Extraction of Maximum Support Rules for the Root Cause Analysis

Tomas Hrycej[1] and Christian Manuel Strobel[2]

[1] Formerly with DaimlerChrysler Research, Ulm, Germany, tomas_hrycej@yahoo.de
[2] University of Karlsruhe (TH), Karlsruhe, Germany, mstrobel@statistik.uni-karlsruhe.de

Summary. Rule extraction for root cause analysis in manufacturing process optimization is an alternative to traditional approaches to root cause analysis based on process capability indices and variance analysis. Process capability indices alone do not allow to identify those process parameters which have the major impact on quality since these indices are only based on measurement results and do not consider the explaining process parameters. Variance analysis is subject to serious constraints concerning the data sample used in the analysis. In this work a rule search approach using Branch and Bound principles is presented, considering both the numerical measurement results and the nominal process factors. This combined analysis allows to associate the process parameters with the measurement results and therefore to identify the main drivers for quality deterioration of a manufacturing process.

1 Introduction

An important group of intelligent methods is concerned with discovering interesting information in large data sets. This discipline is generally referred to as Knowledge Discovery or Data Mining.

In the automotive domain, large data sets may arise through on-board measurements in cars. However, more typical sources of huge data amounts are in vehicle, aggregate or component manufacturing process. One of the most prominent applications is the manufacturing quality control, which is the topic of this chapter.

Knowledge discovery subsumes a broad variety of methods. A rough classification may be into:

- Machine learning methods
- Neural net methods
- Statistics

This partitioning is neither complete nor exclusive. The methodical frameworks of machine learning methods and neural nets have been extended by aspects covered by classical statistics, resulting in a successful symbiosis of these methods.

An important stream within the machine learning methods is committed to a quite general representation of discovered knowledge: the rule based representation. A rule has the form $x \rightarrow y$, x and y being, respectively the *antecedent* and the *consequent*. The meaning of the rule is: if the antecedent (which has the form of a logical expression) is satisfied, the consequent is sure or probable to be true.

The discovery of rules in data can be simply defined as a search for highly informative (i.e., interesting from the application point of view) rules. So the most important subtasks are:

1. Formulating the criterion to decide to which extent a rule is interesting
2. Using an appropriate search algorithm to find those rules that are the most interesting according to this criterion

The research of the last decades has resulted in the formulation of various systems of interestingness criteria (e.g., support, confidence or lift), and the corresponding search algorithms.

T. Hrycej and C.M. Strobel: *Extraction of Maximum Support Rules for the Root Cause Analysis*, Studies in Computational Intelligence (SCI) **132**, 89–99 (2008)
www.springerlink.com

However, general algorithms may miss the goal of a particular application. In such cases, dedicated algorithms are useful. This is the case in the application domain reported here: the root cause analysis for process optimization.

The indices for quality measurement and our application example are briefly presented in Sect. 2. The goal of the application is to find manufacturing parameters to which the quality level can be attributed. In order to accomplish this, rules expressing relationships between parameters and quality need to be searched for. This is what our rule extraction search algorithm based on Branch and Bound principles of Sect. 3 performs. Section 5 shows results of our comparative simulations documenting the efficiency of the proposed algorithm.

2 Root Cause Analysis for Process Optimization

The quality of a manufacturing process can be seen as the ability to manufacture a certain product within its specification limits U, L and as close as possible to its target value T, describing the point where its quality is optimal. A deviation from T generally results in quality reduction, and minimizing this deviation is crucial for a company to be competitive in the marketplace. In literature, numerous *process capability indices* (PCIs) have been proposed in order to provide a unitless quality measures to determine the performance of a manufacturing process, relating the preset specification limits to the actual behavior [6].

The behavior of a manufacturing process can be described by the process variation and process location. Therefore, to assign a quality measure to a process, the produced goods are continuously tested and the performance of the process is determined by calculating its PCI using the measurement results. In some cases it is not feasible to test/measure all goods of a manufacturing process, as the inspection process might be too time consuming, or destructive. Only a sample is drawn, and the quality is determined upon this sample set. In order to predict the future quality of a manufacturing process based on the past performance, the process is supposed to be stable or in control. This means that both process mean and process variation have to be, in the long run, in between pre-defined limits. A common technique to monitor this is control charts, which are an essential part of the Statistical Process Control.

The basic idea for the most common indices is to assume the considered manufacturing process follows a normal distribution and the distance between the upper and lower specification limit U and L equals 12σ. This requirement implies a lot fraction defective of the manufacturing process of no more than 0.00197 ppm $\cong$ 0% and reflects the widespread *Six-Sigma* principle (see [7]). The commonly recognized *basic* PCIs C_p, C_{pm}, C_{pk} and C_{pmk} can be summarized by a superstructure first introduced by Vännman [9] and referred to in literature as $C_p(u, v)$

$$C_p(u,v) = \frac{d - u|\mu - M|}{3\sqrt{\sigma^2 + v(\mu - T)^2}}, \tag{1}$$

where σ is the process standard deviation, μ the process mean, $d = (U-L)/2$ tolerance width, $m = (U+L)/2$ the mid-point between the two specification limits and T the target value. The *basic* PCIs can be obtained by choosing u and v according to

$$\begin{array}{llll} C_p \equiv C_p(0,0); & C_{pk} & \equiv C_p(1,0) \\ C_{pm} \equiv C_p(0,1); & C_{pmk} & \equiv C_p(1,1). \end{array} \tag{2}$$

