INTERFACING QUANTITATIVE NDE WITH COMPUTER ALGORITHMS FOR AUTOMATED STATISTICAL PROCESS CONTROL

E. P. Papadakis

Center for NDE
Iowa State University
Ames, I.A. 50011

INTRODUCTION

In the Factory of the Future (FOF), production will be unified under a system of Computer Integrated Manufacturing (CIM). Automatic computer-integrated control of processes, detection of errors, and determination of corrective action will be necessary because of the proposed level of manpower in the FOF[1,2]. Under these circumstances, a process which might go out of control and remain that way would be highly detrimental. Very rapid Statistical Process Control (SPC) will provide definitive warnings of out-of-control conditions.

SPC involves taking measurements periodically on relatively small sets of specimens, calculating control chart points, and analyzing the control charts by Run Rules[3]. With automatic electronic measurement systems, and with computers to calculate control chart points and analyze them by Run Rules in the computer memory, the "period" between control chart points can be shrunk to any desired degree consistent with the acquisition of data. In particular, the sets of SPC specimens may be sequential, comprising the entirety of production, such that no extra production is performed between control chart points.

The present work explores the implications of one model for computerized SPC and examines the possibilities for improved timeliness of spotting out-of-control conditions. The influence of Type I errors is also calculated. Their frequency is studied by a Monte Carlo computer simulation of SPC run rules during under-control operation. A strategy (algorithm) based on reliability theory applied to the output of the simulation is developed to discriminate against Type I errors in practice.

THEORY

Background and Rules

Control charts are used to measure the probability that a process is out of control. All charts are characterized by an average value and control limits. In using control charts, there are two steps: (1) data gathering on a periodic basis to add one point per period to the chart, and (2) interpretation of the positions of the recent points relative to the control limits. The use of several recent points can be summarized by Run Rules which the operator must observe on the control chart to interpret an out-of-control condition. The Western Electric Co.[3] has listed four Run Rules as follows:

<u>Western Electric Run Rules</u>

1. $\overline{X}$ exceeds $\overline{\overline{X}} \pm 3\sigma_{\overline{X}}$ for one point.
2. $\overline{X}$ exceeds $\overline{\overline{X}} \pm 2\sigma_{\overline{X}}$ for two out of three successive points. (Both positive or both negative with respect to $\overline{\overline{X}}$.)
3. $\overline{X}$ exceeds $\overline{\overline{X}} \pm 1\sigma_{\overline{X}}$ for four out of five successive points. (All four either positive or negative with respect to $\overline{\overline{X}}$.)
4. $\overline{X}$ lies on the same side of $\overline{\overline{X}}$ (either higher or lower) for eight consecutive points.

When all four of these are used together, the probability[4] of making a Type I error (saying good production is out-of-control) is slightly lower than 1% every time a new data point is added to the X-bar chart, and the Run Rules are applied to the most recent eight points.

<u>Operation</u>

With a manual system, N parts are produced and not measured; the next n are measured and $\overline{X}$ is calculated and plotted; then the next N are not measured, and so on. At each new $\overline{X}$ point, the operator determines whether the process is out-of-control.

<u>Reduction of N</u>

It will be necessary in the Factory of the Future and would be advantageous from a control point of view now to reduce the number of parts N produced between samples so that:

1. Out-of-control conditions could be caught promptly, and
2. Quarantining-and-sorting would involve a small number of parts.

One method of reducing N is considered here. The proposed method is to utilize automated control charting with computers to match the rate of the automated production already in place. A hypothetical system embedded in a Local Area Network in a Factory of the Future is shown in Fig. 1 and described below.

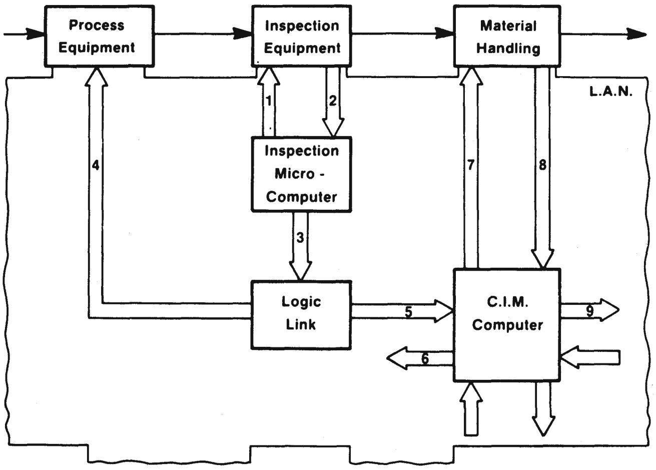

Fig. 1. An automated computer-controlled inspection system for SPC embedded in a Local Area Network with a governing CIM computer.

