultant diffusion profile directly (even when simulated using path-traced volumetric scattering). This was conceptually appealing, especially since it aligned well with the perceptual multiple-scattering definition of the subsurface color parameter described below.

However, in practice, it produced inconsistent results and artifacts due to its tight coupling with the albedo. For example, if the user darkened the albedo, light would travel less far on average due to the increased absorption, so the system would have to increase the translucency of the material to compensate. This was already unintuitive with gray albedos (especially textured ones), but it became even more problematic with colored albedos. Because the albedo affects the shape (peak/tail balance) of the diffusion profile, and because the scale compensation acted independently on each color channel, each color channel would end up with a diffusion profile with both a different shape and a different size. This could result in situations where one color channel would dominate at short distances while another dominated at long distances, resulting in unexpected and unsightly hue shifts (e.g., a red material might look cyan in shadow regions). We observed that this hue inversion cannot physically occur in real-world situations where either the extinction or scattering coefficient is roughly uniform across the visible spectrum, so we decided to avoid this approach.

Ultimately, we decided to use a simpler and more pragmatic approach where the subsurface radius parameter simply specifies the extinction mean free path. That is, `subsurface_radius` \times `subsurface_radius_scale` defines, per RGB channel, the *mean free path* (MFP) r – i.e., the average distance that a ray of light travels through the medium before being absorbed or scattered. The corresponding extinction coefficient $\mu_t = 1/r$ controls the apparent density of the medium. In the limit of zero MFP, the medium tends towards infinite density, and approaches the look of an opaque diffuse surface. Being a length, `subsurface_radius` can be any value greater than or equal to zero. For convenience, we make the soft range $[0, 1]$, thus covering common cases such as skin where the MFP is lower than the scene length units. The `subsurface_radius_scale` controls the color-channel dependence of the MFP, and thus this color is visible in the light transmitted through thinner regions of the subsurface volume.

In this approach, the user-provided radius doesn't map to anything directly observable, but it does map directly to an underlying physical property of the medium, leading to a parameterization that is straightforward to describe and implement. Most importantly, this approach produces results that are familiar for artists, predictable, and free of artifacts.

The medium's phase function is parameterized by `subsurface_scatter_anisotropy`, giving the scalar anisotropy $g \in [-1, 1]$. In practice, implementations may want to internally clamp the anisotropy as the phase function becomes ill-defined and the rendering unstable near the -1 and 1 limits. Furthermore, path-traced volumetric rendering can be made more efficient by leveraging similarity theory as described in Section 3.1.2.

Subsurface color For artist directability, it is important that the input parameters produce renders with colors that match to a reasonable approximation. This corresponds to the requirement that the observed color of the reflected light matches `subsurface_color` given all the other relevant model parameters (i.e., the other volumetric parameters of the subsurface such as anisotropy, and the dielectric IOR and roughness). Renderers should try to achieve that, to the extent possible given their constraints. As shown in Figure 20, in the dense limit, the look of a diffuse base with given `base_color` should closely match the look of the subsurface with equal corresponding `subsurface_color`.

To define what it means to say that the observed color “matches” the `subsurface_color`, we need to separate the Fresnel reflection from the reflection due to the subsurface scattering. To make this precise, we use a similar mechanism to the definition of the albedo of the [glossy-diffuse](#) slab. Denoting the directional reflectance along the normal direction by $E_{\text{subsurface}}$, this can be broken into two components:

$$E_{\text{subsurface}} = E_{\text{spec}} + E_{\text{multi-scatter}} . \quad (43)$$

Here, E_{spec} is the normal-direction reflectance of all energy reflected from the dielectric interface *without* macroscopic transmission. While the *multiple-scattering albedo* $E_{\text{multi-scatter}}$ is the normal-direction reflectance of all remaining reflected energy due to (macroscopic) transmission through the interface, multiple scattering in the subsurface medium, and transmission back out (assuming external vacuum). We then *define* the `subsurface_color` C to parameterize, assuming correct physical light transport and semi-infinite homogeneous slab geometry, the fraction of the energy transmitted through the interface that is transmitted back out and observed, i.e.:

$$E_{\text{multi-scatter}} = (1 - E_{\text{spec}}) C . \quad (44)$$

Thus, since $0 \leq C \leq 1$, $E_{\text{subsurface}} \leq 1$, so energy is always conserved, and if $C = 1$ then $E_{\text{subsurface}} = 1$, which guarantees that a white `subsurface_color` passes the furnace test. According to this definition, the `subsurface_color` C is the observed reflection color (viewed at normal incidence under uniform illumination) in areas where the Fresnel reflection is negligible, and otherwise the observed color is a blend of C with the gray Fresnel reflection that conserves total reflected energy.

If the interface is index-matched (i.e., $E_{\text{spec}} = 0$) then $E_{\text{subsurface}} = \text{subsurface_color}$, so in the index-matched case the color `subsurface_color` drives the base (multi-scatter) albedo directly, as one would expect. Since E_{spec} is grey, in the general case the final albedo $E_{\text{subsurface}}$ has the same hue as `subsurface_color`, while satisfying the properties above, so this provides a reasonable match between the desired and resultant observed color given the constraints.

Albedo inversion In principle, given the required `subsurface_color` C , the single-scattering albedo α which generates the required $E_{\text{multi-scatter}}$ can be determined (assuming the given extinction μ_t and anisotropy g). We do not require any particular theoretical formula for $\alpha(C)$ be used, but the closer it is to satisfying Equation 44 the closer it is to the ground truth. Ideally, this formula should take all the properties of the medium into account as well as the dielectric interface. A number of such approximations have been derived in the literature.

A well-known approximation is due to van de Hulst [KC17; dEo22], which assumes an index-matched boundary (i.e., $E_{\text{spec}} = 0$, thus $E_{\text{multi-scatter}} = C$). According to this, the total reflectance (i.e., multi-scatter albedo) C from a semi-infinite isotropic medium with no Fresnel (vacuum boundary) and diffusion illumination is given by

$$C(s) = \frac{(1 - s)(1 - 0.139s)}{1 + 1.17s} . \quad (45)$$

Where

$$s = \sqrt{1 - \alpha^*} \quad (46)$$

and α^* is the single-scatter albedo for the isotropic case. Inverting Equation 45 gives (to less than a percent error)

$$s(C) = 4.09712 + 4.20863C - \sqrt{9.59217 + 41.6808C + 17.7126C^2} . \quad (47)$$

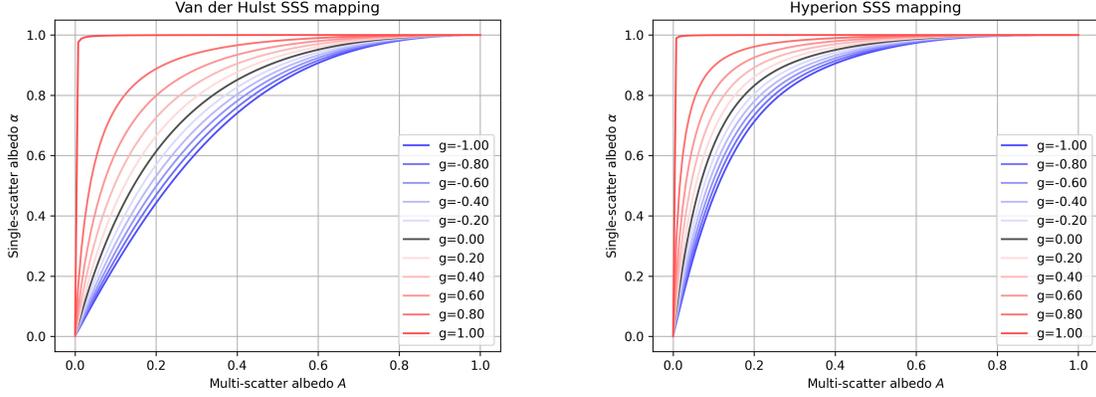


Figure 21: Comparison of the van de Hulst (left) and Hyperion (right) subsurface scattering mappings from desired multi-scatter albedo A to single-scatter albedo α .

This provides the desired single-scatter albedo given multi-scatter albedo C as $\alpha^*(C) = 1 - s^2(C)$. Chiang, Kutz, and Burley [CKB16] found their own fitting function for multi-scatter albedo $\alpha^*(C)$, which was later extended by Burley et al. [Bur+18], taking into account TIR from an interface assumed to have IOR 1.4, producing

$$\alpha^*(C) = 1 - e^{-11.43C + 15.38C^2 - 13.91C^3}. \quad (48)$$

These formulas are necessarily approximate, as for example they neglect certain effects such as the roughness of the dielectric boundary. Various other alternative analytical formulas which attempt to account more accurately for all the relevant effects are presented by d'Eon [dEo22].

To account for anisotropy, the standard approach is to use the following “similarity relation” [ZRB14; Bur+18] between the scattering coefficients in media with different anisotropies g and g^* :

$$\mu_s(1 - g) = \mu_s^*(1 - g^*). \quad (49)$$

The absorption coefficients of the two media are assumed to be equal, i.e., $\mu_a = \mu_a^*$. The extinction coefficient $\mu_t = 1/\tau$ is assumed to set the extinction of the anisotropic medium. The scattering albedo α of the anisotropic medium can then be shown to be (see Appendix C for the derivation)

$$\alpha = \frac{\alpha^*}{1 - g(1 - \alpha^*)}. \quad (50)$$

This formulation for general anisotropy (i.e., scattering albedo given by Equation 50 and Equation 48) is used for example in Arnold’s OpenPBR subsurface implementation.

Figure 21 shows the form of the van de Hulst and Hyperion mappings as a function of anisotropy g . At high anisotropy, a higher single-scatter albedo is required to achieve the same multi-scatter albedo, since light is preferentially scattered in the forward direction so has a higher chance of escaping without scattering again.

Note that the similarity relation described above can also be used to make path-traced volume rendering more efficient by reducing the number of scattering events. This can be done for a volume with anisotropy g by



Figure 22: The subsurface defaults to a `subsurface_radius_scale` of (1, 0.5, 0.25) to approximate the effect of Rayleigh scattering (right), rather than pure white (left).

setting g^* to any value between g and 0 (0 for isotropic scattering), computing μ_s^* using Equation 49, and simply setting $\mu_a^* = \mu_a$ (since the distance traveled along the path and therefore the amount of absorption is essentially unchanged). As proposed by Burley et al. [Bur+18], it is even possible to gradually transition from full anisotropic scattering to reduced isotropic scattering over the first several bounces along a path by setting g^* differently at each bounce, gradually reducing it from g to 0. This can allow an implementation to achieve the best of both worlds of visuals and performance: view-dependent, low-order scattering and efficient high-order scattering.

Rayleigh scattering Note that the default value of `subsurface_radius_scale` is set at (1, 0.5, 0.25) to approximate the effect of Rayleigh scattering (or Tyndall scattering). If we roughly approximate the wavelength corresponding to each of the RGB channels as $\lambda/\text{nm} \approx 650, 550, 450$ and assume the radii scale like the reciprocal of the extinction, then since in Rayleigh scattering the extinction scales like λ^{-4} for light of wavelength λ , the expected relative magnitudes of the radii are $(1, (550/650)^4, (450/650)^4) \approx (1, 0.5, 0.25)$, hence the chosen default. This provides a slightly more realistic default look for the subsurface than resulting from gray radii (Figure 22). We considered a more explicit control for providing a Rayleigh-scattering effect (such as the model from Kutz, Hašan, and Edmondson [KHE21]) but felt that this could always in principle be built of top of the model, and having the default look be reminiscent of Rayleigh scattering was sufficient.

Since the scattered ray may emerge from a different point on the surface than the input point, the material will in general differ there. Thus, on transmitting back out, it will encounter different layer properties. In principle this should be taken into account, but how to model subsurface scattering in the presence of varying subsurface properties is an open question.

3.1.3 Translucent base

As noted earlier, the volume of the base dielectric is parameterized in two ways in OpenPBR, either via [subsurface scattering](#) or using a parameterization more suited to thin volumetric media with visible refraction. We term the latter case the “translucent base” dielectric, whose embedded volume is the “transmission volume” (being parameterized by the transmission parameters).

As for the dielectric slabs of the [glossy-diffuse](#) and [subsurface](#) components, the top interface is described by a rough GGX microfacet BSDF $f_{\text{dielectric}}$ whose “specular” parameters are described in the [Dielectric base](#) section. The bulk of the dielectric, $V_{\text{dielectric}}^{\infty}$ is a volumetric medium supporting absorption and scattering:

$$S_{\text{translucent-base}} = \text{Slab}(f_{\text{dielectric}}, V_{\text{dielectric}}^{\infty}). \quad (51)$$

The index of refraction of $V_{\text{dielectric}}^{\infty}$ is specified by `specular_ior` (as for the entire dielectric base). The volumetric properties of $V_{\text{dielectric}}^{\infty}$ are specified as follows.

The `transmission_depth` λ_T is the distance traveled inside the medium by white light before its color becomes exactly `transmission_color` T by Beer’s law, determining the extinction coefficient of the interior medium μ_t :

$$\mu_t = -\frac{\ln T}{\lambda_T}. \quad (52)$$

Being a length, `transmission_depth` can be any value greater than or equal to zero. For convenience, we make the soft range $[0, 1]$, thus covering common cases. However, when `transmission_depth` λ_T is zero, it is assumed that the interior medium is absent ($\mu_t = 0$) and `transmission_color` is used instead to non-physically tint the dielectric refraction Fresnel factor multiplicatively by a constant amount (ignoring the dielectric energy balance).

The `transmission_scatter` parameter S directly sets the medium scattering coefficient μ_s (as a multiple of the inverse `transmission_depth`):

$$\mu_s = \frac{S}{\lambda_T}. \quad (53)$$

The `transmission_scatter` color thus controls the observed color of the single-scattered light. (Note that in the case `transmission_depth` λ_T is zero, however, the scattering coefficient is ignored.) The absorption coefficient μ_a is then computed as

$$\mu_a = \mu_t - \mu_s. \quad (54)$$

If any component of μ_a is negative, then μ_a is shifted by enough gray to make all of the components positive, i.e. (in an obvious notation):

$$\begin{aligned} &\text{if } \min(\mu_a) < 0 \\ &\mu_a \leftarrow \mu_a - \min(\mu_a). \end{aligned} \quad (55)$$

After this shifting, the final extinction coefficient is given by the new $\mu_a + \mu_s$. This formulation yields volumetric parameters which reproduce reasonably well the independently specified colors of the transmitted and single-scattered light. Finally, the medium phase function anisotropy $g \in [-1, 1]$ is given by `transmission_scatter_anisotropy` (the standard Henyey–Greenstein phase function is assumed). Figure 23 shows the effect of varying the `transmission_scatter_anisotropy` parameter.



Figure 23: The effect of `transmission_scatter_anisotropy`. The left image shows a negative anisotropy of -0.5, the center image shows an isotropic phase function, and the right image shows a positive anisotropy of 0.5.

Dispersion The phenomenon known as *optical dispersion* (i.e., variation of the IOR with wavelength) produces familiar rainbow-like colors when light refracts through dielectrics such as water, glass and diamond. Control over dispersion is grouped with the transmission parameters since this effect is only significant in highly transparent refractive media. Dispersion is parameterized by the *Abbe number* V_d , which is the ratio between differences of refractive indices at short, medium, and long wavelengths as follows:

$$V_d = \frac{n_{\text{medium}} - 1}{n_{\text{short}} - n_{\text{long}}}, \quad (56)$$

where in the standard modern definition $n_{\text{long}} = n_C$, $n_{\text{medium}} = n_d$ and $n_{\text{short}} = n_F$ are the IORs of the material at the wavelengths of the Fraunhofer C, d, and F spectral lines (at $\lambda_C = 656.3$ nm, $\lambda_d = 587.6$ nm, and $\lambda_F = 486.1$ nm, respectively). The amount of dispersion (i.e., angular separation of colors) is roughly proportional to the reciprocal of the Abbe number. The IOR at any wavelength can be determined from the Cauchy empirical formula:

$$n(\lambda) = A + \frac{B}{\lambda^2}. \quad (57)$$

It follows that if the Abbe number V_d and IOR $n_d = n(\lambda_d)$ are given, the coefficients in the Cauchy formula are given by

$$A = n_d - \frac{B}{\lambda_d^2}, \quad B = \frac{n_d - 1}{V_d (\lambda_F^{-2} - \lambda_C^{-2})}. \quad (58)$$

We assume that `specular_ior` (including any modulation via `specular_weight` as in Equation 30) defines $n(\lambda_d)$. Thus the IOR n at any wavelength λ is determined, given V_d . A renderer can use this known $n(\lambda)$ function to model the effect of dispersion, for example by stochastically choosing a wavelength sample and tracing the refracted ray direction according to the corresponding IOR.

However, the Abbe number itself is not very intuitive to work with, since the dispersion effect increases as the Abbe number decreases (zero dispersion occurs at infinite Abbe number). We therefore prefer to use a more artist-friendly parameterization (see Figure 24), where the Abbe number is specified by

$$V_d = \frac{\text{transmission_dispersion_abbe_number}}{\text{transmission_dispersion_scale}}. \quad (59)$$

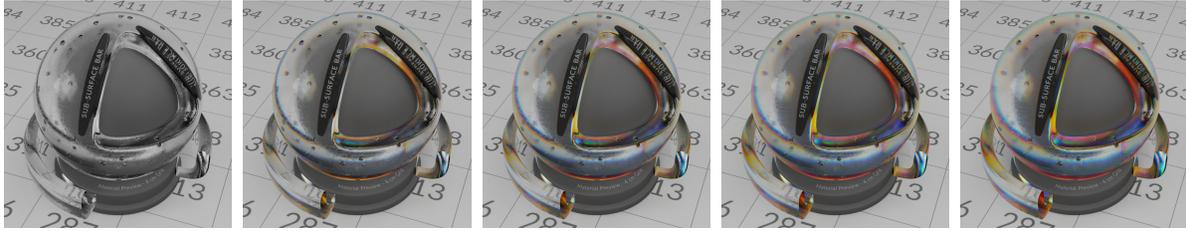


Figure 24: The effect of `transmission_dispersion_scale`, varying over 0, 0.2, 0.4, 0.6, 1. Here, the Abbe number was set (unrealistically low) to 1, via `transmission_dispersion_abbe_number`.

At the default `transmission_dispersion_scale` of zero, the Abbe number is infinite, which corresponds to no dispersion. Conversely, at the maximum of 1, the Abbe number peaks at $V_d = \text{transmission_dispersion_abbe_number}$, which defaults to 20.

The default 20 was chosen since common real materials with the highest dispersion have Abbe numbers roughly greater than or equal to this (for example, different types of glass have Abbe numbers between 20 and 91, water and diamond both have Abbe numbers between 55 and 56, while the oxide mineral rutile, composed of titanium dioxide, exhibits extremely high dispersion with an Abbe number of 9.87 [Pol24]).

In most cases, the `transmission_dispersion_scale` therefore functions as a convenient, roughly linear slider from low to high dispersion. For those special cases where the Abbe number of a specific material is required, the Abbe number itself can be changed via `transmission_dispersion_abbe_number`.

