Efficient control chart calibration by simulated stochastic approximation

Giovanna Capizzi and Guido Masarotto
Department of Statistical Sciences, University of Padua, Italy

ABSTRACT
The accurate determination of control limits is crucial in statistical process control. The usual approach consists in computing the limits so that the in-control run-length distribution has some desired properties; for example, a prescribed mean. However, as a consequence of the increasing complexity of process data, the run-length of many control charts discussed in the recent literature can be studied only through simulation. Furthermore, in some scenarios, such as profile and autocorrelated data monitoring, the limits cannot be tabulated in advance, and when different charts are combined, the control limits depend on a multidimensional vector of parameters. In this article, we propose the use of stochastic approximation methods for control chart calibration and discuss enhancements for their implementation (e.g., the initialization of the algorithm, an adaptive choice of the gain, a suitable stopping rule for the iterative process, and the advantages of using multicores workstations). Examples are used to show that simulated stochastic approximation provides a reliable and fully automatic approach for computing the control limits in complex applications. An R package implementing the algorithm is available in the supplemental materials.

1. Introduction
Stochastic Approximation (SA) is one of the most investigated recursive algorithms for handling many different applications, such as online estimation, regression, and adaptive control problems. In particular, SA as a recursive method for identifying the zero of a function was introduced in the seminal paper by Robbins and Monro (1951), with several subsequent theoretical advances. Today, almost six decades of research on the theoretical and applied developments in SA can be found in survey articles (Ruppert, 1991; Bharath and Borkar, 1999; Lai, 2003) and in several books (Chen, 2002; Kushner and Yin, 2003; Spall, 2003).

Although the literature regarding SA is substantial, the attention on SA algorithms for solving nonlinear root-finding problems has increased in recent years due to the growing number of complex nonlinear models requiring stochastic search and optimization and the enormous advances in simulation methodology and software (see Pasupathy and Kim (2011) for a recent overview of the stochastic root-finding problem).

In our opinion, root-finding SA can find appealing applications in the Statistical Process Control (SPC) framework. As is well known, control charts are statistical tools for testing process stability over time. Their efficiency is usually assessed in terms of run-length, which is the number of process observations necessary to signal a change in one or more process parameters. The parameters of control charts are determined to guarantee a desired run-length performance when the process is stable—that is, In-Control (IC)—and the quickest detection of an Out-of-Control (OC) situation when at least one process parameter has changed. In particular, the control limits define the decision interval to declare an OC situation are determined to attain a specified value of some IC run-length characteristics, such as the IC Average Run-Length (ARL), the IC median, the probability of false alarms, etc.

Performance measures, such as the in-control ARL or the probability of false alarms, obviously depend on the probability distribution of the underlying quality characteristic and, above all, on the implemented control chart. In some standard applications, run-length characteristics can be calculated either analytically or by using appropriate numerical algorithms; this is the case of well-known univariate parametric control charts, such as Shewhart, Exponentially Weighted Moving Average (EWMA), or Cumulative Sum (CUSUM) (Montgomery, 2009), and their derivatives (see, for a recent example and further references, Huang et al. (2014)).

However, due to the more complex reality in the manufacturing and service industries, several quality characteristics are gathered at very high rates. The presence of autocorrelation, the need to simultaneously monitor several process parameters, and/or quality characteristics of a process often lead to the implementation of control charts whose run-length characteristics are too complicated to be computed either analytically or using standard deterministic numerical methods. In these practical situations, Monte Carlo simulation must be used to evaluate the run-length distribution, and we believe that SA algorithms may provide an appealing and simple-to-implement solution for the computation the control limits.

The idea of using SA algorithms to find the control limit has been previously discussed by Yashchin (1993). According to Yashchin (1993), the Robbins–Monro (RM) recursive algorithm can lead to unsatisfactory efficiency and high computation efforts when applied to nonstandard applications, such as
in the autocorrelated case considered in Yashchin (1993). However, as shown in some specific cases by Capizzi and Masarotto (2008, 2009), these concerns can be overcome using advances in simulation methodology and methods accelerating the convergence rate of the SA's itertes.

In this article, starting from these earlier works, we describe and study a general SA-based two-stage algorithm for computing control limits. The algorithm combines the averaging strategy introduced by Ruppert (1991) and Polyak and Juditsky (1992) with a partially adaptive choice of the SA gain. Its main characteristics can be summarized as follows:

1. It is completely automatic and easy to implement. Essentially, users only need to program the run-length simulator and choose the desired accuracy.
2. It can handle the computation of the control limits of single and multi-chart control schemes. Observe that the multi-chart capability reduces the need to select additional design parameters to optimize the OC run-length performance. Indeed, it is possible to choose a set of their values to be effective for a range of shift sizes.
3. It is very robust; in particular, unlike many deterministic nonlinear procedures, our algorithm is insensitive to the choice of the initial values and the scale of the control limits.
4. The number of iterations required to obtain a desired accuracy is determined using a suitable stopping rule.
5. A parallel version is available that can take advantage of the computing power offered by modern multicore workstations.

Our research experience shows that this algorithm can guarantee very accurate estimates of the control limits within a reasonable number of iterations. Furthermore, the computational effort (timing, number of simulations, etc.) is lower than that needed by other, more traditional procedures, such as those using a grid of values of the control limits or those based on a generic numerical solver.

