Received: 11 July 2019 | Revised: 2 September 2019 | Accepted: 4 September 2019

DOI: 10.1002/col.22437

RESEARCH ARTICLE

WILEY

Numerical methods for smoothest reflectance reconstruction

Scott A. Burns

Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois

Correspondence
Scott A. Burns, Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champaign, Urbana, IL.
Email: scottallenburns@gmail.com

Abstract

Three numerical methods are presented for finding the smoothest reflectance curve associated with a given triplet of tristimulus values. The methods differ in how "smooth" is defined, and also differ in the domain of colors over which they are applicable. The first method is very quick and applies to any tristimulus values, but sometimes can yield reflectance curves with portions that fall outside the range 0 to 1. The second method applies to colors within the spectral locus (real colors) and guarantees that the reflectances produced are positive. The third method applies to colors within the object color solid (object colors) and guarantees that the reflectances fall within the range 0 to 1. The methods are shown to create reflectances that closely resemble those of real colors (natural and synthetic). Focus is given to implementing the numerical methods in very short MATLAB/Octave functions and to understanding the numerical behavior of the methods near the limits of their respective domains of applicability in terms of matrix conditioning and discretization artifacts.

KEYWORDS

color science, colorimetry, matrix R, reflectance, reflectance reconstruction

1 | INTRODUCTION

The goal of generating a reflectance distribution over the visible portion of the spectrum from a specified three-dimensional color specifier (eg, tristimulus values) has been the subject of many investigations.[1-36] The solution is not unique; there is an entire "metameric suite" of spectral reflectance functions that share a common triplet of tristimulus values for a particular illuminant.[37] The solution obtained by any one algorithm depends on the assumptions and restrictions imposed during the reconstruction process. Reflectance reconstruction has useful applications, such as modeling subtractive color mixture,[38] adapting multichannel color sensors for full spectral imaging,[7,21] rendering realistic ray-traced computer graphics imagery,[39] or implementing a chromatic adaptation transform as part of a color appearance model.[40]

Many investigators have observed that measured reflectance distributions of natural colors tend to be smooth.[6,9,41-51] The goal of seeking the smoothest reflectance function was advanced by van Trigt, who minimized the integral of slope squared over the visible portion of the spectrum, sometimes adding explicit constraints on reflectance to prevent the values from exceeding the range 0 to 1.[11,12] Li and Luo later implemented van Trigt's approach, using a numerical optimization treatment of van Trigt's functional analysis approach.[15] They used explicit inequality constraints on reflectance magnitude. Imposing explicit constraints on reflectance magnitude can sometimes lead to an unnatural, truncated appearance, with sharp discontinuities in slope at the points where the constraints become active. This will be demonstrated in Section 5.

This article presents three numerical methods for finding the smoothest reflectance curve within the metameric suite associated with a given triplet of tristimulus values. The methods differ from one another in how "smooth" is defined, and also differ in their applicability, that is, the color domain over which they operate. The first method is

very fast, requiring only simple matrix multiplication, but it sometimes yields reflectance curves with elements outside the 0-1 range. It corresponds to van Trigt's "unconstrained" case and can be applied to any tristimulus values. The second method guarantees positive reflectance values by using an exponential transformation. It is intended to be used with colors that fall within the spectral locus (defined here as "real" colors). The third method guarantees reflectance values strictly between 0 and 1 by using a hyperbolic tangent transformation. This method is intended to be used with object colors, that is, colors that fall within the object color solid, which is a subset of the real colors. The second and third methods require more computational effort but give rise to more realistic and useful reflectance curves. None of the three methods require explicit inequality constraints to be imposed on reflectance values, and therefore, avoid the shape truncation and slope discontinuity problems.

Short MATLAB/Octave functions are provided for each of the three methods; each function comprises less than two dozen programming statements. The quality of the reflectance curves produced by each method is assessed by comparing them to spectrophotometrically measured reflectances of Munsell Book of Color samples.[52]

Finally, the behaviors of the methods are examined, particularly toward the boundaries of their respective domains of validity. Numerical considerations are discussed concerning how method 2 (real colors) converges to single wavelength colors near the spectral locus, and how method 3 (object colors) converges to square-wave-shaped "optimal" colors near the object color solid boundary.[53]

2 | METHOD 1: UNCONSTRAINED REFLECTANCE

Consider the following objective function, which seeks to find a reflectance distribution, ρ, that has minimum slope squared, integrated over the visible portion of the spectrum:

$$\min_{\rho} \int_{\text{visible } \lambda} (d\rho/d\lambda)^2 d\lambda. \tag{1}$$

This is van Trigt's objective function mentioned in the previous section. The method presented in this section corresponds to van Trigt's "unconstrained" case, meaning no explicit constraints are imposed on ρ.

The continuous objective function of Equation (1) can be discretized to make it suitable for numerical solution by specifying n wavelength bands over the visible portion of the spectrum and representing all spectral-based entities as $n \times 1$ vectors. We also add an equality constraint that requires the reflectance curve to be associated with a specific triplet of tristimulus values, XYZ_W, which is referenced to an $n \times 1$ illuminant vector, W. This yields the following equality-constrained optimization statement:

$$\begin{aligned} &\min_{\rho} \ \sum_{i=1}^{n-1} (\rho_{i+1} - \rho_i)^2 \\ &\text{s.t.} \quad A_W{}' \, \rho = XYZ_W. \end{aligned} \tag{2}$$

Matrix A is an $n \times 3$ array of color matching functions (CMFs) by columns, $A = [\bar{x}, \bar{y}, \bar{z}]$, and matrix A_W is the illuminant-W-referenced CMFs, computed as $\overline{W}A$, where $\overline{W}$ is an $n \times n$ matrix with W on the main diagonal and zeros elsewhere. Prime denotes matrix transpose. It is assumed that the illuminant has been normalized so that the scalar product $\bar{y}' \cdot W$ equals 1. The use of illuminant-referenced CMFs is a common practice in color science.[54] Tristimulus values follow a $0 \le Y \le 1$ convention in this article, in contrast to the alternative $0 \le Y \le 100$ convention often used.

Equation (2) can be expressed fully in matrix form:

$$\begin{aligned} &\min_{\rho} \quad \frac{1}{2} \rho' D \rho \\ &\text{s.t.} \quad A_W{}' \, \rho = XYZ_W, \end{aligned} \tag{3}$$

where D is an $n \times n$ tridiagonal matrix of finite-differencing constants,

$$D = \begin{bmatrix} 2 & -2 & & & & & \\ -2 & 4 & -2 & & & & \\ & -2 & 4 & -2 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & -2 & 4 & -2 & \\ & & & & -2 & 2 & \end{bmatrix}. \tag{4}$$

A computationally efficient way to solve this equality-constrained quadratic program (QP) is to apply the method of Lagrange multipliers. The Lagrangian function is

$$\mathcal{L}(\rho, \lambda) = \frac{1}{2} \rho' D \rho + \lambda' (A_W{}' \rho - XYZ_W), \tag{5}$$

where λ is a 3×1 vector of Lagrange multipliers. Setting partial derivatives of the Lagrangian to zero, we obtain the stationary conditions:

$$\begin{aligned} \partial \mathcal{L} / \partial \rho &= D\rho + A_W \lambda = 0 \\ \partial \mathcal{L} / \partial \lambda &= A_W{}' \rho - XYZ_W = 0. \end{aligned} \tag{6}$$

The solution of this system of linear equations yields a stationary point, which for this QP is a unique minimum.

Normally the stationary conditions for a least-squares QP would have the solution

$$\rho = D^{-1}A_W\left(A_W{}'D^{-1}A_W\right)^{-1}XYZ_W. \tag{7}$$

In our case, however, D is singular, so we need a different approach to solve for ρ. Note that Equation (6) can be expressed in block matrix form:

$$\begin{bmatrix} D & A_W \\ A_W{}' & 0 \end{bmatrix}\begin{Bmatrix} \rho \\ \lambda \end{Bmatrix} = \begin{Bmatrix} 0 \\ XYZ_W \end{Bmatrix}. \tag{8}$$

Let B represent the inverse of the (nonsingular) square matrix in Equation (8),

$$B = \begin{bmatrix} D & A_W \\ A_W{}' & 0 \end{bmatrix}^{-1}. \tag{9}$$

Premultiplying Equation (8) by B gives

$$\begin{Bmatrix} \rho \\ \lambda \end{Bmatrix} = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}\begin{Bmatrix} 0 \\ XYZ_W \end{Bmatrix} \tag{10}$$

or simply

$$\rho = B_{12}\,XYZ_W. \tag{11}$$

We need only compute matrix B once for a given illuminant and choice of standard observer (A matrix). B_{12} is the upper-right $n \times 3$ portion of B. Once B_{12} is obtained, a least-slope-squared reflectance curve is computed from any arbitrary XYZ_W by simple matrix multiplication (Equation (11)).