Note that dispersion can cause banding artifacts and color noise if not properly managed, however these issues can be mitigated using a careful strategy of wavelength sampling, color-channel selection, and MIS. For sampling wavelengths and incorporating spectral effects inside an RGB renderer, multiple approaches exist, such as the empirical one described by Kutz, Hašan, and Edmondson [KHE21]. The wavelengths from such a “locally spectral” approach can also be used for rendering thin-film iridescence, as mentioned in Section 4. For selecting a color channel to use for the refracted ray direction, it is useful to multiply the path throughput into the probabilities. This way, the channel carrying the most energy along the path so far will be preferentially used for continuing the path. This keeps the magnitude of the path throughput balanced and avoids generating paths that don’t contribute to the image.

For example, if one scattering event selects the red channel and generates a refracted path that is only valid for the red channel, the next scattering event will also use the red channel. While careful color-channel selection can reduce variance, it doesn’t directly target color noise. Color noise is inevitable when individual paths are only valid for a single wavelength, however when dispersion occurs on a rough microfacet surface, the resultant transmission lobes can overlap significantly, so paths can be shared across color channels using MIS (as described by Kutz, Hašan, and Edmondson [KHE21]). Similar approaches can also be leveraged to reduce noise when using path tracing to render volumes with wavelength-varying extinction coefficients: specifically, the path throughput (along with the single-scattering albedo) can be incorporated into the color-channel selection probabilities for distance sampling, and paths can be shared among color channels using MIS (as described by Chiang, Kutz, and Burley [CKB16] and Kutz et al. [Kut+17]).

3.2 Metallic base: the F82-tint model

The metallic base is represented as a separate bulk slab consisting of an opaque GGX microfacet conductor BRDF $f_{\text{conductor}}$ whose NDF is parameterized by `specular_roughness` and `specular_roughness_anisotropy` (overloading the same parameters used for the dielectric BSDF $f_{\text{dielectric}}$, as noted in the [Microfacet](#) section):

$$S_{\text{metal}} = \text{Slab}(f_{\text{conductor}}) . \quad (60)$$

Metals are completely opaque and have a characteristic and familiar form of specularity due to the Fresnel factor for conductors differing from that of dielectrics. The conductor Fresnel reflection curve is parameterized by the colors at normal and near-grazing incidence (`base_color` and `specular_color` respectively, scaled by the corresponding weights). This allows for art-directable variation in reflectivity toward the grazing edges by directly specifying the (texturable) colors at normal and grazing incidence to simulate the dip in reflectivity observed in real metals, or just for artistic effect [[Hof19](#)]. Note that these two color parameters are also used for the non-metallic (i.e., dielectric) [specular](#) and [diffuse](#) BRDFs.

As noted previously, this non-transmissive metallic base is blended as a statistical mixture with the dielectric base according to the `base_metalness` parameter:

$$M_{\text{base-substrate}} = \text{mix}(M_{\text{dielectric-base}}, S_{\text{metal}}, M) , \quad (61)$$

where $M = \text{base_metalness}$.

The specific model we stipulate for the metallic Fresnel factor $F_{\text{metal}}(\mu)$ is the ‘‘F82-tint’’ model of Kutz, Hařan, and Edmondson [[KHE21](#)], which extends previous work by Hoffman [[Hof19](#)]. This is based on the standard Schlick approximation to the metallic Fresnel factor, where F_0 is the RGB reflectivity at normal incidence (i.e., `base_weight * base_color`), and μ is the cosine of the incident angle:

$$F_{\text{Schlick}}(\mu) = F_0 + (1 - F_0)(1 - \mu)^5 . \quad (62)$$

To better approximate the actual Fresnel curve of metals, in the F82-tint model the Schlick approximation is augmented with a correction term:

$$F_{82}(\mu) = F_{\text{Schlick}}(\mu) - \mathbf{b}\mu(1 - \mu)^6 , \quad (63)$$

where

$$\mathbf{b} = \frac{F_{\text{Schlick}}(\bar{\mu}) - F(\bar{\mu})}{\bar{\mu}(1 - \bar{\mu})^6} . \quad (64)$$

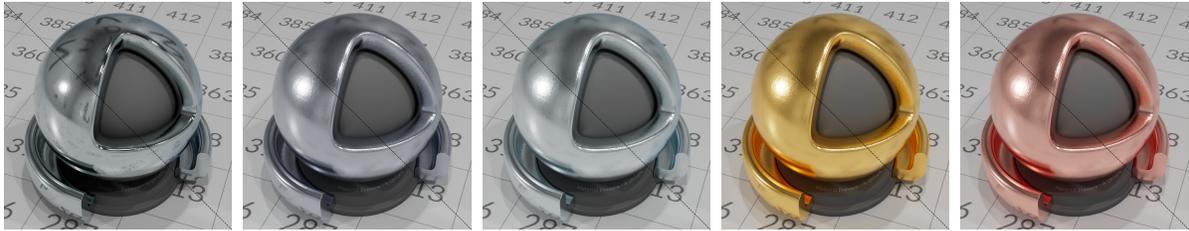


Figure 25: Renders with the F82-tint model of (from left to right) chromium, lead, zinc, gold and copper. Above the diagonal is the full fit to the measured metal IORs, while below the diagonal has the fitted `base_color` but `specular_color` is set to white.

Here, $F(\bar{\mu})$ is the desired metallic reflectivity at the “grazing edge” angle cosine $\bar{\mu} = 1/7$ corresponding roughly to 82° (i.e., around silhouettes), ensuring $F_{82}(\bar{\mu}) = F(\bar{\mu})$. This desired edge reflectivity is user-specified as a fractional tint of the Schlick curve, controlled via $C_s = \text{specular_color}$, i.e.

$$F(\bar{\mu}) = C_s F_{\text{Schlick}}(\bar{\mu}) . \quad (65)$$

The benefit of having `specular_color` function as the tint is that the model reduces to the regular Schlick reflectivity at the default values of `specular_weight` and `specular_color`.

The final metallic Fresnel term we employ is then given by an overall multiplication by $\xi_s = \text{specular_weight}$, ensuring that the entire metallic lobe is suppressed as the weight goes to zero. Similar to the dielectric, the weight can exceed one in order to linearly boost the Fresnel, with a clamp put in place to ensure that it remains bounded in $[0, 1]$ as $\xi_s \rightarrow \infty$:

$$F_{\text{metal}}(\mu) = \text{clamp}(\xi_s F_{82}(\mu), 0, 1) . \quad (66)$$

Note that the clamp at the lower end is applied since $F_{82}(\mu)$ can become negative for some values of μ , which is not physically meaningful [Hof19]. Note that the edge cannot be brighter than the standard Schlick term, but this is generally true in real metals. We consider this a benefit of this parameterization, since it makes it impossible to produce physically implausible metals with excessively bright edges.

Figure 25 shows renders of chromium, lead, zinc, gold and copper, with and without the physically correct `specular_color` F82-tint color. In the cases of chromium and lead, the effect of the F82-tint color is visually significant near the edge. In the case of zinc it is less obvious, and barely perceptible in the case of gold and copper. In Appendix D, we provide a table of computed fits to the F82-tint model for a variety of real metals, using publicly available tabulated IOR data [Pol24].

The hemispherical (or average) albedo of the F82-tint model (i.e., $E_{\text{avg}} \equiv 2 \int_0^1 F_{82}(\mu) \mu \, d\mu$) is required in multiple-scattering compensation schemes, for example. It is given exactly by

$$E_{\text{avg}} = F_0 + (1 - F_0)/21 - \mathbf{b}/126 , \quad (67)$$

where \mathbf{b} is defined in Equation 64. If both $F_0 = \text{base_weight} \times \text{base_color}$ and $C_s = \text{specular_color}$ are white, then $F_0 = 1$ and $\mathbf{b} = 0$ reducing to $E_{\text{avg}} = 1$, thus the metal satisfies the furnace test in this case. Note, however, that $F_{82}(\mu)$ can be slightly negative (for very dark metals), thus it is necessary to clamp it. The albedo of the clamped Fresnel is not exactly given by clamping the formula above, but it is very close (never more than 0.01 incorrect in absolute albedo).

Future work: Decoupled metallic parameterization The use of `base_color` and `specular_color` to control the reflectivity of both the metallic and dielectric base is a convenience that originates from the model of Burley [Bur12], and corresponds to the popular “metalness” workflow where a single set of textures, in conjunction with a metalness map, can be used to define both metallic and non-metallic regions on the same object.

However, while convenient, this approach does introduce some limitations. Figure 26 shows an example of the artifacts that can occur when blending between metallic and non-metallic regions, in this case to produce a rusty look. The left image shows a metalness blend done in the current model, while the right image shows the result with the metal and diffuse colors decoupled (made by altering the shader to allow the metal F_0 to be defined independently from the diffuse color used for rust). The metalness blend produces a brightening “halo” effect in the transition region (slightly blue due to the metal color), while the decoupled result produces a transition that

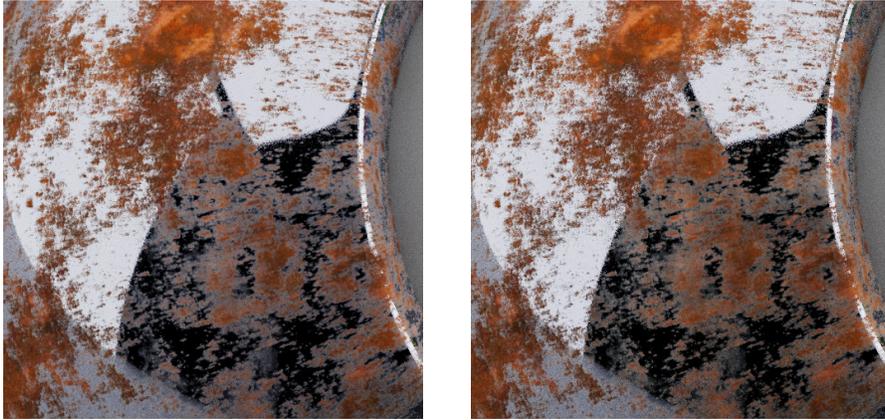


Figure 26: Metalness blending artifacts with the current model (left), which would be fixed with an expanded parameterization (right).

is more plausible for a metal that is partially covered by rust. While this could be avoided (if using a general material layering framework) by adding a completely separate layer on top of the OpenPBR base substrate, it seems preferable to have a single model that can handle blending between metallic and non-metallic regions without artifacts.

To be able to avoid the artifacts, a proposed future extension to the OpenPBR metallic base model is to decouple the metallic and dielectric reflectivities, so that the metalness can be controlled independently of the dielectric reflectivity. This would allow for a more physically plausible transition between metallic and non-metallic regions, as well as more flexibility in defining the material properties.

One proposed model for this decoupling is shown in Figure 27, where the existing convenient controls are retained, but they are used to drive (as indicated by the arrows) an underlying model which provides full decoupling of the metallic and dielectric properties. So, if necessary, one can drop down to the underlying parameters to control the reflectivity of the metal and dielectric separately.



Figure 27: Proposed expanded parameterization for decoupled metalness.

3.3 Emission

It is assumed that the base substrate below the coat may emit light, with a directionally uniform distribution. One can imagine the light being emitted from the interior of the base substrate with an isotropic luminance given by L_e . We put emission below the coating so that emitted light will be tinted due to the absorption in the coat and fuzz layers. This allows for the convenient rendering of low-emission materials that are bounded by a reflective surface (e.g., glow sticks, LEDs, display screens, etc.) without explicit modeling of the emitter and the bounding object.

The intensity of the emission is controlled by a luminance value (in nits) with color and weight multipliers. The color and weight act as multipliers, i.e., the HDR emission in the model color space is defined to have a color given by $\text{emission_weight} \times \text{emission_color} \times \text{emission_luminance}$. The `emission_luminance` parameter thus refers to the luminance the emissive layer would have when the color is white and weight is 1, and in the absence of coat and fuzz. Thus the final resulting luminance may be less than the input parameter, or even zero if the color or weight are zeroed.

Moreover, the overall material luminance may be further reduced in the presence of coat or fuzz, as they can absorb light coming from the emissive layer before it exits the surface. The emission from the top surface should in principle gain a directional dependence due to the combined effects of absorption, total internal reflection (TIR) and multiple bounces in the coat layer, and absorption in the fuzz layer. The combined effect should result mostly in darkening and saturation at grazing angles.

Being an intensity, `emission_luminance` can be any value greater than or equal to zero. For convenience, we make the soft range $[0, 1000]$ nits, corresponding to the typical range of home appliances. (Note that if the renderer does not deal with photometric units internally, a scale factor may need to be applied to bring the emission into a sensible range.)

We found that it is convenient for there to be a simple `emission_weight` control modulating the luminance that can be adjusted via a slider (and also textured with a $[0, 1]$ mask). This is more consistent with the other lobes, which all have an associated weight dialing the strength of the effect. Adding this weight, which defaults to 0, allows `emission_luminance` to default to a reasonable value of 1000 nits.



Figure 28: A scene with an emitting surface (rendered in Arnold on the left, and Adobe's proprietary renderer on the right).

3.4 Future work

Here we detail two proposed extensions to the base substrate model: [hazy-specular](#) and [retro-reflection](#). These are not yet implemented in OpenPBR (as of version 1.1), but are planned for future releases.

3.4.1 Hazy specular

A strongly requested feature is a more configurable [microfacet](#) NDF which can support a “hazy gloss” look. The [coat](#) layer does not provide equivalent functionality, since the coat produces a strong Fresnel effect near grazing, roughens the underlying base, and generates darkening due to inter-reflections. A number of models for a generalized microfacet NDF have been proposed. Ribardi re et al. [Rib+17] introduced a BRDF based on the Student’s t-distribution. Burley [Bur12] introduced the “generalized Trowbridge-Reitz” (GTR) NDF. These add a 1-parameter control over the NDF shape. Barla, Pacanowski, and Vangorp [BPV18] introduced a model with a blend of a primary lobe and secondary lobe differing in roughness, expressed as a sum of a “specular core” and a “surrounding halo”, in such a way that the halo width can be adjusted independently of the core brightness.

An alternative, simpler approach based on Barla, Pacanowski, and Vangorp [BPV18] that we favour is to simply have the BSDF be a blend between microfacet BSDFs with independent roughnesses, the original “core” lobe and an added “haze” lobe with some mix weight $w_h \in [0, 1]$ and effective roughness r_h greater than the core roughness. We noted that this model is actually more expressive and physically plausible than the 1-parameter models, since the mix weight w_h effectively controls the ratio of the energy in the haze relative to the core, while the haze roughness r_h can be varied completely independently of this (supporting weak and strong haze, with independent low or high added roughness).

We plan to provide this via the following parameters:

- `specular_haze`, $w_h \in [0, 1]$: Provides the blend weight of a secondary specular NDF (defaulting to 0).
- `specular_haze_spread`, $\xi_h \in [0, 1]$: Specifies the extra roughness of this secondary lobe, in $[0, 1]$. This is mixed with the primary specular lobe roughness (`specular_roughness`, r) to produce the roughness value for the secondary lobe r_h (with the sum clamped to the $[0, 1]$ range) via $r_h = (1 - \xi_h)r + \xi_h = r_c + \xi_h(1 - r)$. Note that the anisotropy parameters are the same for both primary and secondary specular lobes. Figure 29 shows the effect of varying the `specular_haze_spread` parameter, with a fixed `specular_haze` of 0.5.

The model then consists simply of two microfacet BSDFs f_c (core) and f_h (haze) with roughnesses r and r_h respectively (and equal anisotropy), blended via $(1 - w_h)f_c + w_h f_h$.



Figure 29: With `specular_haze` $w_h = 0.5$, varying the `specular_haze_spread` ξ_h over 0, 0.2, 0.4, 0.6.

3.4.2 Retro-reflection

Rendering of retro-reflective materials is useful in a variety of contexts (typically for safety applications such as road markings, signs, vehicles and clothing). Such materials are usually designed to be retro-reflective via a substructure of elements that preferentially scatter light backwards. A number of CG models have been proposed ([Bel+14; GP17; GGP18]). For practical purposes in visual effects, we are interested in a model which is visually plausible at least, provably energy-conserving, efficient and easy to implement, and ideally a small modification to well-understood microfacet models so it can be implemented as a modification to the specular lobe.

A model we are evaluating is based on the previously published “back-vector” formulation [Bel+14] which meets these requirements. This empirically based approach simply replaces the half-vector \mathbf{h} in the microfacet model with the back-vector \mathbf{b} defined as

$$\mathbf{b}(\mathbf{v}, \mathbf{l}) = \mathbf{h}(\mathbf{v}', \mathbf{l}) = \frac{\mathbf{v}' + \mathbf{l}}{\|\mathbf{v}' + \mathbf{l}\|}, \text{ with } \mathbf{v}' = \text{reflect}(\mathbf{v}, \mathbf{n}). \quad (68)$$

Belcour et al. [Bel+14] demonstrated that this formulation achieves a reasonable match to measured retro-reflective materials. Given an implementation of a regular microfacet BRDF, extending it to retro-reflection is then extremely straightforward:

- Evaluation merely needs to replace \mathbf{v} with \mathbf{v}' upfront. Similarly, importance sampling of \mathbf{l} given \mathbf{v} can be realized by replacing \mathbf{v} with \mathbf{v}' upfront and then importance sampling the regular microfacet BRDF. This may include low variance sampling using the domain of visible microfacets [Hei18].
- As the albedos of the standard BRDF and retro-reflective BRDF are essentially identical, compensating for energy loss in the sense of Kelemen and Szirmay-Kalos [KS01] can be realized using the same data tables.

We tentatively term this model the “minimal retro-reflective microfacet model” (MRRM). It can be shown that this leads to a visually plausible, energy-conserving and reciprocal retro-reflective result (Figure 30). To provide this retro-reflectivity functionality, the BRDF of the base can simply be taken to be a statistical mix of the BRDF with and without the view vector modification, with a mix weight w_{retro} :

$$f_{\text{conductor}} \rightarrow (1 - w_{\text{retro}})f_{\text{conductor}} + w_{\text{retro}}f_{\text{retro}} \quad (69)$$

An open question is whether the back-vector modification should be applied to both the conducting and dielectric microfacet cases, as the behavior of the latter is less plausible as a typical retro-reflective material.

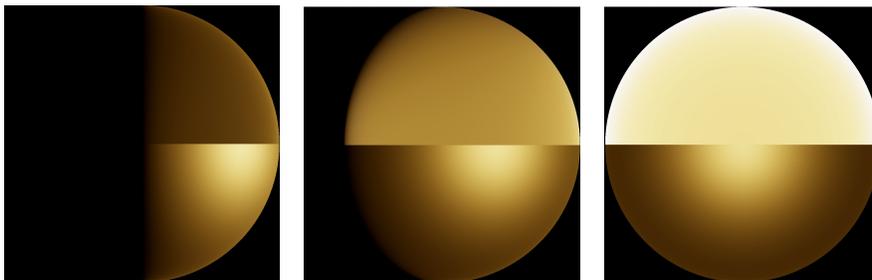


Figure 30: The top row shows the “minimal retro-reflection model” (MRRM) applied to a microfacet conductor, viewed from the side, quarter-on and facing away from the illumination. The bottom row shows the look with the usual non-retro-reflective microfacet conductor model of the same roughness.

4 THIN-FILM IRIDESCENCE

Iridescence is the occurrence of rainbow-like color fringes in the reflection when a thin dielectric film with thickness on the order of the wavelength of light is placed on top of a material, due to interference between light reflected from the film’s top and bottom surfaces, including internal reflections. To model this, we assume such a thin film sits atop the base substrate (whether metal or dielectric), parameterized by

- `thin_film_weight`: the coverage (presence) weight of the film.
- `thin_film_thickness`: the thickness of the film in micrometers (μm). The $[0, 1]$ (soft) range of thicknesses, in micrometer units, then corresponds nicely to the typical thickness range needed to observe interference effects.
- `thin_film_ior`: the index of refraction (IOR) of the film.