The article is organized as follows. In Section 2, we briefly review main results pertaining to the stochastic root-finding problem and illustrate the proposed SA-based approach to the computation of the control limits. In Section 3, the new algorithm is applied to compute the control limits of three control charts: the Adaptive Generalized Likelihood Ratio (AGLR) chart, suggested by Capizzi and Masarotto (2012) for detecting unknown patterned mean shifts; the combination of a $T^2$ chart and multivariate EWMA ($T^2$-MEWMA) for detecting changes in the process mean vector of a multivariate process (Reynolds and Stoumbos, 2010); and the Nonparametric EWMA (NEWMA) introduced by Zou et al. (2008) for monitoring generic profiles. In Section 4, the proposed SA algorithm is compared with the traditional approach suggested in the literature for computing the control limits of a monitoring scheme when the run-length distribution can only be approximated by simulation. In Section 5, we highlight the results related to the parallel implementation of the proposed SA algorithm on a multicore workstation. Concluding remarks are provided in Section 6. An R package implementing the algorithm is available in the supplemental materials.

2. Control limits computation by SA

2.1. The problem

2.1.1. A single control chart

Suppose that at any instant of time $t = 1, 2, \ldots$, (S1) A realization of a $k$-dimensional random vector of quality variables, $\mathbf{x}_t$, is observed.

(S2) A control statistic $Q_t = Q_t(\mathbf{x}_1, \ldots, \mathbf{x}_t)$ is calculated, compared with a control limit $h > 0$ and an OC alarm is triggered if $Q_t > h$.

The run-length of the control chart is given by

$$RL = \inf\{t > 0 : Q_t > h\}.$$  

The control limit $h$ is usually determined as the solution of the equation:

$$E(RL) = ARLO,$$  \hspace{1cm} (1)

where $ARLO$ is an acceptable level of the in-control ARL and the expected value $E(\cdot)$ is computed assuming an IC process.

Observe that, notwithstanding that (S1) and (S2) are based on a constant control limit $h$, the notation is sufficiently general to also accommodate the presence of time-varying limits. For example, consider the EWMA control chart with time-varying limits and fast initial response proposed by Steiner (1999) for monitoring the mean of a univariate process (see also Knoth (2005)). This control chart consists of plotting the statistic

$$\xi_t = (1 - \lambda)\xi_{t-1} + \lambda x_t \quad (\xi_0 = \mu_0),$$

together with the time-varying control limits:

$$LCL_t = \mu_0 - h\sigma_0C_t \quad \text{and} \quad UCL_t = \mu_0 + h\sigma_0C_t,$$

where

$$C_t = \left[1 - \left(\frac{1}{2}\right)^{\frac{t-1}{2}}\right]^{\frac{1}{2}} \sqrt{\frac{\lambda[1 - (1 - \lambda)^{2t}]}{2 - \lambda}},$$

$\mu_0$ and $\sigma_0$ denote the IC mean and standard deviation, respectively, and $\lambda$ is a suitable smoothing constant. This scheme is a particular case of (S1) and (S2) that triggers an alarm when $Q_t = \|\xi_t - \mu_0\|/\sigma_0C_t$ is greater than $h$. In general, the algorithm discussed in the present article can be used for determining the value of $h$ satisfying Equation (1) for control charts that trigger an alarm when $Q_t > L_t(h)$, where $L_t(\cdot)$ are known, monotonically increasing functions of a scalar parameter $h$.

2.1.2. Multi-chart monitoring schemes

Nowadays, several control charts are sometimes combined to monitor more than one process parameter, such as the process mean and variance (see Gan (1995), for example) or to improve the detection power in the presence of small and large mean shifts; examples are (i) the Shewhart-EWMA control charts, discussed for the univariate and multivariate framework in Lucas and Saccucci (1990) and Reynolds and Stoumbos (2010), respectively; and (ii) the CUSUM and EWMA multi-charts studied by Han et al. (2007) and Han and Tsung (2007).

In the multi-chart framework, let $RL_i$ be the run-length of the $i$th control scheme, with $i = 1, \ldots, p$; i.e., $RL_i = \inf\{t > 0 : \}$.
Q_{i,t} > h_i$, where $Q_{i,t}$ and $h_i$ are the control statistic and the control limit of the $i$th control scheme. The combined control chart signals an OC alarm as soon as one of the $p$ control statistics $Q_{i,t}$ is larger than the corresponding control limit $h_i$. Thus, the RL of the combined control chart is $RL = \min(RL_1, \ldots, RL_p)$. Assuming that the $p$ control schemes are equally important, one possibility consists of determining $h = (h_1, \ldots, h_p)^T$, the $p$-dimensional vector of control limits, so that the combined control scheme achieves a desirable value of the in-control ARL, say $ARL_0$, and the single control schemes give the same in-control ARL. Thus, control limits can be determined so that

$$E[\min(RL_1, \ldots, RL_p)] = ARL_0,$$

and

$$E(RL_1) = \cdots = E(RL_p).$$

As (i) ARL-based criteria are the most commonly used; and (ii) the single chart criterion (1) is a special case of Equations (2) and (3) when $p = 1$, we will focus on Equations (2) and (3). However, observe that, as exemplified in subsection 2.3.1, the SA algorithm can be modified to handle different criteria, such as those giving different weights to different schemes or those based on the run-length quantiles.

### 2.2. Nonlinear root finding via simulated SA

Let $s_h = s(RL_1, \ldots, RL_p) \in \mathbb{R}^p$ be a suitable score such that the system of equations (2) and (3) is equivalent to

$$g(h) = E(s_h) = 0.$$  

A particular choice of $s_h$ will be presented in subsection 2.3.1.

The goal of simulated SA algorithms is to find one root, say $h^*$, of systems of equations such as Equation (4) when the expected value $g(h)$ can only be approximated using Monte Carlo techniques. The solution to this root-finding problem, introduced by Robbins and Monro (1951) and generalized to the multidimensional case by Blum (1954), is based on the recursive iteration

$$h_{r+1} = h_r - \frac{1}{r+1}A_s h_r, \quad r = 0, 1, \ldots,$$

where $s_h$, for $r = 0, 1, \ldots$, is obtained by simulating the run-lengths from an IC process, and $A$, the gain of the scheme, is a non-singular matrix.

