TIME EFFICIENT SIMULATION: A SIMPLE ALGORITHM FOR PRODUCING EMPIRICAL DISTRIBUTION FUNCTIONS WITH PREDETERMINED CONSTANT ERROR

Schreiber, F.

Technical University Aachen, Aachen, Federal Republic of Germany

ABSTRACT

Considering the problem "measurement of an unknown distribution function \( F(x) \)" it is shown that recently derived formulae for the so-called objective empirical distribution function \( F_n(x) \) and its relative error \( d_F(x) \) can be made accessible to practical application by implementing a "Constant Relative Error-algorithm" (CRE-algorithm). This algorithm controls the number of independent trials \( n \) and the sorting of measured random data, such that the empirical distribution function \( F_n(x) \) is traced down to a desired minimum value (resolution) \( F_{min} \), whereby the relative error \( d_F(x) \) is kept on a practically constant level below a predetermined maximum value \( d_F_{max} \) and the computertime for sorting is reduced effectively.

Simulation examples demonstrate that the CRE-algorithm can be applied to all random process types of continuous and/or discrete nature and that it is well suited to detect rare event details of \( F(x) \) within the given resolution \( F_{min} \).

1. INTRODUCTION

Following the invention of the early traffic machines simulation techniques on digital computers have become an indispensable tool to investigate complex teletraffic systems /1/. One of the most important tasks in simulation is to determine approximately the a priori unknown d.f. \( F(x) \) of a r.v. \( x \). Often this is achieved by some kind of frequency distribution method, where due to a relatively large number of trials \( n \) the frequency \( k \) events in interval \( u \) of \( m \) intervals \( u=1,2,\ldots,m \) is assumed to be approx. normally distributed. This then leads to the application of conventional t-distribution confidence intervals for the control of error and simulation run length /2/.

In /3/ it has been shown that confidence intervals belong to the subjective methods in statistics and can be favorably replaced by an easy to handle objective error formula derived from the so-called objective Bayes-statistics. In the mean time the simulation tool has been used to put the fundamentals of this statistics (including an "extended Bayes-postulate") on an experimental basis /4/. Thus simulation techniques on computers help to gain improved statistical methods and these methods on the other hand improve simulation techniques.

2. THE OBJECTIVE EMPIRICAL DISTRIBUTION FUNCTION /5/

2.1 Frequency distribution methods suffer from the disadvantage that the information with respect to the individual magnitude of all \( x \)-values collected in a certain interval is lost. Such information loss can be avoided by the following solution to the problem "measurement of an unknown distribution function".

Given a r.v. \( x \) with unknown d.f. \( F(x) \) in the unknown range \( x_1 \leq x \leq x_\Pi \) the \( x \)-values measured in \( n \) independent trials can be sorted to yield the ordered vector

\[
(x_r) = (x_1, x_2, \ldots, x_n); \quad x_r < x_{r+1}; \quad r = 1, 2, \ldots, n-1.
\]

Then for any given point \( x \) on the real axis we observe \( r \) values "left of \( x \)" in the range \( x_1 \leq x \leq x_\Pi \) and \( v = n-r \) values "right of \( x \)" in the range \( x > x \). The evaluation of these observations leads to the so-called objective empirical distribution functions \( F_n(x) \) and \( G_n(x) \) expressing the objective posterior approximations to \( F(x) \) resp. \( G(x) = 1 - F(x) \); in addition we obtain error measures to judge these approximations. The following formulae have been derived in /5/ and are presented here in a form which covers the generalized case that the unknown d.f. \( F(x) \) may contain continuous sections and steps at discrete points /6/.

These formulae depend on the objective Bayes-statistics applied to the case of a binomial random process, see e.g. formula table in /3/. The step functions \( F_n(x) \) and \( G_n(x) \) have in case of a purely continuous r.v. \( x \) a step size \( 1/(n+2) \) and are closely related to but not identical with the empirical distribution function known from literature having a step size \( 1/n \), see e.g. /7/.

