

Software for Complete Mode Structure Analysis of a Light Field

Ivan A. Burenkov^{1,2,*} and Sergey V. Polyakov¹

¹National Institute of Standards and Technology,
Gaithersburg, MD 20899

²Joint Quantum Institute & University of Maryland,
College Park, MD 20742

*This author contributed to this paper as a guest researcher at the National Institute of Standards and Technology

ivan.burenkov@nist.gov
sergey.polyakov@nist.gov

Software DOI: <https://doi.org/10.18434/M3388B>

Software Version: 1.0

Key words: loss inversion; photo-number resolved measurement; quantum light source characterization.

Accepted: June 14, 2017

Published: June 26, 2017

<https://doi.org/10.6028/jres.122.030>

1. Summary

We present a software package aimed at simulating photon-number probability distributions of a range of classical and non-classical states of light. This software can generate arbitrary probability distributions from user-defined mode structure of a light field. It can also solve the reverse problem, i.e., reconstructing the mode structure of a light field from a given probability distribution. The mode structure fully defines a light field. Thus it provides information about the source of light without having to directly access the source. The multimode fields simulated by this software include those comprised of a number of thermal modes, a Poisson mode, and single-photon modes. In addition, conjugated multimode sources (such as those created via parametric downconversion (PDC) or four-wave mixing (FWM)) can be simulated. This software provides for a nearly-perfect reconstruction of multimode fields comprised of several correlated and uncorrelated thermal and Poisson modes. The code is particularly effective at characterizing mesoscopic quantum states of light. This software can be modified to include other types of modes (i.e., governed by other statistical distributions), as needed.

2. Software Specifications

NIST Operating Unit	Physical Measurement Laboratory, Quantum Measurement Division, Quantum Optics Group
Category	Evaluation of probability distributions. Designed to pair with Levenberg-Marquardt non-linear fitting algorithm.
Operating Systems	Generation: Any OS Reconstruction: Any OS supporting GNU Scientific Library
Programming Language	C
Inputs/Outputs	Generation: number and population of light modes of simulated light source/ Square JPD matrix as a column of double numbers ASCII format file Reconstruction: Square JPD matrix as a column of double numbers ASCII format file and an initial guess for a fitting algorithm/ populations of light modes of used for light source reconstruction
Documentation	Source code and binaries: https://github.com/usnistgov/FSMR
Disclaimer	https://www.nist.gov/disclaimer

3. Methods

Using statistical approaches for complex system analysis is a powerful, broadly applicable technique. More specifically, basic sub-systems comprising an object under investigation can be identified by their contribution to that object's statistical properties. We find that this idea works well for mesoscopic and macroscopic states of light [1, 2]. Using the measured photon number statistics of such a state of light, we extract the contributions of its elementary sub-processes, i.e., its constituent optical modes.

In the simplest case of a multimode light field, all the modes are independent; we refer to this type of light source as an uncorrelated light source (ULS). There, however, are particular types of light sources with highly dependent modes. For example, there are sources that generate photons in pairs into separate modes. These sources provide a correlated contribution to their photon statistics. We refer to this type of a light source as conjugated light source (CLS). These two types of multimode fields cover a large class of classical and non-classical sources important in quantum optics. We introduce statistical mode decomposition methods for both types of sources. First, we consider a ULS comprised of one or more modes. In this case, light sources are described by a one-dimensional photon number distribution (PND) expressed as a series of probabilities $P(M)$, where M is the number of photons in the field. Typically, a PND $P(M)$ can be measured directly. The PND contains the information about underlying modes [1]. Second, we extend our method to CLS that generate two correlated beams (called signal and idler for historical reasons). Such sources are described by a two-dimensional joint probability distribution (JPD) $P(n_s, n_i)$, i.e., probabilities to get n_s and n_i photons in the signal and the idler arms correspondingly, also accessible through an experiment. Just like in the case of the PND, the JPD contains rich information about the underlying modes and their correlations [2]. Our software builds PNDs and JPDs of multimode fields from statistical distributions of elementary modes and includes losses. We also implement the reverse algorithm: i.e., finding elementary modes and loss (efficiency) coefficients from a PND or a JPD.