The estimate $h_r$ of $h$ is recursively adapted to meet some goals asymptotically. A rich convergence theory has been developed to establish whether the iterate (5) converges to a solution $h^*$ as $r$ increases. The early contributions provided conditions for almost sure convergence and convergence in probability. In particular, the iterate $h_r$ is shown to converge to a solution $h^*$ as $r \rightarrow \infty$ when $Ag(h)$ points toward $h^*$ (see Blum (1954) and Ruppert (1985) for more details). Although it is relatively simple, at least in some practical cases, to find a gain matrix that ensures the asymptotic convergence of the RM recursive iteration, convergence in itself does not necessarily imply efficiency; that is, a high speed at which the iterate (5) approaches the root $h^*$. An efficient SA algorithm is guaranteed for a gain matrix $A$ equal to the inverse Jacobian matrix computed at $h = h^*$; that is, for

$$A = \left( \frac{\partial g(h)}{\partial h} \right)^{-1} \bigg|_{h = h^*}.$$

Unfortunately, one does not know the Jacobian matrix. Thus, it is also relevant to establish how to choose practically the gain to attain a good finite-sample performance without degrading the SA’s asymptotic performance.

Two approaches have been proposed in the literature.

1. The first consists of adaptively estimating the optimal Jacobian matrix (Ruppert, 1985; Wei, 1987; Spall, 2000).
2. The second, known as iterate averaging and independently proposed by Ruppert (1991) and Polyak and Juditsky (1992), consists in using as an estimate of $h^*$, at the $N$th recursive stage, the sample average

$$\bar{h}_N = \frac{1}{N} \sum_{r=1}^{N} h_r.$$  

Each of the addends in Equation (6) is computed using the following slightly modified recursion

$$h_{r+1} = h_r - \frac{1}{r+1}A_s h_r.$$  

It is possible to show that, if $0.5 < q < 1$, the Polyak–Ruppert (PR) algorithm (6)–(7) obtains the optimal convergence rate without either the knowledge or an estimate of the Jacobian matrix. Thus, the PR approach provides an efficient estimate of $h^*$ for any gain matrix $A$ satisfying the necessary general conditions for convergence.

Furthermore, a constrained version of the PR algorithm can also be implemented. In particular, assuming that $h^* \in H \subset \mathbb{R}^p$, iteration (7) is substituted by

$$h_{r+1} = \Psi \left( h_r - \frac{1}{r+1}A_s h_r \right).$$

where $\Psi(x)$ maps any vector $x \in \mathbb{R}^p$ to its nearest point inside $H$.

As the PR algorithm is easier to use and implement, we will discuss its application in the SPC framework. The suggested implementation uses a gain matrix estimated during the initialization stage. Hence, the resulting algorithm combines the two previously mentioned approaches.

### 2.3. A practical SA algorithm

Here, we describe a two-stage SA algorithm for solving Equations (2) and (3) that can be used when the in-control ARL cannot be easily computed, but we are able to simulate pseudo-observations $x_1, x_2, \ldots$ (and, thus, the run-lengths $RL_1, \ldots, RL_p$) from an IC process. The first stage of the algorithm is based on a fixed-gain SA method, and it is used to (i) move the initial control limits, chosen by the user, to a neighborhood of the solution; and (ii) compute a suitable gain matrix $A$. The second stage uses the constrained PR iterations (6) and (8) to refine the estimates of the control limits until the desired precision is achieved.

The suggested algorithm depends on several constants that can be selected by the users. However, we found that some fixed
The choice of the score

Table 1. Suggested constant values for the implementation of the SA algorithm.

| Parameter | Value |
|-----------|-------|
| \( A_{\text{fixed}} \) | 0.1 |
| \( N_{\text{fixed}} \) | 500 |
| \( \delta \) | 0.1 |
| \( A_{\text{min}} \) | 0.1 |
| \( A_{\text{max}} \) | 100 |
| \( q \) | 0.55 |
| \( z \) | 3 |
| \( N_{\text{min}} \) | 1000 |
| \( C_{\text{maxK}} \) | 10 |

values of these constants (see Table 1) give satisfactory results in various practical cases. Hence, once the user specifies a prescribed value of the overall in-control ARL and the desired degree of accuracy, Table 1 provides guidelines for the practical implementation of the algorithm. Observe that users also need to select an initial value for the recursive process. However, as shown in Section 3, the algorithm's performance is quite robust with respect to this choice. Indeed, these findings motivate the introduction of the first stage in our algorithm.

In the following, subsections 2.3.1, 2.3.2, and 2.3.5 describe aspects of the algorithm that are common to both stages, and subsections 2.3.3 and 2.3.4 provide details on the first and second stages, respectively.

### 2.3.1. Choice of the SA score \( s_h \)

The choice of the score \( s_h \) is not unique and, indeed, we investigated different possibilities. On the basis of the very good numerical results that we obtained, we recommend the use of

\[
\begin{align*}
\tilde{s}_h &= (s_i) = \left( \frac{\min(RL_1, \ldots, RL_p) - ARL_0}{ARL_0} + \frac{RL_i - \overline{RL}}{ARL_0} \right)^T,
\end{align*}
\]

where \( \overline{RL} = (RL_1 + \ldots + RL_p) / p \) denotes the average of the individual run-lengths \( RL_1, \ldots, RL_p \).

In particular, note that when this score is used in an SA-type algorithm,

- the first addend of \( s_h \), which is identical for all \( i \), moves \( h_i \) toward a solution satisfying (2); and
- the second addend of \( s_h \), depending on \( i \), attempts to ensure the holding of condition (3).