1) Abbreviations: r.v. = random variable; p.f. = probability function; p.d.f. = probability density function; d.f. = distribution function; c.d.f. = complementary distribution function. In order to simplify notation the same symbol is used for a r.v. and its value.
Objective empirical distribution function

\[
F_n(x) = \frac{(r+1)}{(n+2)} \\
G_n(x) = 1 - F_n(x) = \frac{(v+1)}{(n+2)}
\]

\[
F(x) = \frac{1}{n+2} (r+1)/(v+1)
\]

Absolute error (standard deviation)

\[
\sigma_F(x) = \sigma_G(x) = \sigma(x) = \frac{1}{n+2} \left[ \frac{(r+1)(v+1)}{n+3} \right]^{1/2}
\]

Relative error (coefficient of variation)

\[
d_F(x) = \frac{\sigma(x)}{F_n(x)} = \left[ \frac{n-r+1}{(n+3)(r+1)} \right]^{1/2}
\]

\[
d_G(x) = \frac{\sigma(x)}{G_n(x)} = \left[ \frac{n-v+1}{(n+3)(v+1)} \right]^{1/2}
\]

2.2 The straightforward application of the objective formulae eq. (2a,b) and (4a,b) gives rise to a problem which can be introduced by the simulation example fig.1 depicting \(G_n(x)\) eq.(2b) and \(d_G(x)\) eq.(4b) in case of an \(E_k\)-distributed r.v. \(x\) (Erlang-distribution of order \(k = 5\)). We find that the measured function \(G_n(x)\) has an unbalanced rel. error curve \(d_G(x)\) leading from a region of low rel. error to a region of high rel. error at the right tail of \(G_n(x)\). If we would be interested in \(F_n(x) = 1 - G_n(x)\) instead of \(G_n(x)\) then due to eq.(4a) the high and low rel. error regions would change sides, see e.g. fig.7 in /5/. Obviously the straightforward application of eq.(2a,b) and (4a,b) results principally in unbalanced rel. error curves.

2.3 Let us assume that the schedule of a simulation run contains two prescribed parameters:

a) the "resolution" \(F_{\min}\) resp. \(G_{\min}\) i.e. the lowest value to be determined of the empirical d.f. \(F_n(x)\) resp. c.d.f. \(G_n(x)\);

b) the maximum admissible value \(d_{\max}\) of the rel. error \(d_F(x)\) within the range \(F_{\min} \leq F_n(x) \leq 1\) resp. \(d_{\max}\) of \(d_G(x)\) within the range

\[
1 \leq G_n(x) \leq G_{\min}\]

From eq.(2a,b) and eq.(4a,b) it is easily derived that these two parameters determine the number of trials \(n\) which are at least necessary for the simulation run

\[
n = \frac{1-F_{\min}}{F_{\min} \cdot d_2^{2} \cdot F_{\max}} \]  

\[ \tag{5a,b} \]

The diagram fig.2 expresses the well known fact that the low error simulation of the "rare event"-tail region of a distribution function with \(F_{\min}\), \(G_{\min} \ll 1\) is expensive with respect to a great number of trials. If we assume e.g. \(F_{\min} = 10^{-3}\) and \(F_{\max} = 10^{-1}\) then due to fig.2 we must execute \(n \propto 10^5\) trials: this is unavoidable. But we must also sort this number of measured \(x\)-values in order to obtain the ordered vector \((x_r)\) eq.(1): this would cause a prohibitive high consumption of computer time for a sorting effort which yields an unneeded extremely low rel. error \(d_F(x)\) in a large part of the empirical distribution function.

2) In this and other formulae eq.\(\ldots\) is associated to the d.f. \(F_n(x)\). The corresponding eq.\(\ldots\) is associated to the c.d.f. \(G_n(x)\) is given by replacing as far as applicable:

\[
F_{\min} + G_{\min} \cdot d_{\max} \geq F_n(x) + G_n(x) ; \quad d_F(x) + d_G(x) ; \quad x + v.
\]

Fig.1 \(E_k\)-distribution; objective empirical c.d.f. \(G_n(x)\) eq.(2b) and rel. error \(d_G(x)\) eq.(4b); ("straightforward sorting")
3. AN ALGORITHM FOR CONSTANT RELATIVE ERROR

