An Addendum to Rational Function Fitting

Laurent Belcour, Oliver Salazar-Celis, Romain Pacanowski

An Addendum to Rational Function Fitting

L. Belcour
Inria Bordeaux Sud-Ouest

O. Salazar-Celis
University of Antwerp

R.Pacanowski
LP2N-CNRS

November 27, 2013

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

$$r_{n,m}(\boldsymbol{x}) = \frac{p_n(\boldsymbol{x})}{q_m(\boldsymbol{x})} = \frac{\sum_{j=1}^{n} p_j\, b_j(\boldsymbol{x})}{\sum_{k=1}^{m} q_k\, b_k(\boldsymbol{x})}, \tag{1}$$

where $b_j(\boldsymbol{x})$ and $b_k(\boldsymbol{x})$ 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

$$\underline{f_i} \leqslant r_{n,m}(\boldsymbol{x}_i) \approx \rho(\boldsymbol{x}_i) \leqslant \overline{f_i}, \qquad i = 0, \ldots, s. \tag{2}$$

Here $[\underline{f_i}, \overline{f_i}]$ are custom intervals that bound the allowed variation between the RF and the $(s+1)$ measured BRDF values $\rho(\boldsymbol{x}_i)$. 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):

$$\arg \min_{\boldsymbol{c} \in \mathbb{R}^{n+m+2}} ||\boldsymbol{c}||_2$$

$$\text{subject to}$$

$$\mathbf{A}^{(j)}_{n,m}\, \boldsymbol{c} - \delta ||\mathbf{A}^{(j)}_{n,m}||_2 \geqslant 0, \qquad j = 1, \ldots, 2s+2$$

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

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

$$\mathbf{A}_{n,m} = \begin{pmatrix} b_0(\boldsymbol{x}_0) & \ldots & b_n(\boldsymbol{x}_0) & -\underline{f_0}\, b_0(\boldsymbol{x}_0) & \ldots & -\underline{f_0}\, b_m(\boldsymbol{x}_0) \\ \vdots & & \vdots & \vdots & & \vdots \\ b_0(\boldsymbol{x}_s) & \ldots & b_n(\boldsymbol{x}_s) & -\underline{f_s}\, b_0(\boldsymbol{x}_s) & \ldots & -\underline{f_s}\, b_m(\boldsymbol{x}_s) \\ -b_0(\boldsymbol{x}_0) & \ldots & -b_n(\boldsymbol{x}_0) & \overline{f_0}\, b_0(\boldsymbol{x}_0) & \ldots & \overline{f_0}\, b_m(\boldsymbol{x}_0) \\ \vdots & & \vdots & \vdots & & \vdots \\ -b_0(\boldsymbol{x}_s) & \ldots & -b_n(\boldsymbol{x}_s) & \overline{f_s}\, b_0(\boldsymbol{x}_s) & \ldots & \overline{f_s}\, b_m(\boldsymbol{x}_s) \end{pmatrix}$$