Furthermore, note that both addends are relative differences with respect to \( ARL_0 \). We found that this standardization makes some details of the algorithm easier.

It should be noted that the application of the same SA algorithm to modified scores allows for easy handling of different criteria. For example, suppose that the \( p \) control charts are designed to detect faults with a different degree of importance to users. In these situations, it is reasonable to substitute Equation (3) with

\[
E(RL_i) = \frac{1}{w_i}E(\overline{RL}), \quad i = 1, \ldots, p,
\]

where \( w_1, \ldots, w_p \) are weights reflecting the relative importance of the different schemes. We can assume, without loss of generality, that the weights are standardized so that

\[
\sum_{i=1}^{p} \frac{1}{w_i} = p.
\]

Thus, the only modification in Equation (9) is the introduction of \( w_i \) in the second addend; that is,

\[
\tilde{s}_h = (s_i) = \left( \frac{\min(RL_1, \ldots, RL_p) - ARL_0}{ARL_0} + \frac{RL_i - \overline{RL}/w_i}{ARL_0} \right)^T.
\]

### 2.3.2. Constrained solution

Since control limits are usually greater than zero, we implement the constrained version of the SA algorithm, see Equation (8), based on the projection operator \( \Psi(x) \) mapping an arbitrary \( p \)-dimensional vector \( x = (x_i) \) to the vector with components max \((0, x_i)\).

#### 2.3.3. First stage

For iterate averaging to be successful, the starting point should not be too far from the solution \( h^* \) (see Spall (2000), chapter 4, and Kushner and Yin (2003), chapter 11). As it can be difficult to specify a good starting point in complex situations, in our implementation, the first set of \( N_{\text{fixed}} \) iterations is done using the fixed-gain SA recursion

\[
\tilde{h}_{r+1} = \Psi(\tilde{h}_r - A_{\text{fixed}}s_h), \quad r = 0, \ldots, N_{\text{fixed}} - 1,
\]

where \( \tilde{h}_0 \) is the initial value chosen by users, \( A_{\text{fixed}} > 0 \) is a scalar constant, and \( s_h \) is a score obtained simulating the run-lengths \( RL_1, \ldots, RL_p \) from an IC process.

During the first stage, we also compute a suitable gain matrix \( A \) for the second stage PR algorithm. Indeed, while the asymptotic efficiency of iterate averaging methods does not depend on the gain, its finite-sample performance does and, in particular, it can be affected by scale differences in the \( h \) elements.

The gain matrix is obtained as follows. Let \( \delta \) be a small number greater than zero and let \( \epsilon_i \in \mathbb{R}^p \) be the standard unit vector in the \( i \)th direction; i.e., all the elements of \( \epsilon_i \) are zero except for the \( i \)th element, which has a value of one. At step \( r \) of the iteration, we also simulate the scores \( s^+_{ij} = (s^{+}_{ij}) \) for the control limits \( h_i \pm \delta \epsilon_i, i = 1, \ldots, p \). Then, a \( p \times p \) diagonal gain matrix \( A = (a_{ij}) \) is computed with

\[
a_{ij} = \begin{cases} 
1 & \text{if } i = j \\
\max \left\{ \frac{1}{N_{\text{fixed}}}, \min \left( \frac{1}{N_{\text{fixed}}}, d_{ij} \right) \right\} & \text{if } i \neq j 
\end{cases}
\]

and

\[
d_{ij} = \frac{1}{2\delta N_{\text{fixed}}} \sum_{r=0}^{N_{\text{fixed}}} (s^+_{rji} - s^-_{rji}).
\]

We recommend simulating the \( 2p \) scores \( s^\pm_{ij}, i = 1, \ldots, p \), using the same IC pseudo-random sequence \( x_1, x_2, \ldots \), used to simulate \( s_h \). This yields two advantages: (i) a reduction in the number of computations since the observations and the control statistics \( Q_{ij} \) have to be computed only once; and (ii) a considerable reduction in the variance of differences \( s^+_{rji} - s^-_{rji} \) (see Kushner and Yin (2003), p. 16).

Table 1 lists the recommended values for \( A_{\text{fixed}}, N_{\text{fixed}}, \delta, A_{\text{min}}, \) and \( A_{\text{max}} \).

#### 2.3.4. Second stage and a stopping rule

The second stage is based on the PR algorithm. In particular, for \( r = 1, 2, \ldots \), a sequence of estimates, \( \tilde{h}_r \), of the control limits is
puting the score

The next step. To avoid this phenomenon, we suggest com-

3.1. Controlcharts

In the following, we consider three different control charts:

1. The AGLR control chart proposed by Capizzi and
   Masarotto (2012) for detecting an unknown arbitrary
   patterned shift in the mean of a univariate random vari-
   able. In particular, we discuss the computation of the
   single control limit $h$ for the parametric version of the
   AGLR control statistic. The other design parameters of
   the AGLR have been set at the default values given in
   Capizzi and Masarotto (2012).

2. The $T^2$-MEWMA control chart, a multi-chart based
   on a combination of the Hotelling’s $T^2$ chart (Mason
   and Young, 2002; Montgomery, 2005) and multivariate
   EWMA (Lowry et al., 1992), also discussed in Reynolds
   and Stoumbos (2010), for detecting changes in the pro-
   cess mean vector. In particular, we will show results for
   the computation of the two control limits $h_1$ ($T^2$) and
   $h_2$ (MEWMA) when there are five variables and when $\lambda$,
   the smoothing constant of the MEWMA, is equal to 0.1.