3.1 Photon Number Distribution of a Multimode Uncorrelated Light Source

Consider the lossless case of a multimode ULS first. The probability to generate k photons in a mode $p_\mu(k)$ is governed by that mode's statistics and its mean photon number μ . As defined, the modes are independent. Therefore, the probability to simultaneously generate k_j photons in each mode j is given by the product of the probabilities:

$$P(k_1, \dots, k_j, \dots) = \prod_j p_{\mu_j}(k_j).$$

We are interested in finding a probability of generating a total of M photons. The additive rule for a probability of mutually exclusive events gives:

$$P(M) = \sum_{\Sigma k_j = M} \prod_j p_{\mu_j}(k_j). \quad (1)$$

In quantum optics losses are modeled as a beam splitter operator acting on a state [3]. Thus, loss factors L can be written in form of binomial coefficients $L_{n,k}(\eta) = \eta^n (1-\eta)^{k-n} k! / ((k-n)! n!)$, where η is an efficiency coefficient that includes propagation losses and detection efficiency (beam splitter transmittance), k is the number of impinging photons, and n is the number of photons at the output of this beam splitter.

$$P(M) = \sum_{k=M}^{\infty} \sum_{\Sigma k_j = k} \prod_j p_{\mu_j}(k_j) L_{m_j, k_j}(\eta_j). \quad (2)$$

Note that loss applied to photon states represented by Poissonian $p_\mu^{\text{Pois}}(k) = \exp(-\mu) \mu^k / k!$, Binomial $p_\mu^{\text{SP}}(0) = (1-\mu)$; $p_\mu^{\text{SP}}(1) = \mu$; $p_\mu^{\text{SP}}(k > 1) = 0$ or Bose-Einstein $p_\mu^{\text{Therm}}(k) = \mu^k / (1+\mu)^{k+1}$ statistics does not change statistics [3]. For these states, the mean photon number simply gets reduced by an efficiency coefficient η . This property allows the introduction of adjusted mean photon numbers for uncorrelated modes, $\tilde{\mu}_j = \eta \mu_j$, significantly simplifying the master equation set:

$$P(M) = \sum_{\Sigma k_j = M} \prod_j p_{\tilde{\mu}_j}(k_j), \quad (3)$$

Note that Eq. (1) is mathematically identical to Eq. (3). Therefore, for our types of statistics, losses cannot be extracted from ULS PNDs. Thus, losses are not separately considered.

3.1.1 PND Generation

To generate a PND from a known set of modes call a C function defined in file *func.c*:

```
inline int gen_rpd_z (double * x_init, int * pp, int nn, double * z);
```

where $pp[3] = \{n_T, n_{SP}, n_P\}$ is the vector containing the number of modes of all three types (Thermal, Single Photon, Poisson);

$p = pp[0] + pp[1] + pp[2]$ is the total number of modes;

$*x_init$ is the vector of mode occupations (average photons per mode per observation time bin) of length p ;

nn is the length of PND (or the maximal photon number plus one);

$*z$ is the PND array of length nn .

Example: To generate a PND for the first 50 photon number states of ULS, comprised of three thermal and one Poisson modes with the average numbers of photons $\mu_1^T = 5$, $\mu_2^T = 3$, $\mu_3^T = 1$, $\mu_4^P = 2.5$, use the following C code:

```

#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <math.h>

#include "func.c"

int main (){
    int i;
    int pp[3]={3,0,1};
    int p = pp[0]+pp[1]+pp[2];
    int nn=50;
    double * x_init = (double *)malloc(p*sizeof(double));
    x_init={5.0, 3.0, 1.0, 2.5};
    double *z = (double *)malloc(nn*sizeof(double));
    ////////////////////////////////////////////////////////////////////
    gen_rpd_z(x_init , pp, nn, z);
    ////////////////////////////////////////////////////////////////////
    for (i=0;i<nn;i++){
        printf("P(%d)=%lf\n",i,z[i]);
    }
    return 0;
}

