Center
for
Economic Research

No. 2000-29

SHORT-TERM ROBUSTNESS OF PRODUCTION MANAGEMENT SYSTEMS: NEW METHODOLOGY

By Jack P.C. Kleijnen and Eric Gaury

March 2000

ISSN 0924-7815

D:\Data\WP\PAPERS\RobEJOR1.WPD

Printed: March 7, 2000 (12:37PM)

Written: February 18, 2000

Short-term robustness of production management systems: new methodology

Jack P.C. Kleijnen [a] & Eric Gaury [b]

[a] Corresponding author
Department of Information Systems (BIK)/
Center for Economic Research (CentER),
Tilburg University (KUB),
Postbox 90153, 5000 LE Tilburg, Netherlands
Phone: +3113-4662029; Fax: +3113-4663377; E-mail: kleijnen@kub.nl
Web: center.kub.nl/staff/kleijnen

[b] BIK/CentER, KUB and
LIMOS/IFMA, Clermont Ferrand, France
Current address:
Technocentre Renault
API: TCR AVA 3 61
1, Avenue du Golf, 78288 Guyancourt Cedex, FRANCE
Phone: +01 34 95 52 03; Fax: +01 34 95 56 25; E-mail: eric.gaury@renault.com

Abstract

This paper investigates the short-term robustness of production planning and control systems. This robustness is defined here as the system’s ability to maintain short-term service probabilities (i.e., the probability that the fill rate remains within a prespecified range), in a variety of environments (scenarios). For this investigation, the paper introduces a heuristic, stagewise methodology that combines the techniques of discrete-event simulation, heuristic optimization, risk or uncertainty analysis, and bootstrapping. This methodology compares production control systems, subject to a short-term fill-rate constraint while minimizing long-term work-in-process (WIP). This provides a new tool for performance analysis in operations management. The methodology is illustrated via the example of a production line with four stations and a single product; it compares Kanban, Conwip, Hybrid, and Generic production control schemes.

Keywords: manufacturing, inventory, risk analysis, robustness and sensitivity analysis, scenarios

1. Introduction

The problem addressed in this paper is the *robustness* or *riskiness* of production management systems; that is, their sensitivity to unexpected changes in the environment. We focus on *pull* production-control systems (PPCSs) such as Kanban systems and their variants.

Quite surprisingly, our elaborate literature survey suggests that PPCS analysts optimize PPCSs, assuming a specific environment (scenario). We, however, claim that in practice the future environment is unknown. Consequently, PPCS performance may be far below the manager’s target. Therefore we wish to account for changes in the environment.

Moreover, traditionally these analysts focus on long-run, steady-state performance metrics. We, however, also consider the *short run*: if the managers’ performance is bad in the short run, they will be fired - at least such performance is not good for their careers. [In the related field of economics, Keynes said: ‘In the long run we are all dead’. For example, supply and demand of labor will be equal in the long run, so no unemployment will persist; this view, however, was cold comfort for the numerous unemployed in the 1930s! We claim that production managers are in the same position as the unemployed were.]

Whereas robustness of PPCSs has been neglected in the literature, robustness of products (e.g., automobiles) has been emphasized - especially under the influence of the Japanese quality guru *Taguchi*. In practice, however, robustness of PPCSs is judged to be a major issue (our personal contacts with managers support this statement).

Our *methodology* combines the following quantitative techniques, in five stages (we give more details in the next sections).

Stage 1: Build a *discrete-event simulation* of the real production system and its PPCS. This is a classic technique, which needs no further comments or references.

Stage 2: Assuming a specific environmental scenario (e.g., the most likely combination of non-controllable input values), try to *optimize* the PPCS of stage 1. This optimization is not simple, since the simulation is stochastic, non-linear, and multi-response. This is a well-known problem.

Stage 3: Apply *risk analysis* (RA) - also called risk assessment, risk management, or uncertainty analysis - to estimate the probability of a specific system performance (output). We do not know of any RA applications in production management, but we do know many applications in other fields. RA uses the Monte Carlo method (i.e., random numbers) to sample from an assumed distribution of inputs; RA feeds these inputs into the simulation model (resulting from stages 1 and 2).

Stage 4: Apply *bootstrapping* to estimate a (say) 90% confidence region for the performance measures. We do not know of any bootstrap applications in PPCSs. The bootstrap is a resampling technique, using Monte Carlo.

Stage 5: Let the managers select a particular PPCS that fits their specific risk attitude.

We illustrate this five-stage methodology through the example of a stochastic production line with four stations and a single product, taken from Bonvik, Couch, and Gershwin (1997). We compare the performance of this line, under four PPCSs (namely, Kanban, Conwip, Hybrid, and Generic, defined below). Our methodology results in a performance ranking that differs from the ranking resulting from other methods that ignore risk!

The remainder of this paper is organized as follows. §2 gives background information on robustness and RA. §3 explains our methodology. §4 illustrates this methodology through an example. §5 summarizes our conclusions. Forty-two references finish this paper.

2. Robustness and risk analysis

2.1 Prior work on robustness and risk analysis ***outside*** *manufacturing*

Typically, Operations Research focuses on optimization, *given a set of input parameter values*. In practice, however, these parameters are unknown. How can one solve this problem?

One type of situation is that the parameters are estimated from historical data. Then, the estimates follow a statistical distribution. One of the few queueing studies accounting for this complication is Haverkort and Meeuwissen (1995). Some simulation studies propose bootstrap methods to account for the variability of the input process: see Barton and Schruben (1993), Cheng and Holland (1997), Chick (1997); also see §2.2. Long ago, Kleijnen (1983) has already proposed to use the central limit theorem to incorporate input variability; also see Kleijnen (1994).

The second type of situation is that there are no historical data on the input process. Then neither the bootstrap technique nor the central limit theorem apply. This problem is addressed by RA! The literature gives several definitions of RA, sometimes differentiating between RA and uncertainty analysis, and between RA and risk management; see Granger Morgan and Henrion (1990). RA is a standard technique in nuclear engineering (see Helton, Anderson, Marietta, and Rechard 1997, and also Balson, Welsh, and Wilson 1992 and Breeding et al. 1992). In computer science, RA has been used in software development (Bennett, Bohoris, Aspinwall, and Hall 1996). Recently, RA has also become accepted in management, because of the widespread availability of software that supplements popular spreadsheet programs (Sugiyama and Chow 1997); we use the well-known program @Risk (see §3). But in management, RA has been applied mainly to investment analysis; case studies are given in Krumm and Rolle (1992).

In both types of situation we can distinguish two sources of uncertainty:
(i) *System uncertainty*: random customer arrivals, random breakdowns, etc.
(ii) *Analyst uncertainty*: the analysts do not know the true arrival and breakdown rates.
Type (i) is modeled though the ‘laws’ of mathematical probability; for example, Poisson distributions. Type (ii) uses objective or subjective input data. In practice, managers see only one source of uncertainty, namely type (i), system uncertainty! For further discussion of these two types of randomness we refer to Helton (1997).

Even if type (i) uncertainty is ignored (as is the case in deterministic simulation), type (ii) still requires consideration of risk. Risk as studied in RA is closely related to *robustness*. The practical importance of robustness has been explicated by *Taguchi*. (A popular example is Toyota; they have succeeded in designing and producing very reliable cars.) We summarize the Taguchian view as follows.

When designing a physical product (such as a car) in the laboratory, there are a number of controllable inputs (e.g., type of material) and a number of environmental inputs (e.g., humidity). When selecting the values of the controllables to obtain good performance, the undesirable effects of the environment should be minimized. This quality management is nowadays accepted in theory and practice; see the panel discussion in Nair (1992).

Though Taguchi's robustness *concept* is now generally accepted, his *methods* are more controversial! Taguchians investigate the effects of the environment through the selection of a relatively small number of combinations of environmental input values (using orthogonal arrays). However, the efficiency of this experiment can be improved by classic design of experiments or DOE (for example, fractional factorial designs), as classical statisticians have pointed out (see again Nair 1992). More important for our investigation is that the performance measure to be optimized, may be improved: Taguchians assume either quadratic loss functions or simple signal-noise functions that combine the mean and variance of the output; also see Myers (1999). We, however, shall use performance measures that make sense from a managerial point of view: see the next subsection (§2.2).

