ProFiDo XML Interchange Format Specification

Falko Bause, Jan Kriege
Informatik IV, TU Dortmund
{falko.bause,jan.kriege}@tu-dortmund.de

July 24, 2014

ProFiDo (Processes Fitting Toolkit Dortmund) [1, 5] is a graphical toolkit that provides a consistent use of commandline-oriented tools for the fitting of stochastic processes and distributions like Phase-type distributions, Markovian Arrival Processes, CAPPs, ARIMA and ARTA Processes. For process descriptions ProFiDo uses an XML based interchange format that is specified in this document.
Sect. 1 describes the general document structure. In Sect. 2 the XML specification of various distributions is presented followed by the definition for the different stochastic processes mentioned above. We expect the reader to have some basic knowledge about the different stochastic processes treated in this document and hence, they are only introduced very briefly providing some additional references to the literature.

1 General Document Structure of the Interchange Format

The document description should begin with an XML declaration like for example `<?xml version="1.0"?>`. The single root element of the process description is called profido with the start tag `<profido>` and the end tag `</profido>`. The process specification itself is expected between these tags.
Note, that a ProFiDo process description may contain several alternative representations of the same process or distribution. For example an Erlang distribution can be described by specifying the number of phases and the rate (cf. Sect. 2) or as a Phase-type distribution specifying the initial probabilities and the transition rate matrix (cf. Sect. 3). A tool that processes these descriptions can choose between the alternatives and select the most appropriate description for its purpose. An example for this is given later in Sect. 3.

2 Distributions

ProFiDo supports several distributions. Every description of a distribution is enclosed by `<dist>` and `</dist>`. The content of the Dist-Tags depends on the type of the distribution. Regarding the parameters of the distributions we followed [14]. For the Johnson distribution the reader is referred to [11] and for the Hyper-Erlang distribution to [17].

- **Exponential distribution:** The description of the exponential distribution has start tag `<expodist>` and end tag `</expodist>`. Only parameter is the mean $\beta > 0$ which is expected inside the tags `<mean>`... `</mean>`.

- **Normal distribution:** The description of the normal distribution is started by `<normaldist>` and ended by `</normaldist>`. The mean and the standard deviation of the distribution have to be specified inside the tags `<mean>`... `</mean>` and `<std>`... `</std>`, respectively.

- **Lognormal distribution:** The lognormal distribution begins with the start tag `<lognormaldist>` and ends with `</lognormaldist>`. It has the two parameters μ which is the mean of a lognormal distributed random variables natural logarithm and $\sigma > 0$ which is the standard deviation of a lognormal distributed random variables natural logarithm. The two parameters can be specified within `<mu>`... `</mu>` and `<sigma>`... `</sigma>`, respectively, so that, e.g., $E[X] = e^{\mu+\sigma^2/2}$.

- **Johnson distribution:** The description of the johnson distribution is started by `<johnsondist>` and ended by `</johnsondist>`. The family of Johnson distributions contains four types of distributions: Bounded, unbounded, lognormal and normal. These types are specified by `<type>`SB `</type>`, `<type>` SU `</type>`, `<type>`SL `</type>` and `<type>`SN `</type>`, respectively. All types of Johnson distributions are parametrized by four parameters: A shape parameter γ (`<shape1>`... `</shape1>`), a shape parameter δ (`<shape2>`... `</shape2>`), a location parameter ξ (`<location>`... `</location>`) and a scale parameter λ (`<scale>`... `</scale>`).

- **Uniform distribution:** The description of the uniform distribution has start tag `<uniformdist>` and end tag `</uniformdist>`. The uniform distribution is defined by a lower limit a and an upper limit b with $a < b$ that can be specified as `<a>` ... `</a>` and `<b>` ... `</b>`.

- **Weibull distribution:** The description of a Weibull distribution is enclosed by `<weibulldist>` and `</weibulldist>`. It has a shape parameter $\alpha > 0$ and a scale parameter $\beta > 0$ that can be defined by `<shape>`... `</shape>` and `<scale>`... `</scale>`, respectively.

- **Triangular distribution:** The description of a triangular distribution is started by the tag `<triangulardist>` and ended by `</triangulardist>`. It has three parameters, the lower limit a, the upper limit b and the mode c with $a < c < b$. The parameters are described within the tags `<a>` ... `</a>`, `<b>` ... `</b>` and `<c>` ... `</c>`, respectively.