Consider an electronic measuring system which could make the same type of measurement made currently for X-bar control charting of some process. This electronic measuring system under computer control and mounted on-line could measure every part passing through it. By a counting procedure the computer could choose the data points to skip and those to use. Then it could compute $\bar{X}$, update its list, and analyze the Run Rules. The entire calculational sequence on set N + 1 through N + n could be performed in the time while the measuring device was waiting for the production line to present it with the next part to measure. Thus, the batches could be shrunk in size so that the number N goes to zero (N=0). This concept of time compression is shown in Fig. 2. With the data for the most recent 8 batches updated and stored in computer memory and examined at the completion of every n-size batch, the time until the detection of an out-of-control condition would be equivalent to 8n production cycle times τ for the longest Run Rule where τ is the time to produce one part. The computer sequence is shown in the flow chart in Fig. 3. As an example, for 600 parts per hour, $\tau = 6s$. With n = 5, $8n\tau = 240s$, i.e., 4 minutes, whereas at one manual point per hour the delay would be one shift. As a result of the timeliness, the number of parts to be quarantined could be 40 instead of 4800 in the above manual example of 8 points each an hour apart.

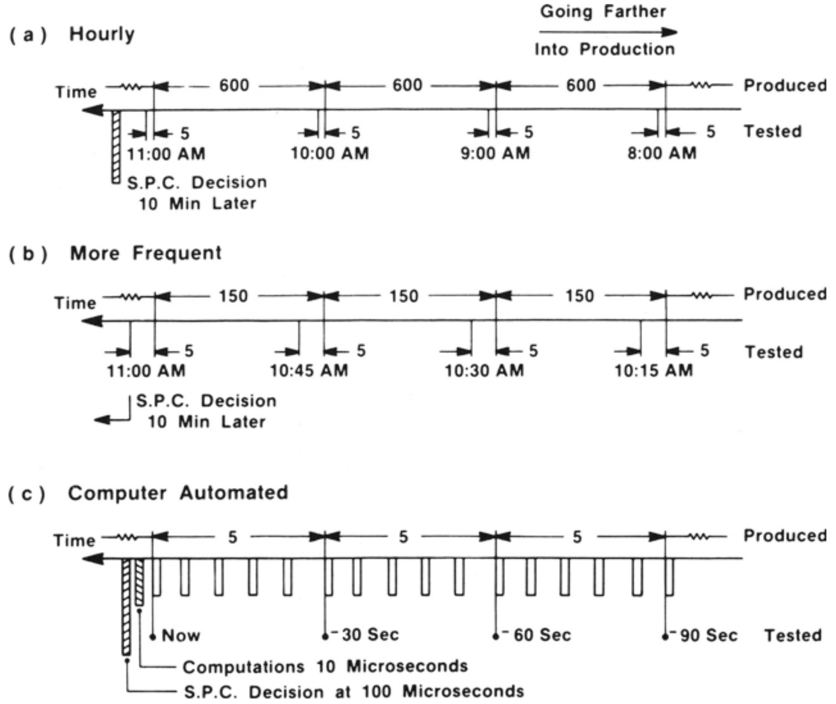

Fig. 2. Time compression of SPC permitted by computer automation. (a) Present manual inspection, computation, and decision. Too slow for FOF. (b) Manual system speeded up to point where it is impractical/uneconomical. (c) Computer automated SPC (CASPC). Decisions made in less than 100 microseconds after the production of each group of n parts. Group size n = 5 here.