A pair of short MATLAB functions implement this first numerical method. These functions also work in the free open-source Octave software.[55] The first function computes the B_{12} matrix needed in preparation for performing the reflectance reconstruction, which takes place in the second function.

```
function B12=method_1_prep(A,W)
% This function computes B12 in preparation for
% the first numerical method (the unconstrained
% reflectance case).
% A is nx3 matrix of color matching functions.
% W is nx1 illuminant vector, scaled arbitrarily.
% B12 creates a reflectance curve: rho=B12*XYZw.
n=size(A,1);
D=full(gallery('tridiag',n,-2,4,-2));
D(1,1)=2; D(n,n)=2;
W_normalized=W/(A(:,2)'*W);
Aw=diag(W_normalized)*A;
B=inv([D, Aw; Aw', zeros(3)]);
B12=B(1:n,n+1:n+3);
```

The second function generates a reflectance vector from a specified triplet of illuminant-W-referenced tristimulus values, using B_{12} obtained above.

```
function rho=method_1(B12,XYZw)
% This function generates a reflectance vector for
% method 1 (the unconstrained reflectance case)
% B12 is nx3 matrix created by method_1_prep.
% XYZw is a 3-element vector of illum-W-referenced
% tristimulus values in the 0 <= Y <= 1 range.
rho=B12*XYZw(:);
```

The method is broken up into two functions because B_{12} needs to be computed only once for a given illuminant. It would be inefficient to compute it repeatedly for a series of reflectance reconstructions under a common illuminant, which is a typical use case. Method 1 can be used with any input tristimulus values, but the resulting reflectance curve may have elements outside the 0-1 range. Methods 2 and 3 keep the reflectances within one or both of these bounds, making them more useful for some applications.

3 | METHOD 2: COLORS WITHIN THE SPECTRAL LOCUS (REAL COLORS)

The previous method can sometimes create reflectance curves with negative elements. In some applications, it is important to avoid negative reflectances, for example, when modeling subtractive color mixture.[38] Reflectance functions with negative components map to points outside the spectral locus in *XYZ* space. These "imaginary" colors are physically meaningless, but they can be useful; the three primaries in the *XYZ* color space are examples.

The method in this section assumes that the input tristimulus values represent "real" colors, which is defined in this context as colors that fall strictly within the spectral locus. These colors have reflectance values that are strictly positive, including those with reflectance values greater than one. Strictly speaking, reflectances greater than one are associated with fluorescent colors. Here we are expanding the notion of reflectance to include arbitrary emissive sources. Although the "reflectance" magnitudes associated with emissive sources are meaningless in the standard physical sense, they are mathematically meaningful since a stimulus vector describing the

emissive source, S, can be computed by multiplying this "reflectance" vector by the illuminant matrix, $S = \overline{W}\rho$. Thus, we are able to represent any color that falls within the spectral locus as a strictly positive reflectance multiplier vector operating on a reference illuminant. If it is desired to produce only reflectance curves that are bounded by 0 and 1, then Method 3 should be used instead.

Method 2 guarantees that the reflectance functions it produces are strictly positive by making a change of variables $\rho = e^z$. Our definition of "smoothest" is modified to retain a quadratic objective function; instead of seeking a least-slope-squared ρ vector, we seek a least-slope-squared z vector:

$$\begin{aligned} \min_z \quad & \frac{1}{2} z' D z \\ \text{s.t.} \quad & A_W{}' \, e^z = XYZ_W, \end{aligned} \tag{12}$$

Observe that z can take on any value, but the reflectance, e^z, will always be positive. We are finding the "smoothest" reflectance curve in the space of $\ln(\rho)$. This equality-constrained QP can again be solved by the method of Lagrange multipliers. The Lagrangian function is

$$\mathscr{L}(z,\lambda) = \frac{1}{2} z' D z + \lambda' (A_W{}' \, e^z - XYZ_W) \tag{13}$$

and the stationary conditions are

$$\begin{aligned} \partial \mathscr{L} / \partial z &= D z + \operatorname{diag}(e^z) A_W \lambda = 0 \\ \partial \mathscr{L} / \partial \lambda &= A_W{}' e^z - XYZ_W = 0. \end{aligned} \tag{14}$$

The stationary conditions now comprise an $(n + 3) \times (n + 3)$ nonlinear system of equations, which is more difficult to solve computationally. However, the system is readily solved by Newton's method. Defining the vector-valued function, F, as

$$F = \begin{Bmatrix} D z + \operatorname{diag}(e^z) A_W \lambda \\ A_W{}' e^z - XYZ_W \end{Bmatrix}, \tag{15}$$

and the Jacobian matrix of first partial derivatives of F as

$$J = \left[\begin{array}{c|c} D + \operatorname{diag}(\operatorname{diag}(e^z) A_W \lambda) & \operatorname{diag}(e^z) \; A_W \\ \hline A_W{}' \; \operatorname{diag}(e^z) & 0 \end{array}\right], \tag{16}$$

the change in the values of z and λ with each Newton iteration is found by solving the linear system

$$J \begin{Bmatrix} \Delta z \\ \Delta \lambda \end{Bmatrix} = -F. \tag{17}$$

Thus, a reconstructed reflectance curve can be found with a series of linear equation solutions, updating z and λ with each iteration: $z^{k+1} = z^k + \Delta z$ and $\lambda^{k+1} = \lambda^k + \Delta\lambda$.

Below is a pair of MATLAB/Octave functions that implement method 2. The first function computes two arrays, D and A_W, that do not depend on the input tristimulus values:

```
function [D,Aw]=method_2_prep(A,W)
% This function computes two arrays in preparation
% for method 2 (the real colors case).
% A is nx3 matrix of color matching functions.
% W is nx1 illuminant vector, scaled arbitrarily.
% D is nxn matrix of finite differencing constants.
% Aw is nx3 matrix of illuminant-W-referenced CMFs.
n=size(A,1);
D=full(gallery('tridiag',n,-2,4,-2));
D(1,1)=2; D(n,n)=2;
W_normalized=W/(A(:,2)'*W);
Aw=diag(W_normalized)*A;
```

The second function generates the reflectance vector for a given W-referenced triplet of tristimulus values, XYZ_W, using the two arrays prepared in advance:

```
function rho=method_2(D,Aw,XYZw)
% D is nxn matrix of finite differencing constants.
% Aw is nx3 matrix of illuminant-W-referenced CMFs.
% XYZw is a 3-element vector of illum-W-referenced
% tristimulus values in the 0 <= Y <= 1 range.
% rho is nx1 reflectance vector (or 0 if failure).
n=size(Aw,1);
rho=zeros(n,1); z=zeros(n,1); lambda=zeros(3,1);
count=0; % iteration counter
maxit=20; % max number of iterations
ftol=1.0e-8; % convergence tolerance
while count <= maxit
    r=exp(z); dr=diag(r); dra=dr*Aw; v=dra*lambda;
    F=[D*z+v; Aw'*r-XYZw(:)];
    J=[D+diag(v), dra; dra', zeros(3)];
    delta=J\(-F); % solve J*delta = -F
    z=z+delta(1:n);
    lambda=lambda+delta(n+1:n+3);
    if all(abs(F)<ftol) % convergence check
        rho=exp(z); return
    end
    count=count+1;
end
```

Note that the two off-diagonal blocks in Equation (16) are transposes of one another. That observation is used for

computational efficiency in the function above. However, for some applications, it is useful to modify one of the two blocks slightly to achieve additional functionality, so that they no longer remain transposes of one another (see section 7 of Reference 40).