```

3.1.2 PND Reconstruction

To reconstruct a PND (i.e., to obtain mode parameters for a given PND) we adapted `gen_rpd_z` in `func.c` to conform with the input and the output format of a Levenberg-Marquardt non-linear fitting algorithm from the GNU Scientific Library (GSL), for details see the GSL manual [4].

```
inline int mode_reconstruction_RPD_f (const gsl_vector * x, void *data, gsl_vector * f)
```

An example of a PND reconstruction source code using the GSL can be found in `rpd_fit.c`.

Example A precompiled binary can be called from a bash command line:

```
./rpd_fit pp[0] pp[1] pp[2] x_init[0] ... x_init[p] /path/to/PND/data.dat
```

This contains the appropriate number of modes `pp` of the three types described in section 3.1.1 and an initial guess for the average number of photons `x_init` in each of these modes. The last argument points to a tab-separated text file containing PND values starting with $P(M=0)$.

3.2 Joint Probability Distribution of a Multimode Conjugated Light Source

We extend our treatment to the characterization of CLS. By CLS here we mean a wide class of parametric down-conversion or four-wave mixing sources resulting in multimode correlated fields and probably containing background, uncorrelated fields. Correlated fields are pairs of fields in which the photons are generated simultaneously: one in each field. As mentioned before, the photon-number statistics of the two outputs is represented by a JPD $P(n_s, n_i)$, i.e., the probability to get n_s and n_i photons in the signal and the idler arms respectively. Again, we start with a lossless case.

The conjugated fields are comprised of correlated “c” and uncorrelated “u” modes. For the probability of generating M photons in all uncorrelated modes one writes:

$$P_{u,s,i}(M) = \sum_{\Sigma k_j=M} \prod_j p_{\mu_j}(k_j).$$

Photons in correlated modes are generated in pairs: the same number in both “s” and “i” fields: $N_s = N_i$. Similarly,

$$P_c(N_s, N_i) = \sum_{\Sigma k_j=N_s=N_i} \prod_j p_{\mu_j}(k_j).$$

Then, the joint probability distribution is given by applying probability multiplication and additive rules:

$$P(n_s, n_i) = \sum_{\substack{N_s + M_s = n_s \\ N_i + M_i = n_i}} P_c(N_s, N_i) P_{u_s}(M_s) P_{u_i}(M_i). \quad (4)$$

This result can be generalized for the case of losses. As before, losses for uncorrelated modes represented by Poissonian, Binomial and Bose-Einstein statistics are described through adjusted mean photon numbers:

$$P_{u_{s,i}}(M) = \sum_{\sum k_j = M} \prod_j p_{\bar{\mu}_j}(k_j), \quad (5)$$

For correlated modes, losses in each arm should be independently considered, therefore introducing effective mean photon numbers is no longer possible:

$$P_c(N_s, N_i) = \sum_{k=0}^{\infty} \sum_{\substack{\sum n_j^s = k \\ \sum n_j^i = k}} \prod_j p_{\mu_j}(k_j) L_{n_j^s, k_j}(\eta_j^s) L_{n_j^i, k_j}(\eta_j^i). \quad (6)$$

Equation (4) is not affected by the losses. Therefore, Eqs. (4), (5), (6) describe multimode fields comprised of modes with Poissonian, Binomial and Bose-Einstein statistics. Using Eqs. (4), (2), (6) generalizes our method to modes with any underlying combinations of statistical distributions.

Our algorithm follows the structure of Eq. (4). We separately compute $P_c(N_s, N_i)$ (including losses) and $P_{u_{s,i}}(M)$. Then, appropriate summations take place.