3.1 The desired algorithm for handling the evaluation of simulated data should yield a balanced rel. error curve \( d_F(x) \) within a given region of \( F_n(x) \) and should also allow an essential reduction of computer time for sorting. For this purpose we combine eq. (2a,b) with eq. (4a,b) eliminating \( n \) and express the rel. error in the form

\[
\frac{d_F(x)}{d_{F_{\text{max}}}} = \left( 1 - F_n(x) \right) \left( 1 + \frac{2}{d_{F_{\text{max}}}} \cdot F_n(x) \right)^{1/2}, \quad (6a,b) \tag{2}
\]

\[
d_{F_{\text{max}}}^2 = \frac{r+1}{r}
\]

see fig. 3. From this equation we conclude that in order to stay below a prespecified maximum rel. error \( d_{F_{\text{max}}} \) resp. \( d_{G_{\text{max}}} \) we may at any point \( x \) of the simulated curve \( F_n(x) \) resp. \( G_n(x) \) limit the number \( r \) resp. \( v \) of measured \( x \)-values in the range \( x \leq x \) resp. \( x > x \) according to the formula

\[
r = \text{ENTIER}(d_{F_{\text{max}}}^2) - 1 \quad (7a,b)
\]

\[
v = \text{ENTIER}(d_{G_{\text{max}}}^2) - 1 \quad \text{not applicable}
\]

We also conclude from eq. (6a,b) that if we apply an algorithm keeping \( r \) resp. \( v \) constant then as shown in fig. 3 we have in the range \( F_n(x) < 10^{-1} \) resp. \( G_n(x) < 10^{-1} \) (which is of predominant interest in simulation) a practically constant rel. error \( d_F(x) \approx d_{F_{\text{max}}} \) resp. \( d_G(x) \approx d_{G_{\text{max}}} \). This then is in effect the desired balanced error curve.

3.2 As shown in fig. 4 the evaluation program for performing the constant error-algorithm proposed here must provide a \( F \)-Sortingmemory \( F_S \) resp. \( G \)-Sortingmemory \( G_S \) and a \( F \)-Resultmemory \( F_R \) resp. \( G \)-Resultmemory \( G_R \). The \( i \)-th result \( x_i \) \( (i=1,2,...) \) stored in \( F_R \) resp. \( G_R \) is obtained after a number of trials \( n_i \) has been executed which is unknown in advance. This indicates the main feature of the balanced error-algorithm: instead of having \( n \) = const. and \( r \) resp. \( v \) a measured variable as in fig. 1 according to the straightforward use of eq. (2a,b) and eq. (4a,b) we have now \( r \) = const. resp. \( v \) = const. eq. (7a,b) whereas the number of trials \( n = n_i \) is a measured variable. We will describe the algorithm now step by step for the general case that \( F_n(x) \) and/or \( G_n(x) \) have to be determined.

3.2.1 Initial phase

a) Given the prespecified maximum rel. error \( d_{F_{\text{max}}} \) resp. \( d_{G_{\text{max}}} \) compute the integers \( r \) resp. \( v \) eq. (7a,b).

b) Execute \( n_1 = r+1 \) resp. \( n_1 = v+1 \) trials, sort the produced random \( x \)-values into \( F_S \) resp. \( G_S \) yielding the initial ordered vector

\[
(x_{F1}, x_{F2}, ..., x_{F_{r+1}}), \quad x_{F_j} \leq x_{F_{j+1}} \quad j = 1,2, ..., r
\]

resp.

\[
(x_{G1}, x_{G2}, ..., x_{G_{v+1}}), \quad x_{G_j} \leq x_{G_{j+1}} \quad j = 1,2, ..., v
\]
c) Store the first result \((i=1)\) \(x_1 = x_{Fr+1}\) and \(F_n(x_1)\) eq. (2a) in \(FR\) resp. \(x_1 = x_{Gl}\) and \(G_n(x_1)\) eq. (2b) in \(GR\).

\[ (8a, b) \]

3.2.3 Output phase

a) The contents of \(FR\) resp. \(GR\) determine the plot of the empirical d.f. resp. c.d.f.

\[ P_n(x) = F_n(x), \quad x = x_{i-1} \]

\[ G_n(x) = G_n(x), \quad x = x_{i-1} \]

\[ i = 1, 2, \ldots, f; \quad x_0 = x_{II} \]

\[ i = 1, 2, \ldots, g; \quad x_0 = x_I \]

where the nonmeasured points \(x_0\) are assumed to be unknown, see section 2.1.

