An Addendum to Rational Function Fitting
Laurent Belcour, Oliver Salazar-Celis, Romain Pacanowski

To cite this version:


HAL Id: hal-00913516
https://inria.hal.science/hal-00913516v2
Submitted on 22 Sep 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
An Addendum to Rational Function Fitting

L. Belcour  
Inria Bordeaux Sud-Ouest  

O. Salazar-Celis  
University of Antwerp  

R. Pacanowski  
LP2N-CNRS  

September 16, 2014

In the paper Rational BRDF [PSCS’12] Pacanowski et al. introduced a new algorithm to fit BRDF data. The fitting algorithm is an extension of the Rational Vertical Segment Interpolation method introduced by Salazar Celis et al. [SCV07]. In this document we clarify some technical and numerical aspects of the fitting procedure that need to be taken into account in order to get satisfactory fitting results.

1 Rational BRDF Fitting Method

Rational BRDF relies on fitting a multivariate Rational Function (RF) \( r_{n,m}(\mathbf{x}) \) to approximate a set of measurements \( \rho(\mathbf{x}_i) \) representing a BRDF (i.e., \( r_{n,m}(\mathbf{x}_i) \approx \rho(\mathbf{x}_i) \)). Here \( \mathbf{x} \) denotes a finite dimensional vector of real variables \( x_i \) and the Rational Function \( r_{n,m}(\mathbf{x}) \) is of the form

\[
r_{n,m}(\mathbf{x}) = \frac{\sum_{j=1}^{n} p_j b_j(\mathbf{x})}{\sum_{k=1}^{m} q_k b_k(\mathbf{x})},
\]

where \( b_j(\mathbf{x}) \) and \( b_k(\mathbf{x}) \) are multivariate basis functions.

The goal of the fitting/approximation procedure is to find a RF of low complexity, counted by the total \((n + m)\) number of coefficients \( p_j \) in the numerator and \( q_k \) in the denominator, and such that

\[
\frac{f_l}{\delta} \leq r_{n,m}(\mathbf{x}_i) \approx \rho(\mathbf{x}_i) \leq \frac{f_u}{\delta}, \quad i = 0, \ldots, s.
\]

Here \([f_l, f_u]\) are custom intervals that bound the allowed variation between the RF and the \((s + 1)\) measured BRDF values \( \rho(\mathbf{x}_i) \). Ideally \( n + m \) is as low as possible \((n + m \ll s)\) to compress the original data efficiently.

For a given a number of coefficient (i.e., \( n + m \) is fixed), one may solve a quadratic problem \( \mathcal{P}(n, m) \) (cf. Section 2.1 in [PSCS’12]) to obtain a RF that satisfies Equation (2):

\[
\arg \min_{\mathbf{c} \in \mathbb{R}^{n+m+2}} ||\mathbf{c}||_2
\]

subject to

\[
\mathbf{A}^{(j)}_{n,m} \mathbf{c} - \delta ||\mathbf{A}^{(j)}_{n,m}||_2 \geq 0, \quad j = 1, \ldots, 2s + 2
\]

with \( \mathbf{c} = (p_0, \ldots, p_n, q_0, \ldots, q_m)^t \)

and \( \mathbf{A}^{(j)}_{n,m} \) denoting the \( j \)-th row of the matrix

\[
\mathbf{A}_{n,m} = \begin{pmatrix}
    b_0(\mathbf{x}_0) & \ldots & b_n(\mathbf{x}_0) & \ldots & -f_0 b_0(\mathbf{x}_0) & \ldots & -f_0 b_m(\mathbf{x}_0) \\
    \vdots & \ddots & \vdots & & \vdots & \ddots & \vdots \\
    b_0(\mathbf{x}_s) & \ldots & b_n(\mathbf{x}_s) & \ldots & -f_s b_0(\mathbf{x}_s) & \ldots & -f_s b_m(\mathbf{x}_s) \\
    -b_0(\mathbf{x}_0) & \ldots & -b_n(\mathbf{x}_0) & \ldots & f_0 b_0(\mathbf{x}_0) & \ldots & f_0 b_m(\mathbf{x}_0) \\
    \vdots & \ddots & \vdots & & \vdots & \ddots & \vdots \\
    -b_0(\mathbf{x}_s) & \ldots & -b_n(\mathbf{x}_s) & \ldots & f_s b_0(\mathbf{x}_s) & \ldots & f_s b_m(\mathbf{x}_s)
\end{pmatrix}
\]
and where $\| \cdot \|_2$ denotes the Euclidean norm. To avoid under- or overflow of the computed coefficients, the real valued tolerance $\delta > 0$ is set to be the reciprocal of the condition number of the matrix $A_{n,m}$.

Although any kind of basis functions can be used for $b_k(x)$ (resp. $b_j(x)$), to solve $P(n,m)$, it is advised to use a polynomial basis which is orthogonal (e.g., tensor products of Legendre or Chebyshev polynomials) on the fitting domain, after rescaling to the domain of orthogonality (e.g., $[-1,1]^2$). This generally improves the conditioning of the matrix $A_{n,m}$ and thus benefits the accuracy of the quadratic programming problem.

If one desires a monomial basis, for instance because of its efficient evaluation (by using the Horner factorization method), it is always possible to recast any polynomial back into monomial form a posteriori. For instance, this can be achieved in a stable manner by inverting a Vandermonde matrix evaluated in the roots of unity. Instead, the monomials are orthogonal on the complex unit circle. Therefore, direct fitting on their domain of orthogonality rather than over the BRDF fitting domain would require complex arithmetic and nonlinear optimization. Further details may be found in [GI87].