The thickness and IOR together affect the intensity, spacing, and hue of the color fringes. The coverage weight acts as a blend between the BSDF with and without the presence of the film, allowing the overall strength of the effect to be adjusted without altering its structure or color. Figure 31 shows the effect of varying the thickness of the thin film. As thickness increases, one sees higher-frequency color banding, and the effect eventually converges toward a “thick film” look where the colors fade to gray as the interference effects wash away.

The currently recommended thin-film model is that of Belcour and Barla [BB17], which pre-integrates interference effects using Fourier-domain convolutions and Gaussian filtering. This method efficiently produces high-quality fringe patterns in an RGB rendering context, but it can be challenging to implement and may introduce inaccuracies in some cases as it assumes that Fresnel amplitude and phase coefficients remain constant across each spectral band, which limits the model’s ability to capture wavelength-dependent dispersion effects.

A more direct alternative that we highlight here (and in more detail in Appendix E) is a “locally spectral” approach that computes reflectance per light path by evaluating the full Fresnel and Airy interference stack – including complex amplitudes, polarizations, and phase shifts – at specific wavelengths sampled per path. This can begin with fixed red, green, and blue wavelengths, but better results are achieved by stochastically sampling wavelengths from approximate camera sensitivity curves (as described in Section 3.1.3). This enables convergence to neutral gray for very thick films and avoids the high-frequency color banding that fixed RGB wavelengths can produce.

The same wavelengths can also be reused to model *dispersion*, while all other BSDF components are free to ignore them and operate in RGB as usual. This approach uses only the Airy summation from Belcour and Barla (Equation 3 from Belcour and Barla [BB17]) – a method whose origins trace back to 19th-century studies



Figure 31: Varying the thickness of the thin film (from left to right: 0.25, 0.5, 0.75, 1, 2 μm).

of thin-plate interference by G.B. Airy and others – but requires additional per-wavelength computations and assembling the necessary formulas from multiple sources rather than a single reference. We attempt to collect the necessary equations and implementation instructions in one place in Appendix E.

Regardless of which approach is chosen, several considerations apply to both:

- The shape and color of the fringe patterns in the reflection from the film will be affected by the complex IOR of the adjacent media above and below the film, which in general are a statistical mix of metal and dielectric below, and of coat and ambient medium above (to which the fuzz is index-matched).

Figure 32 illustrates the eight possible different structures depending on the presence of both the film and coat, each of which leads to different Fresnel effects due to the differing IORs at the interfaces. In principle, the implementation should account for all of these configurations accurately, though the precise modeling of these effects is left to the implementation.

- In practice, this wave-optics effect is most easily incorporated directly into the Fresnel factor of the microfacet BSDFs of both the [metal](#) and [dielectric-base](#) layers. (For this reason, this effect is not represented by incorporating an explicit thin-film Slab into the model.)
- In the case of the dielectric base, the thin film should also generate color fringes in the transmission lobe. This is important for example when rendering soap bubbles (see Belcour and Barla [BB17]).
- In the case of the metallic base, the physics is somewhat ambiguous since the Fresnel factor for metal is defined according to the Schlick-based “F82-tint” parameterization (as described in Section 3.2), which does not specify the underlying physical complex IOR. We suggest here that some reasonable approximation is employed to map the Fresnel factor to the best matching effective complex IOR, for example that described by Gulbrandsen [Gul14].
- Because the thin film is non-absorbing and interference based, it only redistributes the probabilities of reflection and transmission; therefore, it should not violate energy conservation. In the limit, where the base metal or dielectric has a Fresnel factor of 1, the thin film has no effect and passes a white furnace test.

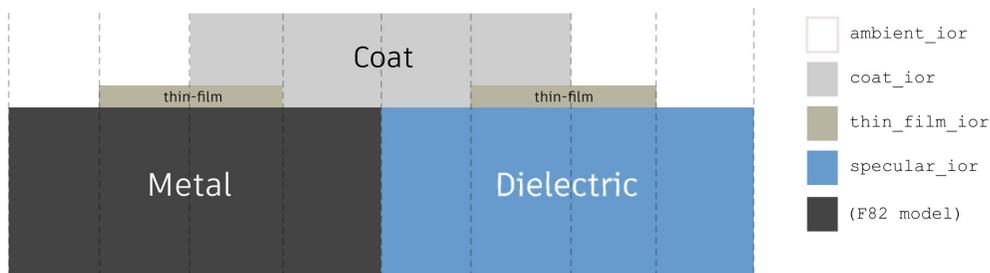


Figure 32: Schematic of all 8 possible IOR configurations, including those involving the thin film.

5 COAT

The coat slab is a layer of dielectric that transmits light without scattering, but with possible absorption. This is intended to support the appearance of objects with a coat of colored varnish or lacquer:

$$S_{\text{coat}} = \text{Slab}(f_{\text{coat}}, V_{\text{coat}}) . \quad (70)$$

The BSDF of the interface f_{coat} is that of a GGX [microfacet](#) dielectric parameterized by `coat_roughness` and `coat_roughness_anisotropy`. The IOR `coat_ior` of this dielectric layer is distinct from that of the base dielectric, as described below. There is also assumed to be an embedded *purely absorbing* medium V_{coat} . The coat is applied on top of the base substrate, with a coverage weight $C = \text{coat_weight}$ as follows:

$$M_{\text{coated-base}} = \text{layer}(M_{\text{base-substrate}}, S_{\text{coat}}, C) . \quad (71)$$

The IOR $n_c = \text{coat_ior}$ of the coat medium V_{coat} will alter the Fresnel factor of both the coat top interface and the underlying metal or dielectric. If there is a fractional `coat_weight` C , then the surrounding IOR of the base dielectric or metal varies statistically across the surface depending on whether the coat is locally present (and the fuzz layer can be assumed to have the ambient IOR n_a). The ratio between the specular IOR $n_b = \text{specular_ior}$ and the surrounding medium can thus be reasonably approximated as

$$\eta_s = \text{lerp}(n_b/n_a, n_b/n_c, C) . \quad (72)$$

This ratio then determines the specular Fresnel factor, as in Equation 30. As discussed [below](#), evaluation of the specular Fresnel factor may need to be further modified to model the refraction of the ray inside the coat.

The absorption of the medium V_{coat} is parameterized by `coat_color`, which is assumed to specify the *square* of the transmittance of the coat at normal incidence (i.e., $T_{\text{coat}}^2 = \text{coat_color}$ in the notation of Equation 8). Thus the observed tint color, at normal incidence, of the underlying base due to absorption in the coat is approximately given by `coat_color` due to the absorption along the incident and outgoing rays. (Note that the specular reflection from the coat itself is *not* tinted.)

In the full light transport within the coat, various physical effects occur which we assume are accounted for in the ground truth appearance:



Figure 33: A wood-textured base with no coat (left), and with a clear-coat (IOR 1.5) with `coat_darkening` $\delta = 0$ (middle) and $\delta = 1$ (right).

- **Darkening:** The observed color of the coated base is darkened and saturated due to multiple internal reflections from the inside of the coat, which causes light to strike the underlying material multiple times and undergo more absorption. This effect is controlled via the `coat_darkening` parameter. Figure 33 shows an example of this effect, where the middle image shows the effect of removing the darkening due to internal reflections in the coat via `coat_darkening` $\delta = 0$, while the right image shows the physically correct darkening effect.
- **View-dependent absorption:** The observed `coat_color` tint also darkens as the incidence angle changes due to the change in path length in the medium.
- **Base roughening:** The presence of a rough coat will increase the apparent roughness of the BSDF lobes of the underlying base.
- **TIR:** Care needs to be taken in the implementation to account for the refraction of the ray direction inside the coat.

In reality, coats can also darken the underlying surface due to a different mechanism where the coat modifies the Fresnel factor of the base due to the coat material filling in air gaps between granules or threads of a porous base material, which reduces the relative IORs at the internal interfaces. This occurs, for instance, when adding water to sand or fabric, or adding a penetrating wood finish. We assume here that this effect explicitly does *not* occur, at present, since we do not have enough knowledge about the properties of the underlying substance to model it. We can only safely assume that the first mechanism of darkening (i.e., internal reflections) occurs.

5.1 Darkening

In Appendix F, we provide the formulas for the physical darkening effect in the case of a smooth coat.

However, this darkening may not always be desirable artistically, as in some applications it is beneficial for the observed color of the coated base to “match” the input base color (in a sense defined more precisely below in Equation 75). We allow for this by introducing a `coat_darkening` parameter, δ . By default $\delta = 1$, in which case the physically correct darkening effect due to internal reflections occurs as normal. In the case $\delta = 0$, however, the base albedo is modified to counteract the darkening effect (see Figure 33).

To describe what we mean by “counteract the darkening”, we write the (physically darkened) coat albedo at normal incidence in the general form

$$\mathbf{E}_c = F_0 + (1 - F_0)\mathbf{E}'_c, \quad (73)$$

where F_0 is the normal incidence Fresnel factor of the coat, and \mathbf{E}'_c represents the albedo due to transmission into the coat medium (and scattering off the base substrate, potentially multiple internal reflections off the coat interface, and re-transmission back out). We then require that the *effect* of $\delta = \text{coat_darkening}$ is to multiply \mathbf{E}'_c by an RGB boost factor

$$\mathbf{B}(\delta) = \text{lerp}\left(\frac{\mathbf{T}_{\text{coat}}^2 \mathbf{E}_b}{\mathbf{E}'_c}, 1, \delta\right), \quad (74)$$

where \mathbf{T}_{coat} is the coat absorption transmittance, and \mathbf{E}_b represents the normal-incidence albedo of the entire base beneath the coat (which can be approximated as the normal-incidence albedos of the individual slabs of the base, blended according to their mix weights).

This is a straightforward implementation of the requirement for the albedo of the “un-darkened” coat to be equal to the usual albedo scaling formula of Equation 8 (which involves no color shift other than that due to the

absorption, and the combination with the Fresnel factor), i.e.

$$\mathbf{E}_c = F_0 + (1 - F_0)\mathbf{E}'_c\mathbf{B}(\delta) \rightarrow F_0 + \mathbf{T}_{\text{coat}}^2 \mathbf{E}_b(1 - F_0) \quad (75)$$

as $\delta \rightarrow 0$.

Note that we have not specified the detailed physical mechanism by which the boosting of \mathbf{E}'_c occurs. Instead, we have only defined what the resultant effective albedo modification (at normal incidence) has to be as `coat_darkening` is varied. While this defines the behavior at the level of the required albedos, in practice implementations will need to develop some specific approximation for the coat darkening effect consistent with their physical approximations. Due to the physical effect of the darkening due to the internal reflections, the base BSDF would generally need to be altered in order for the coated base to achieve the un-darkened albedo. While this can be implementation dependent, we suggest here a reasonably simple, efficient scheme that captures the essential behavior.

According to Appendix F, the physical effect of the inter-reflections in the coat can be modeled by multiplying the naive albedo-scaling formula for the coat layering by a darkening factor (Equation 135):

$$\Delta(\mathbf{E}_b, \eta_c) = \frac{1 - K_0}{1 - \mathbf{E}_b K_0 \mathbf{T}_{\text{coat}}^2} . \quad (76)$$

Here, $K_0 \in [0, 1]$ is the *internal diffuse reflection coefficient*, which corresponds to the fraction of the energy leaving the base that returns, in the case of a clear coat. For a Lambertian base (which should be a reasonable approximation to the rough metal, dielectric, or diffuse cases), $K_0 = K_r$, where

$$K_r = 1 - \left(1 - E_F(\eta_c)\right) / \eta_c^2 \quad (77)$$

with relative coat IOR η_c . As discussed at the end of Appendix F, if the base has specular reflection (due to a smooth metallic or dielectric interface), the appropriate value of the internal diffuse reflection coefficient K_0 should be closer to

$$K_s = F(\omega_o, \eta_c) . \quad (78)$$

We thus recommend taking

$$K_0 = \text{lerp}(K_s, K_r, r_b) , \quad (79)$$

where r_b is an estimate of the effective base roughness, a blend between dielectric r_d and metal r_m roughness estimates according to the `base_metalness` M :

$$r_b = \text{lerp}(r_d, r_m, M) . \quad (80)$$

The base dielectric roughness r_d can be reasonably estimated as a mix between the high roughness of an assumed underlying base and the microfacet dielectric roughness $r = \text{specular_roughness}$, according to the base dielectric Fresnel factor modulated via `specular_weight` ξ_s (see Equation 28 and Equation 30):

$$r_d = \text{lerp}(1, r, \xi_s F_s) , \quad (81)$$

while the metallic roughness can be taken to be $r_m = r$. (Note that, in this formula for r_d , a clamp must be applied to ensure that $\xi_s F_s \in [0, 1]$.)

Listing 2: Example implementation of coat darkening.

```

void AddCoat(const ShaderGlobals& sg, const Params& P, BSDFs& substrate_bsdfs)
{
    float C      = P.coat_weight; if (C <= 0.f) return;
    float delta  = P.coat_darkening;
    vec3 CC      = P.coat_color;
    float n_coat = P.coat_ior;
    vec3 base_color = P.base_weight * P.base_color;
    float M      = P.base_metalness;
    float xi     = P.specular_weight;
    float n_spec = P.specular_ior;
    float r_spec = P.specular_roughness;
    float T      = P.transmission_weight;
    float S      = P.subsurface_weight;
    vec3 Sc      = P.subsurface_color;

    // Compute the base_darkening factor based on the coat properties:
    vec3 base_darkening = WHITE;
    if (delta > 0.f)
    {
        float EF = DielectricFresnelAvg(n_coat); // Hemispherical Fresnel albedo, Eqn.124
        float Kr = 1.f - (1.f - EF)/sqr(n_coat); // (rough case, Eqn.77)
        float Ks = DielectricFresnel(abs(dot(sg.wi, sg.N)), n_coat); // (smooth case, Eqn.78)
        float eta_s = mix(n_spec, n_spec/n_coat, C); // specular/coat IOR ratio
        float F0 = DielectricFresnel(1.f, eta_s); // coat Fresnel at normal incidence
        float Fs = clamp(xi * F0, 0.f, 1.f); // modulated dielectric Fresnel
        float rd = mix(1.f, r_spec, Fs); // estimate roughness of dielectric base
        float rb = mix(rd, r_spec, M); // roughness of entire base, Eqn.80
        float K = mix(Ks, Kr, rb); // internal diffuse reflection coefficient K, Eqn.79
        vec3 E_dielec = mix(mix(base_color, Sc, S), vec3(1.f-F0), T); // dielectric base albedo
        vec3 E_metal = clamp(base_color * xi, 0.f, 1.f); // metallic base albedo
        vec3 E_base = mix(E_dielec, E_metal, M); // entire base albedo
        vec3 Delta = max(1.f - K, 0.f) / (WHITE - E_base*K*CC); // darkening factor, Eqn.76
        base_darkening *= mix(WHITE, Delta, delta); // coat_darkening modulation, Eqn.83
    }

    // Apply albedo scaling to model the layering of the coat on the base.
    // First, the coat BSDF has the coat_weight presence weight functioning as a multiplier:
    BSDF* coat_bsdf = C * DielectricBSDF(sg, n_coat, P.coat_roughness);

    // account for angle-dependent absorption path length in the coat medium:
    float mu_o = dot(sg.wo, P.coat_normal);
    float mu_i = dot(sg.wi, P.coat_normal);
    vec3 absorb_tint = pow(CC, 0.5f*(1.f/mu_i + 1.f/mu_o));

    // The base BSDFs are then multiplied by the coat_throughput factor,
    // which includes the base_darkening effect and the absorption tinting:
    vec3 coat_albedo = Albedo(coat_bsdf);
    vec3 coat_throughput = mix(WHITE, base_darkening * absorb_tint * (WHITE - coat_albedo), C);
    substrate_bsdfs *= coat_throughput;
    substrate_bsdfs.add(coat_bsdf);
}

```

Listing 2: Example implementation of coat darkening.

We can thus approximate $E'_c \approx E_b T_{\text{coat}}^2 \Delta$. The boost factor of Equation 74 then reduces to

$$\mathbf{B}(\delta) = \text{lerp}\left(\frac{1}{\Delta}, 1, \delta\right). \quad (82)$$

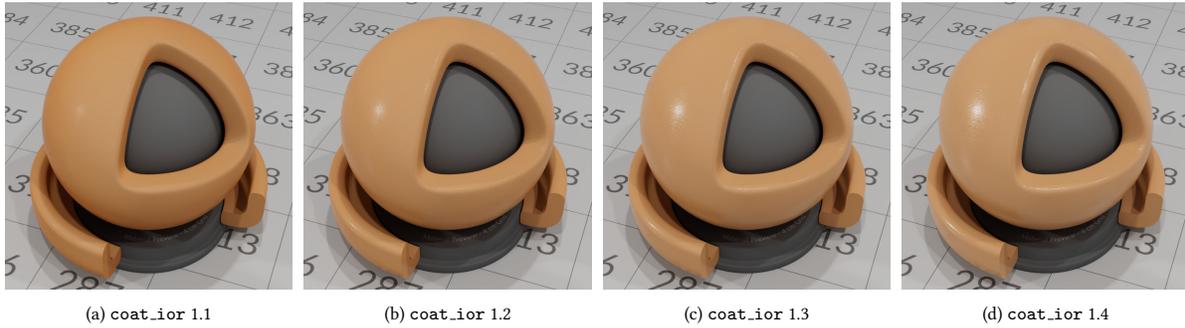


Figure 34: The color of an absorbing coat becomes slightly darker and more saturated at grazing angles, with a stronger effect for low `coat_ior`.

A reasonable approximate scheme, assuming no other compensation is made to approximate the effect, is to multiply the base BSDF by the uniform *modulated darkening factor* (depending on the darkening parameter $\delta = \text{coat_darkening}$):

$$\mathbf{B}(\delta) \Delta T_{\text{coat}}^2 = \text{lerp}(1, \Delta, \delta) T_{\text{coat}}^2 \quad (83)$$

with Δ evaluated via Equation 76 with the internal diffuse reflection coefficient, accounting for base roughness, given by Equation 79. Listing 2 gives pseudocode of an example implementation in an albedo-scaling based layer framework.

5.2 View-dependent absorption

In the case of an absorbing coat, there is also enhanced darkening and saturation at grazing angles due to increased path length within the coat medium. The effect of this can be modeled via a factor in the coat BRDF of

$$T_{\text{coat}}^{1/\mu_i^t + 1/\mu_o^t}, \quad (84)$$

where μ_i^t is the angle cosine of the incident ray refracted into the coat, i.e.

$$\mu_i^t = \sqrt{1 - \frac{1 - \mu_i^2}{\eta_c^2}}, \quad (85)$$

and similarly for μ_o^t . Note that, at normal incidence, this factor reduces to T_{coat}^2 , which we defined to equal `coat_color`. The effect of the absorption is more apparent for low `coat_ior`, as the path length within the coat layer increases (see Figure 34).