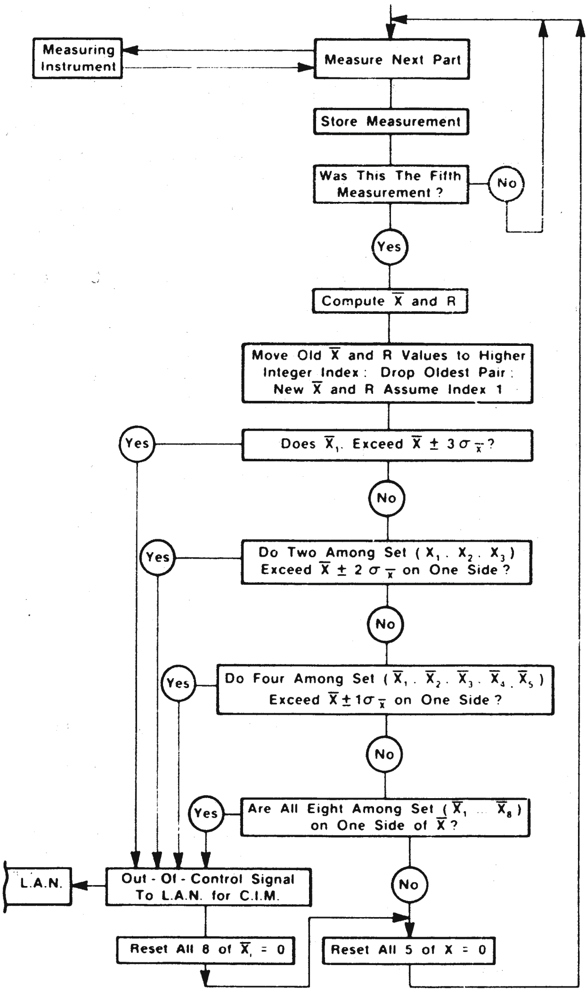

Fig. 3. Flow chart of a computer program to do computer-automated statistical process control (CASPC). Specialized to subsets of five (5) measurements per control chart point and to analysis by Western Electric Run Rules.

Repetitive Run Rule Decisions to Mitigate the Type I Error

Use of the Western Electric Run Rules results in a Type I error of just under 1% at every control chart point[4]. Using $\tau = 6s$ and $n = 5$, the Mean Time Between Type I Errors (MTBTIE) would be 3000s or 0.83 hours. This short time would be unreasonable and unacceptable, and would lead to fallacious down-time.

The strategy proposed here to reduce Type I errors is to utilize a reliability approach by repeating the out-of-control determination with the same Run Rules on further specimens. Reliability theory would indicate that Type I errors would not repeat at short intervals frequently for statistical processes.

An a priori calculations of the probabilities of consecutive or closely spaced repetitive Type I errors is tedious and unwieldy. A much more fruitful appraoch is a computer simulation of the operation of the control charts as they monitor a process which is under control and which shows expected statistical fluctuations. A Monte Carlo run through a multiplicity of possible fluctuations will define the probable repetitiveness of Type I errors. Analysis of the Monte Carlo output should lead to an algorithm for reducing the Type I error accumulation to a manageable frequency.

Monte Carlo Study of Type I Error Repetition

A Monte Carlo computer simulation was performed to test the feasibility of using the method suggested in Section D to mitigate the effects of Type I errors. A computer program was written to analyze control charts according to the Western Electric Run Rules. The flow chart is essentially the same as Fig. 3 except that the random number generator is substituted for the initial measuring of 5 parts. This program was run with normal random number input simulating a process under control. Whenever the program called an out-of-control condition, the occurrence indicated a Type I error.

For 99,999 control chart points, there were 1,104 indications of out-of-control for a probability of about 1 in 90 verifying the WECO prediction of approximately 1 in 100. The computer output yielded also the time between Type I errors in terms of the times between control chart points. A frequency histogram versus the time intervals from one indication to the next indication is given in Fig. 4. Details concerning the frequency distribution in the first ten time intervals is given in Fig. 5 by another histogram. It can be seen from Fig. 5 that very few second indications happen very soon after a first indication.