Whereas Taguchians focus on real-world experiments meant to design robust mass products (such as cars), RA concerns simulation experiments meant to study the chance of a disaster of a unique system (such as a nuclear reactor). Hence Taguchians must limit their experiments to a few scenarios, whereas RA may simulate *hundreds of scenarios*.

We conclude that Taguchi's robustness concept is important; for the study of risk and robustness of systems, however, better methods are simulation and RA.

[Note: In production management, analysts have often applied simulation, including sensitivity analysis through classic DOE and - sometimes - through Taguchian designs (see §2.2). We, however, emphasize that such an analysis selects *extreme* combinations of input values, and these scenarios have low probabilities of realization, in general. An alternative sensitivity technique for simulation in general - namely Schruben and Cogliano's (1987) frequency domain experiments; see Morrice and Bardhan (1995) - also results in too many extreme scenarios. Taguchian robustness of physical products versus simulated systems is further discussed by Mayer and Benjamin (1992).]

2.2 Robustness and risk analysis in production management

For PPCSs we interpret *Taguchian robustness* as follows. When selecting the values of the controllable factors (e.g., number of kanbans/cards) to obtain good performance, the undesirable effects of the environment (e.g., demand variability) should be minimized.

We know only a single application of Taguchi's approach to PPCSs, namely the Kanban study by Moeeni, Sanchez, and Vakharia(1997). Taguchi's approach has also been used for system design including job shops (Benjamin, Erraguntla, and Mayer 1995), and for production planning outside PPCSs (Lim, Kim, Yum, and Hwang 1996).

Besides robustness versus optimality, we consider *short term performance* - as opposed to long term. Traditionally production management focuses on steady-state performance. In such a view events are repetitive, so the classical probability definition applies: a probability is the limit of a frequency. However, consider the probability of a (personal) 'disaster'; for example, being fired as a manager because the environment turns hostile. Such a disaster is a unique event, so a subjectivist definition of probability is more adequate. (A related view is the Bayesian one; see Chick 1997.)

To illustrate our methodology we shall use an example inspired by a Toyota factory and studied in Bonvik et al. (1997). That system may be modeled as a sequence of servers with limited buffers (inventories) and a rather complicated queue priority rule (e.g., Kanban authorization rule); see Buzacott and Shanthikumar (1993).

Traditionally, queueing systems are characterized by their steady-state mean performance. A few studies, however, do consider *transient* performance (e.g., Abate and Whitt 1994, Lin and Cochran 1990, Muppala, Malholtra, and Trivedi 1996, and Tan 1999). Besides expected values, blocking probabilities or buffer overflows are studied for computer and telecommunication systems (Heidelberger 1995).

The Kanban literature uses many performance metrics. Chu and Shih (1992) classify these measures into three categories: overall, inventory related, and due-date related. They state that three criteria have been used frequently, namely capacity utilization, output rate, and WIP. However, we do not use capacity utilization, because the goal of a manufacturing system is not to keep workers and machines busy (Goldrat and Fox, 1986). Further, the output rate should be measured relatively to the demand rate: a production system should not overproduce; yet, it should meet demand very fast. Therefore, a good indicator of system performance is the proportion of demand actually met from stock: the fill rate. So we wish to meet a predetermined fill rate, while minimizing average WIP; see the definitions below. (This metric implies that we do not use a cost function that assumes specific holding and out-of-stock costs; the latter costs would be hard to quantify, in practice.)

We propose the following specific definition of robustness. To give a precise definition, we use the following mathematical notation. Upper case letters denote random variables, lower case letters denote realizations of random variables and deterministic variables, and Greek letters represent parameters to be estimated. So we define

$\mu = E(W)$: expected average WIP ($\mu \geq 0$ as $W \geq 0$);

Y: fill rate per shift ($0 \leq y \leq 1$; percentage of demand per work shift, satisfied from stock);

$\pi = P(Y < c_y)$: probability of Y dropping below the prespecified managerial threshold c_y.

We measure *short-term PPCS performance* by π, and its *long-term performance* by μ. To estimate these two measures, we use discrete-event simulation. This simulation gives

w_t: WIP realized at simulated (continuous) time t;

y_i: fill rate realized in shift i.

We chose to run the simulation for one month or 22 working days, each of 900 minutes ($0 \leq t \leq 19800$), or 44 shifts (2 shifts/day: $i = 1, ..., 44$), plus a warming-up period of three days (2700 minutes). So the estimates for the two performance measures are:

$$\hat{\mu} = \int_{2700}^{2700 + 19800} w_t \; dt/19800 ;$$

$$\hat{\pi} = \sum_{i=1}^{44} I(y_i < c_y)/44 \text{ with indicator function } I(\;).$$

Notice that this definition implies that (say) two shifts each with 1 lost sale is more serious than 1 shift with 2 lost sales. Obviously, the w_t are autocorrelated, and so are the y_i. An example realization is given in Figure 1.

INSERT Figure 1: Simulated w_t (WIP at time t) and y_i (fill rate of shift i)

We speak of a *disaster* whenever y (fill rate/shift) drops below the manager's target c_y. Figure 2 gives an example of the estimated density function and corresponding (cumulative) distribution of Y, and the resulting estimated disaster probability $\hat{\pi} = 0.455$ for $c_y = 0.95$.

INSERT Figure 2: Estimated distribution of y (fill rate per shift) and disaster probability $\pi = P_y(y < 0.95)$: a simulation example

Actually, all these symbols need a subscript denoting the (environmental) *scenario*, as we use RA; that is, we repeat the simulation for different scenarios (say) S:
$\mu_s = E(W| S = s)$;
$Y_s = (Y| S = s)$;
$\pi_s = P(Y < c_y| S = s) = P(Y_s < c_y)$.
In other words, the two performance measures have become random variables if the scenarios are treated as random (input or conditional) variables (Bayesians always use such a world view!). So, by definition, these measures have a joint statistical distribution function.

Management might select a PPCS based on this distribution function (see Figure 7, discussed in §4). However, we think that it is more practical to characterize each of the two marginal functions through a single number. It is quite natural to characterize the estimated marginal distribution of the estimated average WIP through its *average* $\bar{\hat{\mu}}$. The estimated marginal distribution of the estimated disaster probability $\hat{\pi}$, however, we characterize through the estimated probability of $\hat{\pi}$ being higher than another managerial threshold (say) c_π; this is the *probability* $\hat{\rho}$. In other word, we take a risk-averse attitude: we wish to avoid high probabilities of high disaster probabilities (we shall give results for $c_\pi = 0.9$ in §4).

In summary, we define the following *two robustness measures*:
$\eta = \underset{s}{E}(\mathrm{M}_s) = \underset{s}{E}[E(W_s| S = s)]$: mean average WIP averaged over all scenarios (M has realizations μ);
ρ: probability of Π_s exceeding the managerial threshold c_π, under various scenarios.

To estimate η, RA uses a given input distribution of scenarios, resulting in the following estimate.
$\bar{\hat{\mu}} = \sum_{s=1}^{n} \hat{\mu}_s / n$: average WIP averaged over the n scenarios actually sampled.
Actually, we should replace μ by the capital letter M to emphasize the random character of this estimate $\bar{\hat{\mu}}$. (To estimate the randomness of $\bar{\hat{\mu}}$, we use bootstrapping.)

Analogously, the RA estimate of ρ is

$\hat{\rho} = \sum_{s=1}^{n} I(\hat{\pi}_s \geq c_\pi)/n$: fraction of $\hat{\pi}_s$ that exceeds c_π in RA.

The challenge is to meet a constraint on the short-term fill-rate (see $\hat{\rho}$), at minimal long-term WIP (see $\bar{\hat{\mu}}$). Below we shall show how to meet this challenge!