- **Erlang distribution:** The description of an Erlang distribution is enclosed by the tags `<erlangdist>` and `</erlangdist>`. The distribution has two parameters: The integer-valued number of phases (`<phases>`... `</phases>`) and the rate (`<rate>`... `</rate>`).

- **Hyper-Erlang distribution:** The Hyper-Erlang distribution is the convex mixture of m Erlang distributions. The description is enclosed by the two tags

<hypererlangdist> and </hypererlangdist>. It has three parameters each containing m values separated by a blank: <prob>... </prob> contains the probabilities of the branches that must sum up to 1, <phases>... </phases> contains the integer-valued number of phases of each branch and <rates>... </rates> contains the rate for each branch.

- **Gamma distribution:** The description of a Gamma distribution is enclosed by the tags <gammadist> and </gammadist>. The distribution has two parameters: The shape parameter $\alpha > 0$ (<shape>... </shape>) and the scale parameter $\beta > 0$ (<scale>... </scale>).
- **Chi-Square distribution:** The description of the Chi-square distribution is given within the tags <chisquaredist> and </chisquaredist>. Its only parameter are the integer-valued degrees of freedom $k > 0$ which can be specified by <deg>... </deg>.

Example 1 (Exponential Distribution) *An exponential distribution with mean* 0.5 *has the following XML description:*

```xml
<?xml version="1.0"?>
<profido>
  <dist>
    <expodist>
      <mean> 0.5 </mean>
    </expodist>
  </dist>
</profido>
```

Example 2 (Uniform Distribution) *A uniform distribution with lower limit* 0.0 *and upper limit* 1.0 *is described as:*

```xml
<?xml version="1.0"?>
<profido>
  <dist>
    <uniformdist>
      <a> 0.0 </a>
      <b> 1.0 </b>
    </uniformdist>
  </dist>
</profido>
```

Example 3 (Johnson Distribution) *A johnson bounded distribution with the shape parameters* $\gamma = 0.584$ *and* $\delta = 1.386$, *location parameter* $\xi = 0.006$ *and scale parameter* $\lambda = 2.479$ *is described as:*

```xml
<?xml version="1.0"?>
<profido>
  <dist>
    <johnsondist>
      <type> SB </type>
      <shape1> 0.584 </shape1>
      <shape2> 1.386 </shape2>
      <location> 0.006 </location>
      <scale> 2.479 </scale>
    </johnsondist>
  </dist>
</profido>
```

Example 4 (Hyper-Erlang Distribution) *Consider a Hyper-Erlang distribution with* 3 *branches. The first branch has probability* 0.1 *and* 3 *phases with rate* 0.3*. The second branch has probability* 0.4 *and* 2 *phases with rate* 2.0*. The third branch has probability* 0.5 *and* 1 *phase with rate* 1.3*. The XML description of this distribution is:*

```xml
<?xml version="1.0"?>
<profido>
  <dist>
    <hypererlangdist>
      <prob> 0.1  0.4  0.5 </prob>
      <phases> 3  2  1 </phases>
      <rates> 0.3  2.0  1.3 </rates>
    </hypererlangdist>
  </dist>
</profido>
```

3 Phase-Type (PH) Distributions and Markovian Arrival Processes (MAPs)

Phase-Type (PH) distributions [16] of order n are defined by a $n \times n$ matrix D_0 with $D_0(i,j) \geq 0$ for $i \neq j$ and $D_0(i,i) \leq -\sum_{j=1,j\neq i}^{n} D_0(i,j)$ and an initial probability vector π with $\pi(i) \geq 0$ and $\sum_{i=1}^{n} \pi(i) = 1$.

A MAP [13, 15] of order n is a stochastic process defined by two $n \times n$ matrices (D_0, D_1) where D_0 has the same properties as defined for a PH distribution, $D_1 \geq 0$ and $D_0 + D_1$ is the generator of a Markov process. Matrix D_0 contains the rates of internal transitions without an arrival and matrix D_1 contains the rates of transitions generating an arrival.