The MATLAB/Octave statement delta=J\(−F) solves the system of linear equations $J\Delta = -F$ for the vector Δ.[56] The algorithm it uses is more advanced than simple Gaussian elimination; it implements pivoting and other stability-enhancing techniques. If one wishes to program method 2 outside of the MATLAB/Octave environment, a good choice for a linear equation solver library is LAPACK,[57] the library upon which MATLAB's linear equation solver is based.

The method 2 function will fail if the input tristimulus values fall on or outside the spectral locus. It fails for colors outside the locus because these colors must have reflectances with some negative elements. It fails for colors exactly on the locus because those colors can only be represented by reflectances with some nonpositive elements, for example, pure single wavelength colors, which have all zero elements except for one element. The $\rho = e^z$ transformation prevents reflectance values from becoming zero or negative, leading to failure of the method. Therefore, this method is intended only for real colors located strictly within the spectral locus. See the online supplementary documentation for a MATLAB/Octave function to test if a color is within the spectral locus.[58] The output of method 2 will approach the limiting case of a single wavelength spike as we approach the spectral locus from the interior, as discussed in Section 6.

4 | METHOD 3: COLORS WITHIN THE OBJECT COLOR SOLID (OBJECT COLORS)

The object color solid is a subset of the real colors comprising colors that can be represented by reflectance curves having elements ranging from 0 to 1.[53] The outer boundary of the object color solid contains the optimal colors, which can be represented by certain reflectance curves having only 0 and 1 values over the visible portion of the spectrum. The location and shape of the object color solid within the spectral locus depends on the choice of illuminant.

This section presents the third method, which is intended for colors within the object color solid (which excludes fluorescent colors with $\rho \geq 1$). While the previous method guaranteed strictly positive reflectances, this method guarantees reflectances strictly between 0 and 1 by using a hyperbolic tangent change of variables, $\rho = (\tanh(z) + 1)/2$. The hyperbolic tangent (also expressed as $(e^z - e^{-z})/(e^z + e^{-z})$) ranges from −1 to 1, so by adding 1 and dividing by 2, the range is shifted to between 0 and 1. Any real value of z will produce a ρ value between 0 and 1. As with the previous method, we redefine "smoothest" to be the minimum of a quadratic function of z:

$$\begin{aligned} &\min_z && \frac{1}{2} z' D z \\ &\text{s.t.} && A_W'(\tanh(z)+1)/2 = XYZ_W. \end{aligned} \tag{18}$$

Here, we are seeking the smoothest reflectance in the space of $\tanh^{-1}(2\rho - 1)$. Again applying the method of Lagrange multipliers, and noting that the derivative of $\tanh(z)$ is $\text{sech}^2(z)$, we obtain a system of nonlinear equations representing the stationary conditions:

$$F = \begin{Bmatrix} Dz + \text{diag}\left(\text{sech}^2(z)/2\right) A_W \lambda \\ A_W'(\tanh(z)+1)/2 - XYZ_W \end{Bmatrix}, \tag{19}$$

and its associated Jacobian matrix of first partial derivatives (noting that the derivative of $\text{sech}^2(z)/2$ is $-\text{sech}^2(z)\tanh(z)$):

$$J = \left[\begin{array}{c|c} D - \text{diag}\left(\text{diag}\left(\text{sech}^2(z)\ \tanh(z)\right) A_W \lambda\right) & \text{diag}\left(\text{sech}^2(z)/2\right)\ A_W \\ \hline A_W'\ \text{diag}\left(\text{sech}^2(z)/2\right) & 0 \end{array}\right]. \tag{20}$$

As before, Newton's method solves this nonlinear system efficiently by iteratively solving the linear system shown in Equation (17), updating the values of z and λ with each iteration: $z^{k+1} = z^k + \Delta z$ and $\lambda^{k+1} = \lambda^k + \Delta\lambda$.

Below is a pair of MATLAB/Octave functions that implement method 3. The first function computes the two arrays, D and A_W, that do not depend on the input tristimulus values (this is the same as for method 2):

```
function [D,Aw]=method_3_prep(A,W)
% This function computes two arrays in preparation
% for method 3 (the object colors case).
% A is nx3 matrix of color matching functions.
% W is nx1 illuminant vector, scaled arbitrarily.
% D is nxn matrix of finite differencing constants.
% Aw is nx3 matrix of illuminant-W-referenced CMFs.
n=size(A,1);
D=full(gallery('tridiag',n,-2,4,-2));
D(1,1)=2; D(n,n)=2;
W_normalized=W/(A(:,2)'*W);
Aw=diag(W_normalized)*A;
```

The second function generates the reflectance vector for a given W-referenced triplet of tristimulus values, XYZ_W, using the two arrays prepared in advance:

```
function rho=method_3(D,Aw,XYZw)
% D is nxn matrix of finite differencing constants.
% Aw is nx3 matrix of illuminant-W-referenced CMFs.
% XYZw is a 3-element vector of illum-W-referenced
% tristimulus values in the 0 <= Y <= 1 range.
% rho is nx1 reflectance vector (or zeros if failure).
n=size(Aw,1);
rho=zeros(n,1); z=zeros(n,1); lambda=zeros(3,1);
count=0; % iteration counter
maxit=20; % max number of iterations
ftol=1.0e-8; % convergence tolerance
while count <= maxit
    r=(tanh(z)+1)/2;
    d1=diag((sech(z).^2)/2);
    d1a=d1*Aw;
    d2=diag(sech(z).^2.*tanh(z));
    F=[D*z+d1a*lambda; Aw'*r-XYZw(:)];
    J=[D-diag(d2*Aw*lambda), d1a; d1a', zeros(3)];
    delta=J\(-F); % solve J*delta = -F
    z=z+delta(1:n);
    lambda=lambda+delta(n+1:n+3);
    if all(abs(F)<ftol) % convergence check
        rho=(tanh(z)+1)/2; return
    end
    count=count+1;
end
```

Method 3 function will fail if the input tristimulus values fall outside the object color solid or fall on its boundary (the optimal colors). This is because the $\rho = (\tanh(z) + 1)/2$ transformation prevents reflectance values from becoming ≤ 0 or ≥ 1, which must happen with colors that are not strictly within the object color solid. See the online supplementary documentation for a MATLAB/Octave function to check if a color is within the object color solid.[59] The output of method 3 will approach the limiting case of a square-shaped optimal color as we approach the object color solid boundary from the interior, as discussed in Section 6.

5 | COMPARISON TO MEASURED OBJECT REFLECTANCES

This section compares the reflectance distributions produced by the three methods to 1485 spectrophotometrically measured reflectances from the glossy version of the 2007 Munsell Book of Colors.[60] The measurements are collected in 36, 10 nm wavelength bands, spanning 380 nm to 730 nm. First, the illuminant-C-referenced tristimulus values, XYZ_C, for the reflectance curves in this dataset are computed by

$$XYZ_C = A_C{}' \rho, \tag{20}$$

where A_C is the illuminant-C-referenced CMFs obtained from either function *method_2_prep* or *method_3_prep* above. Then the reconstructed reflectance curves are generated from the tristimulus values using each of the three methods.

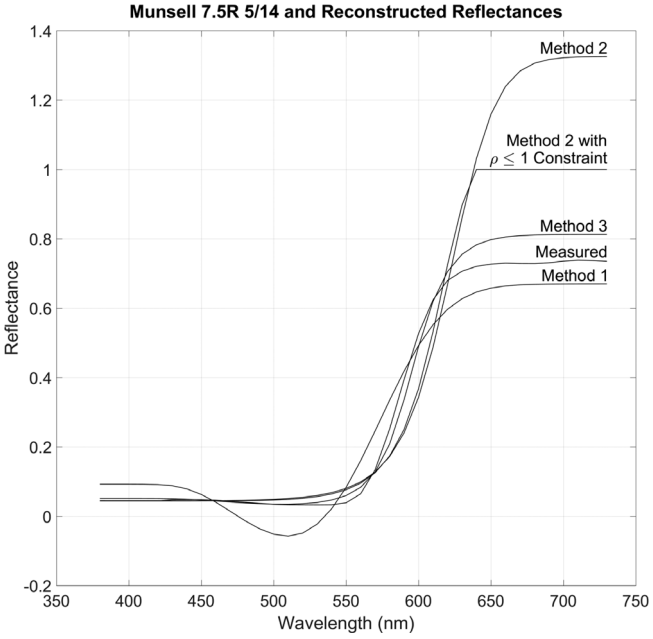


FIGURE 1 Munsell 7.5R 5/14 measured reflectance, reconstructed reflectances from the three methods, and a reconstruction with an explicit inequality constraint on reflectance magnitude

Figure 1 shows the measured reflectance curve for Munsell chip 7.5R 5/14 (slightly yellowish red, value 5, chroma 14) along with the reconstructed reflectances from its XYZ_C triplet for the three methods. This figure demonstrates behaviors that are sometimes observed with the three methods. Method 1 has a region of $\rho < 0$ and method 2 has a region of $\rho > 1$. Method 3 always enforces $0 < \rho < 1$ over the visible wavelengths.