Each underlying probability distribution is coded as a separate function, thus simplifying the possible extension of the code to the more exotic modes. The three functions supplied with this code correspond to thermal modes governed by Bose-Einstein statistics: $p_{\mu}^{\text{Therm}}(k) = \mu^k / (1 + \mu)^{k+1}$, Poisson modes: $p_{\mu}^{\text{Pois}}(k) = \exp(-\mu) \mu^k / k!$, and single-photon modes governed by binomial statistics: $p_{\mu}^{\text{SP}}(0) = (1 - \mu)$; $p_{\mu}^{\text{SP}}(1) = \mu$; and $p_{\mu}^{\text{SP}}(k > 1) = 0$.

3.2.1 JPD Generation

To generate a JPD given the underlying modes and their parameters, call the C function defined in *func.c* file:

```
inline int gen_jpd_z (double * x_init, int * pp, int nn, double * z);
```

where `pp[9]` is a 3x3 matrix (3 vectors that defines the types of modes for each arm: Conjugated (`pp[0..2]`), Signal (`pp[3..5]`) and Idler (`pp[6..8]`)). These vectors describe number of mode of each type (Thermal, Single Photon, Poisson), cf. a PND generation function, described earlier.
`*x_init` is a vector of parameters of length $(p = \sum pp[i] + 2)$ [Average number of photons in each mode + 2 efficiency parameters which describe losses];
`nn` - is the size of the output JPD matrix `nn*nn`;
`*z` - is the array of length `nn*nn` where the output JPD is stored.
The source code that implements the JPD generation can be found in *jpd_gen.c* file.

Example The precompiled binary is called from the bash command line:

```
./gen_jpd pp[0] ... pp[8] nn x_init[0] ... x_init[p]
```

For example, to generate a 20x20 JPD matrix of a source comprised of a single squeezed vacuum mode with an average photon number 5, a Poisson background with an average number of photons equal to 1 and 2 for signal and idler arms respectively, and corresponding efficiencies of 40% and 60% run:

```
./gen_jpd 1 0 0 0 0 1 0 0 1 20 5 1 2 0.4 0.6
```

As a result, a file

gen_JPD20_100_001_001_mu1_5.00000_mu2_1.00000_mu3_2.00000_etaS_0.40000_etaI_0.60000.dat will be generated. The output file contains the 20x20 JPD matrix in the form of a single column of 400 double-precision numbers in ASCII format. The first 20 elements correspond to the first column of JPD giving the probability to get 0 to 19 photons in the *Idler* arm for zero photons in the *Signal* arm, etc.

3.2.2 JPD Reconstruction

To solve the reverse problem i.e., to reconstruct mode structure, our code uses a GNU Scientific Library implementation of Levenberg-Marquardt non-linear fitting algorithm [4]. The JPD generation function is used by the algorithm to fit the input JPD data. The complete example source code for JPD reconstruction can be found in *jpd_fit.c* file. To call the reconstruction algorithm, run:

```
./fit_jpd pp[0] ... pp[8] x_init[0] ... x_init[p] /path/to/JPD/data.dat
```

The call is supplemented with the appropriate number of modes in each arm *pp* (9 parameters) and an initial guess for the average number of photons in each mode (*p*-2 parameters) and detection efficiency (2 parameters). The last argument points to a tab-separated text file containing JPD values starting with $P(0,0)$.

Using the data generated in the previous example, we now reconstruct the mode populations:

```
./fit_jpd 1 0 0 0 0 1 0 0 1 0.5 0.5 0.5 0.5 0.5 ./gen_JPD20_100_001_001_mu1_5.00000_mu2_1.00000_mu3_2.00000_etaS_0.40000_etaI_0.60000.dat
```

Note that in this example we keep the number and types of modes exactly the same as in the generating code, but use an arbitrary initial guess. As a result, a file

fit_JPD20_100_001_001_mu1_5.00000_mu2_1.00000_mu3_2.00000_etaS_0.40000_etaI_0.60000.dat will be generated, containing the reconstructed JPD matrix of the same size as the input JPD in the form of column if double numbers in ASCII format. The console output will display the final fit result (mode populations, detection efficiencies, and associated uncertainties), along with information on each iteration of the fitting algorithm.