MAPs can be extended to account for arrivals from different classes resulting in Multi-class-MAPs (MMAPs) [8]. MMAPs of order n with k classes are described by a $n \times n$ matrix D_0 defined as above and matrices $D_i, i = 1, \ldots, k$ with $D_i \geq 0$ specifying the arrivals of class i.

A PH description is started with `<ph>` and ends with `</ph>`. A MAP or MMAP description is enclosed by the opening tag `<map>` and the closing tag `</map>`. For PH and (M)MAP descriptions the following information is expected:

- `<states>`... `</states>`: The integer-valued order (number of states) n.
- `<classes>`... `</classes>`: The integer-valued number of classes k (only for MAPs and MMAPs). If omitted it is assumed that $k = 1$, i.e. a MAP description (D_0, D_1) is assumed.
- `<d0>` ... `</d0>`: The $n \times n$ matrix D_0 in row-major order. The matrix entries are separated by blanks.
- `<di>` ... `</di>`: For each $i = 1, \ldots, k$ where k is read from the `<classes>` tag the $n \times n$ matrix D_i is expected in row-major order (only for (M)MAPs). The matrix entries are separated by blanks. Thus, for a MAP (D_0, D_1) the entries of matrix D_1 are exptected within `<d1>` ... `</d1>`.
- `<pi>` ... `</pi>`: The initial probability vector π (only for PH distributions). The vector entries are separated by blanks.

Example 5 (Erlang Distribution) *Erlang distributions are a special subclass of PH distributions. As already mentioned in Sect. 1 several representations are possible. An Erlang distribution with* 2 *states and rate* 1.0 *can as well be described as a PH distribution with* $\pi = (1.0, 0.0)$ *and*

$$D_0 = \begin{bmatrix} -1.0 & 1.0 \\ 0.0 & -1.0 \end{bmatrix}$$

The XML description containing the two alternatives is defined as:

```xml
<?xml version="1.0"?>
<profido>
  <dist>
    <erlangdist>
      <phases> 2 </phases>
      <rate> 1.0 </rate>
    </erlangdist>
  </dist>
  <ph>
    <states> 2 </states>
    <pi> 1.0 0.0 </pi>
    <d0>
      −1.0 1.0
      0.0 −1.0
    </d0>
  </ph>
</profido>
```

Of course, only one of the two representations is required, but providing alternative descriptions allows for tools to choose the alternative that is suited best.

Example 6 (MAP) *For a MAP of order 2 defined by the two matrices*

$$D_0 = \begin{bmatrix} -1.0 & 0.7 \\ 0.0 & -2.0 \end{bmatrix}, \qquad D_1 = \begin{bmatrix} 0.1 & 0.2 \\ 1.3 & 0.7 \end{bmatrix}$$

the full XML description is:

```
<?xml version="1.0"?>
<profido>
  <map>
    <states>2</states>
    <d0>
      -1.0  0.7
      0.0  -2.0
    </d0>
    <d1>
      0.1  0.2
      1.3  0.7
    </d1>
  </map>
</profido>
```

Example 7 (MMAP) *For a MMAP of order 2 with 2 classes defined by the three matrices*

$$D_0 = \begin{bmatrix} -2.0 & 0.7 \\ 0.0 & -2.5 \end{bmatrix}, \qquad D_1 = \begin{bmatrix} 0.1 & 0.2 \\ 1.3 & 0.7 \end{bmatrix}, \qquad D_2 = \begin{bmatrix} 0.6 & 0.4 \\ 0.2 & 0.3 \end{bmatrix}$$

the full XML description is:

```
<?xml version="1.0"?>
<profido>
  <map>
    <states>2</states>
    <classes>2</classes>
    <d0>
      -2.0  0.7
      0.0  -2.5
    </d0>
    <d1>
      0.1  0.2
      1.3  0.7
    </d1>
    <d2>
      0.6  0.4
      0.2  0.3
    </d2>
  </map>
</profido>
```

4 ARIMA Processes

A time series with dependent successive values can be interpreted as a series of independent shocks a_t that are distributed Normal with zero mean and variance σ_a^2 called white noise, which is transformed to the process Z_t by a linear filter that takes the weighted sum of previous observations or white noise values. Let $\phi_i, i = 1, ..., p$ and $\theta_j, j = 1, ..., q$ be coefficients to construct the weighted sum of the previous observations and white noise values, respectively. Then a so called $ARMA(p, q)$ [7] model is described as

$$Z_t = \phi_1 Z_{t-1} + ... + \phi_p Z_{t-p} + a_t + \theta_1 a_{t-1} + ... + \theta_q a_{t-q} \tag{1}$$