It was mentioned in the introduction that explicit inequality constraints on reflectance values in the optimization statement can sometimes give undesirable results. Figure 1 includes a reconstructed reflectance using method 2 when an additional set of constraints are added to the optimization, $\rho \leq 1$. This solution was obtained through a general-purpose optimization program. A sharp discontinuity in slope appears as the curve reaches $\rho = 1$, giving it an unnatural, truncated appearance.

We now turn to comparing the reconstructed reflectances for the three methods to the measured reflectances for the entire dataset of 1485 Munsell colors. When comparing two reflectance functions for quality of match, some portions of the visible spectrum are more important than others. A mismatch in the upper or lower extremes of the visible spectrum has little impact on tristimulus values and perceived color. Consequently, a measure of reflectance curve match was

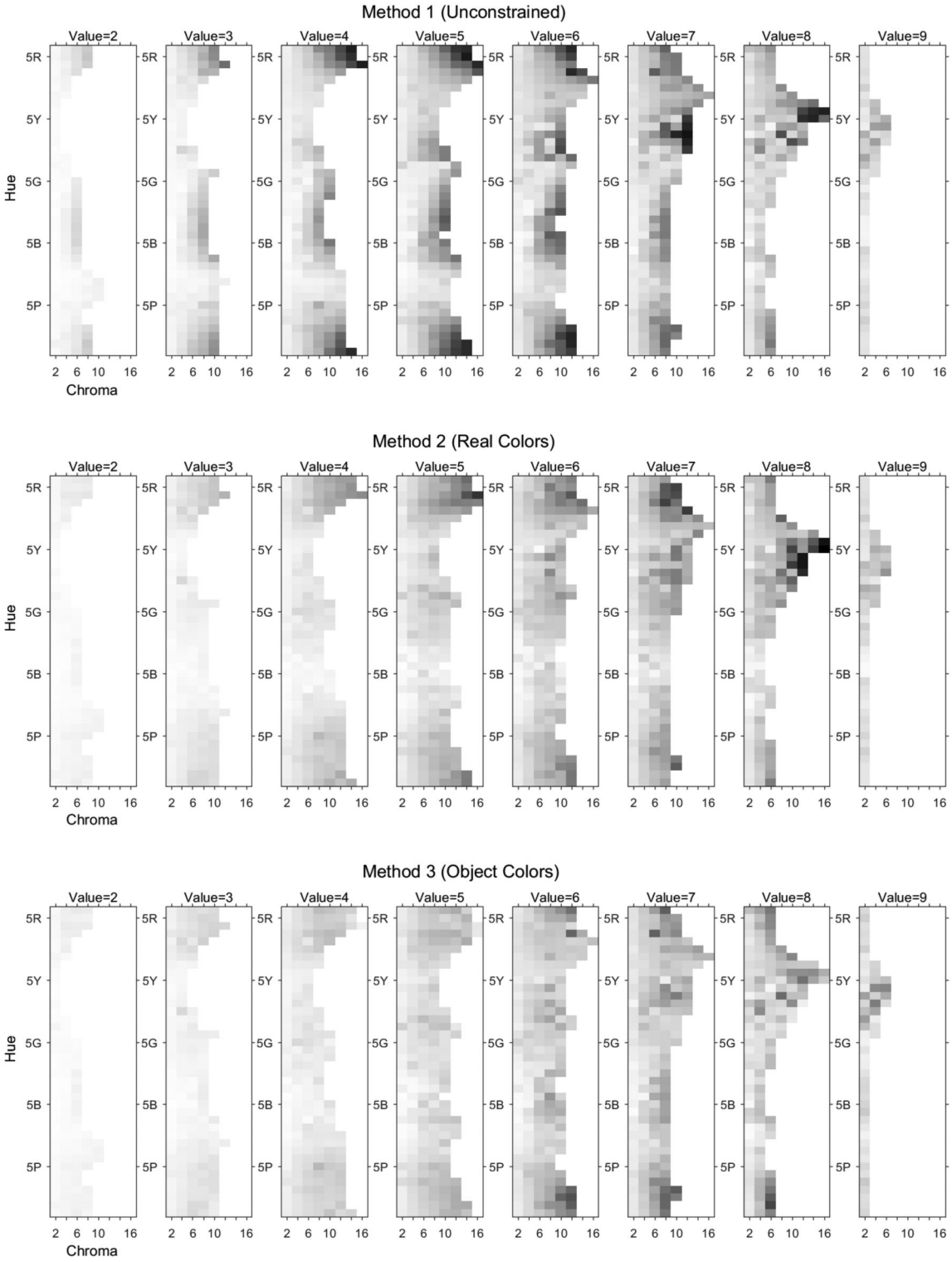
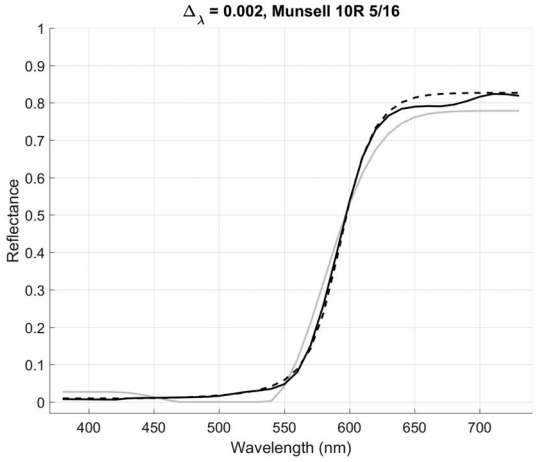
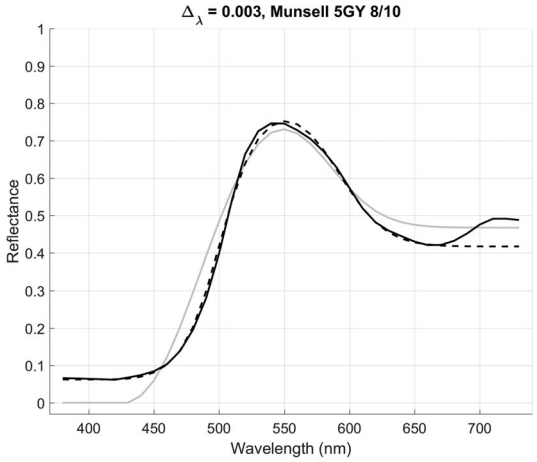


FIGURE 2 Reflectance curve match, Δ_λ, mapped to grayscale for 1485 Munsell colors compared to reconstructed reflectances for the three methods (black is $\Delta_\lambda = 0.036$ and white is $\Delta_\lambda = 0$)