3. New methodology

In this section we detail our methodology that consists of five stages, and - as we go along - we illustrate our method through an example (more details of this example will follow in the next section, §4).

Stage 1: Build a *discrete-event simulation model* of the real production system and its PPCS. In our example the production system is a line with four stations and a single product, taken from Bonvik, Couch, and Gershwin (1997). We compare four PPCS types: Kanban, Conwip, Hybrid, and Generic. The first three PPCSs have already been discussed in the literature; for our methodology it suffices to characterize these PPCSs as follows. Kanban has control loops that connect each production stage with its immediate predecessor. Conwip has a single loop, from the final to the initial production stage; see Spearman, Woodruff, and Hopp (1990). Hybrid simply combines Kanban and Conwip; see Bonvik et al.(1997). Generic is a general PPCS that - in principle - connects each stage with all its predecessors; hence the three other PPCSs as special cases; see Gaury, Pierreval, and Kleijnen (2000). However, Generic does not implement loops with non-restrictive card numbers; for example, Generic reduces to Conwip if the only restrictive loop - given the card numbers of the other loops - is the one that connects the last stage with the first stage. Actually, the number of cards is determined in the next stage.

Stage 2: Assuming a specific scenario, we try to *optimize* the PPCS. Optimization in a random environment is not so straightforward: which solution is preferred in practice or should be preferred by a rational decision maker? In our example we minimize the estimated average long-term WIP criterion $\hat{\mu}$ while satisfying the short-run fill-rate condition $\hat{\pi} = 0$ (i.e., we reject a solution that has an estimated fill rate *Y* below the manager's threshold c_y).

Many optimization techniques have been studied in the simulation literature; see Kleijnen (1998). In our example we use a Genetic Algorithm combined with Response Surface Methodology; for details see Gaury et al. (1999). This yields an optimized Generic with 14 cards for the Conwip loop, 6 and 3 cards for stage 1 and stage 2 respectively; no other loops need to be implemented.

Because we wish to compare our solution with the other three PPCS types, we use the estimated optimal card numbers for Kanban, Conwip, and Hybrid that Bonvik et al. found after an exhaustive search. These card numbers are: 15 for Conwip; 2, 2, 4, and 10 for Kanban; 15 and 2, 3, 5, and 15 for Hybrid (with the first 15 for the Conwip loop, etc.).

Stage 3: We apply *risk analysis* to estimate the probability of particular values for the two performance measures when considering alternative scenarios (besides the base scenario of the preceding stage for which the PPCS is 'optimized', which resulted in one realization $\hat{\mu}$ and $\hat{\pi}$). This RA consists of the following steps.

(a) First we sample a value for each input variable from its input distribution; for simplicity we suppose that the inputs are independent. To increase the accuracy (i.e., reduce statistical variation), we do not use crude Monte Carlo, but *Latin hypercube sampling* (LHS). LHS gives better coverage of the total sample space; see McKay, Beckman, and Conover (1979), and also Helton (1997). LHS is a standard option in @Risk (software add-on that we use).

[Note: LHS is a variance reduction technique developed especially for RA. Nevertheless, LHS has also been used in simulation - albeit rarely; see Avramidis and Wilson (1998) and Chick (1997). First, RA samples values for parameters such as the Poisson arrival rate; next, the discrete-event simulation samples individual arrival times during a simulation run: two sources of uncertainty; see §2.1.]

(b) Next we feed these sampled RA input values into the simulation model that we 'optimized' in the traditional way described in stages 1 and 2. We run this simulation model to obtain one new realization of the two PPCS performance measures, $\hat{\mu}$ and $\hat{\pi}$. We stick to the runlength of one month, selected in §2.2. Because we wish to minimize computer time, we do not replicate the simulation run of one month for a particular scenario.

(c) To estimate the distribution of the outputs ($\hat{\Pi}$, $\hat{M}$), we repeat the RA steps (a) and (b) a number of times; this number is the *LHS sample size n*. All these *n* runs with the discrete-event simulation model start with the same initial conditions, but different random numbers. An illustration is Figure 3, where part (a) shows the estimated marginal density function of the estimated disaster probability $\hat{\Pi}$ (with values $\hat{\pi}$); part (b) does the same for the other criterion $\hat{M}$ (average WIP with values $\hat{\mu}$); part (c) gives the scatter plot that indicates the joint distribution of these two estimators. We shall further discuss this figure in the next section (§4).

INSERT Figure 3: Estimated density function of estimated disaster probability $\hat{\Pi}$ and average WIP $\hat{M}$ - for Kanban, optimized given 97% fill rate target: Bonvik et al’s four-stage example

Figure 3 enables us to estimate the two robustness measures η and ρ through $\bar{\hat{\mu}} = \sum_{s=1}^{n} \hat{\mu}_s/n$ and $\hat{\rho} = \sum_{s=1}^{n} I(\hat{\pi}_s \geq c_\pi)/n$. We emphasize that for different target values c_y and c_π the simulation does not need to be run again: Figures 2 and 3(a) demonstrate that the basic information is available to compute $\hat{\pi}$ and $\hat{\rho}$ for different target values.

Stage 4: We apply *bootstrapping* to estimate a (say) 90% simultaneous confidence region for the two performance measures of each of the four ‘optimized’ PPCSs. The seminal book on bootstrapping (outside simulation and RA) is Efron and Tibshirani (1993). [Another monograph is Shao and Tu (1995); a short introduction is Mooney and Duval (1993).] Bootstrapping in simulation raises an interesting question: instead of using the computer to generate responses through bootstrapping, the computer may be used to generate more simulation responses. In practice, however, replicating a simulation generally requires much more computer time than bootstrapping a simulation. In our example we bootstrap as follows.

A particular scenario (say) *s* - in the original LHS sample of size *n* = 100 - gives a disaster probability $\hat{\pi}_s$ and an average WIP $\hat{\mu}_s$. (These *n* scenarios together give $\hat{\rho}$ and $\bar{\hat{\mu}}$.) Bootstrapping means that the bivariate output $(\hat{\pi}_s, \hat{\mu}_s)$ of scenario *s* is resampled randomly with replacement (technically, the output receives a multinomially distributed weight w_s with values 0, 1, 2, ..., *n* such that $\sum w_s = n$). This yields a new value for the two robustness criteria $\hat{\rho}$ and $\bar{\hat{\mu}}$. To estimate the distribution of these two criteria, we repeat this bootstrap sampling *b* times (for example, 200 times). This gives Figures 8 and 9 (parts a and b), which will be discussed below.

To estimate a 90% simultaneous confidence region for the two estimated robustness criteria of a specific PPCS, we hypothesize that the bootstrapped variables are *bivariate normal*. To test this hypothesis, we apply Johnson and Wichern (1992, pp. 158-164), as follows. Denote the sampled multi-variate observations by X_j with j = 1, ..., b; in our example x equals $(\bar{\hat{\mu}}, \hat{\rho})$. Define the squared generalized distance as

$$D_j^2 = (\boldsymbol{X}_j - \bar{\boldsymbol{X}})'\boldsymbol{S}^{-1}(\boldsymbol{X}_j - \bar{\boldsymbol{X}}) \; \textit{with } j = 1, ..., b \tag{1}$$

with bold letters for matrices and vectors, and the classic estimators $\overline{\boldsymbol{X}} = \sum_{j=1}^{b} \boldsymbol{X}_j$ and $\boldsymbol{S} = \sum_{j=1}^{b} (\boldsymbol{X}_j - \overline{\boldsymbol{X}})(\boldsymbol{X}_j - \overline{\boldsymbol{X}})'/(b - 1)$. Then the hypothesis of υ-variate normality (here υ = 2) is not rejected if

(i) roughly half of the d_j^2 are less than the 50% quantile of the chi-square statistic with υ degrees of freedom (say) $\ ^2(0.50)$, and

(ii) a plot of the *b* ordered d_j^2 versus the *b* quantiles $\ ^2([j - 0.5)/b])$ gives a straight line.