3. A multi-chart based on the combination of four
   NEWMA control charts. The NEWMA is a nonparamet-
   ric scheme, suggested by Zou et al. (2008), for moni-
   toring nonlinear profiles. This control chart is based on
   the application of two statistical tools: (i) a multivariate
   EWMA control chart for accumulating process observa-
   tions over time; and (ii) a standard nonparametric test,
   based on a linear smoother, for detecting an OC condi-
   tion. From a practical point of view, the design of this
   control chart implies the uneasy choice of (i) the smooth-
   ing constant, $\lambda$, of the MEWMA chart; and (ii) the effec-
   tive degrees of freedom, $df$, of the linear smoother (see
   also Qiu and Zou (2010)). On the other hand, a multi-
   chart based on the combination of different $\lambda$s and $dfs$
   provides satisfactory protection against a wide range of
   OC scenarios. Here, we discuss the estimate of the limits
   of the combination of the following four NEWMA con-
   trol schemes:

   $A$: $\lambda = 0.05$, $df = 4$; $B$: $\lambda = 0.2$, $df = 4$;
   $C$: $\lambda = 0.05$, $df = 8$; $D$: $\lambda = 0.2$, $df = 8$.

The independent variables of the nonlinear profiles are
those used by Zou et al. (2008) in their simulation study
(scenario II based on $k = 20$).

3.2. Simulation design and results

We used an in-control ARL equal to 200 and two possible values
for the accuracy: $\gamma = 0.025$ and 0.005. Then, the SA algorithm
was run to obtain 200 independent estimates of the control lim-
its. Each estimate $\hat{\text{h}}_i^{(l)}$, $l = 1, \ldots, 200$, was obtained by initial-
izing the procedure with control limit values randomly generated
from a uniform distribution between 1000 and 2000. Such a wide
range of initial values is considered to show the robustness of the
suggested method to the choices of the initial values that are very
far from the solution. Finally, to study the convergence of the SA
solutions to the desired control limits, the in-control ARL was
estimated as the average over 1000 000 simulated run-lengths
for each value $\hat{\text{h}}_i^{(l)}$.

Table 2 lists the main summary statistics of the estimated con-

2.3. Avoiding useless run-length simulations

When, as suggested here, a stochastic recursive algorithm is
used, it is not possible to guarantee that for some $r$, the
elements of $\mathbf{h}_r$ do not become too large, implying a very
high computational effort for simulating the run-lengths in
the next step. To avoid this phenomenon, we suggest com-
puting the score $s_h$, using the truncated run-lengths $RL_i^* = \min \{ RL_i, C_{maxRL} ARL_0 \sqrt{\gamma} \}$ in place of $RL_i$. Since the truncation
point diverges with $r$, this modification does not introduce any
bias. As can be seen in Table 1, the suggested value of $C_{maxRL}$ is
10.

3. Some examples

3.1. Control charts

In the following, we consider three different control charts:

1. The AGLR control chart proposed by Capizzi and
   Masarotto (2012) for detecting an unknown arbitrary
   patterned shift in the mean of a univariate random vari-
   able. In particular, we discuss the computation of the
   single control limit $h$ for the parametric version of the
   AGLR control statistic. The other design parameters of
   the AGLR have been set at the default values given in
   Capizzi and Masarotto (2012).

2. The $T^2$-MEWMA control chart, a multi-chart based
   on a combination of the Hotelling’s $T^2$ chart (Mason
   and Young, 2002; Montgomery, 2005) and multivariate
   EWMA (Lowry et al., 1992), also discussed in Reynolds
   and Stoumbos (2010), for detecting changes in the pro-
   cess mean vector. In particular, we will show results for
   the computation of the two control limits $h_1$ ($T^2$) and
   $h_2$ (MEWMA) when there are five variables and when $\lambda$,
   the smoothing constant of the MEWMA, is equal to 0.1.

3. A multi-chart based on the combination of four
   NEWMA control charts. The NEWMA is a nonparamet-
   ric scheme, suggested by Zou et al. (2008), for moni-
   toring nonlinear profiles. This control chart is based on
   the application of two statistical tools: (i) a multivariate
   EWMA control chart for accumulating process observa-
   tions over time; and (ii) a standard nonparametric test,
   based on a linear smoother, for detecting an OC condi-
   tion. From a practical point of view, the design of this
   control chart implies the uneasy choice of (i) the smooth-
   ing constant, $\lambda$, of the MEWMA chart; and (ii) the effec-
   tive degrees of freedom, $df$, of the linear smoother (see
   also Qiu and Zou (2010)). On the other hand, a multi-
   chart based on the combination of different $\lambda$s and $dfs$
   provides satisfactory protection against a wide range of
   OC scenarios. Here, we discuss the estimate of the limits
   of the combination of the following four NEWMA con-
   trol schemes:

   $A$: $\lambda = 0.05$, $df = 4$; $B$: $\lambda = 0.2$, $df = 4$;
   $C$: $\lambda = 0.05$, $df = 8$; $D$: $\lambda = 0.2$, $df = 8$.

The independent variables of the nonlinear profiles are
those used by Zou et al. (2008) in their simulation study
(scenario II based on $k = 20$).

2.3. Avoiding useless run-length simulations

When, as suggested here, a stochastic recursive algorithm is
used, it is not possible to guarantee that for some $r$, the
elements of $\mathbf{h}_r$ do not become too large, implying a very
high computational effort for simulating the run-lengths in
the next step. To avoid this phenomenon, we suggest com-
puting the score $s_h$, using the truncated run-lengths $RL_i^* = \min \{ RL_i, C_{maxRL} ARL_0 \sqrt{\gamma} \}$ in place of $RL_i$. Since the truncation
point diverges with $r$, this modification does not introduce any
bias. As can be seen in Table 1, the suggested value of $C_{maxRL}$ is
10.

