Efficient Control Chart Calibration by Simulated Stochastic Approximation

GIOVANNA CAPIZZI and GUIDO MASAROTTO
Department of Statistical Sciences
University of Padua
Italy

XITH INTERNATIONAL WORKSHOP ON INTELLIGENT STATISTICAL QUALITY CONTROL
August 20–23, 2013
Sydney, NSW Australia
1. Statistical Process Control (SPC)

2. Stochastic Approximation - Nonlinear stochastic root finding

3. A two-stage Robbins and Monro algorithm

4. Examples

5. Conclusions
General framework

In-Control and Out-of-Control

- $x_1, x_2, \ldots$ observations on a quality characteristic (or several characteristics...)
- Process is In-Control (IC) before $\tau$
  \[
  (x_t | x_{t-1}, \ldots, x_1) \sim p_{\theta_0}(x_t | x_{t-1}, \ldots, x_1) \quad (t = 1, \ldots, \tau - 1).
  \]
- Process is Out-of-Control (OC) after $\tau$
  \[
  (x_t | x_{t-1}, \ldots, x_1) \sim p_{\theta_1}(x_t | x_{t-1}, \ldots, x_1) \quad (t = \tau, \tau + 1, \ldots).
  \]
- The time of the change, $\tau$, is unknown.
At time $t$ a decision function (control statistic) is evaluated

$$Z_t = Z_t(x_t, x_{t-1}, \ldots, x_1)$$

Control chart performance evaluated in terms of Run-Length distribution, where

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

$h$ is determined to give prescribed values of the in-control RL characteristics:
- In-Control ARL;
- quantiles, false alarm probability, etc.

**Standard applications:** the RL characteristics can be computed either analytically or numerically.
Detection Algorithms (Control Charts)

Examples of nonstandard applications

- Multivariate statistical monitoring
  - several quality characteristics (multivariate control charts)
  - several process parameters (combined control charts)
- High frequency sampling

Nonstandard applications: the RL characteristics can be computed via Monte Carlo simulation.
Assume that it is possible to simulate

\[ s = \frac{RL - ARL_0}{ARL_0} \sim P_h \]

from \( P_h \), with \( h \in \mathbb{R} \).

Find \( h \), the root to

\[ g(h) = E_h(s) = 0 \]

where \( E_h(\cdot) \) is computed with respect to \( P_h \),

Monte Carlo techniques can be used to compute \( E_h(\cdot) \).
SA: class of stochastic recursions for on-line estimation and root finding

Recent overview (Pasupathy and Kim, 2011)

- Rich arsenal of theories and techniques concerning asymptotic and finite-sample performance
- New important issues related to emerging application areas and requiring
  - stochastic search and optimization
  - performance measures that can only be estimated by simulation
  - solving non-linear root-finding problems
- Easier implementation with modern advances in simulation methodology and software.
Why a stochastic approximation algorithm for SPC applications?

- High efficiency of on-line SA estimation:
  - fast updating;
  - minimum number of simulations to attain a given precision of the on-line estimate
  - no storage/space cost: observations and control statistics do not need to be saved.
  - recursive algorithms can take advantage of parallel computation.

- Our previous research:
  - the SA-based design offers a simple to implement solution when the run-length characteristics can only be simulated.
  - Control limits can be estimated via SA using several criteria: in-control ARL, quantiles, false alarm probability, etc.
The Robbins and Monro (RM) iteration

Stochastic Root-finding problem

Simulate

\[ s_r = \frac{RL_r - ARL_0}{ARL_0}, \quad r = 0, 1, \ldots \]

Find \( h^* \) the root to

\[ g(h) = E_h(s) = 0 \]

by using the recursive iteration

\[ h_{r+1} = h_r - \frac{1}{r+1} As_r, \quad r = 0, 1, \ldots, \]

A the gain of the scheme.
Many theoretical results ensure the asymptotically convergence, i.e.
\[ h_r \to h^* \]
as \( r \to \infty \) and \( A g(h) \) points toward \( h^* \).