The simulation output was analyzed further in terms of the frequency of three "calls" closely following one another. The data are shown in Table 1. Only three (3) occurrences were noted of three calls with 10 or fewer time intervals between #1 and #2 and between #2 and #3. If the fourth Run Rule is neglected, the number reduces to two (2) occurrences. Thus, the requirement of finding three closely-following out-of-control indications (spaced apart by up to 10 time intervals) improves the reliability to 3 parts in 100,000 or better. The MTBTIE becomes 280 to 420 hours (35 to 52 shifts), quite acceptable in view of the timeliness of the indication which would be 10 minutes at most (6s/part x 5 parts/point x 20 points max ÷ 60s/min). The output to be quarantined would be only 140 parts at most.

Table 1. Time intervals for Triple Type I Calls (99,999 control chart points; Run Rule in parenthesis)

To First Call	Between First Call and Second Call	Between Second Call and Third Call
Over 10	5(2)	3(2)
Over 10	8(4)	8(4)
Over 10	8(1)	8(1)

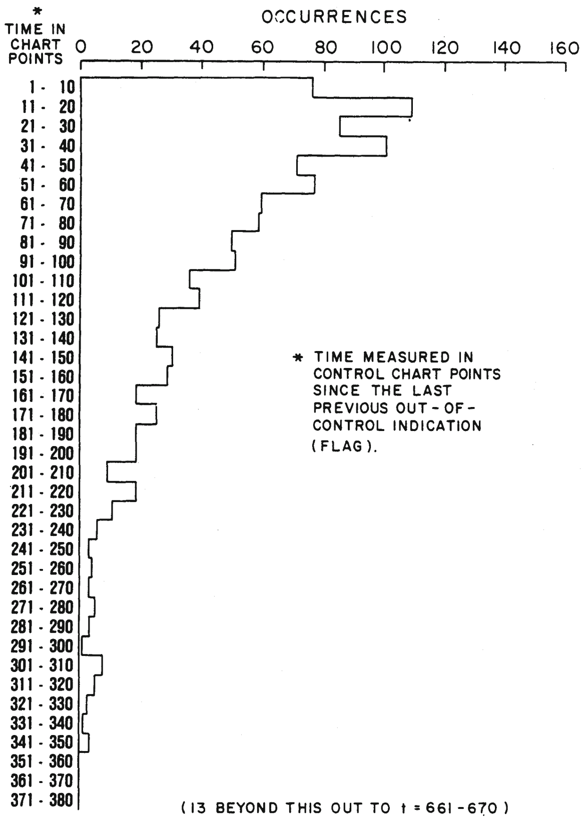

Fig. 4. Frequency histogram of the times of occurences of Type I errors. Output of a Monte Carlo computer simulation of the operation of the Western Electric Company Run Rules for $\overline{X}$ control charts.

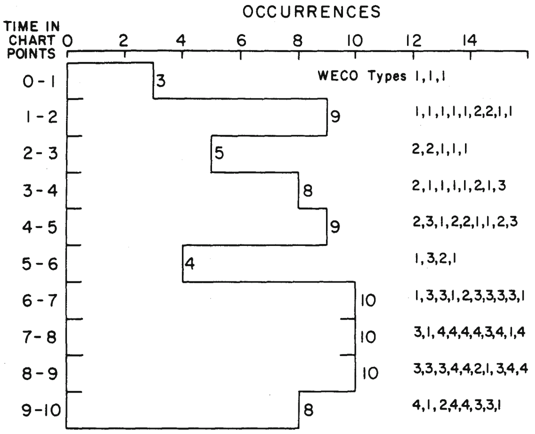

Fig. 5. Detail of Figure 4 for the first ten (10) time intervals. Each time interval is one control chart point.

These simulation results strongly suggested that three successive out-of-control indications within 10 time intervals of each other should be used as the algorithm to mitigate the effects of Type I errors in computer-automated statistical process control. See the flow chart in Fig. 6 for an embodiment of the algorithm.

Byproduct Benefits

1. Just-in-Time Inventory - Minimal interruptions.
2. Statistical Data - Ongoing streams of data upon all parts.
3. Potential for Sorting In Situ - Accept and reject automatically.

POTENTIAL FOR IMPLEMENTATION

General

Actual human measurements are being supplanted by electronic measurements at a rapid rate. These include NDE variables which are related to material properties. All the electronic measurement instruments have the potential for being linked with computers for control and data acquisition; many of the measurement systems are configured that way currently. Thus, automated SPC is possible.