3. Some examples

3.1. Control charts

In the following, we consider three different control charts:

1. The AGLR control chart proposed by Capizzi and
   Masarotto (2012) for detecting an unknown arbitrary
   patterned shift in the mean of a univariate random vari-
   able. In particular, we discuss the computation of the
   single control limit $h$ for the parametric version of the
   AGLR control statistic. The other design parameters of
   the AGLR have been set at the default values given in
   Capizzi and Masarotto (2012).

2. The $T^2$-MEWMA control chart, a multi-chart based
   on a combination of the Hotelling’s $T^2$ chart (Mason
   and Young, 2002; Montgomery, 2005) and multivariate
   EWMA (Lowry et al., 1992), also discussed in Reynolds
   and Stoumbos (2010), for detecting changes in the pro-
   cess mean vector. In particular, we will show results for
   the computation of the two control limits $h_1$ ($T^2$) and
   $h_2$ (MEWMA) when there are five variables and when $\lambda$,
   the smoothing constant of the MEWMA, is equal to 0.1.

3. A multi-chart based on the combination of four
   NEWMA control charts. The NEWMA is a nonparamet-
   ric scheme, suggested by Zou et al. (2008), for moni-
   toring nonlinear profiles. This control chart is based on
   the application of two statistical tools: (i) a multivariate
   EWMA control chart for accumulating process observa-
   tions over time; and (ii) a standard nonparametric test,
   based on a linear smoother, for detecting an OC condi-
   tion. From a practical point of view, the design of this
   control chart implies the uneasy choice of (i) the smooth-
   ing constant, $\lambda$, of the MEWMA chart; and (ii) the effec-
   tive degrees of freedom, $df$, of the linear smoother (see
   also Qiu and Zou (2010)). On the other hand, a multi-
   chart based on the combination of different $\lambda$s and $dfs$
   provides satisfactory protection against a wide range of
   OC scenarios. Here, we discuss the estimate of the limits
   of the combination of the following four NEWMA con-
   trol schemes:

   $A$: $\lambda = 0.05$, $df = 4$; $B$: $\lambda = 0.2$, $df = 4$;
   $C$: $\lambda = 0.05$, $df = 8$; $D$: $\lambda = 0.2$, $df = 8$.

The independent variables of the nonlinear profiles are
those used by Zou et al. (2008) in their simulation study
(scenario II based on $k = 20$).
The root of the approximating system of equations (14) is obtained using a generic numerical solver; for example, classic bisection search has been often used when \( p = 1 \).

It is possible to demonstrate (Kushner and Yin, 2003) that, when \( B \) is large, this approach produces estimates with the same accuracy offered by the SA algorithm stopped after \( B \) iterations. However, observe that computing \( \hat{g}(\mathbf{h}) \) for a single \( \mathbf{h} \) requires the same number of simulated run-lengths needed by SA to produce the final estimates. Hence, given a desired precision, the SA algorithm considerably reduces the computational burden.

We have conducted a simulation experiment to document the following points:

1. For comparing the SA approach with the bisection method in the \( p = 1 \) case, we computed 200 independent estimates of the control limit of the AGLR scheme using, for solving Equation (14), the standard approach with \( B = 50,000 \), and the bisection search using \([0, 10]\) as initial interval.

2. For comparing the SA approach with a numeric solver in the \( p > 1 \) case, we computed 200 independent estimates of the control limits of the \( T^2 \)-MEWMA chart (\( k = 5, \lambda = 0.1 \)) solving Equation (14) with the function replicates of the control limits exhibit a relatively small variability leading to tightly clustered values of the in-control ARLs, as shown from the closeness of the low and high quantiles to the target values. Furthermore, note that the differences between the \( ARL_0 \) and the ARLs based on the estimated control limits are, as expected, within the specified tolerance \( \gamma \). Of course, the computational effort increases with the desired accuracy, as shown by the summary statistics related to the stopping times \( N_{PR} \) that, as expected, are much higher for the smallest value of \( \gamma \).

### 4. Comparisons with the traditional approach

When the run-length distribution can only be studied by simulation, the standard approach suggested in the literature (see Qiu (2013) for some examples) involves obtaining an estimate of the control limits \( \mathbf{h} \), substituting the target system of equations \( E(s_0) = 0 \) with

\[
\hat{g}(\mathbf{h}) = \frac{1}{B} \sum_{i=1}^{B} s_{h_i} = 0, \tag{14}
\]

where \( B \) is a large positive integer and the scores \( s_{h_i}, i = 1, \ldots, B \), are simulated in the IC scenario using the control limits \( \mathbf{h} \). The

| \( \gamma \) | Mean | SD   | min  | \( Q_{10} \) | \( Q_{90} \) | Max   |
|---|---|---|---|---|---|---|
| 0.025 | \( h \) | 2.920 | 0.004 | 2.910 | 2.913 | 2.926 |
|   | ARL  | 200.220 | 1.602 | 196.614 | 197.770 | 202.972 |
|   | \( N_{PR} \) | 13,376,815 | 235,149 | 12,700,000 | 12,987,850 | 13,778,150 |
| 0.005 | \( h \) | 2.920 | 0.001 | 2.918 | 2.918 | 2.921 |
|   | ARL  | 200.002 | 0.368 | 198.942 | 199.373 | 200.555 |
|   | \( N_{PR} \) | 327,881,195 | 112,206 | 324,866,000 | 326,023,700 | 329,695,350 |

**Table 2.** Control limits of the AGLR, \( T^2 \)-MEWMA, and combined NEWMA control charts computed via stochastic approximation.
BBsolve, initialized using $h_1 = h_2 = 10$, of the R package BB (Varadhan and Gilbert, 2009). Here, again, we set $B = 50\, 000$. We also attempted to use the function nleqslv from the homonymous R package. However, in many cases, this function, based on a quasi-Newton algorithm, was unable to find the solution of Equation (14) due to the numerical singularities of the Jacobian matrix.