Efficient choice of the gain scheme

\[
A = \left( \frac{\partial g(h)}{\partial h^T} \bigg|_{h=h^*} \right)^{-1}
\]
The Robbins and Monro (RM) iteration

Problem/Research question

The Jacobian matrix is unknown

How to “tweak” the Robbins-Monro procedure to obtain a satisfactory finite-sample performance?

Proposal of an efficient SA algorithm to estimate $h$ with a given accuracy
How to accelerate the RM algorithm?

A two-stage SA algorithm

1. **First stage (initialization):** use a **fixed-gain** SA method
   - specify a good starting point: an **arbitrary** initial value is moved to a neighborhood of the solution;
   - the gain matrix is estimated **adaptively**.

2. **Second stage (estimate):** use the **iterate averaging method**, PR, (Ruppert, 1991; Polyak and Judtisky, 1992)
   - the sample average of $N$ recursive iterations is used to estimate the control limit $h$;
   - the iterative algorithm is stopped when a given level of accuracy is attained.
First Stage

Choice of a good starting point and adaptive estimate of $A$

1. simulate pseudo-observations and run-lengths

2. update the estimate, for $N_{fixed}$ iterations, according to

$$\tilde{h}_{r+1} = \tilde{h}_r - A_{\text{fixed}} s_r, \quad r = 0, \ldots, N_{fixed} - 1$$

- $\tilde{h}_0$ initial value chosen by users;
- $A_{\text{fixed}} > 0$ fixed constant;
- $s_r$, score simulated at $\tilde{h}_r$.

3. update the estimate of the gain matrix as a function of the score $s_r$.

4. use $\tilde{h}_{N_{\text{fixed}}}$ as initial value for the second stage.
Second Stage: the PR algorithm

How to obtain an estimate with a desired accuracy?

1. Compute a sequence of estimates $\overline{h}_r$, using a PR algorithm

\[
\begin{align*}
    h_{r+1} &= h_r - \frac{1}{(r+1)q} A s_r \\
    \overline{h}_{r+1} &= \overline{h}_r + \frac{1}{r+1} (h_r - \overline{h}_r)
\end{align*}
\]

- $h_0 = \tilde{h}_{N_{\text{fixed}}}$
- $\overline{h}_0 = 0$
- $A$ comes from the first stage

2. Stop the iterative PR algorithm when

\[
| g(\overline{h}_r) | = | E_{\overline{h}_r}(s) | \leq \gamma
\]

with $\gamma$ a desired level of accuracy.
A stopping rule for the PR algorithm

When $r$ is large, at least approximately

$$g(\bar{h}_r) \sim N\left(0, \frac{1}{r}E_{h^*}\left[s^2\right]\right).$$

Proposed stopping rule

$$N_{PR} = \inf \left\{ N > N_{min} : N \geq \left(\frac{z}{\gamma}\right)^2 \frac{1}{N} \sum_{r=1}^{N} S_i^2 \right\}$$

- $z$ is such that $P(-z \leq N(0,1) \leq z) = 1 - \alpha$
- $N_{min}$ minimum number of iterations

$\bar{h}_{NPR}$ is used as the final estimate of $h$
Multivariate and multiple control charts

- $RL_i$, RL of the $i$-th control scheme
  $$RL_i = \inf \{ t > 0 : z_{i,t} > h_i \}, \ i = 1, \ldots, p$$
- $RL$, RL of the combined control chart
  $$RL = \min(RL_1, \ldots, RL_p).$$

Determine $\vec{h} = (h_1, \ldots, h_p)^T$ so that
$$E_{\vec{h}}[\min(RL_1, \ldots, RL_p)] = ARL_0 \text{ and } E_{\vec{h}}(RL_1) = \cdots = E_{\vec{h}}(RL_p).$$

Solve so that $E_{\vec{h}}(\vec{s}) = 0$, where
$$\vec{s} = (s_i) = \left( \min(RL_1, \ldots, RL_p) - ARL_0 \right) + \frac{RL_i - \overline{RL}}{ARL_0} \right)^T$$
and $\overline{RL} = (RL_1 + \cdots + RL_p)/p$. 