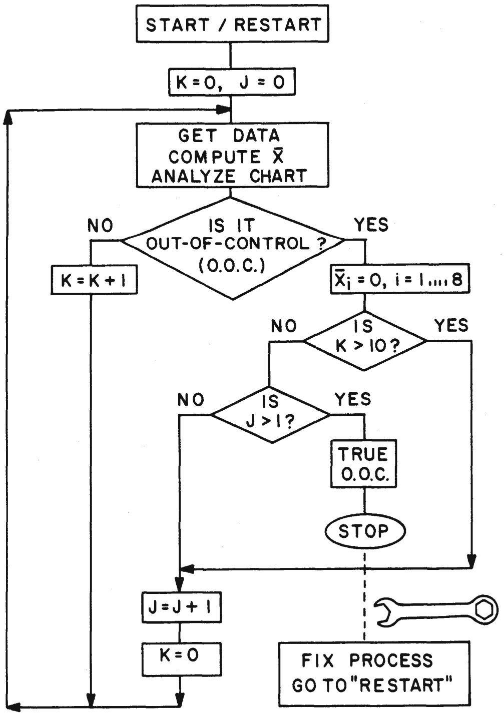

Fig. 6. Flow chart for computer algorithm to test for three out-of-control calls less than 11 X-bar points apart.

Computer Software

The author has written a subroutine in Fortran IV for the analysis of control chart points according to the Western Electric Run Rules. The output of the control chart analysis subroutine is a flag indicating out-of-control and a pointer showing the number of the Run Rule which detected the condition. Thus, this control chart analysis subroutine tells the operator when to stop production and how far back in time to quarantine the product, assuming the Run Rules are considered adequate. Programs with similar capabilities are available from at least two vendors of software[5,6]. These must be interfaced with metrology and/or NDT systems on a custom basis. One vendor[7] of machine vision systems builds multi-sensor units which measure dimensions on production lines. SPC information is measured and reported for every piece, and out-of-control conditions are flagged. One NDT and metrology systems vendor[8] has a gaging system with a computerized statistical package which plots control charts on a video monitor for visual decisions about points exceeding $\pm 3\sigma_{\bar{x}}$.

SUMMARY AND CONCLUSIONS

Points for control charts can be calculated and the charts can be analyzed for out-of-control conditions as fast as production occurs providing that the measurements are made by electronic instruments interfaced with computers. It is shown in this paper that an out-of-control condition in production could be detected by computer-automated SPC after the production of only forty (40) parts using the longest Run Rule in the literature with 5 parts per data point. Certain statistical techniques can be used to check on the validity of out-of-control indications which might be Type I errors, and to reduce the probability of taking unneeded corrective action in the presence of Type I errors. The analysis in this paper suggests that the use of three (3) successive out-of-control indications within 10 control chart point times of each other would be adequate to discriminate against Type I errors. With the suggested algorithm, the probability would be 3 per 100,000 chart points. Only 140 parts would be produced within the algorithm time. An added benefit would be the automatic sorting of faulty parts during the period between the inception of the out-of-control condition and its detection by the Run Rules in the SPC computer software.

ACKNOWLEDGMENT

The author is indebted to R. Hough, D. Hobson, H. Arnold, and W. W. Scherkenbach for critical comments and enlightening discussions. R. Hough and W. W. Scherkenbach kindly reviewed the manuscript in preliminary form. D. O. Thompson encouraged and supported this presentation.

REFERENCES

1. J. Tukloff, Industrial Engineering 16 (2), 46-52 (1984).
2. M. E. Merchant, IEEE Spectrum 20 (5), 36-39 (1983).
3. Western Electric Company (1956), Statistical Quality Control Handbook (Western Electric Co., Newark, New Jersey), pp. 24-28 et al.
4. Western Electric Company (1956), op. cit., pp. 180-183.
5. Advanced Systems and Designs, Inc., Dearborn, Michigan.
6. BBN Software Products, Northbrook, Illinois.
7. Perceptron, Inc., Farmington Hills, Michigan.
8. K. J. Law Engineerings, Inc., Farmington Hills, Michigan.