2 Illustrative Example

To illustrate our statement consider the approximation of the following analytical function:

$$f(x) = 1000e^{-x^2}x^2 + 0.1e^{-100(x-9)^2}x^3 + 0.1$$

As shown in Figure 1, this function is composed of two asymmetric peaks. One of them varies slowly compared to the other.

We used the Matlab Optimization toolbox to solve the quadratic program, and set the algorithm to be interior point convex.

In the Figure 2 and Figure 3, we compare the resulting fits when interpolating $f(x)$ using 40 coefficients (20 for the numerator and 20 for the denominator), for different basis functions $b_k$:

- Legendre polynomials with the domain of the function scaled to $[-1,1]$.
- Chebychev polynomials with the domain of the function scaled to $[-1,1]$. 
and monomials with the domain of the function symmetrically centered around 0. Such centered monomials are partially orthogonal on a symmetric domain since the dot product between even degree monomials and odd degree monomials is always zero.

In Figure 2, we display the resulting interpolants over the input domain and the vertical segment envelop. Matlab provides a solution for all of the quadratic problems. However, due to ill-conditioning, we clearly notice that the constraints are in fact not satisfied in the case of the centered monomials basis. In Figure 3, we confirm this by displaying the error of each fit to the target curves.

The (reciprocals of the) condition numbers of each of the formulations are reflected by the $\delta$ parameter. Intuitively, $\delta$ is a global shift of the constraints to avoid a solution with a very small or very large norm. We output the different $\delta$ as well as the minimum and maximum singular values for the three cases:

<table>
<thead>
<tr>
<th>Basis</th>
<th>$\delta$</th>
<th>min</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Legendre</td>
<td>1.4e-05</td>
<td>0.10</td>
<td>7160</td>
</tr>
<tr>
<td>Chebychev</td>
<td>1.0e-05</td>
<td>0.11</td>
<td>11023</td>
</tr>
<tr>
<td>Monomials</td>
<td>1.8e-10</td>
<td>1.5e-06</td>
<td>8173</td>
</tr>
</tbody>
</table>

Although the maximum singular value of $A$ seems to be stable (around $10^4$), the minimum singular value dramatically drops when monomials are used. Consequently, the associated condition number is huge. Results from such ill-conditioned problems should not be trusted. As a rule of thumb, remember that the precision in terms of decimal digits is approximately 7 digits when using single (32 bits) precision floating point value whereas it is approximately 15 digits when using double precision (64 bits).

### 3 Rational Fitting of CDFs

Rational interpolation of vertical segments can be used to perform fitting of cumulative density functions (cdf). These functions are useful to generate sampling patterns in the context of importance sampling.
Pacanowski et al. give the following constraints (cf. Equation 4 of [PSCS*12]) to fit a rational function that represents a cdf:

$$\forall x_i, \quad p'(x_i) \geq \int f_i q'(x_i) \quad \text{and} \quad p'(x_i) \geq \int f_i q'(x_i)$$

(3)

In this section we explain where these constraints are coming from. Since a BRDF is positive real valued function ($\forall x_i, \rho(x_i) \geq 0$). Therefore, the fitted rational function $r(x)$ must also be positive:

$$\forall x_i, \quad r(x_i) \approx \frac{p(x_i)}{q(x_i)} \geq f_i \geq 0 \iff \begin{cases} p(x_i) \geq 0 \quad \text{and} \quad q(x_i) > 0 \quad (a) \\ p(x_i) \leq 0 \quad \text{and} \quad q(x_i) < 0 \quad (b) \end{cases}$$

Let us remind the reader that since a cumulative density function (CDF) is the integral of the probability density function, the CDF is a positive monotonic increasing function. Therefore we are looking for contraints that guarantees that the derivative of the fitted rational function, $r'(x)$, is positive or null:

$$\forall x_i, \quad r'(x_i) \geq 0 \iff \frac{p'(x_i)q(x_i) - p(x_i)q'(x_i)}{q^2(x_i)} \geq 0$$

$$\iff p'(x_i)q(x_i) - p(x_i)q'(x_i) \geq 0$$

$$\iff p'(x_i)q(x_i) \geq p(x_i)q'(x_i)$$

(4)

If we consider case (a) where $q(x) > 0$, then we can divide the last inequality of Equation 4 by $q(x)$ yielding:

$$\forall x_i, \quad q(x) > 0 \implies p'(x_i) \geq \frac{p(x_i)}{q(x_i)} q'(x_i) \iff p'(x_i) \geq \max \left( \frac{\int f_i q'(x_i)}{\int i q'(x_i)} \right)$$

(5)

(because $\forall x_i, \quad f_i \leq \frac{p(x_i)}{q(x_i)} \leq \int i$).
Note that if we consider case (b) where $q(x) < 0$, then the inequality changes:

$$\forall x, q(x) < 0 \implies p'(x_i) \leq \frac{p(x_i)}{q(x_i)} q'(x_i) \iff p'(x_i) \leq \min \left( f_i q'(x_i), f_i q'(x_i) \right)$$  \hspace{1cm} (6)

In the rational BRDF paper, Pacanowski et al. [PSCS*12] considered case (a) and did not implement the constraint by taking the maximum but by enforcing it as the double constraint of Equation 3 even if one is redundant.

4 Conclusion

To fit Rational Functions accurately, we advise to use an orthogonal basis, and to scale and shift the data to match its associated domain of orthogonality. After the fitting, it is always possible to transform the fitted coefficients back to the monomial basis over the original domain to get the maximum performance when evaluating. This conversion will not change the number of coefficients as the maximum degrees are preserved in the numerator and in the denominator.

Acknowledgements This addendum has been supported by the ALTA project (ANR-11-BS02-006).

References