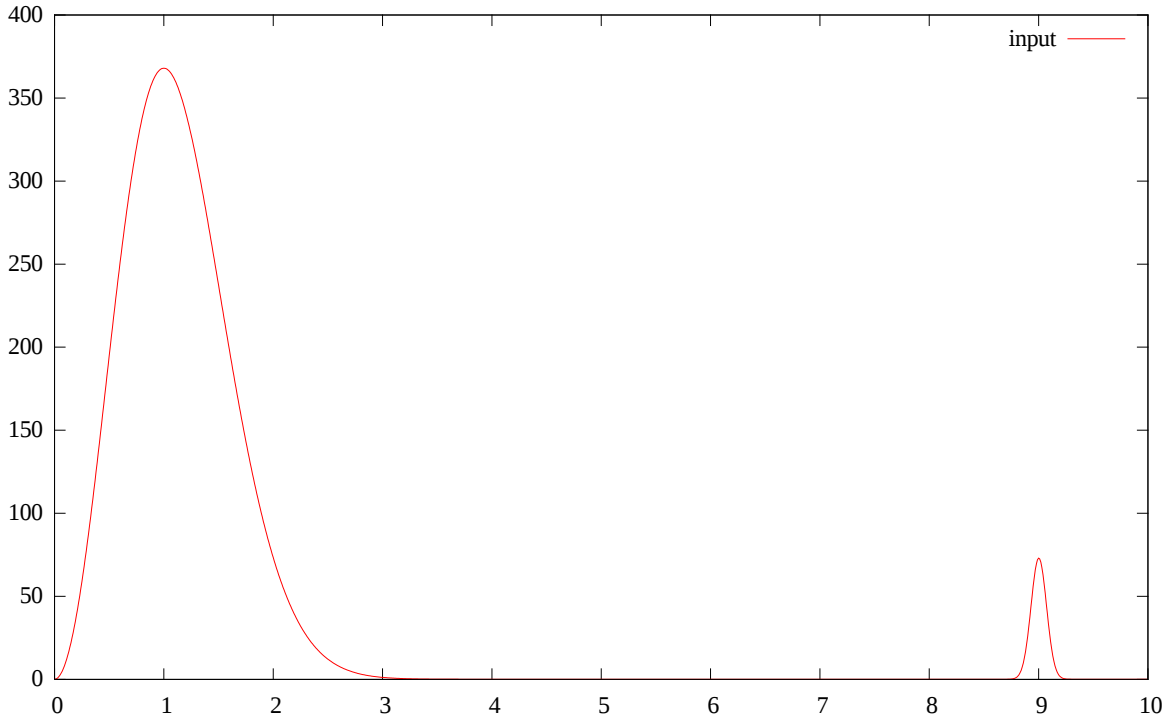

Figure 1: The function we used for our test.

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 $\mathbf{A}_{n,m}$.

Although any kind of basis functions can be used for $b_k(\boldsymbol{x})$ (resp. $b_j(\boldsymbol{x})$), to solve $\mathcal{P}(n,m)$, 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 $\mathbf{A}_{n,m}$ 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 orthogonallity rather than over the BRDF fitting domain would require complex arithmetic and nonlinear optimization. Further dertails 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 $[-1,1]$.

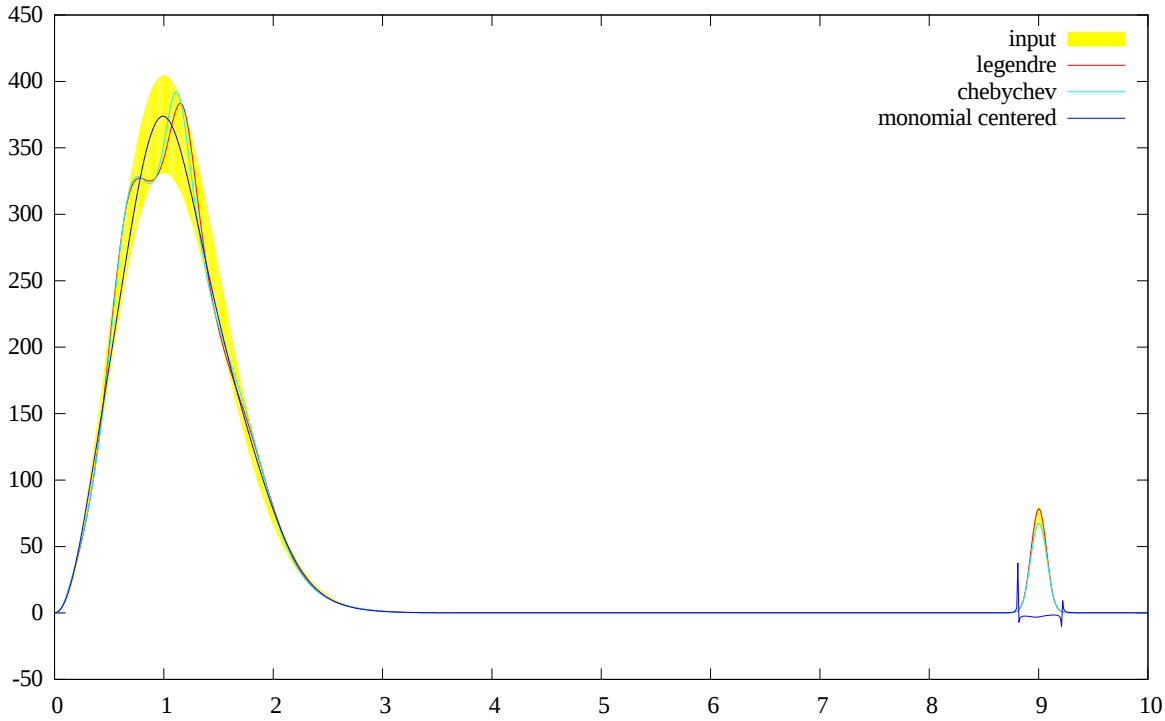

Figure 2: When using a set of vertical segments with a relative width of 0.1, we obtain the following fit of the rational function using Legendre, Chebychev and centered monomial basis. Due to ill-conditioning, the centered monomial basis fails to meet the constraints of the second peak and contains two poles.