Applying Johnson and Wichern (1992, p. 189), we derive a 1 -   confidence region for the two bootstrapped robustness criteria (say)   = (η, ρ):

$$b(\overline{\boldsymbol{X}} - \quad)'\boldsymbol{S}^{-1}(\overline{\boldsymbol{X}} - \quad) \le [2(b - 1)/(b - 2)]f_{2;\, b - 2}(1 - \quad) \qquad (2)$$

where $f_{2,\, b - 2}(1 - \quad)$ denotes the 1 -   quantile (upper   point) of the F-statistic with degrees of freedom 2 and *b* - 2.

Stage 5: We consider our methodology as a *decision support system* (DSS); that is, the methodology does not make the final selection of a particular PPCS. Instead, short-term risk in terms of fill rate versus long-term costs in terms of WIP are presented to the managers so they can select a particular PPCS that fits their specific risk attitude.

Moreover, *risk management* may be supported as follows. Once we have finished the RA, we may try to identify the important inputs. In RA it is customary to make scatter plots per input; two examples are given in Figure 4 (with $n = 100$ points per plot). We find that the most important parameter (with first-order and higher-order effects) is the demand rate: part (a) suggests that a low demand interarrival time increases the disaster probability; part (b) indicates that changes in the average processing time at the last production stage do not have a systematic effect on the disaster probability. The first conclusion makes sense: high demand tends to decrease the fill rate. This conclusion support our model's credibility (Details on the statistical analysis of scatter plots to identify important factors in large-scale simulations can be found in Kleijnen and Helton 1999.)

INSERT Figure 4: Scatter plot of (a) an important, and (b) an unimportant input parameter

4. Illustration and results

To illustrate our methodology, we use the example in Bonvik et al. (1997)), so our example is not biased to favor our methodology. The base scenario for our discrete-event simulation model uses the same *assumptions* as Bonvik et al. did (see stage 2 of our methodology in §3). These assumptions are the following.

Delivery of raw materials is continuous and infinite. Movements of products and cards are instantaneous. Inventory value is constant over the production line (value added is ignored). Processing times at each station are lognormal with a mean of 0.98 (minutes) and a standard deviation of 0.02. Demand interarrival time is constant, namely 1 (the system is feeding an assembly line that is modeled as a deterministic demand process consuming one part per minute.) If no finished product is available, then demand is lost; so it is essential to have a fill rate close to 100%. Actually the *fill-rate target* is 99.9%. Machines have times between failures and repair times that are exponentially distributed with means of 1,000 and 3 respectively. The runlength is 240,000 simulated time units, of which the first 9,600 time units are estimated to show transient behavior so statistics collected during this transient period are discarded. (Our RA, however, uses shorter runlength: see §2.2.)

Bonvik et al. compare Kanban, Conwip, and Hybrid (besides two more systems that we do not examine, namely, so-called minimal blocking and Base stock). They try to achieve the fill-rate target with minimal WIP. Their 'optimized' Hybrid outperforms Kanban (Hybrid's advantage grows as the demand rate increases). Conwip's performance is between Kanban's and Hybrid's. We verify our simulation model by comparing its simulated output with results in Bonvik et al.; indeed we succeed in reproducing the estimated performance for their three optimized PPCSs. Moreover, we do reproduce the results for Generic in Gaury et al. (1999). Actually, our Generic outperforms their Hybrid, when we use their performance criteria.

Now we proceed to RA (stage 3 in §3). In our example we have no information on the likelihood of the various scenarios, so we assume that all scenarios are equally likely. Hence we use a uniform (prior) distribution per input, and assume independent inputs. There are the following 17 inputs: the processing time's mean and variance, MTBF (mean time between failures) and MTR (mean time to repair) per stage, and the demand rate. In our RA we vary these 17 inputs over a range of ±5% around their base values.

Figure 3 has already shown an example of our RA output, which we now discuss further. This figure concerns Kanban optimized for the base scenario, given a fill-rate threshold of 97% (c_y = 0.97). The results are quite surprising, we think. The disaster density function turns out to have a *bathtub* shape: under many scenarios no disasters occur (left-hand side in part a: $\hat{\pi}$ = 0; ample line capacity); under many other scenarios the optimized Kanban system never gives the target fill rate of 97% (right hand: $\hat{\pi}$ = 1; lack of capacity). Part c shows that - unlike we conjectured - low disaster probabilities do not necessarily go together with high WIPs: the coefficient of determination R^2 is only 0.0121.

What happens to the disaster probability $\hat{\pi}$, when we change the *number of cards*? For Conwip the optimal number of cards under the base scenario is 15. Figure 5 illustrates the effect of increasing the number of cards. Of course, the probability of zero disaster probability is highest when the number of cards is largest (see c = 50; left-hand side). Nevertheless, even with this number of cards, 18 out of 100 scenarios lead to a disaster probability of 1 (right-hand side).

INSERT Figure 5: Effect of number of cards c on estimated disaster probability $\hat{\pi}$, in Conwip, estimated from n = 100 scenarios

Further, from the same simulation run we estimate the ‘disaster’ probability for several fill-rate *target values* (c_y), namely 95%, 97%, and 99.9%. Obviously, the lower the threshold is, the higher is the probability of no ‘disaster’: see Figure 6, left-hand side.

INSERT Figure 6: Effect of fill-rate target c_y on estimated disaster probability $\hat{\pi}$

Figures 3 and 5 have already illustrated the bathtub shape of the estimated density function of the estimated disaster probability $\hat{\pi}$ in Kanban and Conwip respectively. However, to compare the four PPCSs, we prefer the *cumulated* density functions; see Figure 7 (which uses a fill-rate target value c_y of 0. 97). This figure shows that - whatever PPCS is used - some scenarios certainly give a disaster (right-hand side). Most disaster scenarios are characterized by mean demands that exceed production rates (remember Figure 4).

INSERT Figure 7: Estimated density function of estimated disaster probability $\hat{\pi}$ and average WIP $\hat{\mu}$ for four PPCSs

Though the distribution functions in Figure 7 might enable management to select a PPCS, we prefer to characterize each function through a single number, namely the average $\overline{\hat{\mu}}$ for WIP, and the probability $\hat{\rho}$ for fill rate (with threshold c_{π} = 0.9). These two robustness criteria may be used by management to select a particular PPCS based on their personal trade-off between these criteria.

However, the analysts should also quantify the uncertainty of their numerical results; that is, they should provide confidence intervals. First we use bootstrapping to obtain Figure 8.

INSERT Figure 8: Bootstrapped joint density function of the two robustness criteria ($\overline{\hat{\mu}}$, $\hat{\rho}$) for Generic

Next we estimate a confidence interval. The estimated marginal density functions of the two individual criteria suggest that normality holds (even for the estimated probability $\hat{\rho}$); see the two upper parts of Figure 9. The lowest part corresponds with the test defined in equation (1). This test suggests that the normality assumption may indeed be used for Generic. For simplicity's sake we do not test normality for the other three PPCSs of this example, but simply assume that this assumption also holds for these PPCSs.

INSERT Figure 9: Testing normality of the bootstrapped $\overline{\hat{\mu}}$ and $\hat{\rho}$ for Generic

A 1 - confidence region for the two bootstrapped criteria follows from equation (2). We might apply this formula to each of the four PPCSs with a type-I error rate of α. However, our selection of a PPCS depends on all four confidence regions simultaneously. Therefore we use *Bonferroni*'s inequality: we replace α by α /4, which keeps the overall type-I error rate below α. Taking α = 0.10 yields Figure 10. This figure shows that - though Bonferroni's inequality is conservative - our example gives four non-overlapping confidence intervals for the WIP criterion $\overline{\hat{\mu}}$. However, the differences for the service criterion $\hat{\rho}$ are not significant.