5.3 Roughening

If the coat is rough, the microfacet BSDF lobes of the underlying base substrate (metal and dielectric) are also effectively roughened. If this is not otherwise accounted for by the light transport, it can instead be reasonably approximated by directly altering the NDF of the base BSDFs. Figure 35 illustrates the (approximate) effect of varying the coat roughness on the appearance of the base material.

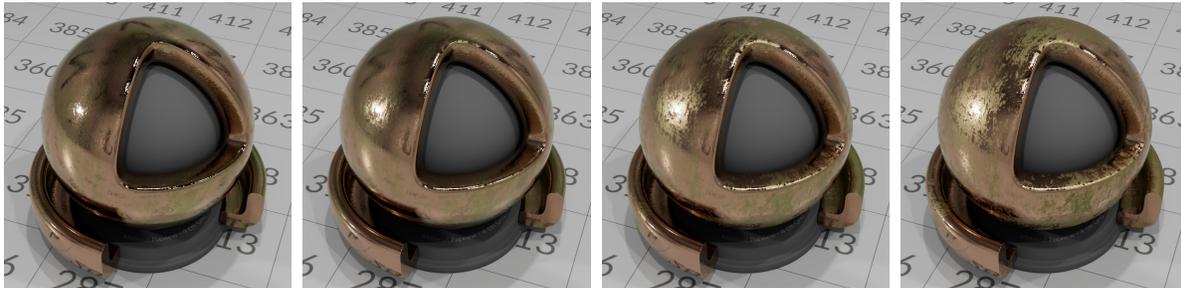


Figure 35: Appearance of a smooth metallic base with a coat “patina” (with textured weight) of varying roughness, for $r = 0$ (left), $r = 0.25$ (middle left), $r = 0.5$ (middle right), and $r = 0.75$ (right).

A heuristic we recommend for this is obtained by identifying the NDF of each microfacet lobe as corresponding approximately to a Gaussian in slope space with variance given by $\alpha_t^2 + \alpha_b^2 = r^4$ (in the notation of Section 2.2). The effect of the coat roughening can then be modeled as the convolution of these Gaussian NDFs, which corresponds to adding the variances (double counting the coat variance, since the reflection passes through the coat boundary twice). The IOR ratio of the coat and ambient medium η_{ca} also needs to be accounted for, since as $\eta_{ca} \rightarrow 1$, the roughening due to the coat goes to zero. This leads to the following suggested approximation for the modified roughness r'_B of the base due to the coat:

$$r'_B = \min(1, r_B^4 + 2x_C r_C^4)^{\frac{1}{4}} \quad \text{with } x_C = 1 - \min(\eta_{ca}, 1/\eta_{ca}), \quad (86)$$

where $r_B = \text{specular_roughness}$ and $r_C = \text{coat_roughness}$. Of course, the presence weight of the coat ($C = \text{coat_weight}$) also needs to be taken into account, ideally by blending between the effect with and without the coat present. Alternatively, a cruder approximation would be to just set the roughness of the base to $\text{lerp}(r_B, r'_B, C)$.

We intend to provide better approximations in the future. For example, a derivation of the roughening due to transmission through a rough GGX dielectric interface was provided by Belcour [Bel18], which could provide a more physically accurate result.

5.4 Total internal reflection

A technical issue which can cause difficulties in the implementation of the BRDF of the coated dielectric base should also be mentioned. See Figure 36 for a schematic of a ray refracting from the ambient exterior medium (with IOR n_a) into the coat (with IOR n_c), and then into the base dielectric (with IOR n_b).

If the coat IOR n_c exceeds the IOR of the base dielectric n_b , then rays incident to the base can in general undergo total internal reflection (TIR) at the base-coat boundary surface S_{bc} (with IOR ratio $\eta_{bc} = n_b/n_c$). In this case, the reflection from the base (the dashed line) is enhanced since no refraction into the base along the μ_b direction occurs.

However, if the surfaces are smooth and parallel, it is easy to show that *no such TIR is possible* for a ray incident from the exterior due to the refraction of the ray at the upper coat-ambient boundary S_{ca} (with IOR ratio $\eta_{ca} = n_c/n_a$). In the general case of rough boundaries, TIR can occur though it will be increasingly suppressed as the roughness decreases. Thus if the implementation of the **dielectric-base** BSDF does not account for the refraction

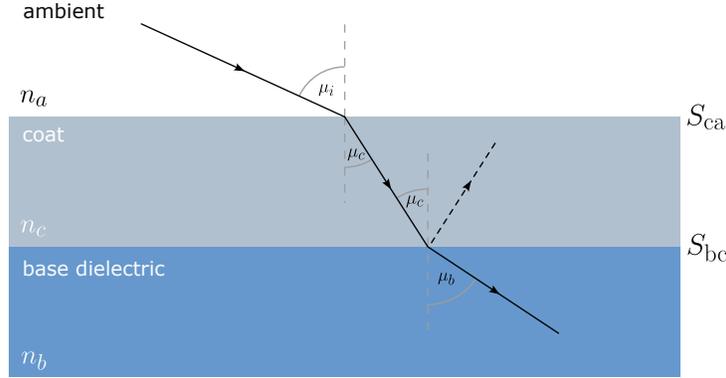


Figure 36: Ray configuration at coat-base boundary.

of the incident ray in the coat, spurious TIR will be generated, which produces an obviously incorrect appearance manifesting as a bright “ring” near grazing angles. In an implementation which reduces the model to a mixture of independent lobes sampled as a function of incident direction, it is non-obvious how to account for this.

We suggest as a reasonable practical approximation (following Kutz, Hařan, and Edmondson [KHE21]) to simply invert the IOR ratio at the coat-base boundary, i.e., replace η_{bc} as follows:

$$\eta_{bc} \rightarrow \begin{cases} \eta_{cb} = n_c/n_b & \text{if } n_c > n_b \\ \eta_{bc} & \text{if } n_c \leq n_b . \end{cases} \quad (87)$$

This eliminates the TIR while plausibly maintaining the specular reflection. Then, to account for partial coat coverage, this modified η_{bc} is used in place of the n_b/n_c term in Equation 72.

An alternative approach is to explicitly modify the refraction angle cosine of the incident direction to the base. Assuming, for simplicity, that in the local space of the base microfacet the overlying coat microfacet has matching micronormal, the angle cosine μ_c to the base micronormal is given by refraction of the incident angle cosine μ_i as follows:

$$\mu_c^2 = 1 - (1 - \mu_i^2)/\eta_{ca}^2 . \quad (88)$$

The resultant Fresnel factor can then be approximated as a blend of the Fresnel factors evaluated at the original and refracted incident directions, according to the presence weight of the coat:

$$\text{lerp}(F(\mu_i, \eta_{ba}), F(\mu_c, \eta_{bc}), C) , \quad (89)$$

where $\eta_{ba} = n_b/n_a$, and this blended Fresnel factor is then substituted into the microfacet BRDF (this replaces the Fresnel calculation of Equation 72, which ignores the coat refraction). If $\eta_{ca} < 0$ then TIR at S_{ca} may occur (i.e., $\mu_c^2 \leq 0$ in Equation 88), in which case the Fresnel factor $F(\mu_c, \eta_{bc})$ can be assumed to be zero.

Both of these schemes produce a plausible appearance that eliminates the spurious TIR. A more physically correct approach would require something similar to the Weidlich–Wilkie layering model [WW07], or the layered material model described in Pharr, Jakob, and Humphreys [PJH16], where the scattering through the layers is explicitly modeled via Monte Carlo simulation.

6 FUZZ

The *fuzz* BRDF f_{fuzz} and corresponding volumetric distribution function (VDF) V_{fuzz} are modeled as a homogeneous scattering layer composed of anisotropic microflakes with a fiber-like distribution – often referred to as *sheen* in the literature.

Previous approaches include the microfacet-based sheen of Estevez and Kulla [EK17] as used by the Autodesk Standard Surface model [Geo+19], and the single-scattering microflake model of Adobe Standard Material [KHE21]. In OpenPBR, we adopt the multi-scattering volumetric microflake model of Zeltner, Burley, and Chiang [ZBC22], which uses the SGGX microflake representation of Heitz et al. [Hei+15] and provides efficient evaluation and importance sampling via linearly transformed cosines (LTCs) [Hei+16a].

While structurally similar to Adobe Standard Material’s single-scattering approach, Zeltner’s model captures energy-conserving multiple scattering and supports principled importance sampling and layering.

Key features of the *fuzz*/*sheen* model in OpenPBR:

- **Volumetric foundation:** A homogeneous, purely scattering layer with a fiber-like SGGX microflake phase function [Hei+15], providing a physically plausible appearance.
- **Multiple scattering and LTC fitting:** Reflection is approximated by an LTC fitted to full volumetric multi-scattering simulations, enabling efficient evaluation and importance sampling.
- **Layering model:** The layer has unit optical thickness, is index-matched to the exterior medium (eliminating Fresnel at the interface), and conserves energy – all unreflected light transmits uncolored, as there is no absorption, which allows for principled layering (unlike the Estevez and Kulla [EK17] model).
- **Control parameters:**
 - `fuzz_weight`: Controls the layer’s coverage weight.
 - `fuzz_roughness`: Adjusts microflake shape – low values yield tight fiber-like sheen (e.g., silk), while high values yield diffuse, dust-like scattering (see Figure 37).
 - `fuzz_color`: Tints reflected light, allowing the sheen to lighten or darken.



Figure 37: From left to right, varying the `fuzz_roughness` over 0.25, 0.5 (default), 0.75 produces a progressively more dusty appearance.

The mathematical form of this model is the following (with μ_i, μ_o the angle cosines to the normal of ω_i, ω_o):

$$\mu_i f_{\text{fuzz}}(\omega_i, \omega_o) = \mathbf{F} E_{\text{fuzz}}(\mu_o, \alpha) D(\mu_i | \mu_o, \alpha), \quad (90)$$

where $\mathbf{F} = \text{fuzz_color}$, $E_{\text{fuzz}}(\mu_o, \alpha)$ (termed R by Zeltner, Burley, and Chiang [ZBC22]) is the reflectance at angle cosine μ_o given roughness $\alpha = \text{fuzz_roughness} \in [0, 1]$, and $D(\mu_i | \mu_o, \alpha)$ is a lobe defined by linear transformations of a cosine lobe (LTCs), where the transformation matrices (and E_{fuzz}) are fitted (over μ_o and α) to a simulation of the scattering in the volumetric fuzz microflake layer. Since the LTC lobe D is a normalized PDF over the hemisphere, the resulting albedo of f_{fuzz} is $\mathbf{F} E_{\text{fuzz}}(\mu_o, \alpha)$.

If using the albedo-scaling interpretation of layering, a reasonable approximation of the reflection from the fuzz layer combined with the reflection from the base is to take

$$\text{layer}(M_{\text{coated-base}}, S_{\text{fuzz}}) \rightarrow f_{\text{fuzz}} + (1 - E_{\text{fuzz}}) f_{\text{coated-base}}, \quad (91)$$

where the albedo-scaling is explicitly modified to not tint the base, since the tint \mathbf{F} appears only in the first term via f_{fuzz} . Then, accounting for the coverage weight of the fuzz layer, $\mathbf{F} = \text{fuzz_weight}$, gives

$$\text{layer}(M_{\text{coated-base}}, S_{\text{fuzz}}, \mathbf{F}) \rightarrow \mathbf{F} f_{\text{fuzz}} + \text{lerp}(1, 1 - E_{\text{fuzz}}, \mathbf{F}) f_{\text{coated-base}}. \quad (92)$$

The fuzz shading normal is assumed to inherit from that of the substrate layer, the physical picture being that the fuzz volume settles and conforms to the geometry of the substrate. The substrate is generally a mixture of coat and uncoated base. Thus physically the fuzz model should be evaluated with each of the `geometry_coat_normal` and `geometry_normal` separately (if they differ), and the final result blended according to the `coat_weight`. As a practical consideration, it may be more convenient and efficient to instead approximate the fuzz normal by interpolating the coat and base normal according to `coat_weight`.

Future work: roughening heuristics The scattering within the fuzz layer will have the effect of roughening the appearance of the substrate beneath it. A simple suggested approximation for this can be adapted from the formula used to model the coat roughening, in Equation 86. If we consider the fuzz layer to generate roughening by scattering, we can approximate its effective roughness as being proportional to the albedo of the layer, as well as to the tint color (since darker fuzz will physically scatter less and absorb more). This leads to the following heuristic for the modified roughness r'_B of the substrate lobe:

$$r'_B = \min(1, r_B^4 + 2R_F^4)^{\frac{1}{4}} \quad \text{with } R_F = \text{lum}(\mathbf{F} E_{\text{fuzz}}), \quad (93)$$

where r_B is the original substrate roughness, and $\text{lum}(\dots)$ computes the luminance of the RGB argument. This should be applied to both the coat (if present) and the base lobes. If both the fuzz and the coat are present, then the base lobe roughness will be broadened by both the coat and fuzz formulas successively. The presence weights of the fuzz and coat should be accounted for appropriately. Figure 38 shows the approximate effect of varying the fuzz roughness on the appearance of a smooth [metallic base](#) with a fuzz layer using this heuristic.

The described fuzz roughening heuristic above ignores the fact that, in reality or in a full simulation, the directional distribution of the transmitted light is more complex: some light transmits completely through the fuzz layer without interacting, and the light that is scattered is scattered anisotropically (i.e., preferentially in the backward direction). This should be taken into account when combining with the base lobe.

As for the [coat roughening](#) effect, we would like to provide such improved heuristics in the future.



Figure 38: Appearance of a smooth metallic base with a fuzz layer (with textured weight) modeling dust of varying roughness, for $r = 0.7$ (left), $r = 0.8$ (middle left), $r = 0.9$ (middle right), and $r = 1.0$ (right).

Future work: generalized fuzz We would find a slightly more generalized model of fuzz a useful improvement, though this is not presently available.

In the model of Zeltner, Burley, and Chiang [ZBC22], the optical depth is at most 1, which means the opacity at normal incidence is at most $1 - \exp(-1) = 0.63$, i.e., at least 36% of the light gets through. In really thick dust, the opacity could be arbitrarily close to 1. If the model were to set the optical depth to a higher value (e.g., 3), thicker fuzz could be represented, though that would make the fuzz opacity more uniform across all viewing angles. Ideally, the optical depth would be exposed as a parameter. In other words, it would be helpful to be able to control the density and thickness in order to make a fully opaque layer of dust.

Ideally the weight parameter $F = \text{fuzz_weight}$ would control the total optical depth τ of the slab via $\tau = -\ln(1 - F)$, so weight 1 would be like scattering from a half-space of fuzz of the given microflake roughness. That would increase the brightness of the low-roughness fuzz, compared to the current look which assumes an optical depth 1 in all cases. The weight then also functions naturally as the main control for the fuzz effect, instead of having to treat it as a coverage weight.

Ideally the fuzz color would also correspond to the true multi-scatter albedo (at, say, normal incidence), from which the single-scatter albedo of the flakes is derived, rather than an ad hoc tint multiplier applied to a purely scattering volume.

7 THIN-WALLED MODE

If the `geometry_thin_walled` boolean is enabled, then the surface is assumed to be in a “thin-walled” mode. In this case, we make the assumption that the surface is essentially the same as the bulk structure but mirrored around the base, with the slabs at the base assumed to be thin enough that macroscopically the material can be treated as a two-dimensional sheet with no interior. This sheet thus appears identical viewed from either side.

A convenient rough approximation of this is to ignore the coat and fuzz layers on the underside, and assume that the surface is automatically oriented so that the incident ray is always entering the surface from the top, so the surface appears the same viewed from above or below. Though a crude approximation, this is quite convenient as the light transport can always be dealt with essentially the same as in the bulk case except with a modified base layer (i.e., incident rays enter top-down, and leave from the base layer without further interaction).

As noted earlier, `geometry_opacity` makes physical sense as a fractional value in the thin-walled case (unlike the bulk case), controlling the overall presence weight of the thin wall.

The metal and glossy-diffuse slabs remain but are considered double-sided, with the top BRDF mirrored to the bottom (and a totally opaque, albeit infinitesimally thin, interior). Emission should apply to both sides of a thin-walled surface. This needs to be taken into account in importance sampling of direct illumination; we can no longer assume that emissive geometry only emits from the front. The thin-walled interpretation of the `translucent_base` and `subsurface` differ.

7.1 Thin-wall dielectric

In thin-walled mode, the `translucent-base` slab can be considered an infinitesimally thin sheet of dielectric (with an embedded absorbing but non-scattering medium), with the BSDF $f_{\text{dielectric}}$ on both sides. A ladder of inter-reflections occurs inside this slab, producing a reflected lobe and undeflected refracted lobe. The `transmission_color` can be assumed to give the transmittance through the thin sheet at normal incidence due to absorption (with `transmission_depth` ignored). In the smooth case, the BRDF and BTDF of this sheet can be solved exactly by summing over a geometrical series of terms containing Fresnel and absorption factors, and this can be extended to a good approximation of the rough case by appropriately roughening the transmission lobe (as described by Kulla and Conty [KC17]). This model of thin-walled glass is a cheaper, much more convenient way to render windows than a finite thickness non-thin-walled mesh.

This model is similar to the `ThinDielectricBxDF` of PBRT, except with absorption and roughness accounted for [KC17; PJH16]. The roughening effect of the thin wall can be modeled by considering the broadening of the incident and transmitted rays. Belcour [Bel18] provides a useful heuristic for this, but a more complete model



Figure 39: Thin-wall glass, with `specular_roughness` varying over 0.1–0.5. Notice the absence of refraction.



Figure 40: Varying the `subsurface_anisotropy` over $-1, -0.5, 0$ (default), $0.5, 1$. Note that the specular reflection of the embedding dielectric slab is present.

would be a welcome advancement. Figure 39 shows the effect of varying the `specular_roughness` parameter on the appearance of a thin-wall glass material.

7.2 Thin-wall subsurface

The subsurface slab is considered to degenerate into an infinitesimally thin sheet of dense scattering material (bounded by dielectric interfaces $f_{\text{dielectric}}$), which scatters a fraction $S = \text{subsurface_color}$ of the incident light, split between a diffuse reflection lobe f_{diffuse}^R and diffuse transmission lobe f_{diffuse}^T according to $g = \text{subsurface_scatter_anisotropy} \in [-1, 1]$. That is, where f_+, f_- are albedo 1 diffuse lobes in the positive and negative hemisphere, respectively:

$$\begin{aligned} f_{\text{diffuse}}^R &= \frac{1}{2}S(1-g)f_+, \\ f_{\text{diffuse}}^T &= \frac{1}{2}S(1+g)f_-. \end{aligned} \quad (94)$$

This ensures total energy conservation, i.e., the sum of the reflection and transmission albedos is less than 1:

$$E_R[f_{\text{diffuse}}^R] + E_T[f_{\text{diffuse}}^T] = S \leq 1. \quad (95)$$

At the default of zero anisotropy ($g = 0$) the energy is balanced equally between diffuse reflection and transmission. The diffuse transmission lobe shape (in both hemispheres) is assumed to be controlled by the `base_diffuse_roughness` parameter. Typically the diffuse lobes f_+, f_- will be represented by an Oren–Nayar (EON) lobe flipped into the appropriate hemisphere. This model is useful for rendering cases such as light scattering through a thin sheet of paper.