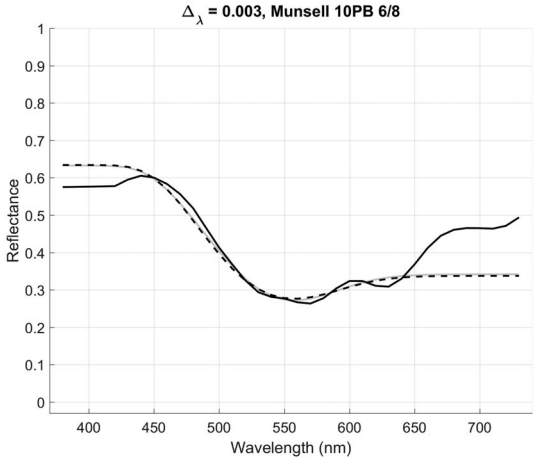
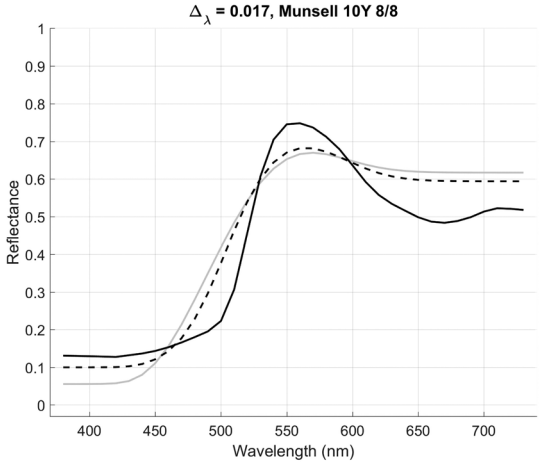
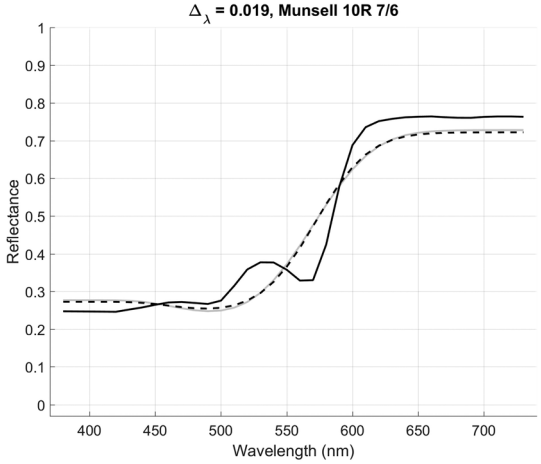


FIGURE 3 Three examples of some of the best method 3 matches (dashed) compared to the measured reflectance (solid black). Solid gray lines are smoothest reconstructions produced by the method of Li and Luo





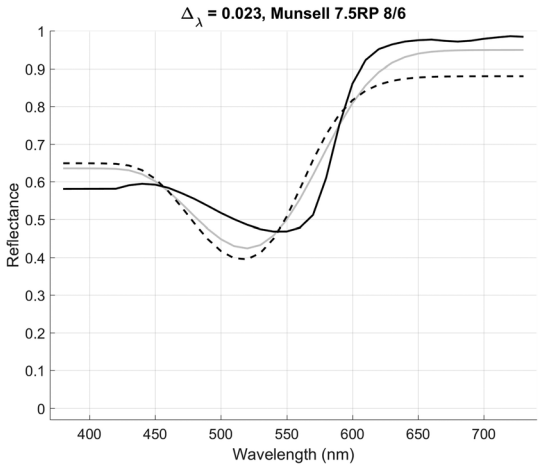


FIGURE 4 Three examples of some of the worst method 3 matches (dashed) compared to the measured reflectance (solid black). Solid gray lines are smoothest reconstructions produced by the method of Li and Luo

devised, Δ_λ, that gives greater emphasis to the central portion of the visible spectrum, using the luminosity function vector, $\bar{y}$, as a weighting function:

$$\Delta_\lambda = \bar{y}' |\rho_1 - \rho_2| / n. \tag{21}$$

This reflectance difference measure should not be confused with color difference measures such as ΔE^*_{94} or CMC

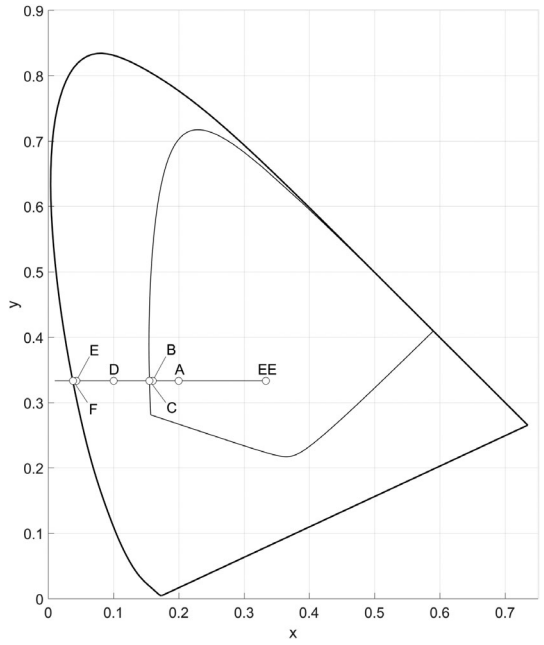


FIGURE 5 Spectral locus (outer curve) and $Y = 0.5$ slice of the object color solid with equal energy illuminant (inner curve). The horizontal line is the path taken through the color space

l:c.[61] These color difference measures are all exactly zero for the reflectance curves being compared here because the colors are metameric and share identical tristimulus values.

Figure 2 presents plots of Δ_λ for the three methods applied to all 1495 Munsell colors. Each of the three subfigures (methods 1, 2, 3) is divided into a series of isovalue subfigures, plotting hue against chroma in each. The gray level used to represent Δ_λ for each Munsell color is linearly scaled so that black is mapped to the largest Δ_λ experienced by any of the methods (=0.036), and white is mapped to $\Delta_\lambda = 0$.

As expected, the best fit to the Munsell colors is with method 3. The maximum Δ_λ for method 3 is 0.023, which is 65% of the maximum value over all three methods. The mean Δ_λ for methods 1-3 are 0.0054, 0.0045, and 0.0039, respectively. Figure 2 is useful for identifying which colors match the best/worst for each method. For the highly chromatic warm colors, the large Δ_λ values are caused by method 1 tending to dip into the negative region and method 2 tending to overshoot the measured curves by a considerable margin.

Figure 3 presents three examples of some of the best matches for method 3 in terms of Δ_λ. The solid line is the measured reflectance and the dashed line is the method 3 reconstruction. The light gray solid line is the smoothest reconstruction that is obtained by the method presented in Li and Luo,[15] which for all 1485 Munsell samples has a maximum Δ_λ of 0.028 and a mean Δ_λ of 0.0051. Note that the slope discontinuities and shape truncation occur in some of these examples where $\rho = 0$. Figure 4 presents some of the worst matches for method 3. See the online supplementary documentation for additional comparison

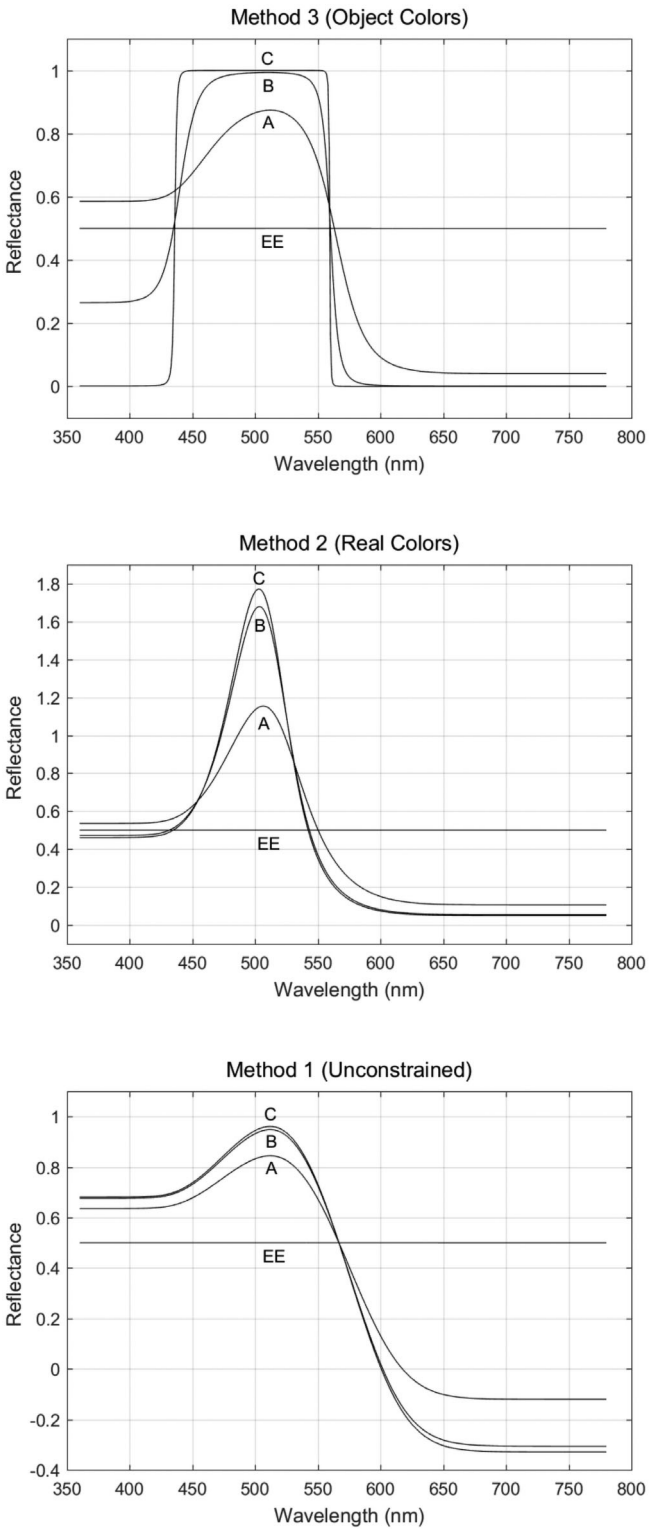


FIGURE 6 Reconstructed reflectance curves along path EE → A → B → C. Point C is just inside the object color solid slice and method 3 approaches the shape of an optimal color