Efficient Control Chart Calibration by Simulated Stochastic Approximation 16/23
### Suggested constant values

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_{\text{fixed}}$</td>
<td>$0.1$</td>
</tr>
<tr>
<td>$N_{\text{fixed}}$</td>
<td>$500$</td>
</tr>
<tr>
<td>$\delta$</td>
<td>$0.1$</td>
</tr>
<tr>
<td>$A_{\text{min}}$</td>
<td>$0.1$</td>
</tr>
<tr>
<td>$A_{\text{max}}$</td>
<td>$100$</td>
</tr>
<tr>
<td>$q$</td>
<td>$0.55$</td>
</tr>
<tr>
<td>$z$</td>
<td>$3$</td>
</tr>
<tr>
<td>$N_{\text{min}}$</td>
<td>$1000$</td>
</tr>
<tr>
<td>$C_{\text{maxRL}}$</td>
<td>$10$</td>
</tr>
</tbody>
</table>
Univariate control chart: *AGLR* (Capizzi and Masarotto, 2012), for detecting unknown arbitrary patterned mean shift;

Multivariate control chart: *$T^2$-MEWMA* (Reynolds and Stombous, 2010), for detecting changes in a process mean vector;

Combined control chart: *NEWMA* (Zou et al., 2008), for monitoring nonlinear profiles (MEWMA + a nonparametric test).
Simulation experiment

- $ARL_0=200$, $\gamma = 0.025, 0.005$
- 200 estimates $\bar{h}$ of the control limit for each control scheme (single and/or combined)
- initial values randomly generated from $U(1000, 2000)$;
- in-control ARL: average over 1000000 run-lengths simulated for each estimate of the control limit.
### Results

**Accuracy:** $\gamma = 0.005$

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>SD</th>
<th>Min</th>
<th>$Q_{10}$</th>
<th>$Q_{90}$</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>AGLR</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$h$</td>
<td>2.920</td>
<td>0.001</td>
<td>2.918</td>
<td>2.918</td>
<td>2.921</td>
<td>2.922</td>
</tr>
<tr>
<td>ARL</td>
<td>200.002</td>
<td>0.368</td>
<td>198.942</td>
<td>199.373</td>
<td>200.555</td>
<td>200.981</td>
</tr>
<tr>
<td>$N_{PR}$</td>
<td>327881.195</td>
<td>1125.206</td>
<td>324866.000</td>
<td>326023.700</td>
<td>329695.350</td>
<td>331088.000</td>
</tr>
<tr>
<td><strong>$T^2$-MEWMA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$h_1$ (MEWMA)</td>
<td>16.368</td>
<td>0.003</td>
<td>16.360</td>
<td>16.363</td>
<td>16.374</td>
<td>16.379</td>
</tr>
<tr>
<td>$h_2$ ($T^2$)</td>
<td>18.272</td>
<td>0.003</td>
<td>18.263</td>
<td>18.268</td>
<td>18.276</td>
<td>18.279</td>
</tr>
<tr>
<td>ARL (Combined)</td>
<td>200.011</td>
<td>0.264</td>
<td>199.228</td>
<td>199.619</td>
<td>200.469</td>
<td>200.731</td>
</tr>
<tr>
<td>$ARL_1$ (MEWMA)</td>
<td>381.057</td>
<td>0.594</td>
<td>379.676</td>
<td>380.124</td>
<td>382.045</td>
<td>382.858</td>
</tr>
<tr>
<td>$ARL_2$ ($T^2$)</td>
<td>381.100</td>
<td>0.577</td>
<td>379.197</td>
<td>380.223</td>
<td>382.004</td>
<td>382.533</td>
</tr>
<tr>
<td>$N_{PR}$</td>
<td>972555.315</td>
<td>3932.864</td>
<td>966914.000</td>
<td>968371.650</td>
<td>976695.900</td>
<td>1016871.000</td>
</tr>
<tr>
<td><strong>NEWMA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$h_1$ ($A$)</td>
<td>14.186</td>
<td>0.003</td>
<td>14.178</td>
<td>14.181</td>
<td>14.190</td>
<td>14.194</td>
</tr>
<tr>
<td>$h_2$ ($B$)</td>
<td>16.423</td>
<td>0.003</td>
<td>16.417</td>
<td>16.419</td>
<td>16.429</td>
<td>16.431</td>
</tr>
<tr>
<td>$h_2$ ($C$)</td>
<td>22.168</td>
<td>0.003</td>
<td>22.162</td>
<td>22.163</td>
<td>22.175</td>
<td>22.179</td>
</tr>
<tr>
<td>$h_4$ ($D$)</td>
<td>24.743</td>
<td>0.003</td>
<td>24.734</td>
<td>24.739</td>
<td>24.748</td>
<td>24.753</td>
</tr>
<tr>
<td>ARL (Combined)</td>
<td>200.013</td>
<td>0.248</td>
<td>199.390</td>
<td>199.619</td>
<td>200.419</td>
<td>200.734</td>
</tr>
<tr>
<td>$ARL_1$ ($A$)</td>
<td>463.193</td>
<td>0.674</td>
<td>461.069</td>
<td>462.048</td>
<td>464.207</td>
<td>465.388</td>
</tr>
<tr>
<td>$ARL_2$ ($B$)</td>
<td>463.125</td>
<td>0.689</td>
<td>461.649</td>
<td>461.931</td>
<td>464.378</td>
<td>465.152</td>
</tr>
<tr>
<td>$ARL_3$ ($C$)</td>
<td>463.219</td>
<td>0.628</td>
<td>461.168</td>
<td>462.267</td>
<td>464.314</td>
<td>464.758</td>
</tr>
<tr>
<td>$ARL_4$ ($D$)</td>
<td>463.176</td>
<td>0.635</td>
<td>461.607</td>
<td>462.211</td>
<td>464.351</td>
<td>464.892</td>
</tr>
<tr>
<td>$N_{PR}$</td>
<td>1432964.750</td>
<td>3552.637</td>
<td>1420855.000</td>
<td>1426866.950</td>
<td>1438141.050</td>
<td>1441659.000</td>
</tr>
</tbody>
</table>
Simple parallel implementation