The thin-wall subsurface is interpreted as being embedded with a finite (but small) thickness dielectric slab. Thus a specular lobe is generated as usual due to reflection from this dielectric interface, with the IOR of the slab given by `subsurface_ior`. Similarly to the glossy-diffuse slab, the effective albedo of diffuse base should be unaffected by the physical effect of inter-reflections in the dielectric slab. (This detail needs clarification in the existing OpenPBR specification, and is being improved.)

Figure 40 shows the effect of varying the `subsurface_anisotropy` parameter on the appearance of a thin-wall subsurface material. The specular reflection from the dielectric slab is visible in all cases, and the diffuse lobes are anisotropic, with the anisotropy increasing from left to right.

Future work: generalized thin-wall model Currently we treat the thin-wall limit of the translucent volume and the subsurface as two distinct cases (essentially absorbing only and scattering-dominated, respectively). However, it would be useful to have a more general model where the thin-wall interior medium is a scattering volume with configurable transmittance and albedo. If the scattering is not totally dominant then there is an intermediate regime between “thin-wall dielectric” (which roughens due to the refractive boundaries but does not scatter), and “thin-wall subsurface” that produces essentially diffuse reflection and transmission lobes. However, at present, there isn’t a well-defined model for this intermediate case. Furthermore, this generalized thin-wall model would make more sense in the context of a unified base volume, which is also considered future work.

It should also be noted that there currently isn’t a completely satisfactory model for the case of the described absorbing only thin-wall dielectric with *rough* boundaries, for example the `ThinDielectricBxDF` of PBRT [PJH16] simplified to the case of smooth boundaries. There are suggested heuristics [KC17; Bel18] but a fully worked out model would be beneficial.

Future work: Two-sided thin-walled materials We do not currently explicitly support the case of a thin-walled material where the properties differ on each side, for example a leaf represented by thin-walled SSS with fuzz on the bottom and a coat on top (but no coat on the bottom, and no fuzz on top).

A more general interpretation would be to allow each side of the thin wall to be defined by a distinct thin-walled OpenPBR Surface (one associated with the top, in the direction of the normal, and one with the bottom in the opposite direction) so the material consists of two thin walls sandwiched together, and the light propagates through the whole structure. Then if there are different [translucent-base](#) or [subsurface](#) parameters on each side, these need to be resolved somehow (e.g., the parameters could be blended, or the light transport could model the presence of the two distinct thin layers of dielectric).

Defining this more general thin-walled model is a topic for future work, and we would like to hear from the community about whether such a more general thin-walled model would be useful for them, or perhaps it is niche enough to not be needed in an uber-shader and a bespoke shader would instead be sufficient.

8 CONCLUSION AND FUTURE WORK

In this report, we have given a detailed technical overview of the OpenPBR model, complementary to the official specification available at <https://academysoftwarefoundation.github.io/OpenPBR/>. The following future investigations are planned for the OpenPBR model:

- Continued interaction with the CG community to help adoption, and gather feedback and suggestions for improvements to the model, ensuring it meets the needs of artists and developers in various industries.
- Further alignment and synchronization of the OpenPBR model with the MaterialX standard, ensuring the features required to implement OpenPBR are fully supported in MaterialX.
- Completion of the [hazy-specular](#) and [retro-reflection](#) parameterizations.
- Further investigation of the proposed [decoupling](#) of the metallic and dielectric parameters to avoid certain artifacts.
- Development of a more comprehensive [generalized](#) fuzz model that accounts for varying optical depths and anisotropic scattering. Also, further work on developing more accurate physically based heuristics for [roughening](#) effects (for both coat and fuzz).
- Thin-walled materials: further investigation of a [generalized thin-wall](#) model, and possible support for [two-sided thin-walled](#) materials.
- There is interest in adding explicit support for a model for rendering glints [[Jak+14b](#); [Yan+14](#)] or flakes [[AK16](#)], which are important effects for certain use cases, such as car paint.
- Development (currently in collaboration with Palmqvist [[Pal25](#)]) of a set of example materials suitable as presets, and for demonstration of the model’s capabilities.
- More work on facilitating translation of the OpenPBR model to and from other shading models (to provide an upgrade path, or to map the model to existing shading models more suitable for certain use cases).
- Further elucidation and exploration of the use of the model in real-time rendering applications, such as video games or interactive simulations, to achieve high-quality visuals while maintaining performance.
- Potential extension of the effort to include non-surface material models in need of standardization, such as volumetric materials or hair.

9 ACKNOWLEDGEMENTS

We thank our fellow co-authors of the OpenPBR specification for extensive discussions over the past three years: Zap Andersson, Paul Edmondson, Julien Guertault, Adrien Herubel, Alan King, Andréa Machizaud, Frédéric Servant and Jonathan Stone.

We would also like to thank the following individuals for their valuable feedback and contributions: François Beaune, Henrik Edstrom, Eugene d’Eon, Jerry Gamache, Iliyan Georgiev, Dhruv Govil, Lee Griggs, Larry Gritz, Niklas Harrysson, Miloš Hašan, Thiago Ize, Lee Kerley, Chris Kulla, Vladimir Koylazov, Bernard Kwok, Anders Langlands, Frankie Liu, Thomas Makryniotis, Thomas Mansencal, Arnon Marcus, André Mazzone, Nikie Monteleone, Michael Nickelsky, Anton Palmqvist, Nick Porcino, Guido Quaroni, Nathan Reed, Anthony Salvi, Brian Sharpe, Rob Slater, Lukas Stockner, Masuo Suzuki, Brecht van Lommel, Andrea Weidlich and Nicolas Wirmann.

The “shader playground” renders shown in [Figure 1](#) and [Figure 28](#) were created by Nikie Monteleone.

REFERENCES

- [AK16] A. Atanasov and V. Koylazov. “A practical stochastic algorithm for rendering mirror-like flakes”. In: *ACM SIGGRAPH 2016 Talks*. SIGGRAPH ’16. Anaheim, California: Association for Computing Machinery, 2016. ISBN: 9781450342827. DOI: [10.1145/2897839.2927391](https://doi.org/10.1145/2897839.2927391). URL: <https://doi.org/10.1145/2897839.2927391>.
- [And16] Z. Andersson. *Physical Material - Whitepaper*. Jan. 2016. DOI: [10.13140/RG.2.2.18987.21280](https://doi.org/10.13140/RG.2.2.18987.21280).
- [APS00] M. Ashikhmin, S. Premoze, and P. Shirley. “A microfacet-based BRDF generator”. In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2000, New Orleans, LA, USA, July 23-28, 2000*. Ed. by J. R. Brown and K. Akeley. ACM, 2000, pp. 65–74. DOI: [10.1145/344779.344814](https://doi.org/10.1145/344779.344814). URL: <https://doi.org/10.1145/344779.344814>.
- [BB17] L. Belcour and P. Barla. “A practical extension to microfacet theory for the modeling of varying iridescence”. In: *ACM Trans. Graph.* 36.4 (July 2017). ISSN: 0730-0301. DOI: [10.1145/3072959.3073620](https://doi.org/10.1145/3072959.3073620). URL: <https://doi.org/10.1145/3072959.3073620>.
- [Bel+14] L. Belcour, R. Pacanowski, M. Delahaie, A. Laville-Geay, and L. Eupherte. “BRDF Measurements and Analysis of Retroreflective Materials”. In: *Journal of the Optical Society of America*. JOSA A 31.12 (Dec. 2014), pp. 2561–2572. URL: <https://inria.hal.science/hal-01083366>.
- [Bel18] L. Belcour. “Efficient rendering of layered materials using an atomic decomposition with statistical operators”. In: *ACM Trans. Graph.* 37.4 (July 2018). ISSN: 0730-0301. DOI: [10.1145/3197517.3201289](https://doi.org/10.1145/3197517.3201289). URL: <https://doi.org/10.1145/3197517.3201289>.
- [BPV18] P. Barla, R. Pacanowski, and P. Vangorp. “A Composite BRDF Model for Hazy Gloss”. In: *Computer Graphics Forum* 37.4 (2018), pp. 55–66. DOI: <https://doi.org/10.1111/cgf.13475>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13475>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13475>.
- [Bur+18] B. Burley, D. Adler, M. J.-Y. Chiang, H. Driskill, R. Habel, P. Kelly, P. Kutz, Y. K. Li, and D. Teece. “The Design and Evolution of Disney’s Hyperion Renderer”. In: *ACM Trans. Graph.* 37.3 (July 2018). ISSN: 0730-0301. DOI: [10.1145/3182159](https://doi.org/10.1145/3182159). URL: <https://doi.org/10.1145/3182159>.
- [Bur12] B. Burley. “Physically Based Shading at Disney”. In: *SIGGRAPH Courses: Practical Physically Based Shading in Film and Game Production*. 2012.
- [BW] M. Born and E. Wolf. *Principles of Optics*. 7th ed. Cambridge University Press.
- [Chr15] P. H. Christensen. “An Approximate Reflectance Profile for Efficient Subsurface Scattering”. In: *ACM SIGGRAPH 2015 Talks*. SIGGRAPH ’15. Los Angeles, California: Association for Computing Machinery, 2015. ISBN: 9781450336369. DOI: [10.1145/2775280.2792555](https://doi.org/10.1145/2775280.2792555). URL: <https://doi.org/10.1145/2775280.2792555>.
- [CKB16] M. J.-Y. Chiang, P. Kutz, and B. Burley. “Practical and Controllable Subsurface Scattering for Production Path Tracing”. In: *ACM SIGGRAPH 2016 Talks*. SIGGRAPH ’16. Anaheim, California: Association for Computing Machinery, 2016. ISBN: 9781450342827. DOI: [10.1145/2897839.2927433](https://doi.org/10.1145/2897839.2927433). URL: <https://doi.org/10.1145/2897839.2927433>.
- [DB23] J. Dupuy and A. Benyoub. “Sampling Visible GGX Normals with Spherical Caps”. In: *Computer Graphics Forum* (2023). ISSN: 1467-8659. DOI: [10.1111/cgf.14867](https://doi.org/10.1111/cgf.14867).
- [dEo22] E. d’Eon. *A Hitchhiker’s Guide to Multiple Scattering: Exact Analytic, Monte Carlo and Approximate Solutions in Transport Theory*. Nov. 2022.
- [EK17] A. C. Estevez and C. Kulla. “Production Friendly Microfacet Sheen BRDF”. In: *SIGGRAPH Courses: Practical Physically Based Shading in Theory and Practice*. 2017.

- [ESM01] M. Elias, L. Simonot, and M. Menu. “Bidirectional reflectance of a diffuse background covered by a partly absorbing layer”. In: *Optics Communications* 191.1 (2001).
- [Geo+19] I. Georgiev, J. Portsmouth, Z. Andersson, A. Herubel, A. King, S. Ogaki, and F. Servant. *Autodesk Standard Surface*. Tech. rep. Autodesk, 2019. URL: <https://autodesk.github.io/standard-surface/>.
- [GGP18] J. Guo, Y.-w. Guo, and J.-g. Pan. “A retroreflective BRDF model based on prismatic sheeting and microfacet theory”. In: *Graphical Models* 96 (2018), pp. 38–46. ISSN: 1524-0703. DOI: <https://doi.org/10.1016/j.gmod.2018.01.002>. URL: <https://www.sciencedirect.com/science/article/pii/S1524070318300018>.
- [GHZ18] Y. Guo, M. Hašan, and S. Zhao. “Position-free monte carlo simulation for arbitrary layered BSDFs”. In: *ACM Trans. Graph.* 37.6 (Dec. 2018). ISSN: 0730-0301. DOI: [10.1145/3272127.3275053](https://doi.org/10.1145/3272127.3275053). URL: <https://doi.org/10.1145/3272127.3275053>.
- [GP17] J. Guo and J. Pan. “A physically-based BRDF model for retroreflection”. In: *Proceedings of the Computer Graphics International Conference*. CGI ’17. Yokohama, Japan: Association for Computing Machinery, 2017. ISBN: 9781450352284. DOI: [10.1145/3095140.3095176](https://doi.org/10.1145/3095140.3095176). URL: <https://doi.org/10.1145/3095140.3095176>.
- [Gri+10] L. Gritz, C. Stein, C. Kulla, and A. Conty. “Open Shading Language”. In: *ACM SIGGRAPH 2010 Talks*. SIGGRAPH ’10. Los Angeles, California: Association for Computing Machinery, 2010. ISBN: 9781450303941. DOI: [10.1145/1837026.1837070](https://doi.org/10.1145/1837026.1837070). URL: <https://doi.org/10.1145/1837026.1837070>.
- [Gul14] O. Gulbrandsen. “Artist Friendly Metallic Fresnel”. In: *Journal of Computer Graphics Techniques (JCGT)* 3.4 (Dec. 2014), pp. 64–72. ISSN: 2331-7418. URL: <http://jcgt.org/published/0003/04/03/>.
- [HD24] S. Hillaire and C. De Rousiers. “Authoring Materials That Matter - Substrate in Unreal Engine 5”. In: *ACM SIGGRAPH 2023 Courses*. SIGGRAPH ’23. Los Angeles, California: Association for Computing Machinery, 2024. ISBN: 9798400701450. DOI: [10.1145/3587423.3607881](https://doi.org/10.1145/3587423.3607881). URL: <https://doi.org/10.1145/3587423.3607881>.
- [Hei+15] E. Heitz, J. Dupuy, C. Crassin, and C. Dachsbacher. “The SGGX microflake distribution”. In: *ACM Trans. Graph.* 34.4 (July 2015). ISSN: 0730-0301. DOI: [10.1145/2766988](https://doi.org/10.1145/2766988). URL: <https://doi.org/10.1145/2766988>.
- [Hei+16a] E. Heitz, J. Dupuy, S. Hill, and D. Neubelt. “Real-time polygonal-light shading with linearly transformed cosines”. In: *ACM Trans. Graph.* 35.4 (July 2016). ISSN: 0730-0301. DOI: [10.1145/2897824.2925895](https://doi.org/10.1145/2897824.2925895). URL: <https://doi.org/10.1145/2897824.2925895>.
- [Hei+16b] E. Heitz, J. Hanika, E. d’Eon, and C. Dachsbacher. “Multiple-scattering microfacet BSDFs with the Smith model”. In: *ACM Trans. Graph.* 35.4 (July 2016). ISSN: 0730-0301. DOI: [10.1145/2897824.2925943](https://doi.org/10.1145/2897824.2925943). URL: <https://doi.org/10.1145/2897824.2925943>.
- [Hei14] E. Heitz. “Understanding the Masking-Shadowing Function in Microfacet-Based BRDFs”. In: *Journal of Computer Graphics Techniques (JCGT)* 3.2 (June 2014), pp. 48–107. ISSN: 2331-7418. URL: <http://jcgt.org/published/0003/02/03/>.
- [Hei18] E. Heitz. “Sampling the GGX Distribution of Visible Normals”. In: *Journal of Computer Graphics Techniques (JCGT)* 7.4 (Nov. 2018), pp. 1–13. ISSN: 2331-7418. URL: <http://jcgt.org/published/0007/04/01/>.
- [HK93] P. Hanrahan and W. Krueger. “Reflection from Layered Surfaces due to Subsurface Scattering”. In: *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*. 1st ed. New York, NY, USA: Associa-

- tion for Computing Machinery, 1993. ISBN: 9798400708978. URL: <https://doi.org/10.1145/3596711.3596743>.
- [Hof19] N. Hoffman. “Fresnel Equations Considered Harmful”. In: *Workshop on Material Appearance Modeling*. Ed. by R. Klein and H. Rushmeier. The Eurographics Association, 2019. ISBN: 978-3-03868-080-2. DOI: [10.2312/mam.20191305](https://doi.org/10.2312/mam.20191305).
- [Jak+14a] W. Jakob, E. d’Eon, O. Jakob, and S. Marschner. “A comprehensive framework for rendering layered materials”. In: *ACM Trans. Graph.* 33.4 (July 2014). ISSN: 0730-0301. DOI: [10.1145/2601097.2601139](https://doi.org/10.1145/2601097.2601139). URL: <https://doi.org/10.1145/2601097.2601139>.
- [Jak+14b] W. Jakob, M. Hašan, L.-Q. Yan, J. Lawrence, R. Ramamoorthi, and S. Marschner. “Discrete stochastic microfacet models”. In: *ACM Trans. Graph.* 33.4 (July 2014). ISSN: 0730-0301. DOI: [10.1145/2601097.2601186](https://doi.org/10.1145/2601097.2601186). URL: <https://doi.org/10.1145/2601097.2601186>.
- [KC17] C. Kulla and A. Conty. “Revisiting physically based shading at imageworks”. In: *SIGGRAPH Course, Physically Based Shading 2.3* (2017).
- [Ket+15] L. Kettner, M. Raab, D. Seibert, J. Jordan, and A. Keller. “The Material Definition Language”. In: *Workshop on Material Appearance Modeling*. Ed. by R. Klein and H. Rushmeier. The Eurographics Association, 2015. ISBN: 978-3-905674-83-5. DOI: [10.2312/mam.20151195](https://doi.org/10.2312/mam.20151195).
- [KHE21] P. Kutz, M. Hašan, and P. Edmondson. *Novel Aspects of the Adobe Standard Material*. White Paper. Adobe white paper. Adobe, 2021. URL: <https://helpx.adobe.com/substance-3d-general/adobe-standard-material/asm-technical-documentation.html>.
- [KS01] C. Kelemen and L. Szirmay-Kalos. “A Microfacet Based Coupled Specular-Matte BRDF Model with Importance Sampling”. In: *Eurographics 2001 - Short Presentations*. Eurographics Association, 2001. DOI: [/10.2312/egs.20011003](https://doi.org/10.2312/egs.20011003).
- [Kut+17] P. Kutz, R. Habel, Y. K. Li, and J. Novák. “Spectral and decomposition tracking for rendering heterogeneous volumes”. In: *ACM Trans. Graph.* 36.4 (July 2017). ISSN: 0730-0301. DOI: [10.1145/3072959.3073665](https://doi.org/10.1145/3072959.3073665). URL: <https://doi.org/10.1145/3072959.3073665>.
- [McD18] W. McDermott. “The PBR Guide”. In: *Adobe White Paper* (2018).
- [ON94] M. Oren and S. K. Nayar. “Generalization of Lambert’s reflectance model”. In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH ’94. New York, NY, USA: Association for Computing Machinery, 1994, pp. 239–246. ISBN: 0897916670. DOI: [10.1145/192161.192213](https://doi.org/10.1145/192161.192213). URL: <https://doi.org/10.1145/192161.192213>.
- [Pal25] A. Palmqvist. *Physically Based - The PBR Values Database*. 2025. URL: <https://physicallybased.info>.
- [PJH16] M. Pharr, W. Jakob, and G. Humphreys. *Physically Based Rendering: From Theory to Implementation*. 3rd. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2016. ISBN: 0128006455.
- [PKH25] J. Portsmouth, P. Kutz, and S. Hill. “EON: A practical energy-preserving rough diffuse BRDF”. In: *Journal of Computer Graphics Techniques (JCGT)* 14.1 (Mar. 2025), pp. 116–139. ISSN: 2331-7418. URL: <http://jcgt.org/published/0014/01/06/>.
- [Pol24] M. N. Polyanskiy. “Refractiveindex.info database of optical constants”. In: *Sci. Data* 11.94 (2024). URL: <https://refractiveindex.info>.
- [Rib+17] M. Ribardière, B. Bringier, D. Meneveau, and L. Simonot. “STD: Student’s t-Distribution of Slopes for Microfacet Based BSDFs”. In: *Computer Graphics Forum* 36.2 (2017), pp. 421–429. DOI: <https://doi.org/10.1111/cgf.13137>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13137>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13137>.
- [SB02] C. M. Schmidt and B. Budge. “Simple Nested Dielectrics in Ray Traced Images”. In: *Journal of Graphics Tools* 7.2 (2002), pp. 1–8.