INSERT Figure 10: Estimated 90% simultaneous confidence regions for the two criteria ($\overline{\hat{\mu}}$, $\hat{\rho}$) for the four PPCSs

Any reasonable risk attitude implies that managers prefer low WIP, provided the risk is acceptable. Figure 10 suggests that Hybrid dominates the other PPCSs: it minimizes both criteria. Nevertheless, since Hybrid requires the implementation of both Kanban and Conwip, managers might prefer Kanban: the latter PPCS is easier to implement in practice, and only slightly increases both criteria values. Obviously, Conwip gives excessive WIP (Conwip has a single control loop) - without decreasing the risk of a ’disaster’. Generic gives a WIP that is relatively high compared with Hybrid and Kanban, while it does not decrease risk. However, when risk were ignored, then the ranking from best to worst PPCS would be: Generic, Hybrid, Conwip, Kanban; for details we refer to Gaury et al. (1999). *So risk considerations do make a difference.*

The performance results of the four PPCSs differ because these PPCSs have different control loops with different card numbers. Hence, the PPCSs have different total WIPs, and different WIP allocations along the production line.

5. Conclusions

We emphasized that in production control, managers should realize that the actual performance of their system varies with the environmental conditions or *scenarios* that turn up in practice. Traditionally, however, performance is predicted for one particular base scenario only! Technically, our view implies that performance measures have a joint probability distribution that depends on the prior distribution of the inputs that together form the scenario.

To characterize this output distribution, we emphasized the *short-term* view, besides the traditional long-term view. More specifically, we apply these two views to two widely used performance measures, namely WIP and fill rate: for the long-term we used $\mu = E(W)$, the expected average WIP; for the short-term we proposed $\pi = P(Y < c_y)$, the probability of Y (fill rate per shift) dropping below c_y (a prespecified manager’s threshold).

To implement these measures in managerial practice, we developed a *new methodology* for performance analysis in operations management. This methodology combines the techniques of discrete-event simulation, heuristic optimization, risk analysis (RA), and bootstrapping. The methodology proceeds in stages that are detailed in this paper.

RA has never been used in production management (to the best of our knowledge). We apply RA to estimate the distributions of the two performance criteria under different scenarios. Each of these two distributions may be characterized through a single measure. We proposed $\bar{\bar{\mu}}$ - average WIP per simulation run, averaged over scenarios - and $\hat{\rho}$ - probability of

$\hat{\pi}$ exceeding the threshold c_π, under different scenarios.

Besides RA, we applied *bootstrapping* - a technique not much applied in production management - to obtain confidence intervals for the estimated PPCS robustness measures, $\overline{\hat{\mu}}$ and $\hat{\rho}$. Using these intervals, management may select a PPCS that fits their risk attitude.

We illustrated our methodology through *Bonvik et al. (1997)'s example*. They, however, ignored short-term robustness - and did not consider Generic. We found that Hybrid is best when risk is not ignored; otherwise, Generic is best. So we conclude that *risk considerations do make a difference*. Selecting the appropriate PPCS may affect a manager's survival of bad times! Therefore we conclude that methods for performance analysis in operations management should account for robustness when recommending a specific PPCS.

How *general* are the conclusions of our example? In practice, the production system may be different from our's; for example, the system may be a job shop instead of a production line. To manage this system, not a PPCS but a different control system may be used (for example, one of the proprietary systems sold by Baan, i2, SAP, etc.). In addition, managers may assume scenarios with associated input distributions that differ from the ones we used in our example. Finally, they may select other cost functions and have different risk attitudes. Therefore we hope that our article will stimulate other researchers and practitioners to apply our methodology to their specific systems. The ensuing extensive experimental studies may lead researchers to refine our methodology, and may provide managers with general conclusions about the robustness of production control systems!

References

Abate, J. and W. Whitt (1994), Transient behavior of the M/G/1 workload process. *Operations Research*, 42, no. 4, pp. 750-764

Avramidis, A. and J.R. Wilson (1998), Correlation-induction techniques for estimating quantiles in simulation experiments. *Operations Research*, 46, no. 4, pp. 574-591

Balson, W.E., J.L. Welsh and D.S. Wilson (1992), Using decision analysis and risk analysis to manage utility environmental risk. *Interfaces*, 22, no. 6, pp. 126-139

Barton, R.R. and L.W. Schruben (1993), Uniform and bootstrap resampling of empirical distributions. *Proceedings of the 1993 Winter Simulation Conference,* edited by G.W. Evans, M. Mollaghasemi, E.C. Russell, and W.E. Biles, IEEE, Piscataway, N.J., pp.

503-508

Benjamin, P.C., M. Erraguntla, and R.J. Mayer (1995), Using simulation for robust system design. *Simulation,* 65, no. 2, pp. 116-127

Bennett, J.C., G.A. Bohoris, E.M. Aspinwall, and R.C. Hall (1996), Risk analysis techniques and their application to software development. *European Journal of Operational Research,* 95, no. 3, pp. 467-475

Bonvik, A.M., C.E. Couch, and S.B. Gershwin (1997). A comparison of production-line control mechanisms, *International Journal of Production Research,* 35, 3, 789-804

Breeding R.J. et al. (1992), Summary description of the methods used in the probabilistic risk assessments for NUREG-1150. *Nuclear Engineering and Design,* 135, pp. 1-27

Buzacott, J. and J. Shanthikumar (1993), *Stochastic models of manufacturing systems*. Prentice-Hall, Englewood Cliffs, New Jersey

Cheng, R.C.H. and W. Holland (1997), Sensitivity of computer simulation experiments to errors in input data *Journal Statistical Computation and Simulation,* 57, numbers 1-4, pp. 219-242

Chick, S.E. (1997), Bayesian analysis for simulation input and output, *1997 Proceedings of the Winter Simulation Conference,* edited by S. Andradóttir, K., Healy, D. Withers, and B. Nelson, IEEE, Piscataway, N.J., pp. 253-260

Chu, C.-H. and Shih, W.-L. (1992), Simulation studies in JIT production*, International Journal of Production Research,* 30, 11, pp. 2573-2586

Efron, B. and R.J. Tibshirani (1993), *An introduction to the bootstrap*. Chapman & Hall, New York

Gaury, E.G.A., H. Pierreval, and J.P.C. Kleijnen (2000), An evolutionary approach to select a pull system among Kanban, Conwip, and Hybrid. *Journal of Intelligent Manufacturing* (in press)

--- (1999), New species of hybrid pull systems. CentER Discussion Paper, no. 9831 (submitted for publication)

Goldrat, E.M. and Fox, R.E. (1986), *The race,* North River Press. New York

Granger Morgan M. and M. Henrion (1990), *A guide to dealing with uncertainty in quantitative risk and policy analysis*. Cambridge University Press

Haverkort, B.R. and A.M.H. Meeuwissen (1995), Sensitivity and uncertainty analysis of Markov-reward models. *IEEE Transactions on Reliability,* 44, no. 1, pp. 147-154

Heidelberger, P. (1995), Fast simulation of rare events in queueing and reliability models. *ACM Transactions on Modeling and Computer Simulation*, 5, no. 1, pp. 43-85

Helton, J.C. (1997), Uncertainty and sensitivity analysis in the presence of stochastic and subjective uncertainty. *Journal Statistical Computation and Simulation*, 57, pp. 3-76

Helton, J.C., D.R. Anderson, M.G. Marietta, and R.P. Rechard (1997), Performance assessment for the waste isolation pilot plant: from regulation to calculation for 40 CFR 191.13. *Operations Research*, 45, no. 2, pp. 157-177

Johnson, R.A. and D.W. Wichern (1992), *Applied multivariate statistical analysis*. Prentice-Hall International, Englewood Cliffs, New Jersey

Kleijnen, J.P.C. (1998), Experimental design for sensitivity analysis, optimization, and validation of simulation models. Chapter 6 in: *Handbook of Simulation*, edited by J. Banks, Wiley, New York, pp. 173-223

--- (1994), Sensitivity analysis versus uncertainty analysis: when to use what? *Predictability and nonlinear modelling in natural sciences and economics*, edited by J Grasman and G. van Straten, Kluwer, Dordrecht, the Netherlands, pp.322-333