Remark. With the const. rel. error algorithm the objective statements on basis of eq. (2a,b) are restricted to the points \(x_i\) stored in \(FR\) resp. \(GR\), because only for these points the integer \(r\) resp. \(v\) and the number of trials \(n_i\) are known quantities. Between two consecutive points \(x_i, x_{i+1}\) we are principally free to interpolate in accordance with the fundamental character of a

![Fig. 4](image-url)

**Fig. 4** Structure of memories to be implemented in the evaluation program for the constant error-algorithm, see section 3.2.

3.2 Recursive phase

a) After the \(i\)-th result \(x_i\), at a total number of \(n_i\) trials \((i=1, 2, \ldots)\), has been obtained execute as many further trials \(n_{i+1}, n_{i+2}, \ldots\) until, at a certain number of trials \(n_{i+1} > n_i\), the generated \(x\)-value fulfills for the first time the relation \(x \leq x_{Fr+1}\) resp. \(x \leq x_{Gl}\).

b) Sort \(x\) into \(FS\) resp. \(GS\) thereby eliminating the previous largest value \(x_{Fr+1}\) resp. smallest value \(x_{Gl}\).

c) Store the \((i+1)\)-st result \(x_{i+1} = x_{Fr+1}\) and \(F_n(x_{i+1})\) eq. (2a) in \(FR\) resp. \(x_{i+1} = x_{Gl}\) and \(G_n(x_{i+1})\) eq. (2b) in \(GR\).

d) Assuming the lowest value \(F_{min}\) resp. \(G_{min}\) according to 2.3a) to be known check the relation \(F_n(x_{i+1}) < F_{min}\) resp. \(G_n(x_{i+1}) < G_{min}\):

- if this relation is false set \(i := i+1\) and go back to 3.2.2a);

- if it is true stop simulation and continue with 3.2.3a). The final number of measured values stored in \(FR\) resp. \(GR\) will be \(f\) resp. \(g\);

the final number of trials \(n_f\) resp. \(n_g\) is in practice slightly above the least value computed by eq. (5a,b).

![Fig. 5](image-url)

**Fig. 5** \(E_0\)-distribution: empirical c.d.f. \(G_n(x)\) eq. (6b) and associated rel. error \(d_G(x)\) eq. (6b).
distribution function. But taking into account the possibility that the a priori unknown d.f. \( F(x) = 1 - G(x) \) may contain steps it is advisable to plot at first the above conservative empirical step functions eq.(8a,b) and to inspect these functions carefully before any extra- and interpolation takes place, see section 4 in /6/.

b) By introducing eq.(8a,b) in eq.(6a,b) the rel. error \( d_F(x) \) resp. \( d_G(x) \) can be plotted for control purposes. In practical applications this is not necessary because the error is guaranteed to be below the predescribed maximum value \( d_{F\text{max}} \) resp. \( d_{G\text{max}} \), see section 3.1.

c) In order to support the extrapolation of the tails the information of the final contents \( (x_{Fj}) = (x_{F1}, x_{F2}, \ldots, x_{Fv}) \) in FR resp. \( (x_{Gj}) = (x_{G1}, x_{G2}, \ldots, x_{Gv}) \) in GR can be used to plot the left tail of \( F_n(x) \) resp. right tail of \( G_n(x) \) whereby of course the rel. error \( d_F(x) \) resp. \( d_G(x) \) will exceed the limit \( d_{F\text{max}} \) resp. \( d_{G\text{max}} \). To obtain these tails eq.(2a,b) and eq.(4a,b) can be applied in a straightforward manner, the number of trials being constant now: \( n = n_F \) resp. \( n = n_G \).