- [SS16] D. Smythe and J. Stone. “MaterialX: An Open Standard for Network-Based CG Object Looks”. In: *MaterialX spec* (2016). URL: <https://materialx.org>.
- [Ste+24] S. Steinberg, R. Ramamoorthi, B. Bitterli, E. d’Eon, L.-Q. Yan, and M. Pharr. “A Generalized Ray Formulation For Wave-Optical Light Transport”. In: *ACM Trans. Graph.* 43.6 (Nov. 2024). ISSN: 0730-0301. DOI: [10.1145/3687902](https://doi.org/10.1145/3687902). URL: <https://doi.org/10.1145/3687902>.
- [SY22] S. Steinberg and L.-Q. Yan. “Rendering of Subjective Speckle Formed by Rough Statistical Surfaces”. In: *ACM Trans. Graph.* 41.1 (Feb. 2022). ISSN: 0730-0301. DOI: [10.1145/3472293](https://doi.org/10.1145/3472293). URL: <https://doi.org/10.1145/3472293>.
- [TR75] T. S. Trowbridge and K. P. Reitz. “Average irregularity representation of a rough surface for ray reflection”. In: *J. Opt. Soc. Am.* 65.5 (May 1975), pp. 531–536. DOI: [10.1364/JOSA.65.000531](https://doi.org/10.1364/JOSA.65.000531). URL: <https://opg.optica.org/abstract.cfm?URI=josa-65-5-531>.
- [Tur19] E. Turquin. *Practical multiple scattering compensation for microfacet models*. White Paper. Industrial Light & Magic, 2019. URL: https://blog.selfshadow.com/publications/turquin/ms_comp_final.pdf.
- [Vea98] E. Veach. “Robust monte carlo methods for light transport simulation”. AAI9837162. PhD thesis. Stanford, CA, USA, 1998. ISBN: 0591907801.
- [Wal+07] B. Walter, S. R. Marschner, H. Li, and K. E. Torrance. “Microfacet models for refraction through rough surfaces”. In: *Proceedings of the 18th Eurographics Conference on Rendering Techniques*. EGSR’07. Grenoble, France: Eurographics Association, 2007, pp. 195–206. ISBN: 9783905673524.
- [Wei23] A. Weidlich. “Reflecting on the appearance of water”. In: *SIGGRAPH Courses: Path tracing in Production: The Path of Water*. 2023.
- [WW07] A. Weidlich and A. Wilkie. “Arbitrarily layered micro-facet surfaces”. In: *Proceedings of the 5th International Conference on Computer Graphics and Interactive Techniques in Australia and Southeast Asia*. GRAPHITE ’07. Perth, Australia: Association for Computing Machinery, 2007, pp. 171–178. ISBN: 9781595939128. DOI: [10.1145/1321261.1321292](https://doi.org/10.1145/1321261.1321292). URL: <https://doi.org/10.1145/1321261.1321292>.
- [Yan+14] L.-Q. Yan, M. Hašan, W. Jakob, J. Lawrence, S. Marschner, and R. Ramamoorthi. “Rendering glints on high-resolution normal-mapped specular surfaces”. In: *ACM Trans. Graph.* 33.4 (July 2014). ISSN: 0730-0301. DOI: [10.1145/2601097.2601155](https://doi.org/10.1145/2601097.2601155). URL: <https://doi.org/10.1145/2601097.2601155>.
- [ZBC22] T. Zeltner, B. Burley, and M. J.-Y. Chiang. “Practical Multiple-Scattering Sheen Using Linearly Transformed Cosines”. In: *ACM SIGGRAPH 2022 Talks*. SIGGRAPH ’22. 2022. DOI: [10.1145/3532836.3536240](https://doi.org/10.1145/3532836.3536240).
- [ZJ18] T. Zeltner and W. Jakob. “The layer laboratory: a calculus for additive and subtractive composition of anisotropic surface reflectance”. In: *ACM Trans. Graph.* 37.4 (July 2018). ISSN: 0730-0301. DOI: [10.1145/3197517.3201321](https://doi.org/10.1145/3197517.3201321). URL: <https://doi.org/10.1145/3197517.3201321>.
- [ZRB14] S. Zhao, R. Ramamoorthi, and K. Bala. “High-order similarity relations in radiative transfer”. In: *ACM Trans. Graph.* 33.4 (July 2014). ISSN: 0730-0301. DOI: [10.1145/2601097.2601104](https://doi.org/10.1145/2601097.2601104). URL: <https://doi.org/10.1145/2601097.2601104>.

A PARAMETERIZATION

Here we provide a summary of the parameters of the OpenPBR layered material model, including their types, ranges, and default values.

Parameter	Type	Range	Default	Units
base_weight	float	[0, 1]	1	-
base_color	color3	[0, 1] ³	(0.8, 0.8, 0.8)	-
base_metalness	float	[0, 1]	0	-
base_diffuse_roughness	float	[0, 1]	0	-
specular_weight	float	[0, ∞]	1	-
specular_color	color3	[0, 1] ³	(1, 1, 1)	-
specular_roughness	float	[0, 1]	0.3	-
specular_roughness_anisotropy	float	[0, 1]	0	-
specular_ior	float	[1, 3]	1.5	-
transmission_weight	float	[0, 1]	0.0	-
transmission_color	color3	[0, 1] ³	(1, 1, 1)	-
transmission_depth	float	[0, ∞)	0	scene units (e.g., meters)
transmission_scatter	color3	[0, 1] ³	(0, 0, 0)	-
transmission_scatter_anisotropy	float	[-1, 1]	0	-
transmission_dispersion_scale	float	[0, 1]	0	-
transmission_dispersion_abbe_number	float	[0, ∞)	20	-
subsurface_weight	float	[0, 1]	0	-
subsurface_color	color3	[0, 1] ³	(0.8, 0.8, 0.8)	-
subsurface_radius	float	[0, ∞)	1	scene units (e.g., meters)
subsurface_radius_scale	color3	[0, 1] ³	(1, 1, 1)	-
subsurface_anisotropy	float	[-1, 1]	0	-
coat_weight	float	[0, 1]	0	-
coat_color	color3	[0, 1]	(1, 1, 1)	-
coat_roughness	float	[0, 1]	0	-
coat_ior	float	[1, 3]	1.6	-
coat_darkening	float	[0, 1]	1	-
fuzz_weight	float	[0, 1]	0.0	-
fuzz_color	color3	[0, 1]	(1.0, 1.0, 1.0)	-
fuzz_normal	normal	-	-	-
emission_weight	float	[0, 1]	0	-
emission_color	color3	[0, ∞) ³	(1, 1, 1)	-
emission_luminance	float	[0, ∞)	1000	nits (cd/m ²)
thin_film_weight	float	[0, 1]	0	-
thin_film_thickness	float	[0, 1]	0.5	micrometers (μm)
thin_film_ior	float	[1.0, 3.0]	1.4	-
geometry_opacity	float	[0, 1]	1.0	-
geometry_thin_walled	bool	{true, false}	false	-
geometry_normal	vector3	-	-	-
geometry_tangent	vector3	-	-	-
geometry_coat_normal	vector3	-	-	-
geometry_coat_tangent	vector3	-	-	-

B A MIXTURE MODEL EXAMPLE

In various practical models such as Disney’s “Principled” shader [Bur12], Autodesk 3ds Max’s Physical Material [And16] and Autodesk Standard Surface [Geo+19], the model is defined as a mixture (i.e., linear combination) of BSDF/BSSRDFs corresponding to the constituent BSDF lobes of the reflection and transmission. The light transport between the layers and overall energy balance is approximated via the mix weights. Given such a representation, the integration into a renderer is relatively straightforward as the individual lobes can be importance sampled according to their albedos and combined with direct lighting estimates via standard techniques such as multiple importance sampling (MIS) [Vea98].

As an example, we will give a brief derivation of a mixture model representation analogous to Autodesk Standard Surface, from the stated material structure of OpenPBR. Following Autodesk Standard Surface, we assume here that layering is implemented via the non-reciprocal albedo-scaling of Equation 6.

Consider first the non-thin-walled case (i.e., `geometry_thin_walled` is false). For brevity, in the following we suppress all the direction arguments, and use the notation of the tree diagram in the [Model](#) section for the weight factors, i.e.:

$$\begin{aligned}
 \alpha &= \text{geometry_opacity} \\
 F &= \text{fuzz_weight} \\
 C &= \text{coat_weight} \\
 M &= \text{base_metalness} \\
 T &= \text{transmission_weight} \\
 S &= \text{subsurface_weight}
 \end{aligned} \tag{96}$$

The base substrate is a mix, which can be mapped to a BSDF as follows (where the [blue](#) color indicates a primitive BSDF lobe):

$$f_{\text{base-substrate}} = \text{lerp}(f_{\text{dielectric-base}}, f_{\text{conductor}}, M) . \tag{97}$$

Similarly, the dielectric-base mix can be written as

$$f_{\text{dielectric-base}} = \text{lerp}(f_{\text{opaque-base}}, f_{\text{translucent-base}}, T) , \tag{98}$$

and the opaque-base mix as

$$f_{\text{opaque-base}} = \text{lerp}(f_{\text{glossy-diffuse}}, f_{\text{subsurface}}, S) . \tag{99}$$

All of $f_{\text{translucent-base}}$, $f_{\text{subsurface}}$ and $f_{\text{glossy-diffuse}}$ represent the BSDF of a microfacet dielectric interface bounding the dielectric interior and its volumetric media. In each case, the BSDF can be represented as the sum of a “primary specular” BRDF f_{specular}^R corresponding to reflection from the dielectric surface without interaction with the internal medium, and a substrate lobe corresponding to the effect of transmission into and scattering within the medium:

$$\begin{aligned}
 f_{\text{translucent-base}} &= f_{\text{specular}}^R + (1 - E[f_{\text{specular}}^R]) f_{\text{specular}}^T , \\
 f_{\text{subsurface}} &= f_{\text{specular}}^R + (1 - E[f_{\text{specular}}^R]) f_{\text{SS}} , \\
 f_{\text{glossy-diffuse}} &= f_{\text{specular}}^R + (1 - E[f_{\text{specular}}^R]) f_{\text{diffuse}} ,
 \end{aligned} \tag{100}$$

where, as described in the [Slabs](#) section, $E[f_X]$ denotes the directional albedo of f_X .

Here, the substrate lobes f_{specular}^T and f_{SSS} are technically BSSRDFs, which model the entry into the internal medium via the dielectric interface, transport of light from entry point to exit points including absorption and scattering processes, and exit from the medium back through the interface, generating both a reflection and a transmission component. The “specular” BTDF/BSSRDF f_{specular}^T corresponds to transmission into the medium parameterized in the [Translucent base](#) section, and BSSRDF f_{SSS} corresponds to transmission into the medium parameterized in the [Subsurface](#) section. In the case of $f_{\text{glossy-diffuse}}$, the BSSRDF degenerates into the BRDF f_{diffuse} as described in the [Glossy-diffuse](#) section.

Note that in this albedo-scaling approximation, the transmission Fresnel factor associated with f_{specular}^T and f_{SSS} can be *omitted* as the energy conservation of the dielectric BSDF as a whole is maintained automatically, even without explicit multiple-scattering compensation or in the presence of modifications to the reflection Fresnel factor via `specular_color`.

Since f_{specular}^R appears in each of the three component slabs of the [dielectric base](#), it follows that on collecting terms, $f_{\text{dielectric-base}}$ reduces to

$$f_{\text{dielectric-base}} = f_{\text{specular}}^R + (1 - E[f_{\text{specular}}^R]) f_{\text{dielectric-base}}^T, \quad (101)$$

where $f_{\text{dielectric-base}}^T$, the total effective transmission lobe of the dielectric base, is

$$\begin{aligned} f_{\text{dielectric-base}}^T &= T f_{\text{specular}}^T + (1 - T) (S f_{\text{SSS}} + (1 - S) f_{\text{diffuse}}) \\ &= \text{lerp} \left(\text{lerp}(f_{\text{diffuse}}, f_{\text{SSS}}, S), f_{\text{specular}}^T, T \right). \end{aligned} \quad (102)$$

Next, the coat is layered on top of the base substrate with the coverage weight C , where the BRDF of the coat dielectric interface is taken to be f_{coat} , with a transmittance T_{coat} (applied both on entry and exit). As in Equation 15, this can be expressed as

$$f_{\text{coated-base}} = C f_{\text{coat}} + \text{lerp} \left(1, T_{\text{coat}}^2 (1 - E[f_{\text{coat}}]), C \right) f_{\text{base-substrate}}. \quad (103)$$

Similarly, the fuzz layer is applied with coverage weight F (with albedo-scaling adjusted to account for the gray transmission according to Equation 92), producing

$$f_{\text{surface}} = F \overline{f_{\text{fuzz}}} + \text{lerp} \left(1, 1 - E[\overline{f_{\text{fuzz}}}], F \right) f_{\text{coated-base}}, \quad (104)$$

where $\overline{f_{\text{fuzz}}}$ is the fuzz BRDF with the tint color set to white.

Finally, the opacity mix operation is applied, producing

$$f_{\text{PBR}} = \alpha f_{\text{surface}} + (1 - \alpha) f_{\text{transparent}}, \quad (105)$$

where $f_{\text{transparent}}$ is understood to denote a delta-function BSDF corresponding to the absence of any surface interaction.

If E represents the isotropic emission luminance from the base, then the total EDF lobe L_e can be modeled according to the absorption in the coat layer as

$$\begin{aligned} L_e &= (1 - C) E + C T_{\text{coat}}^2 E \\ &= \text{lerp}(1, T_{\text{coat}}^2, C) E, \end{aligned} \quad (106)$$

which, in principle, is a function of direction due to the varying transmittance. This can also be thought of as a lobe in its own right (representing light self-emitted, rather than reflected or transmitted).

To summarize, we have thus expressed the model as the following linear combination of component BRDF/BTDF/BSSRDF lobes (and a separate EDF lobe):

$$\begin{aligned} f_{\text{PBR}} &= \text{lerp}(f_{\text{transparent}}, f_{\text{surface}}, \alpha), \\ f_{\text{surface}} &= F f_{\text{fuzz}} + \text{lerp}\left(1, 1 - E[f_{\text{fuzz}}], F\right) f_{\text{coated-base}}, \\ f_{\text{coated-base}} &= C f_{\text{coat}} + \text{lerp}\left(1, T_{\text{coat}}^2 (1 - E[f_{\text{coat}}]), C\right) f_{\text{base-substrate}}, \\ L_e &= \text{lerp}(1, T_{\text{coat}}^2, C) E, \\ f_{\text{base-substrate}} &= \text{lerp}(f_{\text{dielectric-base}}, f_{\text{conductor}}, M), \\ f_{\text{dielectric-base}} &= f_{\text{specular}}^R + (1 - E[f_{\text{specular}}^R]) f_{\text{dielectric-base}}^T, \\ f_{\text{dielectric-base}}^T &= \text{lerp}(\text{lerp}(f_{\text{diffuse}}, f_{\text{SSS}}, S), f_{\text{specular}}^T, T). \end{aligned}$$

Where the component lobes are listed below:

Lobe name	Lobe symbol	Description	Parameters
Transparency	$f_{\text{transparent}}$	pass-through (delta BTDF)	as in the Geometry section
Coating	f_{coat}	coat BRDF	as in the Coat section
Emission	L_e	emission EDF	as in the Emission section
Metal	$f_{\text{conductor}}$	metal BRDF	as in the Metal section
Specular reflection	f_{specular}^R	specular BRDF	as in the Dielectric base section
Specular transmission	f_{specular}^T	specular BTDF/BSSRDF	as in the Translucent base section
Fuzz	f_{fuzz}	fuzz BRDF	as in the Fuzz section
Subsurface scattering	f_{SSS}	subsurface BSSRDF	as in the Subsurface section
Diffuse reflection	f_{diffuse}	diffuse BRDF	as in the Glossy-diffuse section

Table 2: Primitive BSDF/BSSRDF/EDF lobes in the OpenPBR mixture model.

C SIMILARITY THEORY OF SUBSURFACE ANISOTROPY

For completeness, we give here a brief derivation of the similarity relation of Equation 50 for the scattering albedo α of an anisotropic medium, given the isotropic medium scattering albedo α^* , extinction coefficient of the anisotropic medium μ_t and anisotropy g . This formulation allows the mapping between single and multi-scatter albedo to be determined for the case of an isotropic medium, then adapted to the case of any anisotropic medium.

As noted, there is the following “similarity relation” [ZRB14; Bur+18] between the scattering coefficients in media with different anisotropies g and g^* :

$$\mu_s(1-g) = \mu_s^*(1-g^*), \quad (107)$$

where all starred quantities refer to the isotropic medium, and unstarred quantities refer to the anisotropic medium. We can set $g^* = 0$ (i.e., corresponding to the isotropic medium), and g is the anisotropy of the medium we are interested in. The absorption coefficients of the two media can be assumed to be equal (i.e., $\mu_a = \mu_a^*$).

The scattering albedo α of the anisotropic medium can then be computed as follows. From the definition of the extinction coefficients in the isotropic and anisotropic cases, we have:

$$\mu_t = \mu_a + \mu_s, \quad (108)$$

$$\mu_t^* = \mu_a^* + \mu_s^*. \quad (109)$$

Subtracting the two equations, since the absorption coefficients are equal we have

$$\mu_t - \mu_t^* = \mu_s - \mu_s^*. \quad (110)$$

Applying $\mu_s = \alpha\mu_t$ and $\mu_s^* = \alpha^*\mu_t^*$:

$$1 - \alpha = (1 - \alpha^*) \frac{\mu_t^*}{\mu_t}. \quad (111)$$

Now, applying the similarity relation of Equation 107, we have

$$\begin{aligned} \mu_t &= \mu_a + \mu_s \\ &= \mu_a^* + \mu_s^* / (1-g) \\ &= \mu_t^* \left[1 - \alpha^* + \frac{\alpha^*}{1-g} \right]. \end{aligned}$$

This allows the unknown μ_t^* to be eliminated from Equation 111, and thus solving for α we obtain Equation 50, the relation between the scattering albedos of the anisotropic medium (α) and the isotropic medium (α^*):

$$\alpha = \frac{\alpha^*}{1 - g(1 - \alpha^*)}.$$

This allows the single-scattering albedo, which generates the required observed color $E_{\text{multi-scatter}}$, to be determined in the isotropic case (via a variety of formulas, such as Equation 48), and then adapted to general anisotropy via Equation 50.