In both cases, we also computed the control limits using the SA algorithm stopped after $N_{PR} = 49\, 500$ iterations of the PR algorithm and initialized using $N_{fixed} = 500$ simulated run-lengths during the first stage. We started the algorithm setting $h = 10$ for the AGLR chart and $h_1 = h_2 = 10$ for the $T^2$-MEWMA scheme.

Panels (a1) to (a3) of Fig. 1 show the boxplots of the 200 estimates of the control limits obtained using the SA algorithm and the traditional approach based either on bisection or BBsolve. The results clearly indicate that the three approaches offer the same accuracy. However, as shown in panels (b1) and (b2) of Fig. 1, the SA-based algorithm is computationally more efficient. The following points should be noted.

1. Each SA estimate was obtained using 50 000 simulated run-lengths.
2. The bisection algorithm required from 12 to 14 evaluations of $\hat{g}(h)$; since each evaluation is based on 50 000 simulated run-lengths, the bisection estimates are based on 600 000–700 000 simulated run-lengths.
3. BBsolve evaluated the right-hand side of Equation (14) from 64 to 102 times; hence, it required from 3500 000 to 5000 000 simulated run-lengths.

Beyond reducing the simulation effort, the SA algorithm described in Section 2.3 offers two other advantages with respect to the standard approach.

1. Users directly specify the desired accuracy, and the algorithm adaptively determines the required number of simulations.
2. The algorithm is insensitive to completely incorrect starting point choices. This feature is particularly relevant in the multivariate setting. For example, in the case of the $T^2$-MEWMA control chart, we verified that the BBsolve almost always fails to converge if—as we did without problems for SA in the Monte Carlo study reported in Section 3—the algorithm is initialized using random drawn from a uniform distribution between 1000 and 2000.

5. A simple parallel implementation

As discussed in the previous two sections, the proposed SA algorithm shows an efficient finite-time performance. Indeed, the algorithm uses the smallest number of iterates to estimate the control limits with the prescribed degree of precision. However, when high accuracy is desired, the required number of simulations can be high. This drawback can be successfully overcome...
by using modern computer systems. Indeed, multiple-processor machines are often available (multicore workstations or dedicated clusters), and it would be worthwhile to use these modern resources to efficiently handle the computation of very precise estimates of the control limits.

A simple possibility, which we recommend when $M$ cores, or CPUs, are available, consists of

- running the SA algorithm in parallel on the $M$ CPUs, setting the precision equal to $\gamma \sqrt{M}$; and
- computing the mean of the $M$ different estimates of the control limits.

This approach, which provides the same final accuracy of the non-parallel version, is very simple to implement since it does not require any coordination between the task assigned to the different CPUs.

To show the results of the parallel implementation, we repeated the computation of the control limits of the $T^2$-MEWMA control chart. In particular, Table 3 is analogous to Table 2, but the estimates have been obtained by running the SA algorithm on four CPUs. The results clearly show that the parallel version reaches the prescribed level of precision. In addition, the $N_{PR}$ values in Table 3, which are the sums of the stopping times obtained by the four CPUs, are comparable with those presented in Table 2. In this case, using four parallel CPUs, the algorithm takes one-third of the time to estimate the control limits with the desired accuracy.

6. Conclusions

In this article, we discuss the advantages of using a stochastic approximation algorithm to find the control limits of control charts, whose run-lengths can only be obtained via simulation. In particular, we introduce an automatic, two-stage method based on an adaptive estimate of the gain matrix and the iterative averaging procedure. We also suggest a suitable rule for stopping the iterative search of the control limits when user-specified precision is achieved. As shown by the application to a single and two multi-chart schemes, the proposed procedure yields a satisfactory finite-case performance in terms of the minimum number of simulations within a prescribed tolerance. Furthermore, it outperforms the traditional approach, and we showed that when very high accuracy is requested by users, the related intensive computational effort can be successfully handled using a simple parallel implementation on a multicore workstation.

In practice, control chart design is more complicated than the determination of its control limits. During the first steps of the analysis, the available historical data and engineering knowledge are used to understand process variation and identify an IC reference model. Then, chosen a suitable control chart, the control limits and any other additional parameters have to be determined.

In our practical experience, it often more important to design schemes able to offer a good and balanced performance against various fault types and shift sizes than to pick up the control chart optimal for a particular OC situation. In these situations, the proposed algorithm seems to be useful for its robustness and automatic characteristics. It also has the following desirable characteristics:

1. It allows easy comparisons of the performance of different control charts under a common constraint on their IC behavior and in a variety of OC scenarios.
2. It handles, with no additional effort, the computation of the limits of multi-chart schemes based on the combination of different charts, each offering a good performance against a particular type of shifts.

Furthermore, given the algorithm efficiency and, in particular, since it requires the minimum number of IC simulated run-lengths for obtaining a given precision, we have also used the SA algorithm as a “inner-loop” for minimizing the OC average run-length or other OC performance measures under a constraint on the in-control ARL. In these cases, the “outer-loop” is responsible for the optimization with respect to the other chart parameters and the “inner-loop” for the computation of the control limits satisfying the IC constraints. It should be noted that these problems can also be tackled using SA-based optimization algorithms (Chen, 2002; Kushner and Yin, 2003; Spall, 2003) that are outside the scope of the present article but, in our opinion, deserve further research.

Acknowledgement

The authors thank the Department Editor and referees for their timely review and helpful comments that improved the article. This research was partially funded by the UNIPD CPDA128413/12 grant.