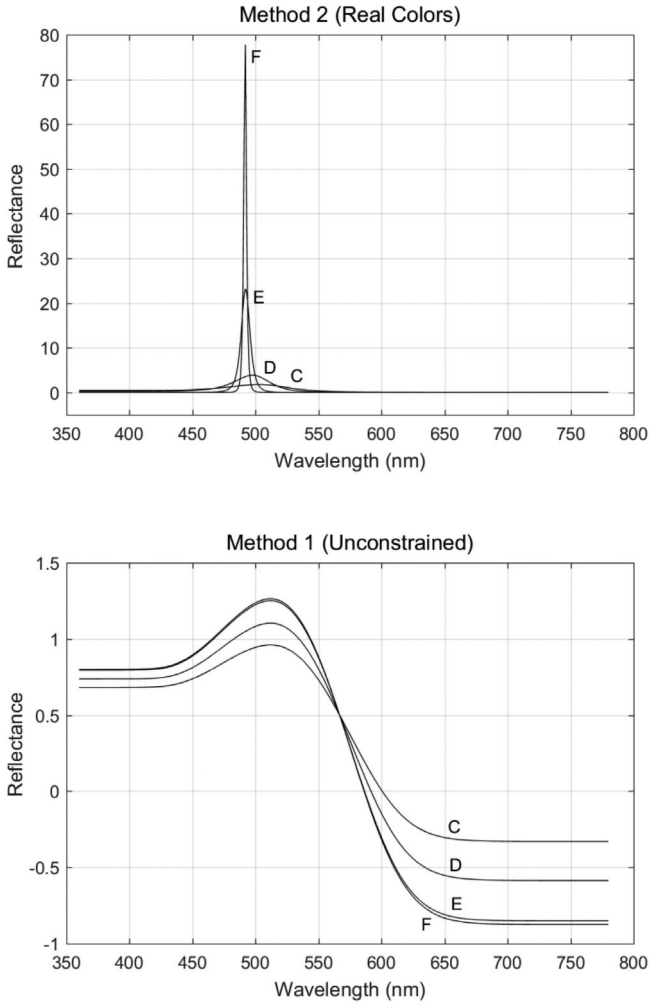


FIGURE 7 Reconstructed reflectance curves along path C → D → E → F. Point F is just inside the spectral locus and method 2 approaches the shape of pure, single-wavelength color

information, including comparisons to Liquitex brand acrylic paints.[62]

6 | BEHAVIOR OF THE METHODS AS BOUNDARIES ARE APPROACHED

Methods 2 and 3 are designed to operate in different regions of the tristimulus color space. Method 2 (real colors) requires that colors be within the spectral locus and method 3 (object colors) requires that colors be within the object color solid. Once these boundaries are reached, the methods will fail to converge, typically due to the Jacobian matrix (Equations (16) and (20)) becoming singular. This section examines how the methods behave as their respective boundaries are approached. It also examines the numerical behavior of the Jacobian matrix in the immediate vicinity of the boundaries.

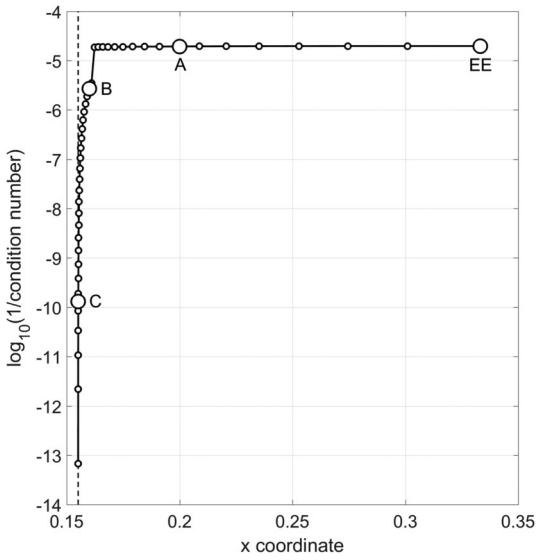


FIGURE 8 Inverse of the condition number of the Jacobian matrix along path EE → A → B → C for method 3. The dashed line is the x-coordinate of where the path intersects the $Y = 0.5$ slice of the object color solid

Figure 5 shows the spectral locus and a $Y = 0.5$ slice of the object color solid (using the 1931 CIE CMFs). We examine the behavior of the three methods as we move along the horizontal line shown, starting at equal energy (the assumed illuminant) and moving along the path EE → A → B → C → D → E → F. Point C is just inside the object color solid boundary (≈99.9% from EE to the boundary) and point F is just inside the spectral locus.

Figure 6 shows the evolution of the reflectance shapes as we approach the object color solid boundary along the path EE → A → B → C. Methods 1 and 2 produce curves that are not object colors (having $\rho < 0$ or $\rho > 1$). Method 3 converges toward a square-wave-shaped optimal color.

Figure 7 shows the evolution of the reflectance shapes for methods 1 and 2 as we move toward the spectral locus along the path C → D → E → F. Method 1 continues to produce solutions with $\rho < 0$, while method 2 converges toward a single wavelength pure color, as one would expect on the spectral locus. Method 3 has no solutions, as there are no object colors in this interval.

As a limiting boundary is approached, the Jacobian matrix becomes increasingly ill-conditioned until it eventually becomes singular at the boundary. One measure of the conditioning of a matrix is its condition number, or more commonly used, the inverse of its condition number (ICOND), which approaches zero as the boundary is approached. Figure 8 plots ICOND of the Jacobian matrix along the horizontal path shown in Figure 5 for method 3.

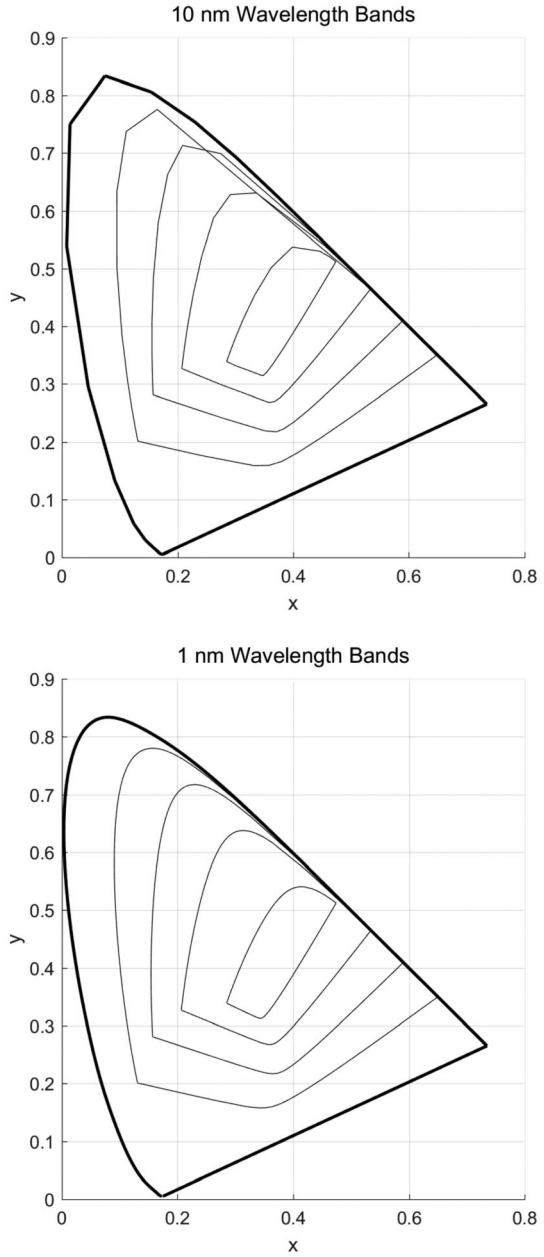


FIGURE 9 A demonstration of 10 nm and 1 nm wavelength band widths affecting the shapes of the spectral locus and a series of object color solid *Y*-slices