If a time series does exhibit non-stationary behavior and does not vary about a fixed mean, it may be expressed by an $ARIMA(p, d, q)$ model. In this case the degree of differencing is given by d.
$ARIMA(p, d, q)$ processes are described within the tags `<arima>` and `</arima>`. They consist of p autoregressive coefficients $\phi_i, i = 1, ..., p$. The number of coefficients is specified between `<arcount>`... `</arcount>` and coefficients between `<ar>` ... `</ar>` separated by blanks. The q moving average coefficients $\theta_j, j = 1, ..., q$ are described in a similar way: `<macount>`... `</macount>` contains the number of coefficients and `<ma>` ... `</ma>` the coefficients separated by blanks. The degree of differencing d is described between `<d>` ... `</d>`. The variance of the white noise σ_a^2 is specified within `<variance>`... `</variance>`. The model described by Eq. (1) has zero mean. If the process should fluctuate around another mean this can be specified by `<mean>`... `</mean>`.
If $q = d = 0$ the ARIMA model becomes an $AR(p)$ process, if $p = d = 0$ the ARIMA model becomes an $MA(q)$ process and if only $d = 0$ we have an $ARMA(p, q)$ process.

Example 8 (ARMA(p,q) process) *Consider an* $ARMA(3, 2)$ *model with* $\phi = (0.0402, 0.4567, -0.4164)$ *and* $\theta = (0.6602, -0.1109)$. *Furthermore,* $\sigma_a^2 = 0.1711$ *and the mean of the process is* 0.0. *Then the XML description is:*

```xml
<?xml version="1.0"?>
<profido>
  <arima>
    <arcount> 3 </arcount>
    <ar> 0.0402 0.4567 -0.4164 </ar>
    <macount> 2 </macount>
    <ma> 0.6602 -0.1109 </ma>
    <d> 0.0 </d>
    <variance> 0.1711 </variance>
    <mean> 0.0 </mean>
  </arima>
</profido>
```

5 ARTA Processes

An ARTA process [9] combines an $AR(p)$ base process with an arbitrary marginal distribution. It is defined as a sequence

$$Y_t = F_Y^{-1}[\Phi(Z_t)], t = 1, 2, ...$$

where Φ is the standard normal cumulative distribution function and $\{Z_t; t = 1, 2, ...\}$ is a stationary $ARMA(p, q)$ process as defined in Sect. 4. The XML description is enclosed by `<arta>`... `</arta>` and contains the specification of an ARMA model as described in Sect. 4 and a distribution as described in Sect. 2.

Example 9 (ARTA process) *For an ARTA process that combines an exponential distribution with mean* 0.5 *with an* $AR(2)$ *process with* $\phi = (0.2, 0.15)$ *and* $\sigma_a^2 = 0.93$ *the XML description is:*

```
<?xml version="1.0"?>
<profido>
  <arta>
    <dist>
      <expodist>
        <mean> 0.5 </mean>
      </expodist>
    </dist>
    <arima>
      <arcount> 2 </arcount>
      <ar> 0.2 0.15 </ar>
      <macount> 0 </macount>
      <ma> </ma>
      <d> 0.0 </d>
      <variance> 0.93 </variance>
      <mean> 0.0 </mean>
    </arima>
  </arta>
</profido>
```

6 CHEP and CAPP Models

CAPPs (Correlated Acyclic Phase-Type Processes) [12] are related to ARTA processes and combine an acyclic Phase-Type distribution with an ARMA process. They are defined as a sequence

$$Y_t = \sum_{i=1}^{m} \delta(\Phi(Z_t), i) X_t^{(\Lambda_i)}, t = 1, 2, \ldots$$

where $X_t^{(\Lambda_i)}$ are sequences of independent and identically distributed random variables with Hypo-Exponential distribution with S_i phases and rates Λ_i that correspond to the

m elementary series [10] of the Phase-Type distribution, $\{Z_t; t = 1, 2, ...\}$ is a stationary $ARMA(p, q)$ process as defined in Sect. 4, Φ is the standard normal cumulative distribution function and $\delta(\Phi(Z_t), i)$ is a function that is 1 if $\Phi(Z_t)$ falls within the subinterval on $(0, 1)$ that corresponds to elementary series i and 0 otherwise. CHEPs (Correlated Hyper-Erlang Processes) are a subclass of CAPPs where the PH distribution is Hyper-Erlang.