Notes on contributors

Giovanna Capizzi is an Associate Professor of Statistics at the Department of Statistical Sciences, University of Padua, Italy. She holds a Ph.D. in Statistics from Padua University. Her publications have appeared in referred
journals including Environmetrics, IIE Transactions, Journal of Quality Technology, Technometrics, Statistics and Computing. She is serving as an Associate Editor of Technometrics and she is on the Editorial Board of Journal of Quality Technology. Her current research is focused on change-point models, statistical process monitoring, and quality control.

Guido Masarotto is a Professor of Statistics at the Department of Statistical Sciences, University of Padua, Italy. His current research includes statistical process monitoring and statistical computing. His research has been published in various scientific journals including Biometrika, Biostatistics, Statistics and Computing, Annals of Applied Statistics, IIE Transactions, Journal of Quality Technology, and Technometrics.

References

Bharath, B. and Borkar, V.S. (1999) Stochastic approximation algorithms: Overview and recent trends. Sudhana-Academy, Proceedings in Engineering Sciences, 24, 425–452.
Blum, J.R. (1954) Multidimensional stochastic approximation methods. Annals of Mathematical Statistics, 25, 737–744.
Capizzi, G. and Masarotto, G. (2008) Practical design of generalized likelihood ratio control charts for autocorrelated data. Technometrics, 50, 357–370.
Capizzi, G. and Masarotto, G. (2009) Bootstrap-based design of residual control charts. IIE Transactions, 41, 275–286.
Capizzi, G. and Masarotto, G. (2012) Adaptive generalized likelihood ratio control charts for detecting unknown patterned mean shifts. Journal of Quality Technology, 44, 281–303.
Chen, H. (2002) Stochastic Approximation and Its Applications, Kluwer Academic Publishers, New York, NY.
Gan, F. F. (1995) Joint monitoring of process mean and variance using exponentially weighted moving average control charts. Technometrics, 37, 446–453.
Han, D. and Tsung, F. (2007) Detection and diagnosis of unknown abrupt changes using CUSUM multi-chart schemes. Sequential Analysis, 26, 225–249.
Han, D., Tsung, F., Hu, X., and Wang, K. (2007) CUSUM and EWMA multicharts for detecting a range of mean shifts. Statistica Sinica, 17, 1139–1164.
Huang, W., Shu, L., and Su, Y. (2014) An accurate evaluation of adaptive exponentially weighted moving average schemes. IIE Transactions, 46, 457–469.
Knoth, S. (2005) Fast initial response features for EWMA control charts. Statistical Papers, 46, 47–64.
Kushner, H.J. and Yin, G.G. (2003) Stochastic Approximation and Recursive Algorithms and Applications, Springer, New York, NY.
Lai, T. L. (2003) Stochastic approximation (invited paper). Annals of Statistics, 31, 391–406.
Lowry, C.A., Woodall, WH., Champ, C.W., and Rigdon, S.E. (1992) A multivariate exponentially weighted moving average control chart. Technometrics, 34, 46–53.
Lucas, J.M. and Saccucci, M.S. (1990) Exponentially weighted moving average control schemes: properties and enhancements. Technometrics, 32, 1–29.
Mason, R.L. and Young, J.C. (2002) Multivariate Statistical Process Control with Industrial Applications, SIAM, Philadelphia, PA.
Montgomery, D.C. (2005) Introduction to Statistical Quality Control, fifth edition, Wiley, New York, NY.
Montgomery, D.C. (2009) Introduction to Statistical Quality Control, sixth edition, Wiley, New York, NY.
Paspapathy, R. and Kim, S. (2011) The stochastic root-finding problem: overview, solutions, and open questions. ACM Transactions on Modeling and Computer Simulations, 21, 1–23.
Polyak, B.T. and Juditsky, A.B. (1992) Acceleration of stochastic approximation by averaging. SIAM Journal of Control Optimization, 30, 838–855.
Qiu, P. (2013) Introduction to Statistical Process Control, Chapman & Hall, Boca Raton, FL.
Qiu, P. and Zou, C. (2010) Control chart for monitoring nonparametric profiles with arbitrary design. Statistica Sinica, 20, 1655–1682.
Reynolds, M.R. and Stoumbos, Z. (2010) Multivariate monitoring of the process mean and variability using combinations of Shewhart and MEWMA control charts, in Frontiers in Statistical Quality Control 9, Lenz, H., Wilrich, P.R., and Schmid, W. (eds), Physica-Verlag, Heidelberg, Germany, pp. 37–54.
Robbins, H. and Monro, S. (1951) A stochastic approximation method. Annals of Mathematical Statistics, 22, 400–407.
Ruppert, D. (1985) A Newton-Raphson version of the multivariate Robbins-Monro procedure. The Annals of Statistics, 13, 236–245.
Ruppert, D. (1991) Stochastic approximation, in Handbook of Sequential Analysis, Ghosh, B.K. and Sen, P.K. (eds), Marcel Dekker, New York, NY, pp. 503–529.
Spall, J.C. (2000) Adaptive stochastic approximation by the simultaneous perturbation method. IEEE Transactions on Automatic Control, 45, 1839–1853.
Spall, J.C. (2003) Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control, Wiley, Hoboken, NJ.
Steiner, S.H. (1999) EWMA chart with time varying control limits and fast initial response. Journal of Quality Technology, 31, 75–86.
Varadhan, R. and Gilbert, P. (2009) BB: An R package for solving a large system of nonlinear equations and for optimizing a high-dimensional nonlinear objective function. Journal of Statistical Software, 32, 1–26.
Wei, C.Z. (1987) Multivariate adaptive stochastic approximation. The Annals of Statistics, 15, 1115–1130.
Yashchin, E. (1993) Performance of CUSUM control schemes for serially correlated observations. Technometrics, 35, 35–52.
Zou, C., Tsung, F., and Wang, Z. (2008) Monitoring profiles based on nonparametric regression methods. Technometrics, 50, 512–526.