D F82-TINT MODEL FITS FOR A VARIETY OF REAL METALS

We present here fits to the F82-tint model used for the [metallic](#) slab. These were computed using publicly available tabulated IOR data [Pol24]. Example programs for computing these fits can be found here:

- <https://github.com/portsmouth/F82-tint-generator/>
- <https://github.com/peterkutz/metal-colors>
- <https://github.com/natyh/material-params>

Metal	F0 (sRGB)	F0 (ACEScg)	F82-tint (sRGB)	F82-tint (ACEScg)	F0 (sRGB)	F82-tint (sRGB)
aluminium	0.916 0.923 0.924	0.918 0.922 0.923	0.91 0.936 0.959	0.921 0.934 0.955		
beryllium	0.539 0.533 0.534	0.537 0.534 0.534	0.731 0.738 0.755	0.734 0.738 0.752		
brass	0.962 0.713 0.464	0.857 0.728 0.502	0.971 0.994 1.019	0.979 0.992 1.015		
caesium	0.702 0.555 0.256	0.633 0.561 0.299	1.087 1.18 1.44	1.128 1.175 1.394		
chromium	0.654 0.685 0.701	0.666 0.682 0.698	0.688 0.728 0.798	0.706 0.726 0.788		
cobalt	0.699 0.704 0.671	0.699 0.703 0.676	0.727 0.772 0.823	0.747 0.77 0.815		
copper	0.932 0.623 0.522	0.811 0.643 0.542	0.982 0.947 0.945	0.97 0.95 0.946		
germanium	0.5 0.517 0.465	0.504 0.515 0.472	0.62 0.653 0.701	0.635 0.651 0.694		
gold	1.059 0.773 0.307	0.929 0.788 0.374	0.971 1.018 0.994	0.987 1.013 0.997		
iridium	0.745 0.734 0.704	0.739 0.734 0.709	0.759 0.781 0.81	0.768 0.78 0.806		
iron	0.53 0.513 0.494	0.523 0.514 0.497	0.765 0.767 0.802	0.767 0.768 0.797		
lead	0.626 0.64 0.693	0.633 0.64 0.686	0.758 0.773 0.799	0.765 0.772 0.795		
lithium	0.916 0.89 0.807	0.902 0.89 0.819	0.985 0.998 1.027	0.991 0.998 1.023		
magnesium	0.956 0.953 0.95	0.955 0.953 0.951	0.954 0.964 0.977	0.958 0.963 0.975		
manganese	0.606 0.592 0.573	0.6 0.592 0.576	0.796 0.834 0.889	0.813 0.832 0.88		
mercury	0.781 0.78 0.778	0.781 0.78 0.779	0.813 0.852 0.902	0.83 0.85 0.895		
molybdenum	0.589 0.612 0.594	0.597 0.61 0.596	0.683 0.696 0.726	0.689 0.695 0.721		
nickel	0.697 0.641 0.563	0.672 0.643 0.575	0.815 0.834 0.871	0.824 0.833 0.865		
palladium	0.734 0.704 0.662	0.721 0.705 0.669	0.811 0.836 0.872	0.822 0.835 0.866		
platinum	0.765 0.73 0.676	0.749 0.732 0.684	0.793 0.815 0.84	0.802 0.814 0.836		
potassium	0.983 0.956 0.906	0.971 0.957 0.914	1.002 1.011 1.032	1.006 1.01 1.029		
rubidium	0.919 0.859 0.747	0.892 0.862 0.764	1.016 1.042 1.105	1.028 1.041 1.096		
silicon	0.345 0.369 0.426	0.357 0.368 0.418	0.72 0.701 0.663	0.711 0.702 0.669		
silver	0.991 0.985 0.974	0.988 0.985 0.975	0.994 0.995 0.998	0.995 0.995 0.998		
sodium	0.977 0.962 0.936	0.97 0.962 0.94	0.998 1.002 1.011	1. 1.002 1.01		
steel	0.669 0.639 0.598	0.656 0.64 0.604	0.789 0.823 0.87	0.803 0.821 0.863		
titanium	0.441 0.4 0.361	0.424 0.403 0.367	0.865 0.906 0.946	0.882 0.903 0.94		
tungsten	0.537 0.536 0.519	0.536 0.535 0.522	0.695 0.704 0.714	0.699 0.703 0.713		
zinc	0.808 0.844 0.865	0.822 0.842 0.861	0.762 0.833 0.896	0.793 0.829 0.887		

E IMPLEMENTING THIN-FILM IRIDESCENCE FROM FIRST PRINCIPLES

In this section, we bring together the core math and physics required to implement thin-film iridescence from first principles. The exposition aims to bridge the gap between theory and practice – combining foundational math often omitted in rendering literature with practical guidance for translating formulas into code and integrating them into a renderer.

Rather than the approach of Belcour and Barla [BB17] in which the thin-film interference effects were pre-integrated over color matching functions, we favour a more direct and general “locally spectral” approach that computes reflectance per light path by evaluating the full Fresnel and Airy interference stack – including complex amplitudes, polarizations, and phase shifts – at specific wavelengths sampled per path. This can begin with fixed red, green, and blue wavelengths, but better results are achieved by stochastically sampling wavelengths from approximate camera sensitivity curves (as described in Section 3.1.3). This enables convergence to neutral gray for very thick films and avoids the high-frequency color banding that fixed RGB wavelengths can produce.

The same wavelengths can also be reused to model dispersion, while all other BSDF components are free to ignore them and operate in RGB as usual. This approach uses only the Airy summation (Equation 3 from Belcour and Barla [BB17]) – a method whose origins trace back to 19th-century studies of thin-plate interference by G.B. Airy and others – but requires additional per-wavelength computations and assembling the necessary formulas from multiple sources rather than a single reference. We attempt to collect the necessary equations and implementation instructions here for convenience.

The geometry of the thin film is assumed to follow the model described in Belcour and Barla [BB17], as illustrated in Figure 41. Table 3 summarizes the relevant mathematical quantities. The film is taken to be applied at each microfacet of the base substrate, so can be assumed to be a flat slab with normal given by the local micronormal \mathbf{m} , and thickness d . The film is assumed to have real IOR n_2 and thickness d , and to be surrounded by ambient medium above with (real) IOR n_1 , and substrate material with IOR n_3 (conducting or dielectric, where n_3 is complex in the conducting case). The incident light from direction ω_i is at angle θ_1 to \mathbf{m} , while the light refracted into the film makes angle θ_2 with \mathbf{m} . Given this setup, we need to determine the albedo of the resulting configuration for incident light with wavelength λ , accounting for interference effects.

Symbol	Description
n_1	Refractive index of incident medium (real; e.g., air)
n_2	Refractive index of thin film (real)
n_3	Refractive index of substrate (real for dielectrics, complex for conductors)
θ_1, θ_2	Angles from surface normal in incident medium and thin film
d	Film physical thickness
λ	Wavelength in vacuum
$r_{ij}^{(s,p)}, t_{ij}^{(s,p)}$	Complex Fresnel amplitude coefficients between media $i \rightarrow j$
(s, p)	Polarization state index of amplitudes (s-polarized or p-polarized)
δ	Round-trip phase shift in the film

Table 3: Nomenclature for thin-film iridescence.

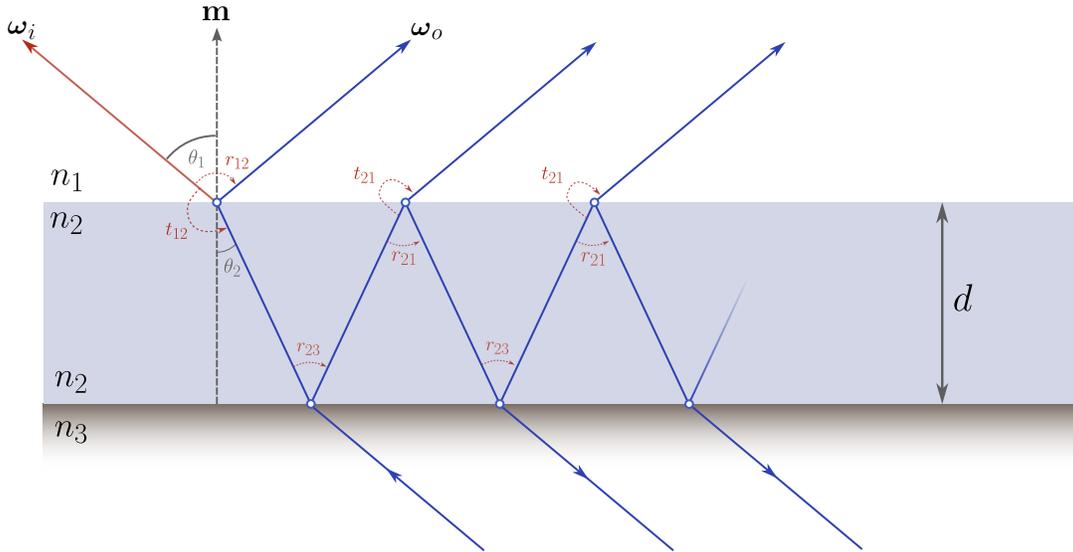


Figure 41: Schematic of the thin-film model (following [BB17]).

Prerequisites and assumptions

- A complex-number type is required, supporting addition, multiplication, complex square root, and $\exp(i\theta)$, preserving both magnitude and phase.
- Refractive indices (IORs) must be stored as complex values; n_i may be real (for dielectrics, written as $n_i + i, 0$) or complex (for conductors, $n_i + i, k_i$ where $k_i > 0$ represents the extinction coefficient).
- Amplitude Fresnel coefficients r_{ij} and t_{ij} must be stored as complex values.
- Input wavelengths: one or more λ values (e.g., monochromatic, RGB triplet, or spectral samples).
- All formulas are evaluated per wavelength.
- Wavelength λ and film thickness d must use consistent units (e.g., nanometers or microns).
- All angles θ are in radians and measured from the surface normal.

1. Angles and Snell's law For each interface $i \rightarrow j$, given a real incidence angle θ_i , compute the transmitted angle using Snell's law:

$$\sin \theta_j = \frac{n_i}{n_j} \sin \theta_i, \quad \cos \theta_j = \sqrt{1 - \sin^2 \theta_j}.$$

The square root must be evaluated using the branch such that $\text{Im}(n_j \cos \theta_j) \geq 0$, ensuring that the transmitted wave behaves correctly for absorbing materials. We assume that the incident medium (n_i) is a real-valued dielectric and that the incidence angle θ_i is real; therefore, $\sin \theta_i$ and $\cos \theta_i$ are real-valued. However, if n_j is complex or if total internal reflection occurs, then $\sin \theta_j$ and $\cos \theta_j$ may become complex (and must be computed using complex arithmetic). If n_j is real and $\sin \theta_j > 1$, then $\cos \theta_j$ is imaginary, indicating total internal reflection

(TIR) and evanescent transmission with no net energy transport. In all cases, the results of Snell's Law can be safely input to the Fresnel equations, which will handle all regimes correctly when evaluated using complex arithmetic.

2. Amplitude Fresnel coefficients Using $\cos \theta_i$ and $\cos \theta_j$ from Step 1, compute for each polarization (s, p) :

$$r_{ij}^{(s)} = \frac{n_i \cos \theta_i - n_j \cos \theta_j}{n_i \cos \theta_i + n_j \cos \theta_j}, \quad t_{ij}^{(s)} = \frac{2 n_i \cos \theta_i}{n_i \cos \theta_i + n_j \cos \theta_j}, \quad (112)$$

$$r_{ij}^{(p)} = \frac{n_j \cos \theta_i - n_i \cos \theta_j}{n_j \cos \theta_i + n_i \cos \theta_j}, \quad t_{ij}^{(p)} = \frac{2 n_i \cos \theta_i}{n_j \cos \theta_i + n_i \cos \theta_j}. \quad (113)$$

Compute for $(i, j) = (1, 2)$ and $(2, 3)$ to get r_{12}, t_{12} and r_{23}, t_{23} . In the TIR regime (for dielectrics), these formulas yield $|r_{ij}| = 1$ and $t_{ij} = 0$ automatically (if and only if they are performed using complex arithmetic; otherwise, TIR must be manually detected and handled with a separate code path).

3. Reciprocity for reverse direction To avoid re-running the Fresnel equations, one can use the following symmetry relations between the reflection and transmission amplitudes [BW]:

$$r_{21}^{(s,p)} = -r_{12}^{(s,p)}, \quad t_{21}^{(s,p)} = t_{12}^{(s,p)} \frac{n_2 \cos \theta_2}{n_1 \cos \theta_1}. \quad (114)$$

(Note that this works even for complex Fresnel coefficients; the minus sign accounts for the phase shift for reflection under the assumption of planar, non-magnetic media.)

4. Phase shift inside the film The round-trip phase shift is

$$\delta = \frac{4\pi n_2 d \cos \theta_2}{\lambda}. \quad (115)$$

5. Airy summation The total reflectance amplitude is computed by summing the contributions of each output reflection mode depicted in Figure 41. This is derived from the following geometric series expansion (known as *Airy summation*) of the reflected wave contributions for the total reflectance amplitude $r_{\text{tot}}^{(s,p)}$ for each polarization (s, p) , where the first term is the direct reflection at the first interface, and the second term accounts for the multiple reflections within the film:

$$\begin{aligned} r_{\text{tot}}^{(s,p)} &= r_{12}^{(s,p)} + t_{12}^{(s,p)} r_{23}^{(s,p)} t_{21}^{(s,p)} e^{i\delta} + t_{12}^{(s,p)} r_{23}^{(s,p)} r_{21}^{(s,p)} r_{23}^{(s,p)} t_{21}^{(s,p)} e^{2i\delta} + \dots \\ &= r_{12}^{(s,p)} + \sum_{k=1}^{\infty} t_{12}^{(s,p)} r_{23}^{(s,p)} \left(r_{21}^{(s,p)} r_{23}^{(s,p)} \right)^{k-1} t_{21}^{(s,p)} e^{ik\delta} \\ &= r_{12}^{(s,p)} + \frac{t_{12}^{(s,p)} r_{23}^{(s,p)} t_{21}^{(s,p)} e^{i\delta}}{1 - r_{21}^{(s,p)} r_{23}^{(s,p)} e^{i\delta}}. \end{aligned} \quad (116)$$

The denominator ensures convergence of the series, provided that $|r_{21}^{(s,p)} r_{23}^{(s,p)} e^{i\delta}| < 1$.

Power reflectance per-polarization follows as

$$R_{s,p} = |r_{\text{tot}}^{(s,p)}|^2. \quad (117)$$

6. Unpolarized reflectance Assuming unpolarized incident light, the total reflectance is the average of the powers of the two polarization states:

$$R = \frac{1}{2}(R_s + R_p). \quad (118)$$

Implementation tips and common pitfalls

- *Complex square root branches:* When computing $\cos \theta_j = \sqrt{1 - \sin^2 \theta_j}$, choose the branch such that $\text{Im}(n_j \cos \theta_j) \geq 0$. This ensures the transmitted wave behaves correctly for absorbing materials, and is essential for energy conservation and reciprocity at interfaces.
- *Transmission prefactor for power:* To convert amplitude transmission coefficients to power, multiply by $\text{Re}(n_j \cos \theta_j / n_i \cos \theta_i)$ for s-polarization (perpendicular), and $\text{Re}(n_j \cos^* \theta_j / n_i \cos \theta_i)$ for p-polarization (parallel). The complex conjugate in the p-term is essential for handling complex angles correctly. This prefactor ensures that $R + T = 1$ at each interface, even when the transmitted medium is absorbing.
- *Interface energy conservation:* Fresnel reflection and transmission at each *interface* must conserve energy, even for lossy media such as metals. This means numerically verifying that $0 \leq R, T \leq 1$ and $R + T = 1$ at every boundary and for all polarizations. For conductors, the transmitted light is absorbed *after* entering the medium, so interface-level energy conservation must still hold.
- *Denominator safety:* Clamp any denominator magnitude to ϵ before division. In particular, the denominator in Equation 116 can approach zero when the reflected waves interfere constructively (a resonance condition), even though the final reflectance remains physically bounded.
- *Phase/magnitude extraction:* You never need to explicitly extract magnitude and phase from the complex amplitude coefficients – compute r via complex arithmetic and convert to power with $|r|^2$ at the end.
- *Early exit for TIR:* At the exterior-film interface, if TIR occurs, reflectance is unity and transmission zero – return $R = 1$ directly without further evaluation.
- *Complex Fresnel unification:* Using complex n covers both dielectrics and conductors, including evanescent waves; a small TIR branch for real dielectrics (e.g., $\sin \theta_t \geq 1$) can shortcut evaluation (a separate function using real arithmetic may also be used for dielectrics if more performant, in which case an early exit is required).
- *Dielectric shortcut:* Alternatively, use a dedicated real-valued Fresnel function for pure dielectrics (with its own TIR early-exit) to avoid complex arithmetic when indices are real. Snell's Law calculations can also use real numbers for dielectrics where applicable.
- *Composite substrates:* For mixed substrates, compute $R_{23}^{(s,p)}$ per component (steps 1–6) and either blend immediately by mixing weights, or return separate dielectric vs. metal reflectances for caller-side blending before Airy summation.
- *$t_{23}^{(s,p)}$ is never needed:* The transmitted amplitude into medium 3 produced by the Fresnel equations is not used in the Airy summation (Equation 116) and can be discarded in all cases, regardless of whether the substrate is a dielectric or conductor.
- *Integration in a renderer:* Use the computed R from Step 6 to replace your microfacet Fresnel term (or to modulate it via the thin-film weight in the material model).

- *Vectorize wavelengths and polarizations*: Evaluate each formula per sample and polarization in parallel (e.g., RGB/spectral, s/p). This helps minimize branching and improves SIMD or GPU parallelism.
- *Energy conservation across layers*: For multi-layer systems (e.g., thin film on metal), the total reflected and transmitted power must satisfy $R + T \leq 1$. Any shortfall from 1 indicates absorption within the film or substrate. If $R + T < 1$ but no absorption is expected, this suggests an implementation error.

Albedo of a thin film Figure 42 plots the albedo of a thin film, computed according to the procedure described, with $n_1 = 1$, $n_2 = 1.4$, real part of $n_3 = 1.8$, and $d = 400$ nm. In the left panel, the wavelength is fixed at 500 nm, and the albedo as a function of angle of incidence shown for a variety of conductor extinction coefficients k .

The albedo tends to 1 as the conductor extinction increases, which is expected physically since in this limit (also known as a *Gires–Tournois etalon*) the Fresnel reflection from the metal tends to 1 leading to no energy loss. Thus a useful check of a thin-film implementation is to verify that a perfectly reflecting metal with a thin film passes a furnace test, regardless of the thin-film properties.

In the right panel, the light is incident along the normal, the conductor absorption is fixed at 5, and the albedo is plotted as a function of wavelength for two different film thicknesses. At high thickness (relative to the wavelength), the albedo exhibits strong oscillations as a function of wavelength. This results in the color fringes disappearing at high thin-film depth (relative to optical wavelengths), as the oscillations integrated over the color matching functions integrate to gray.

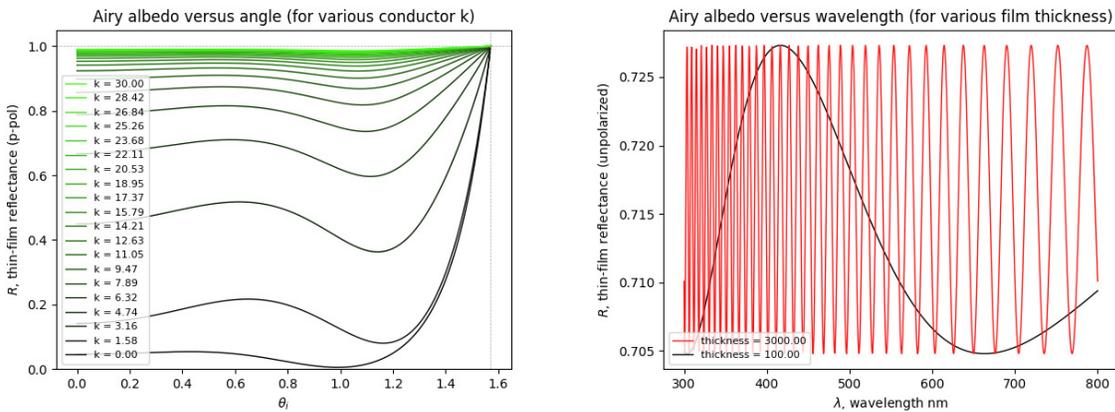


Figure 42: Albedo of a thin film on a conducting substrate, versus angle at 500 nm (left) and versus wavelength at normal incidence (right).