### 4. MEASUREMENT RESULTS WITH THE CRE-ALGORITHM

The CRE-algorithm is demonstrated by three examples. In all cases the same parameter values were chosen: \( d_{F\text{max}} = 10^{-1} \); \( F_{\text{min}} = G_{\text{min}} = 10^{-3} \), see section 2.3. Due to eq.(5a,b) this leads inevitably to a final number of trials in the order of magnitude \( n_F \) resp. \( n_G \approx 10^5 \). In accordance with eq.(6a,b) we note in all cases that in the range \( F_n(x) \) resp. \( G_n(x) \approx 10^{-1} \) we have practically a constant rel. error \( d_F(x) < d_{F\text{max}} \).

In order to judge the quality of the approximations the ideal function \( F(x) \) resp. \( G(x) \) has been added to each diagram; in normal practice of course this comparison cannot be made because \( F(x) = 1 - G(x) \) is unknown.

4.1 In fig.5 the CRE-algorithm has been applied to the \( E_g \)-distribution. Compared to the diagram fig.1 obtained by the straightforward application of eq.(2b) we note in fig.5 that the logarithmic step sizes of \( G_n(x) \) do not increase toward the tail of the curve. This is a typical feature of all CRE-diagrams with logarithmic ordinate.

4.2 The measurement fig.6 shows an example where the CRE-algorithm has traced a generalized c.d.f. \( G_n(x) \) with continuous sections and steps, compare the straightforward evaluation of the same random process in /6/.

4.3 The measurement fig.7 is of special interest because it demonstrates that the CRE-algorithm is well suited to detect rare event details of a random process: in this example a mixed \( E_\text{g-} E_{\text{g5}} \)-distribution with a two peak p.d.f. \( f(x) \) where the first peak is very weak compared to the second, see fig.9. Indeed once the resolution level \( F_{\text{min}} \) resp. \( G_{\text{min}} \) (and the maximum rel. error) have been set then the CRE-algorithm will trace automatically all details within this level. Theoretically rare event details could also be detected by the straightforward application of eq.(2a,b) but this would fail in practice because of the excessive expense for sorting, see section 2.2 and fig.8.

In fig.7 the rare event detail is displayed within the left tail of the d.f. \( F_n(x) \) and cannot be detected within the c.d.f. \( G_n(x) \). From this we conclude that the rare event analysis of both tails of a truly unknown distribution function requires the application of the CRE-algorithm for both empirical functions: the d.f. \( F_n(x) \) and the c.d.f. \( G_n(x) \). Using all memory devices fig.4 simultaneously this can be achieved in a single evaluation run.

We also conclude that the rel. error \( d_F(x) \) resp. \( d_G(x) \) eq.(4a,b) and not the abs. error \( \sigma(x) \) eq.(3) is relevant for the control of the evaluation.

---

4) We introduce here the abbreviation:

CRE = Constant Relative Error
5. REDUCED EXPENSE FOR SORTING

The total computer time $T_O$ for performing a certain simulation resp. statistical evaluation program will contain a certain amount $T_{\text{sort}}$ for sorting the measured random $x$-values. If the speed of a computer is roughly characterized by its average number of operations per time unit ("operation intensity") $\mu_{\text{op}}{[\text{MOp/s}]}$, then the normalized sorting time

$$\frac{T_{\text{sort}}}{\mu_{\text{op}}{[\text{MOp}]}}$$

is the average number of operations for performing a specified sorting job and represents a sorting expense measure which is to some extent independent of a special computer.

Based on measurement of $T_{\text{sort}}$ the quantity $\mu_{\text{op}}{[\text{MOp}]}T_{\text{sort}}$ has been depicted in fig. 8 as a function of $d_{F_{\text{max}}}$ resp. $d_{G_{\text{max}}}$.

a) for straightforward sorting of all measured $x$-values as in fig. 1;

b) for selected sorting according to the CRE-algorithm as in fig. 5 to fig. 7.

Using eq. (8a, b) it is found in case a) that $T_{\text{sort}} \sim n^2$ with good approximation. By application of more effective sorting techniques /8/ we could achieve at best $T_{\text{sort}} \sim n \cdot \log_2(n)$ but even this affords a high expense of computer time for e.g. $n = 10^5$ or $n = 10^6$ trials.