--- (1983), Risk analysis and sensitivity analysis: antithesis or synthesis. *Simuletter*, 14, no. 1-4, pp. 64-72

--- and J.C. Helton (1999), Statistical analyses of scatter plots to identify important factors in large-scale simulations, 1: review and comparison of techniques. *Reliability Engineering and Systems Safety*, 65, no. 2, pp. 147-185

Krumm, F.V. and C.F. Rolle (1992), Management and application of decision and risk analysis in Du Pont. *Interfaces*, 22, no. 6, pp. 84-93

Lim, J., K. Kim, B. Yum, and H. Hwang (1996), Determination of an optimal configuration of operating policies for direct-input-output manufacturing systems using the Taguchi method. *Computers Industrial Engineering*, 31, no.3/4, pp. 555-560

Lin, L. en Cochran, J.K. (1990), Metamodels of production line transient behaviour for sudden machine breakdowns. *International Journal of Production Research*, 28, no. 10, pp. 1791-1806

McKay, M.D., R.J. Beckman, and W.J. Conover (1979), A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21, no. 2, pp. 239-245

Mayer, R.J. and P.C. Benjamin (1992),Using the Taguchi paradigm for manufacturing system design using simulation experiments. *Computers and Industrial Engineering*, 22, no. 2, pp. 195-209

Moeeni, F., S.M. Sanchez, and A.J. Vakharia (1997), A robust design methodology for Kanban system design. *International Journal Production Research*, 35, no. 10, pp. 2821-2838

Mooney, C.Z. and R.D. Duval (1993), *Bootstrapping: a nonparametric approach to statistical inference*. Sage Publications, Newbury Park, California 91320

Morrice, D.J. and I.R. Bardhan (1995), A weighted least squares approach to computer simulation factor screening. *Operations Research*, 43, no. 5, pp. 792-806

Muppala, J.K., M Malholtra, and K.S. Trivedi (1996), Markov dependability models for complex systems: analysis techniques. *Reliability and Maintenance of Complex Systems*, edited by S. Ozekici et al., Springer-Verlag, Berlin, pp. 442-486

Myers, R.H. (1999), Response surface methodology - current status and future directions. (Including Discussion.) *Journal of Quality Technology*, 31, no. 1, pp. 30-74.

Nair, V.N. editor (1992), Taguchi's parameter design: a panel discussion. *Technometrics*, 34, no. 2, pp. 127-161

Schruben, L.W. and V.J. Cogliano (1987), An experimental procedure for simulation response surface model identification. *Communications ACM*, 30, no. 8, pp. 716-730

Shao, J and D. Tu (1995), *The jackknife and bootstrap*. Springer-Verlag, New York

Spearman, M.L., D.L. Woodruff, and W.J. Hopp (1990). CONWIP: a pull alternative to Kanban, *International Journal of Production Research*, 28, 5, 879-894

Sugiyama, S.O. and J.W. Chow (1997), @Risk, Riskview and BestFit. *OR/MS Today*, 24, no. 2, pp. 64-66

Tan, B. (1999), Variance of the output as a function of time: production line dynamics. . *European Journal of Operational Research*, 117, no. 3, p. 470-484

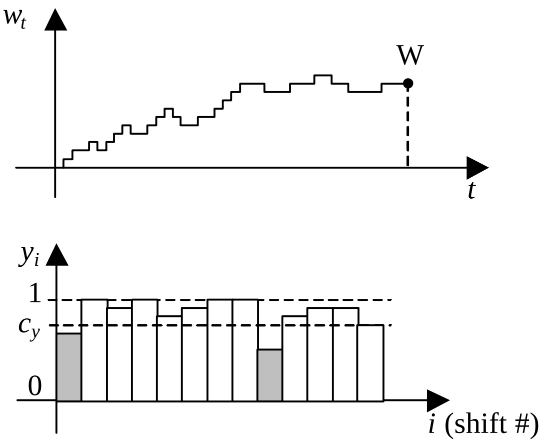


Figure 1. Simulated w_t (WIP at time t) and y_i (fill rate of shift i)

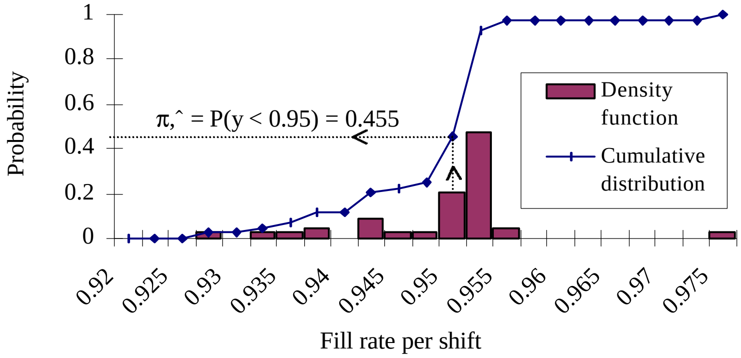


Figure 2. Density function of y (fill rate per shift) and disaster probability $\pi = P_y(y < 0.95)$: a simulation example

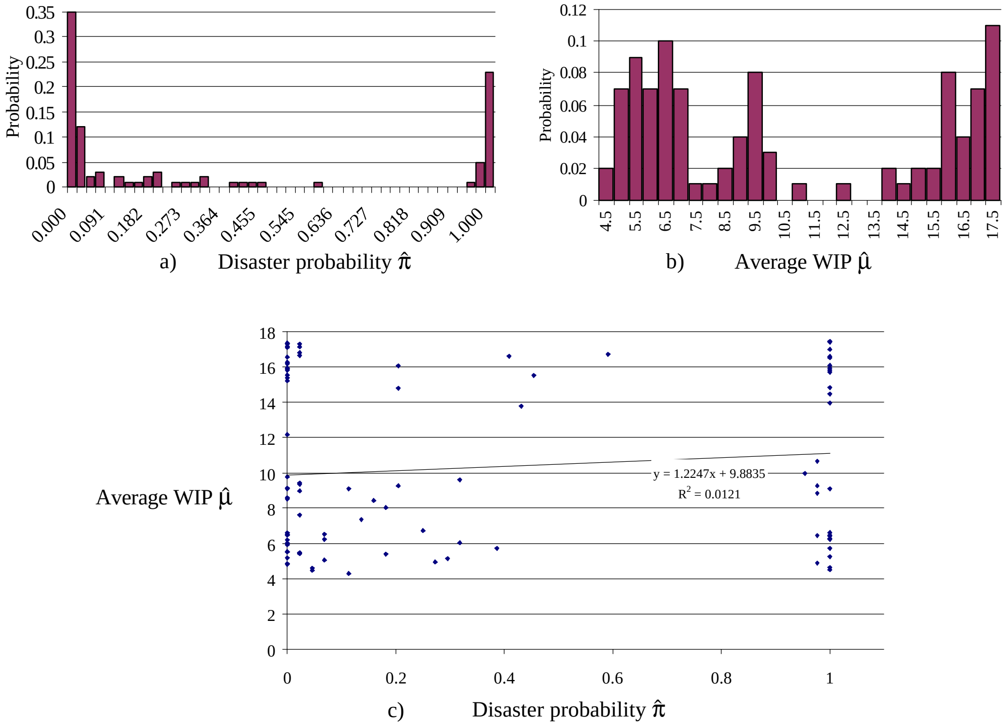
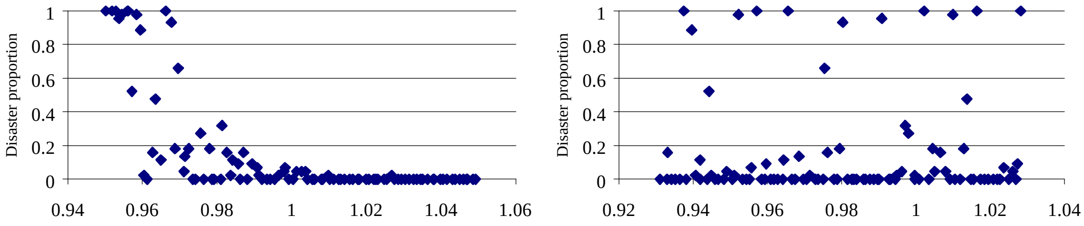