F THE PHYSICS OF COAT DARKENING

We present here some formulas for the physical darkening effect of light transport within a dielectric coat. In the case of a smooth dielectric coat on top of a Lambertian substrate (the so-called “interfaced Lambertian” model), there is an exact solution available, which we describe in detail below. This was derived using radiative transfer theory by Elias, Simonot, and Menu [ESM01], and the result is also stated by d’Eon [dEo22]. The notation used is shown in Figure 43.

The ratio of the IOR of the coat dielectric (n_c) to that of the exterior medium (n_a) is denoted η . The Lambertian base albedo is denoted ρ_d . The dielectric layer absorption is parameterized by the transmittance along the normal direction, T . Light is incident from the upper hemisphere \mathcal{H}_+ at some incident angle cosine μ_i . It enters the dielectric layer, bounces within, then exits back into \mathcal{H}_+ with some outgoing angle cosine μ_o . Due to the azimuthal symmetry, the BRDF must be a function only of these two angle cosines, i.e., $f(\mu_i, \mu_o)$. By reciprocity we must have $f(\mu_i, \mu_o) = f(\mu_o, \mu_i)$. We assume that the incident light is unpolarized, and remains unpolarized after reflection and transmission (i.e., following the usual assumption in radiative transfer⁵, as employed by Elias, Simonot, and Menu [ESM01]). The formulas are all written assuming the spectral frequency is fixed.

We consider here only the case of a perfectly smooth dielectric coat. The extension to the rough case is briefly discussed by d’Eon [dEo22].

In the presence of the Lambertian base, the full BRDF has the form [ESM01]

$$f(\mu_i, \mu_o) = f_s(\mu_i, \mu_o) + f_b(\mu_i, \mu_o) . \quad (119)$$

The first term, denoted $f_s(\mu_i, \mu_o)$, is the specular mirror reflection from the coat (technically a delta function), scaled by the *Fresnel factor* (or *Fresnel reflection coefficient*) $F_R(\mu, \eta)$, which gives the fraction of the energy of an

⁵Though technically this is inconsistent, since even unpolarized incident light will become polarized after Fresnel transmission into the coat. This polarization is ignored and radiation assumed to be unpolarized, as in a standard RTE treatment of the problem.

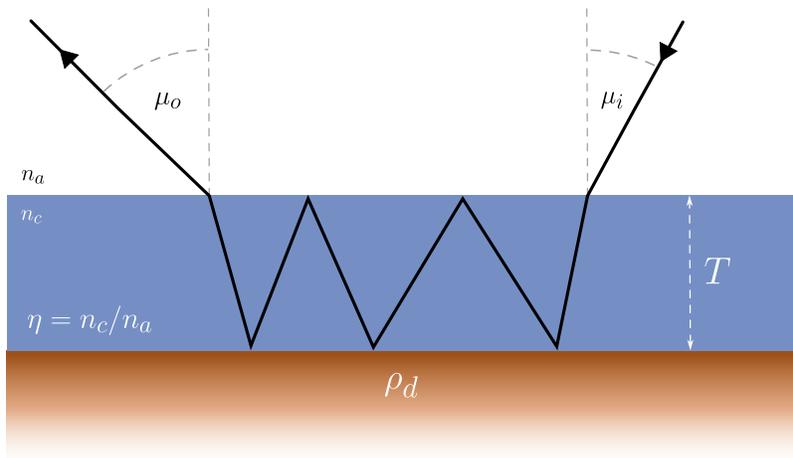
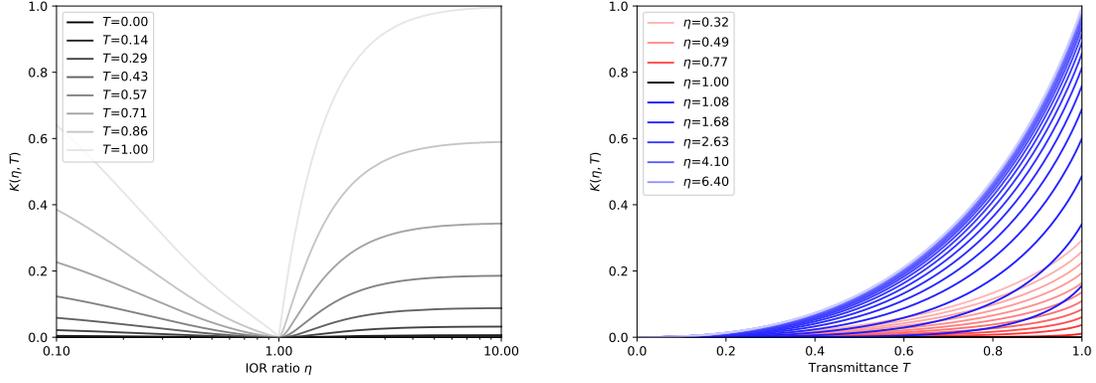


Figure 43: Schematic of the light transport inside a smooth dielectric coat on top of a diffuse base substrate.


 Figure 44: $K(\eta, T)$ versus IOR ratio η or transmittance T .

incident beam from angle-cosine μ which is reflected from a smooth dielectric interface with IOR ratio η . In this case, the directional albedo $R(\mu) = \int_{\mathcal{H}_+} f(\mu, \mu_o) d\omega_o^\perp$ reduces to $F_R(\mu, \eta)$.

The second term is given by

$$f_b(\mu_i, \mu_o) = \frac{\rho_d}{\pi\eta^2} \frac{F_T(\mu_i, \eta)F_T(\mu_o, \eta) T^{\frac{1}{\mu_i} + \frac{1}{\mu_o}}}{1 - \rho_d K(\eta, T)}, \quad (120)$$

where $F_T(\mu, \eta) = 1 - F_R(\mu, \eta)$ is the *Fresnel transmission coefficient*. The two F_T factors account for the transmission into and out of the coat interface, for paths en route to the base. The last factor in the numerator accounts for the coat transmittance along the refracted input and output ray directions, where μ_i^t, μ_o^t are the angle cosines of the incident and output ray refracted into the coat:

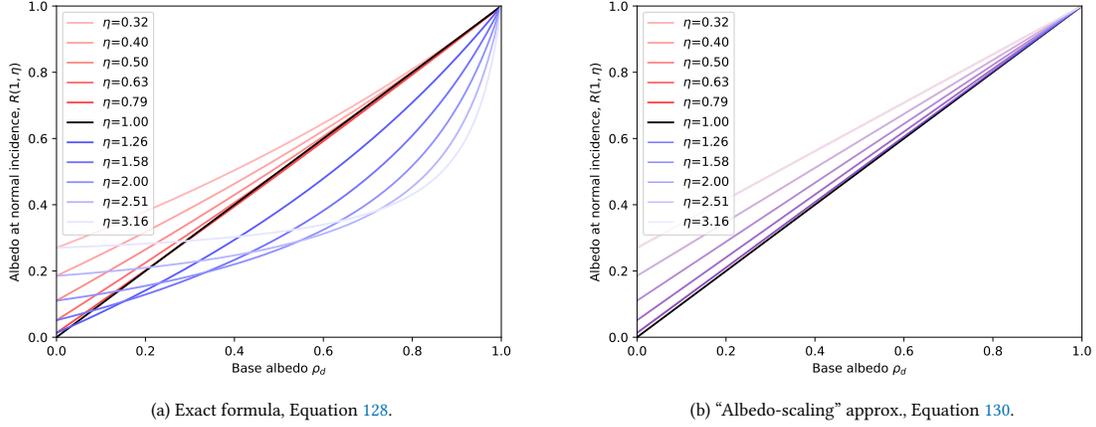
$$\mu_i^t = \sqrt{1 - \frac{1 - \mu_i^2}{\eta^2}}, \quad \mu_o^t = \sqrt{1 - \frac{1 - \mu_o^2}{\eta^2}}. \quad (121)$$

We can assume a solution exists for each, since if either the incident or outgoing directions are in a TIR configuration, then the whole second term is zero since one of the F_T factors is zero.

The denominator accounts for the bounces due to internal reflections from the coat. The factor K in the denominator is known as the *internal diffuse reflection coefficient*, given by

$$K(\eta, T) = 2 \int_0^1 T^{2/\mu} F_R(\mu, 1/\eta) \mu d\mu. \quad (122)$$

It corresponds to the fraction of the energy leaving the base that returns to the base due to internal reflection from the coat. The integral for $K(\eta, T)$ does not have a simple closed-form solution. Its behavior is shown in Figure 44. In the limits $\eta \rightarrow 1$ and $T \rightarrow 0$ (i.e., no coat reflection, or a completely absorbing medium), $K(\eta, T) \rightarrow 0$.


 Figure 45: Clear-coat layer directional albedo at normal incidence, $R(1)$.

The directional albedo of the full BRDF is then given by

$$R(\mu_o) = F_R(\mu_o, \eta) + \frac{\rho_d}{\eta^2} F_T(\mu_o, \eta) \frac{1 - E_F(\eta)}{1 - \rho_d K(\eta, T)}. \quad (123)$$

where $E_F(\eta) = 2 \int_0^1 F_R(\mu, \eta) \mu d\mu$ is the hemispherical (or average) albedo of the Fresnel factor.⁶

Clear coat In the limit $T \rightarrow 1$ (a transparent clear coat), the internal diffuse reflection coefficient reduces to

$$K_0(\eta) = 2 \int_0^1 F_R(\mu, 1/\eta) \mu d\mu = 1 - 2 \int_0^1 F_T(\mu, 1/\eta) \mu d\mu. \quad (125)$$

Applying the following symmetry relation for the Fresnel hemispherical albedo:

$$1 - E_F(\eta) = \eta^2 \left(1 - E_F(1/\eta)\right), \quad (126)$$

this further reduces to

$$K_0(\eta) = 1 - (1 - E_F(\eta))/\eta^2. \quad (127)$$

The directional albedo thus simplifies to (as quoted by d'Eon [dEo22]):

$$R(\mu_o) = F_R(\mu_o, \eta) + \frac{\rho_d}{\eta^2} \frac{(1 - F_R(\mu_o, \eta))(1 - E_F(\eta))}{1 - \rho_d(1 - (1 - E_F(\eta))/\eta^2)}. \quad (128)$$

⁶A good approximation to $E_F(\eta)$ in the limited range $\eta \in [1, 3]$ is given by d'Eon [dEo22], who reports it to be accurate to within 0.15% over that IOR range:

$$E_F(\eta) \approx \ln\left(\frac{10893\eta - 1438.2}{-774.4\eta^2 + 10212\eta + 1}\right). \quad (124)$$

As required, $R(\mu_o) \rightarrow 1$ as base albedo $\rho_d \rightarrow 1$, since in this case no energy is absorbed. Also of course as $\eta \rightarrow 1$, $R(\mu_o) \rightarrow \rho_d$. The corresponding hemispherical albedo is given by

$$E = E_F(\eta) + \frac{\rho_d}{\eta^2} \frac{(1 - E_F(\eta))^2}{1 - \rho_d(1 - (1 - E_F(\eta))/\eta^2)}. \quad (129)$$

In Figure 45a, the directional albedo of the clear coat at normal incidence (i.e., Equation 128 evaluated at $\mu_o = 1$) is plotted as a function of base albedo ρ_d . This quantity represents the observed reflected radiance at normal incidence given unit white uniform illumination. (The black line represents the reflection in the case $\eta = 1$, i.e., no coat, in which case it reduces to the base albedo.)

For comparison in Figure 45b, we plot the so-called ‘‘albedo-scaling’’ approximation of the clear-coat layer albedo, given by

$$R(\mu_o) \approx F_R(\mu_o, \eta) + \rho_d(1 - F_R(\mu_o, \eta)). \quad (130)$$

This corresponds to assuming the clear-coat layer BRDF is approximated by

$$f(\mu_i, \mu_o) \approx f_s(\mu_i, \mu_o) + \frac{\rho_d}{\pi}(1 - F_R(\mu_o, \eta)). \quad (131)$$

This is non-reciprocal, but at least satisfies $R(\mu_o) \rightarrow 1$ as $\rho_d \rightarrow 1$. Note that, in this approximation, the η and $1/\eta$ cases look identical at normal incidence. For the $\eta > 1$ case, the approximation is clearly extremely poor. In both the exact formula and the approximation, for a dark base ($\rho_d = 0$) the reflection is equal to the Fresnel reflection from the coat, and for a white base ($\rho_d = 1$) the reflection is white (i.e., the furnace test passes).

Figure 46 shows the appearance at normal incidence of the exact solution for smooth clear coat (under uniform white sky illumination) if the base is textured. The effect appears as ‘‘varnish’’-like saturation and darkening for IORs below 2, while beyond this the base is increasingly masked by the Fresnel reflection from the coat.

Absorbing coat The exact BRDF of Equation 120 can be expressed as a modification of the albedo-scaling formula for the diffuse BRDF as follows:

$$f_b(\mu_i, \mu_o) = \frac{\rho_d}{\pi}(1 - F_R(\mu_o, \eta))\Delta(\rho_d, \mu_i, \eta), \quad (132)$$

where Δ is a darkening factor given by

$$\Delta(\rho_d, \mu_i, \eta) = \frac{1}{\eta^2} \frac{F_T(\mu_i, \eta)}{1 - \rho_d K(\eta, T)}. \quad (133)$$

This suggests an improvement of the albedo-scaling approximation, which is to replace $\Delta(\rho_d, \mu_i, \eta)$ with its average over the hemisphere. That is, take

$$f(\mu_i, \mu_o) \approx f_s(\mu_i, \mu_o) + \frac{\rho_d}{\pi}(1 - F_R(\mu_o, \eta))T^{\frac{1}{\mu_i} + \frac{1}{\mu_o}}\Delta(\rho_d, \eta), \quad (134)$$

where

$$\Delta(\rho_d, \eta) = \frac{1 - K_0(\eta)}{1 - \rho_d K(\eta, T)}. \quad (135)$$

This BRDF, though non-reciprocal, has the same directional albedo as the exact result (i.e., Equation 128).

A further, very rough approximation of $K(\eta, T)$ accounting for absorption is to ignore the angular dependence of T inside the integral, producing $K(\eta, T) \approx T^2 K_0(\eta)$.

Figure 46: Appearance of textured base with clear coats of different IOR η .

Specular base If the base has specular reflection (due to a perfectly smooth metallic or dielectric interface) then the solution described in the previous sections no longer applies.

The result can be obtained straightforwardly using a geometric series summation quite similar to the Airy summation of Equation 116, except ignoring phase factors since we are working in the geometrical optics limit. Denoting the Fresnel factor of the reflection of the light incident to the coat as F_R (and corresponding transmission coefficient $F_T = 1 - F_R$), the Fresnel factor (i.e., albedo) of the reflection of the refracted ray from the base as ρ_b , and absorption factor $C = T^2/\mu_i^t$, we can write the total directional albedo of the coated base as (using the fact that the Fresnel factors of the internal and external reflections from the coat both equal F_R assuming a mirror reflection from the base):

$$\begin{aligned}
 R &= F_R + F_T C \rho_b F_T + F_T C \rho_b (C F_R \rho_b) F_T + \dots \\
 &= F_R + \sum_{k=1}^{\infty} F_T C \rho_b (C F_R \rho_b)^{k-1} F_T \\
 &= F_R + \frac{F_T^2 C \rho_b}{1 - F_R C \rho_b}.
 \end{aligned} \tag{136}$$

As required by energy conservation, $R \rightarrow 1$ as $C, \rho_b \rightarrow 1$. Re-expressing this in the form of a darkening factor

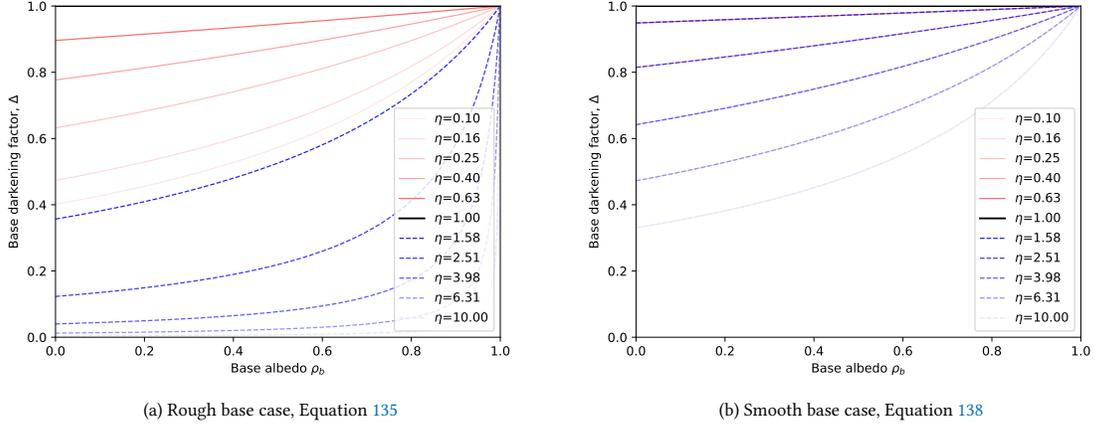


Figure 47: Darkening factor Δ of a base substrate under a smooth clear-coat layer, for rough and smooth base.

Δ applied to albedo scaling, we have

$$R = F_R + (1 - F_R) C \rho_b \Delta, \quad (137)$$

where

$$\Delta = \frac{1 - F_R}{1 - F_R C \rho_b}. \quad (138)$$

This (exact result) has a very similar form to the approximate darkening factor in the case of a Lambertian base, Equation 135, except with K_0 replaced by the Fresnel factor of the coat, and T^2 replaced by the more accurate C .

Figure 47 compares the coat darkening factor Δ in the cases of a rough and smooth base substrate (with a clear coat), for various coat IOR ratios η . A smooth base leads to less darkening (as well as less tinting in the absorptive case) since there is no TIR at all regardless of the angle of incidence, since all rays exit the coat at the same angle they entered (which couldn't have been a TIR direction). In contrast, when diffuse scattering occurs, a very high proportion of rays end up undergoing TIR. This difference in the darkening of a coat for a smooth versus a rough base was also noted by Weidlich [Wei23].

REVISION HISTORY

- v1.0 (**Aug 10, 2025**) – Initial version for SIGGRAPH 2025 course notes.
- v1.1 (**Aug 13, 2025**) – Fixed minor issues.
- v1.2 (**Oct 27, 2025**)
 - Fixed more typos, broken links, and formatting issues.
 - Added a [derivation](#) of the coat darkening effect for a specular base.
 - Added brass and steel to the [table](#) of fits to measured metals.
- v1.3 (**Jan 13, 2026**)
 - The coat darkening example code (Listing 2) was improved to account for the view-dependent absorption as well (and some typos in it were fixed).
 - Fixed broken alpha in some images.