Points EE, A, B, and C are shown (large circles), indicating ICOND at those points. As the boundary of the object color solid is approached along x (vertical dashed line), ICOND drops rapidly. MATLAB will issue warnings when ICOND drops below machine epsilon (the smallest machine number for which $1 + \text{epsilon} \neq 1$), eventually raising an error condition that the matrix is singular to machine precision. In our example, we are able to get good solutions beyond point C, up to an x that is 99.998% from EE toward the boundary (small circles are successful solutions).

It should be noted that the level of discretization impacts the boundary behavior. The examples in this section use CMFs with 1 nm wavelength bands. In contrast, the comparison to Munsell colors in Section 5 used 10 nm bands, since that was the resolution of the measured color samples. Figure 9 demonstrates that the 10 nm discretization yields a spectral locus and object color solid slices that are more polygonal in shape than the 1 nm versions.

The polygonal boundaries shown on the top in Figure 9 are the limiting boundaries for methods 2 and 3 when the 10 nm CMFs are used. Since these boundaries fall somewhat inside the corresponding true continuous boundaries, there will be valid real/object colors very near the true boundaries that are outside the polygonal boundaries, which will cause methods 2 and 3 to fail. This means that when considering colors very near the boundaries using method 2 or 3, it is advisable to use the highest resolution CMFs available for best results.

One final note concerns the illuminant white point. The smoothest reflectance curve corresponding to the illuminant white point is a uniform $\rho = 1$ over the entire visible spectrum. Methods 1 and 2 will produce this result. However, method 3 cannot produce it because ρ must be strictly less than 1 for this method, which excludes this solution. Instead the Jacobian matrix will become singular and the method will fail. If it is desirable to have method 3 return $\rho = 1$ for *XYZ* equal to the white point, it can be implemented as a special case in the code when $Y = 1$, since there is only one point on the object color solid with that Y value. (For robustness, we should also capture Y values slightly less than 1 that cause conditioning errors.)

7 | SUMMARY

Three numerical methods have been presented for reconstructing a reflectance distribution from a specified set of tristimulus values. The three methods identify the smoothest reflectance curve, each using a different measure of smoothness. Each method has a different domain of colors over which it is applicable.

Method 1 applies to any color, real or imaginary. It is the fastest of the three, requiring only a simple matrix multiplication. It minimizes the integral of reflectance slope squared over the visible wavelengths but can produce reflectance values that fall outside the 0-1 range normally associated with reflectances.

Method 2 applies to colors within the spectral locus, or real colors. It minimizes slope squared in the space of $\ln(\rho)$ and guarantees that the reflectance curves are strictly positive. When the method is applied to colors near the spectral locus, the reflectance curves resemble spiked, single

wavelength colors, as expected from colors on the spectral locus.

Method 3 applies to colors within the object color solid, or object colors. It minimizes slope squared in the space of $\tanh^{-1}(2\rho - 1)$ and guarantees that the reflectance curves have values strictly between 0 and 1. When the method is applied to colors near the object color solid boundary, the reflectance curves resemble the square-wave-shaped distributions typical of the optimal colors found on the boundary of the object color solid.

The reflectance curves produced by the second and third methods have a good resemblance to measured object colors, specifically, reflectance curves measured from Munsell Book of Color. The third method provides the best match of the three. Other studies have shown similar results compared to commercial paints and pigments.

Care should be taken when using any of the methods in the immediate vicinity of their respective boundaries of applicability. The linear system can become ill-conditioned or singular if operated too close to the boundary. The width of the wavelength bands chosen to discretize the system can exacerbate boundary-vicinity problems, so smaller bands are better when operating very near a boundary.

One final note is that the three methods presented here can be viewed in historical context if we consider Cohen's notion of a "fundamental metamer."[63] This discussion can be found in the online supplementary documentation.[64]

ORCID

Scott A. Burns https://orcid.org/0000-0002-1251-7758

REFERENCES

[1] Moon P. Polynomial representation of reflectance curves. *J Opt Soc Am*. 1945;35(9):597-600.
[2] Morris R. Metameric formulation. *J Opt Soc Am*. 1947;37(8):669.
[3] Brewer WL, Holly FR. Syntheses of spectral-distribution curves. *J Opt Soc Am*. 1948;38(10):858-874.
[4] Stiles WS, Wyszecki GW. Counting metameric object colors. *J Opt Soc Am*. 1962;52(3):313-328.
[5] Takahama K, Nayatani Y. New method for generating metameric stimuli of object colors. *J Opt Soc Am*. 1972;62(12):1516-1520.
[6] Stiles WS, Wyszecki G, Ohta N. Counting metameric object-color stimuli using frequency-limited spectral reflectance functions. *J Opt Soc Am*. 1977;67(6):779-784.
[7] Park SK, Huck FO. Estimation of spectral reflectance curves from multispectral image data. *Appl Optics*. 1977;16(12):3107-3114.
[8] Ohta N. A simplified method for formulating pseudo-object colors. *Color Res Appl*. 1982;7(2):78-81.
[9] Maloney LT. Evaluation of linear models of surface spectral reflectance with small numbers of parameters. *J Opt Soc Am A*. 1986;3(10):1673-1683.
[10] Maloney LT, Wandell BA. Color constancy: a method for recovering surface spectral reflectance. *J Opt Soc Am A*. 1986;3(1):29-33.
[11] Trigt CV. Smoothest reflectance functions. I. Definition and main results. *J Opt Soc Am A*. 1990;7(10):1891-1904.
[12] Trigt CV. Smoothest reflectance functions. II. Complete results. *J Opt Soc Am A*. 1990;7(12):2208-2222.
[13] Hawkyard CJ. Synthetic reflectance curves by subtractive colour mixing. *J Soc Dyers Colourists*. 1993;109:246-251.
[14] Hawkyard CJ. Synthetic reflectance curves by additive mixing. *J Soc Dyers Colourists*. 1993;109:323-329.
[15] Li C, Luo MR. The estimation of spectral reflectances using the smoothness constraint condition. Paper presented at: Proceedings of IS&T/SID Ninth Color Imaging Conference; November 1999; Scottsdale, AZ.
[16] Smits B. An RGB-to-spectrum conversion for reflectances. *J Graph Tools*. 1999;4(4):11-22.
[17] Dupont D. Study of the reconstruction of reflectance curves based on tristimulus values: comparison of methods of optimization. *Color Res Appl*. 2002;27(2):88-99.
[18] Sharma G, Wang S. Spectrum recovery from colorimetric data for color reproductions. Paper presented at: Proceedings of SPIE 4663, Color Imaging: Device-Independent Color, Color Hardcopy, and Applications VII; December 28, 2001; San Jose, CA.
[19] Zuffi S, Schettini R. Reflectance function estimation from tristimulus values. Paper presented at: Proceedings of SPIE 5293, Color Imaging IX: Processing, Hardcopy, and Applications; December 18, 2003; San Jose, CA.
[20] Wang G, Li C, Luo MR. Improving the Hawkyard method for generating reflectance functions. *Color Res Appl*. 2005;30(4):283-287.
[21] Morovic P, Finlayson GD. Metamer-set-based approach to estimating surface reflectance from camera RGB. *J Opt Soc Am A*. 2006;23(8):1814-1822.
[22] Zhao Y, Berns RS. Image-based spectral reflectance reconstruction using matrix R method. *Color Res Appl*. 2007;32(5):343-351.
[23] Abed FM, Amirshahi SH, Abed MRM. Reconstruction of reflectance data using an interpolation technique. *J Opt Soc Am A*. 2009;26(3):613-624.
[24] Attarchi N, Amirshahi SH. Reconstruction of reflectance data by modification of Berns' Gaussian method. *Color Res Appl*. 2009;34(1):26-32.
[25] Amirshahi SH, Amirhahi SA. Adaptive non-negative bases for reconstruction of spectral data from colorimetric information. *Opt Rev*. 2010;17(6):562-569.
[26] Bianco S. Reflectance spectra recovery from tristimulus values by adaptive estimation with metameric shape correction. *J Opt Soc Am A*. 2010;27(8):1868-1877.
[27] Babaei V, Amirshahi SH, Agahian F. Using weighted pseudo-inverse method for reconstruction of reflectance spectra and analyzing the dataset in terms of normality. *Color Res Appl*. 2011;36(4):295-305.
[28] Kim BG, Han J, Park S. Spectral reflectivity recovery from the tristimulus values using a hybrid method. *J Opt Soc Am A*. 2012;29(12):2612-2621.