The XML description of a CAPP is enclosed by `<capp>`... `</capp>` and contains the specification of an ARMA model as described in Sect. 4 and a PH distribution as described in Sect. 3.

The XML description of a CHEP is enclosed by `<chep>`... `</chep>`. It consists of the specification of an ARMA model as described in Sect. 4 and the specification of an Hyper-Erlang distribution as given in Sect. 2. Since every CHEP is a CAPP the CHEP description may also contain a CAPP part as alternative representation.

Example 10 (CHEP) *The description of a CHEP with a Hyper-Erlang distribution consisting of two branches where the first branch has* 1 *state and the second has* 2 *states with probabilities* 0.091 *and* 0.909*, rates* 0.254050 *and* 2.838471 *and* $ARMA(5, 3)$ *process is given by*

```
<?xml version="1.0"?>
<profido>
  <chep>
    <dist>
      <hypererlangdist>
        <prob> 0.091 0.909 </prob>
        <phases> 1 2 </phases>
        <rates> 0.254050 2.838471 </rates>
      </hypererlangdist>
    </dist>
    <arima>
      <arcount> 5 </arcount>
      <ar> 0.891436 0.552937 −0.424392
        0.0488508 −0.0780046 </ar>
      <macount> 3 </macount>
      <ma> −0.258672 −0.36477 −0.209127 </ma>
      <d> 0.0 </d>
      <variance> 0.218322 </variance>
      <mean> 0.0 </mean>
    </arima>
  </chep>
</profido>
```

Example 11 (CAPP) *The equivalent CAPP description is given by*

```
<?xml version="1.0"?>
<profido>
  <capp>
    <ph>
      <states> 3 </states>
      <pi> 0.091 0.909 0.0 </pi>
      <d0>
        −0.254050 0.0 0.0
        0.0 −2.838471 2.838471
        0.0 0.0 −2.838471
      </d0>
    </ph>
    <arima>
      <arcount> 5 </arcount>
      <ar> 0.891436 0.552937 −0.424392
        0.0488508 −0.0780046 </ar>
      <macount> 3 </macount>
      <ma> −0.258672 −0.36477 −0.209127 </ma>
      <d> 0.0 </d>
      <variance> 0.218322 </variance>
      <mean> 0.0 </mean>
    </arima>
  </capp>
</profido>
```

7 Files

ProFiDo has the policy that only XML files are interchanged between the different programs of a workflow. Various possible processes and distributions have been described in the previous sections. In addition to these descriptions it is necessary to include trace files as input and for example images with plots of the processes as output of the workflow. For a consistent handling of these files the XML description may contain links to existing files.
A file description starts with the tag `<file>` and ends with `</file>`. Between those tags the path to the file is given inside `<ref>`... `</ref>` and the type is specified as `<type>`... `</type>`. Possible types are currently XML−Dist for files containing the description of a distribution, XML−PH for files containing the description of a phase type distribution, XML−MAP for files containing the description of a MAP, XML−CAPP for files containing the description of a CAPP, XML−ARTA for files containing the description of an ARTA process, XML−ARIMA for files containing the description of an ARIMA process, XML−PS for PostScript files, XML−PDF for PDF files, XML−PNG for PNG files, XML−Latex for LaTeX files and XML−Trace for trace files. For trace files the number of entries in the trace is given as `<valuecount>`... `</valuecount>`. The tags `<suffix>`... `</suffix>` specify the

mandatory suffix of the file's name.

