# Kurtosis-based adaptive common noise rejection and spectral normalization for unknown intensity offset and gain

Gabriel M. Delgado, Matheus B. Martinelli, Ariane D. Rossi, Fabricio S. da Cruz, George C. Cardoso
*Universidade de São Paulo, Ribeirão Preto - SP*

Normalization of spectra and time series is a recurrent problem in optical signal processing. A typical example is the determination of transmission spectrum of some liquid. For a given sample, solvent and light source used have a transmission spectrum ($I_o$). If the spectrum of an unknown solute ($I_s$) is to be obtained in the same experimental conditions as spectrum ($I_o$), normalization of the spectrum is trivial. However, laser noise rejection cannot occur if the measurements are not performed simultaneously – and typically the measurements cannot be done simultaneously if the same experimental conditions are to be maintained. In addition, if measurements for $I_s$ and for $I_o$ are not performed in the same experimental conditions, for example, when measurements are done with different sensors for common noise rejection (laser fluctuation or liquid motion fluctuation rejection) offsets and gains might be different and unknown. In such conditions, the true value of $I_o$ cannot be obtained by simple normalization.

We present a practical method to normalize spectra or time series, where the spectrum used for normalization might have unknown gain and unknown offset. The physical principle used is the central limit theorem: the histogram of a mixed signal is more Gaussian than the histogram of a pure signal. In the method proposed, gain and offsets are adaptively changed to find the minimum excess kurtosis of the target spectrum $I_s$. We have shown excellent results using the proposed method. It runs in a simple script in less than one second on a regular laptop. The method proposed has comparable results to principal component analysis or variance statistics methods, while being simpler both experimentally (number of measurements) and computationally (code complexity).