$M=4$ CPUs, precision $\gamma \sqrt{M_{CPU}}$, $T^2$-MEWMA chart

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>SD</th>
<th>Min</th>
<th>$Q_{10}$</th>
<th>$Q_{90}$</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_1$ (MEWMA)</td>
<td>16.368</td>
<td>0.004</td>
<td>16.355</td>
<td>16.362</td>
<td>16.374</td>
<td>16.378</td>
</tr>
<tr>
<td>$h_2$ ($T^2$)</td>
<td>18.272</td>
<td>0.003</td>
<td>18.262</td>
<td>18.267</td>
<td>18.278</td>
<td>18.281</td>
</tr>
<tr>
<td>ARL (Combined)</td>
<td>200.030</td>
<td>0.295</td>
<td>199.236</td>
<td>199.500</td>
<td>200.464</td>
<td>200.938</td>
</tr>
<tr>
<td>ARL$_1$ (MEWMA)</td>
<td>381.094</td>
<td>0.657</td>
<td>378.554</td>
<td>380.016</td>
<td>382.149</td>
<td>382.889</td>
</tr>
<tr>
<td>ARL$_2$ ($T^2$)</td>
<td>381.150</td>
<td>0.658</td>
<td>379.304</td>
<td>380.056</td>
<td>382.124</td>
<td>382.912</td>
</tr>
<tr>
<td>$N_{PR}$</td>
<td>976788.115</td>
<td>14286.952</td>
<td>968629.000</td>
<td>970709.950</td>
<td>981628.150</td>
<td>1168400.000</td>
</tr>
</tbody>
</table>

- Averages of the $M$ estimates of the control limits all attain the same degree of accuracy
- Sums of the stopping times
- The parallel version takes a third of the time to estimate the control limits with the desired accuracy.
Conclusions

- Relatively small variability of the replicates of the control limits
- Tighty clustered values of the IC ARLs to the target value
- Efficient finite-time performance: smallest number of iterates with a prescribed degree of precision of the estimates
- The computational effort increases as the required degree of precision increases. This drawback can be efficiently overcome using a simple parallel implementation.
THANKS...

**Since 1222**

*Universa Universis Patavina Libertas*

*(Paduan Freedom is Complete and for Everyone)*

---

**Anatomy Theatre (1594)**

**Galileo Galilei’s desk (≈1605)**