- 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 δ parameter. Intuitively, δ is a global shift of the constraints to avoid a solution with a very small or very large norm. We output the different δ as well as the minimum and maximum singular values for the three cases:

Basis	δ	min	max
Legendre	1.4e-05	0.10	7160
Chebychev	1.0e-05	0.11	11023
Monomials	1.8e-10	1.5e-06	8173

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 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

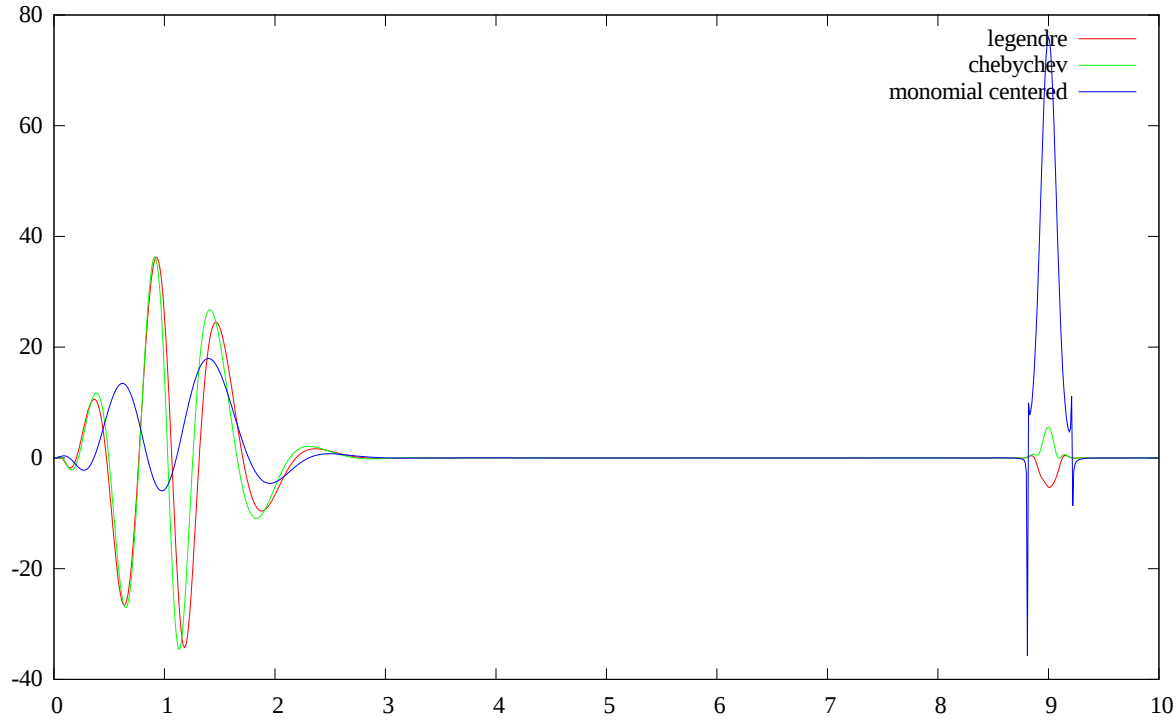

Figure 3: We display the error (computed as the difference) of each fit compared to the ground truth function. The centered monomials fail to interpolate correctly the second peak but correctly interpolate the first one.

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

[GI87] Gautschi W., Inglese G.: Lower bounds for the condition number of vandermonde matrices. *Numerische Mathematik 52*, 3 (1987), 241–250.

[PSCS*12] Pacanowski R., Salazar-Celis O., Schlick C., Granier X., Pierre P., Annie C.: Rational BRDF. *IEEE Transactions on Visualization and Computer Graphics 18*, 11 (Feb. 2012), 1824–1835.

[SCV07] Salazar Celis O., Cuyt A., Verdonk B.: Rational approximation of vertical segments. *Numerical Algorithms 45* (2007), 375–388.