Figure 3. Estimated density function of estimated disaster probability Π,ˆ and average WIP M,ˆ - for Kanban optimized given a 97% fill rate target: Bonvik *et al*'s four-stage example



(a) Demand interarrival time

(b) Average processing time at last stage

Figure 4. Scatter plot of (a) an important and (b) an unimportant input parameter

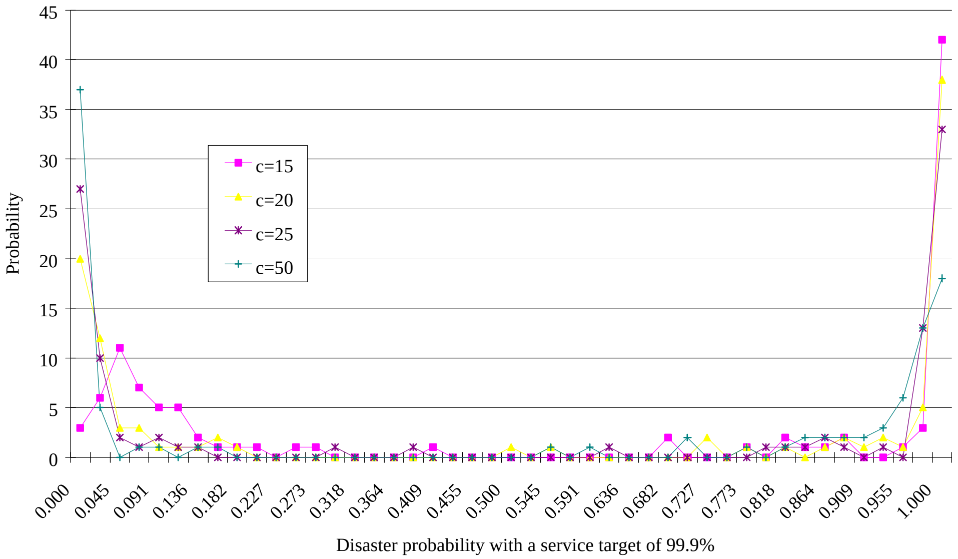


Figure 5. Effect of the number of cards *c* on the estimated disaster probability π,ˆ, in Conwip, estimated from *n* = 100 scenarios

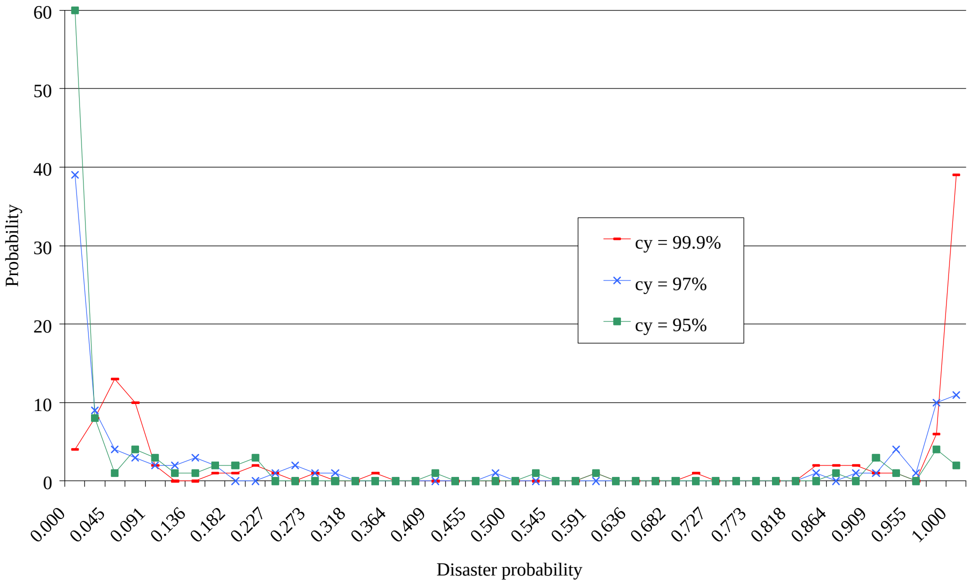


Figure 6. Effect of fill rate target c_y on estimated disaster probability π,ˆ

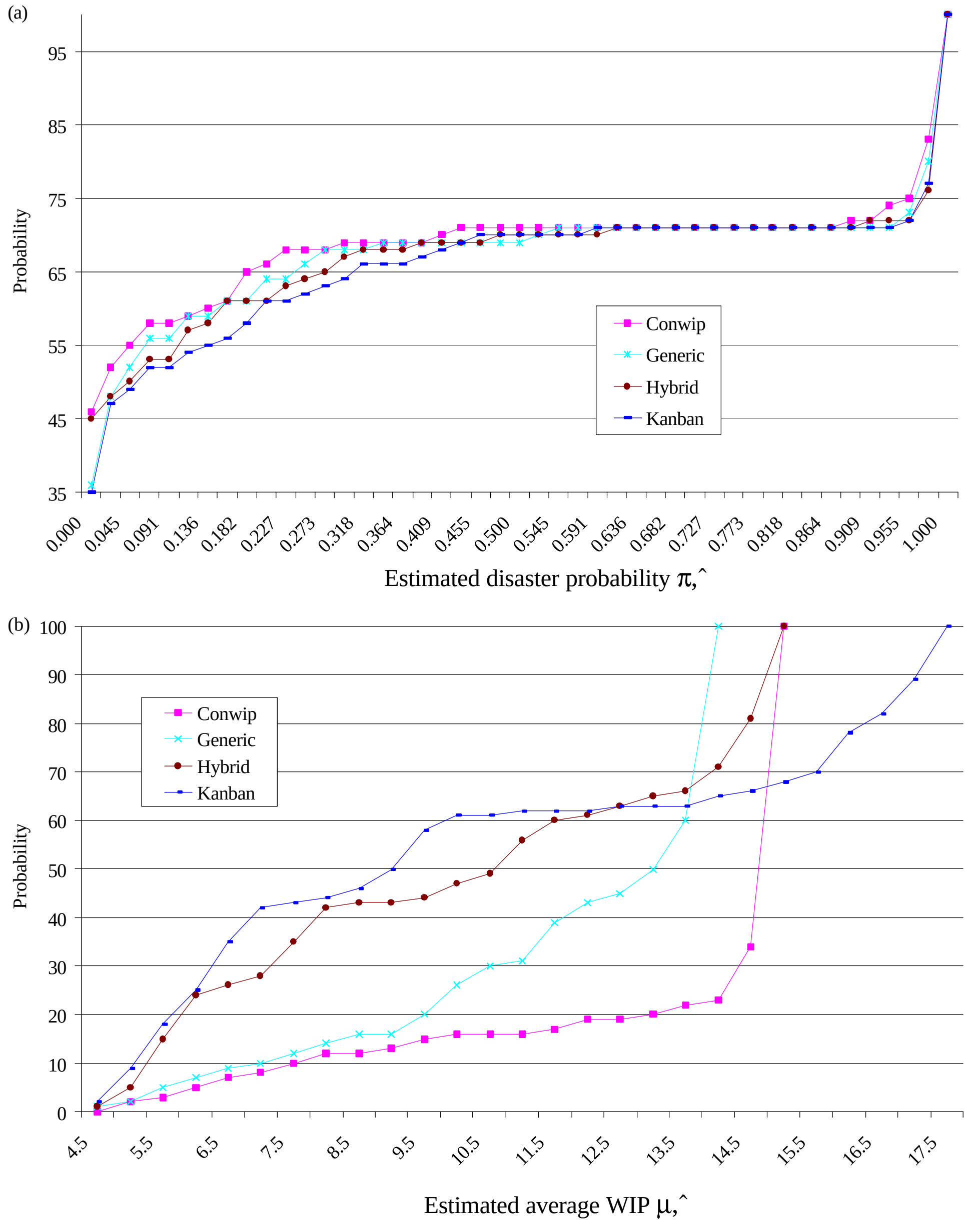


Figure 7. Estimated density function of estimated disaster probability π,ˆ and average WIP μ,ˆ for four PPCSs