3.3 Intended Usage: An Unknown Mode Structure Reconstruction Example

We found [2] that a JPD reconstruction through photon-number statistics works well when the number of modes and their types are known beforehand. Furthermore, adding unoccupied modes to the reconstruction only requires expanding the experimental data sets to achieve the same accuracy, but otherwise does not negatively affect the reconstruction. However, not including all modes present in a reconstruction leads to significant errors in the entire set of recovered parameters. In the most general case, it is useful to establish a method to identify an *a priori* unknown mode structure based on a series of reconstructions, a situation typical for the experimental data.

We propose to use reduced probability distributions (RPD) to determine the number and type of modes in each arm. RPD is a particular case of a PND for ULS obtained from JPD for CLS by summation over the *Signal* or *Idler* channels. RPDs fit faster and potentially with better accuracy because the uncertainty of the RPD values is typically lower than that of the JPD data. In most cases, it is important to determine if a Poisson mode is present, or a distribution can be well-described by a finite number of thermal modes. If reconstructions yield mode populations that changes with the number of thermal modes allowed, the number of thermal modes in the reconstruction should be increased (and/or a Poisson mode allowed) until the reconstruction no longer depends on the number of modes allowed, see [2] for details. The size of the experimental data set, including both the maximum photon number detected and the uncertainty of JPD data, will ultimately limit the number of modes that can be included in an accurate reconstruction.

3.3.1 Example

We start from an arbitrary JPD provided in file *data.dat*. First, we obtain RPDs:

```
./jpd2rpd data.dat
```

This step produces the two files *data.dats* and *data.dati* that contain RPDs for signal and idler arms correspondingly. We recover mode structure in each arm by iteratively choosing a trial number and type of modes (but not their brightness) and applying RPD fitting algorithm:

```
./rpd_fit pp[0] pp[1] pp[2] x_init[0] ... x_init[p] data.dati
```

```
./rpd_fit pp[0] pp[1] pp[2] x_init[0] ... x_init[p] data.dats
```

As a result, we get two arrays of types and number of modes *pp*, one for each arm. These are all the modes (correlated and uncorrelated) that exist in the system. We now need to sort these modes into correlated and uncorrelated by fitting a full JPD to a model given by the previous step.

The accuracy of the full reconstruction is seen from the fit's uncertainty (its P-value) and comparison with the mode brightness obtained in an RPD measurement.

For more detailed discussion on experimental data reconstruction, mode number extraction and reconstruction robustness versus shot noise see our paper [2].

4. References

- [1] Goldschmidt EA, et al. (2013) Mode reconstruction of a light field by multiphoton statistics. *Phys Rev A* 88:013822. <https://doi.org/10.1103/PhysRevA.88.013822>
- [2] Burenkov IA, et al. (2017) Full statistical mode reconstruction of a light field via a photon-number-resolved measurement. *Phys Rev A* 95:053806. <https://doi.org/10.1103/PhysRevA.95.053806>
- [3] Mandel L, Wolf E (1995) *Optical Coherence and Quantum Optics* (Cambridge University Press).
- [4] Gough B (2009) *GNU Scientific Library Reference Manual - Third Edition* (Network Theory Ltd.), 3rd Ed. <https://www.gnu.org/software/gsl/manual/gsl-ref.html>.

About the authors: *The authors' interests include application of quantum information technologies to measuring and characterizing faint states of light of any origin: from single-photon sources to optically-active biological systems. Dr. Ivan A. Burenkov is a guest researcher in the Quantum Optics group in the Quantum Measurement Division of the Physical Measurement Laboratory, and Post-Doctoral Associate in the Joint Quantum Institute at University of Maryland.*

Dr. Sergey V. Polyakov is a physicist in the Quantum Optics group in the Quantum Measurement Division of the Physical Measurement Laboratory.

The National Institute of Standards and Technology is an agency of the U.S. Department of Commerce.