The CRE-algorithm on the other hand keeps $T_{\text{sort}}$ comparatively low. It can be shown in case b) that a simple sorting technique with right-side input into memory FS resp. left-side input into GS in fig. 4 is practically optimal because the contents of FS resp. GS represent sorted $x$-values of the left resp. right tail of the distribution function in each interim state $i = 1, 2, \ldots, f$ resp. $g$.

6. Final Remarks

6.1 The CRE-algorithm based on the considerations /4/, /5/ and /6/ is not confined to the evaluation of simulated random data but represents a general method in statistics to obtain an
a priori unknown distribution function from measured data under control of an objective error measure. For example the large volume data obtained by traffic measurements in telephone or data networks /9/, /10/ could be evaluated favorably by means of the CRE-algorithm, provided that the independence requirement is fulfilled, see section 6.3.

6.2 The CRE-algorithm can be developed further to perform a window-mechanism: given the parameters due to section 2.3a,b) we might be interested to observe the details of a certain segment $F_1$ to $F_2$ ($F_{\min} < F_1 < F_2 < 1$) of the curve $F_n(x)$ with higher accuracy, i.e. with a reduced max.rel. error $\delta_{F_{12}} < \delta_{F_{\max}}$. This can be achieved if the evaluation program opens the window at $F_1$ by switching the integer $r$ to the value

$$r = \text{ENTIER}(d^{-2}_{p12}) - 1$$

and closes it at $F_2$ by switching $r$ back to eq.(7a). This concept can be generalized to allow for several windows along $F_n(x)$ resp. $G_n(x)$.

6.3 Simulation of queueing networks usually leads to correlated i.e. dependent random values which can be transformed to approximately independent values by some "decorrelation"-method e.g. by the linear autoregressive approach /2/; thereafter the CRE-algorithm could be applied in order to gain the proper empirical distribution function. The combination of decorrelation and CRE-algorithm is a matter of further studies.

APPENDIX

A random process with two mixed $E_k$-distributions is described by the p.d.f.

$$f(x) = F_1^{-a_1} \cdot \frac{(x/a_1)^{k_1-1}}{(k_1-1)!} \cdot \exp(-x/a_1)$$

$$+ (1-F_1)^{-a_2} \cdot \frac{(x/a_2)^{k_2-1}}{(k_2-1)!} \cdot \exp(-x/a_2)$$

In order to gain a process with a rare event detail for the measurement fig.7 we have used the parameter set of the density $f(x)$ fig.9 where the left peak is much weaker than the right peak (logarithmic scale!). In a previous paper the same parameter set has been used except $F_1 = 0.3$ yielding a density $f(x)$ with two peaks of comparable "strength", see fig.9 in /5/. For details of the software random generators see the appendices in /5/ and /6/.

REFERENCES


/10/ Evers, R., Analysis of traffic flows on subscriber-lines dependent of time and subscriber-class. Proc. 8th ITC 1976 (Melbourne), 345/1-8.
Q.1 (L. Kosten)

In today's practice simulation is mostly applied to very complex systems, where RNG's are used at various places. Under those circumstances poor system analysts overlooking necessary correlations are a far greater danger than poor RNG's showing unwanted autocorrelation. What is the author's opinion?

A.1 (F. Schreiber)

In general we agree with your opinion, but we are not so much concerned with complex simulation problems that we could give a detailed answer to this question.

Therefore I would suggest to put this question to a simulation specialist.

Of course we don't know the special simulation problem in question but in principle it is possible to produce random number generators which could fulfill the wanted properties if special conditions are taken into account.
CORRECTION NOTE

Session: 2.4
Paper: 4

In this paper the term:
"CRE-algorithm" (CRE Constant Relative Error)

should be replaced by the term:
"LRE-algorithm" (LRE Limited Relative Error),
because this characterizes better the meaning of the algorithm.
In a forth coming paper describing an extended application of the method only the latter abbreviation will be used.
ITC 10

Summary of Questions/Answers

Date: 10 June 1983
Session: 2.4
Paper: 4

Q.1 (Phuoc Tran-Gia)

Is the presented measurement algorithm applicable to the simulation of nonstationary random processes?

A.1 (Prof. F. Schreiber)

No, it is not applicable, unless the unstationarity has a time constant which allows it to gather sufficient data of an approximately stationary character.