[29] Amiri MM, Amirshahi SH. A hybrid of weighted regression and linear models for extraction of reflectance spectra from CIEXYZ tristimulus values. *Opt Rev*. 2014;21(6):816-825.
[30] Kim BG, Werner JS, Siminovitch M, Papamichael K, Han J, Park S. Spectral reflectivity recovery from tristimulus values using 3D extrapolation with 3D interpolation. *J Opt Soc Korea*. 2014;18(5):507-516.
[31] Amiri MM, Amirshahi SH. A step by step recovery of spectral data from colorimetric information. *J Opt*. 2015;44(4):373-383.
[32] Meng J, Simon F, Hanika J, Dachsbacher C. Physically meaningful rendering using tristimulus values. *Comput Graphics Forum*. 2015;34(4):31-40.
[33] Inoue K, Hara K, Urahama K. Reflectance spectra recovery with non-negativity constraints. Paper presented at: 2016 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS); 2016; Phuket.
[34] Amiri MM, Fairchild MD. Use of spectral sensitivity variability in reflectance recovery from colorimetric information. *J Opt Soc Am A*. 2017;34(7):1224-1235.
[35] Cao B, Liao N, Li Y, Cheng H. Improving reflectance reconstruction from tristimulus values by adaptively combining colorimetric and reflectance similarities. *Opt Eng*. 2017;56(5):053104.
[36] Otsu H, Yamamoto M, Hachisuka T. Reproducing spectral reflectances from tristimulus colours. *Comput Graph Forum*. 2018;37 (6):370-381.
[37] Cohen JB, Kappauf WE. Color mixture and fundamental metamers: theory, algebra, geometry, application. *Am J Psychol*. 1985;98(2):171-259.
[38] Burns SA. Subtractive Color Mixture Computation. http://scottburns.us/subtractive-color-mixture/. Accessed July 11, 2019.
[39] Peercy MS. Linear color representations for full spectral rendering. Paper presented at: Proceedings of the ACM SIGGRAPH '93 Conference on Computer Graphics; August 1–6, 1993; Anaheim, CA.
[40] Burns SA. Chromatic adaptation transform by spectral reconstruction. *Color Res Appl*. 2019;44(5):682-693.
[41] Wandell BA. The synthesis and analysis of color images. *IEEE Trans Pattern Anal Mach Intell*. 1987;9(1):2-13.
[42] Worthey JA. Calculation of metameric reflectances. *Color Res Appl*. 1988;13(2):76-84.
[43] Jaaskelainen T, Parkkinen J, Toyooka S. Vector-subspace model for color representation. *J Opt Soc Am A*. 1990;7(4):725-730.
[44] Dannemiller JL. Spectral reflectance of natural objects: how many basis functions are necessary? *J Opt Soc Am A*. 1992;9(4): 507-515.
[45] García-Beltrán A, Nieves JL, Hernández-Andrés J, Romero J. Linear bases for spectral reflection functions of acrylic paints. *Color Res Appl*. 1998;23(1):39-45.
[46] Bonnardel V, Malone LT. Daylight, biochrome surfaces, and human chromatic response in the Fourier domain. *J Opt Soc Am A*. 2000;17(4):677-686.
[47] Kohonen O, Parkkinen J, Jääskeläinen T. Databases for spectral color science. *Color Res Appl*. 2006;31(5):381-340.
[48] Lehtonen J, Parkkinen J, Jaaskelainen T. Optimal sampling of color spectra. *J Opt Soc Am A*. 2006;23(12):2983-2988.
[49] Zuffi S, Santini S, Schettini R. From color sensor space to feasible reflectance spectra. *IEEE Trans Signal Proces*. 2008;56(2): 518-531.
[50] Flinkman M, Laamanen H, Vahimaa P, Hauta-Kasari M. Number of colors generated by smooth nonfluorescent reflectance spectra. *J Opt Soc Am A*. 2012;29(12):2566-2575.
[51] Stiebel T, Merhof D. The importance of smoothness constraints on spectral object reflectances when modeling metamer mismatching. Paper presented at: IEEE International Conference on Computer Vision Workshops; October 22–29, 2017; Venice, Italy.
[52] *The Munsell Book of Color*. New Windsor, NY: Gretag Macbeth LLC; 2004.
[53] Wyszecki G, Stiles WS. *Color Science, Concepts and Methods, Quantitative Data and Formulae*. 2nd ed. New York, NY: John Wiley & Sons; 1982:179-183.
[54] Wyszecki G, Stiles WS. *Color Science, Concepts and Methods, Quantitative Data and Formulae*. 2nd ed. New York, NY: John Wiley & Sons; 1982:768-775.
[55] GNU Octave Scientific Programming Language. https://www.gnu.org/software/octave/. Accessed April 16, 2019.
[56] Mathworks MATLAB Documentation: mldivide. https://www.mathworks.com/help/matlab/ref/mldivide.html. Accessed July 1, 2019.
[57] Wikipedia: LAPACK. https://en.wikipedia.org/wiki/LAPACK. Accessed July 11, 2019.
[58] Supplementary Documentation: 3. Method 2: Colors Within the Spectral Locus (Real Colors). http://scottburns.us/suppl-docs-num-meth-smoothest-refl-recon/#Sec3. Accessed July 11, 2019.
[59] Supplementary Documentation: 4. Method 3: Colors Within the Object Color Solid (Object Colors). http://scottburns.us/suppl-docs-num-meth-smoothest-refl-recon/#Sec4. Accessed July 11, 2019.
[60] Spectral reflectance measurements of X-Rite's 2007 Munsell Book of Color (Glossy Finish). http://www.munsellcolourscienceforpainters.com/MunsellResources/SpectralReflectancesOf2007MunsellBookOfColorGlossy.txt. Archived at https://bit.ly/2FiJKpp. Accessed July 11, 2019.
[61] Wikipedia: Color Difference. https://en.wikipedia.org/wiki/Color_difference. Accessed July 11, 2019.
[62] Supplementary Documentation: 5. Comparison to Measured Object Reflectances. http://scottburns.us/suppl-docs-num-meth-smoothest-refl-recon/#Sec5. Accessed July 11, 2019.
[63] Cohen JB. Color and color mixture: scalar and vector fundamentals. *Color Res Appl*. 1988;13(1):5-39.
[64] Supplementary Documentation: 7. Summary. http://scottburns.us/suppl-docs-num-meth-smoothest-refl-recon/#Sec7. Accessed September 2, 2019.

AUTHOR BIOGRAPHY

Scott A. Burns is a retired engineering professor at the University of Illinois at Urbana-Champaign (UIUC). He has been interested in colorimetry since the 1980s, when he was introduced to the topic by his colleague Jozef Cohen. His other research interests include engineering design optimization, structural design, mechanism design, mechatronics, and data visualization. He received a National Science Foundation Presidential Young

Investigator award in 1989, a UIUC College of Engineering Everitt Award for Teaching Excellence in 1990, was named a fellow in the UIUC Center for Advanced Study in 1992, and received two State-of-the-Art in Civil Engineering Awards from ASCE in 1998 and 2004.

How to cite this article: Burns SA. Numerical methods for smoothest reflectance reconstruction. *Color Res Appl*. 2020;45:8–21. https://doi.org/10.1002/col.22437