Example 12 (Trace File) *For a trace file, named* lbl3.trace*, having* 1789994 *elements and being located in directory* /home/fred *the XML specification is:*

```
<?xml version="1.0"?>
<profido>
  <file>
    <ref>/home/fred/lbl3.trace</ref>
    <type>XML-Trace</type>
    <valuecount>1789994</valuecount>
  </file>
</profido>
```

Example 13 (Postscript File) *For a postscript file, named* GFITexample_Plot1.ps *and being located in directory* /home/fred *the XML specification is:*

```
<?xml version="1.0"?>
<profido>
  <file>
    <ref>/home/fred/GFITexample_Plot1.ps</ref>
    <type>XML-PS</type>
  </file>
</profido>
```

8 Additional Information

To account for possible extensions in the format specification the `<info>` tag is defined. The `<info>` tag may be used at any level of the XML specification and may contain further tags. Currently this tag is only used by the DistFit node in ProFiDo (see [2] for details).

References

[1] Falko Bause, Peter Buchholz, and Jan Kriege. ProFiDo - The Processes Fitting Toolkit Dortmund. In *Proc. of the 7th International Conference on Quantitative Evaluation of SysTems (QEST) 2010*, pages 87–96, 2010.

[2] Falko Bause, Souffian El-Baba, Philipp Gerloff, Alparslan Kirman, Jan Kriege, and Moussa Oumarou. ProFiDo - The Processes Fitting Toolkit Dortmund - Manual, 2014. http://www4.cs.uni-dortmund.de/profido.

[3] Falko Bause, Philipp Gerloff, Alparslan Kirman, Jan Kriege, and Daniel Scholtyssek. ProFiDo XML Configuration Format Specification, 2014. http://www4.cs.uni-dortmund.de/profido.

[4] Falko Bause, Philipp Gerloff, Alparslan Kirman, Jan Kriege, and Daniel Scholtyssek. ProFiDo XML Workflow Format Specification, 2014. http://www4.cs.uni-dortmund.de/profido.

[5] Falko Bause, Philipp Gerloff, and Jan Kriege. ProFiDo - A Toolkit for Fitting Input Models. In Bruno Müller-Clostermann, Klaus Echtle, and Erwin P. Rathgeb, editors, *Proceedings of the 15th International GI/ITG Conference on Measurement, Modelling and Evaluation of Computing Systems and Dependability and Fault Tolerance (MMB & DFT 2010)*, volume 5987 of *LNCS*, pages 311–314. Springer, 2010.

[6] Falko Bause and Jan Kriege. ProFiDo XML Interchange Format Specification, 2014. http://www4.cs.uni-dortmund.de/profido.

[7] G.E.P. Box and G.M. Jenkins. *Time Series Analysis - forecasting and control*. Holden-Day, 1970.

[8] Peter Buchholz, Peter Kemper, and Jan Kriege. Multi-class Markovian arrival processes and their parameter fitting. *Performance Evaluation*, 67(11):1092–1106, 2010.

[9] Marne C. Cario and Barry L. Nelson. Autoregressive to anything: Time-series input processes for simulation. *Operations Research Letters*, 19(2):51–58, 1996.

[10] A. Cumani. On the Canonical Representation of Homogeneous Markov Processes Modeling Failure-Time Distributions. *Micorelectronics and Reliability*, 22(3):583–602, 1982.

[11] David J. DeBrota, Stephen D. Roberts, James J. Swain, Robert S. Dittus, James R. Wilson, and Sekhar Venkatraman. Input modeling with the johnson system of distributions. In *WSC '88: Proceedings of the 20th conference on Winter simulation*, pages 165–179, New York, NY, USA, 1988. ACM.

[12] Jan Kriege and Peter Buchholz. Correlated phase-type distributed random numbers as input models for simulations. *Performance Evaluation*, 68(11):1247–1260, 2011.

[13] G. Latouche and V. Ramaswami. *Introduction to Matrix Analytic Methods in Stochastic Modeling*. Society for Industrial and Applied Mathematics, 1999.

[14] A. M. Law and W. D. Kelton. *Simulation modeling and analysis*. McGraw-Hill, Boston, 3. edition, 2000. ISBN 0-07-059292-6.

[15] M. F. Neuts. A versatile Markovian point process. *Journal of Applied Probability*, 16:764–779, 1979.

[16] M. F. Neuts. *Matrix-geometric solutions in stochastic models*. Johns Hopkins University Press, 1981.

[17] A. Thümmler, P. Buchholz, and M. Telek. A novel approach for phase-type fitting with the EM algorithm. *IEEE Trans. Dep. Sec. Comput.*, 3(3):245–258, 2006.