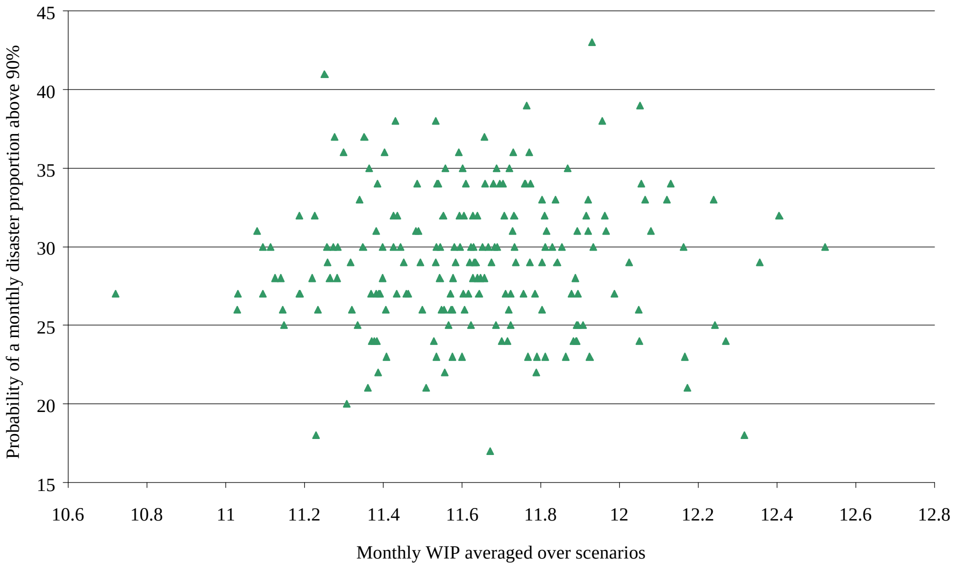


Figure 8. Bootstrapped joint density function of the two robustness criteria (μ,ˆ,¯, ρ,ˆ) for Generic

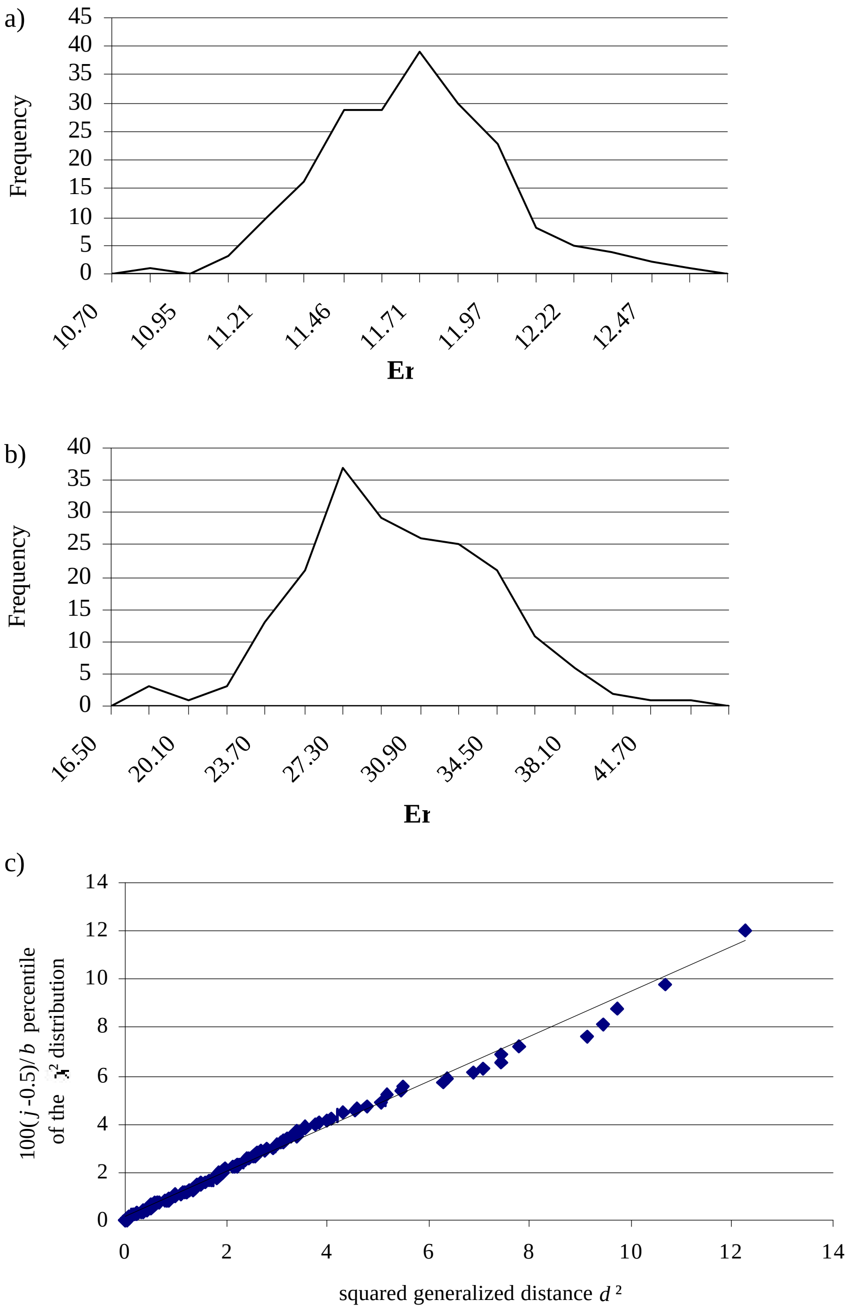


Figure 9. Testing normality of the bootstrapped μ,ˆ, ¯ and ρ,ˆ for Generic

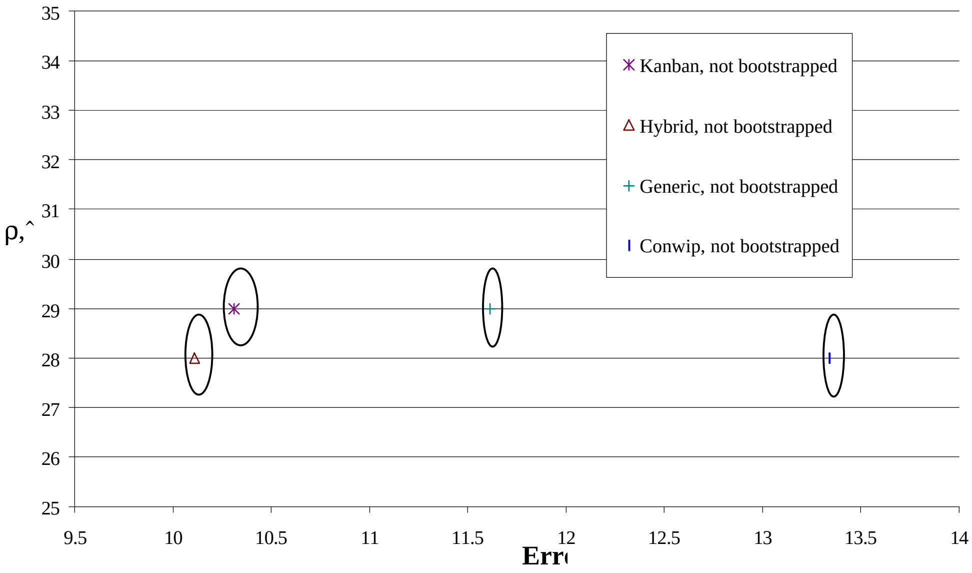


Figure 10. Estimated 90% simultaneous confidence regions for the two criteria (μ,ˆ, ¯ , ρ,ˆ) for the four PPCSs