The estimators for these indices are obtained by substituting μ by the sample mean $\bar{X} = \sum_{i=1}^{n} X_i/n$ and σ by the sample variance $S^2 = \sum_{i=1}^{n}(X_i - \bar{X})^2/(n-1)$. They provide stable and reliable point estimators for processes following a normal distribution. However, in practice, normality is hardly encountered. Consequently the basic PCIs as defined in (1) are not appropriate for processes with non-normal distributions. What is really needed are indices which do not make assumptions about the distribution, in order to be useful for measuring quality of a manufacturing process

$$C'_p(u,v) = \frac{d - u|m - M|}{3\sqrt{[\frac{F_{99.865} - F_{0.135}}{6}]^2 + v(m-T)^2}}. \tag{3}$$

In 1997, Pearn and Chen introduced in their paper [8] a non-parametric generalization of the PCIs superstructure (1) in order to cover those cases in which the underlying data does not follow a Gaussian distribution. The authors replaced the process standard deviation σ by the 99.865 and 0.135 quantiles of the empiric distribution function and μ by the median of the process. The rationale for it is that the difference between the $F_{99.865}$ and $F_{0.135}$ quantiles equals again 6σ or $C'_p(u,v) = 1$, under the standard normal distribution with $m = M = T$. As an analogy to the parametric superstructure (1), the special non-parametric PCIs C'_p, C'_{pm}, C'_{pk} and C'_{pk} can be obtained by applying u and v as in (2).

Assuming that the following assumptions hold, a class of non-parametric process indices and a particular specimen thereof can be introduced: Let $\mathbf{Y} : \Omega \to \mathbb{R}$ be a random variable with $\mathbf{Y}(\omega) = (Y^1, \ldots, Y^m) \in \mathcal{S} = \{S^1 \times \cdots \times S^m\}$, $S^i \in \{s^i_1, \ldots, s^i_{m_i}\}$ where $s^i_j \in \mathbb{N}$ describe the possible influence variables or process parameters. Furthermore, let $X : \Omega \to \mathbb{R}$ be the corresponding measurement results with $X(\omega) \in \mathbb{R}$. Then the pair $\mathcal{X} = (X, \mathbf{Y})$ denotes a manufacturing process and a class of *process indices* can be defined as

Definition 1. *Let $\mathcal{X} = (X, \mathbf{Y})$ describe a manufacturing process as defined above. Furthermore, let $f(x, y)$ be the density function of the underlying process and $w : \mathbb{R} \to \mathbb{R}$ an arbitrary measurable function. Then*

$$Q_{w,\mathcal{X}} = E(w(x)|\mathbf{Y} \in \mathcal{S}) = \frac{E(w(x)\mathbb{1}_{\{\mathbf{Y}\in\mathcal{S}\}})}{P(\mathbf{Y} \in \mathcal{S})} \tag{4}$$

defines a class of process indices.

Obviously, if $w(x) = x$ or $w(x) = x^2$ we obtain the first and the second moment of the process, respectively, as $P(\mathbf{Y} \in S) = 1$. However, to determine the quality of a process, we are interested in the relationship between the designed specification limits U, L and the process behavior described by its variation and location. A possibility is to choose the function $w(x)$ in such way that it becomes a function of the designed limits U and L. Given a particular manufacturing process $\mathcal{X}$ with $(x_i, \mathbf{y}_i), i = 1, \ldots, n$ we can define

Definition 2. *Let $\mathcal{X} = (X, Y)$ be a particular manufacturing process with realizations $(x_i, \mathbf{y}_i), i = 1, \ldots, n$ and U, L be specification limits. Then, the* Empirical Capability Index *(E_{ci}) is defined as*

$$\hat{E}_{ci} = \frac{\sum_{i=1}^{n} \mathbb{1}_{\{L\le x_i\le U\}}\mathbb{1}_{\{\mathbf{y}_i\in\mathcal{S}\}}}{\sum_{i=1}^{n} \mathbb{1}_{\{\mathbf{y}_i\in\mathcal{S}\}}}. \tag{5}$$

By choosing the function $w(x)$ as the identity function $\mathbb{1}_{(L\le x\le U)}$, the E_{ci} measures the percentage of data points which are within the specification limits U and L. A disadvantage is that for processes with a relatively good quality, it may happen that all sampled data points are within the Six-Sigma specification limits (i.e., $C'_p > 1$), and so the sample E_{ci} becomes one. To avoid this, the specification limits U and L have to be relaxed to values realistic for the given sample size, in order to get "further into the sample", by linking them to the behavior of the process. One possibility is to choose empirical quantiles

$$[\bar{L}, \bar{U}] = [F_\alpha, F_{1-\alpha}].$$

The drawback of using empirical quantiles as specification limits is that $\bar{L}$ and $\bar{U}$ do not depend anymore on the actual specification limits U and L. But it is precisely the relation of the process behavior and the designed limits which is essential for determining the quality of a manufacturing process. A combined solution, which on one hand depends on the actual behavior and on the other hand incorporates the designed specification limit U and L can be obtained by

$$[\bar{L}, \bar{U}] = \left[\hat{\mu}_{0,5} - \frac{\hat{\mu}_{0,5} - LSL}{t}, \hat{\mu}_{0,5} + \frac{USL - \hat{\mu}_{0,5}}{t}\right]$$

with $t \in \mathbb{R}$ being a adjustment factor. When setting $t = 4$ the new specification limits incorporate the *Six-Sigma* principle, assuming the special case of a centralized normally distributed process.

As stated above, the described PCIs only provide a quality measure but do not identify the major influence variables responsible for poor or superior quality. But knowing these factors is necessary to continuously

Table 1. Measurement results and process parameters for the optimization at a foundry of an automotive manufacturer

Result	Tool	Shaft	Location
6.0092	1	1	Right
6.008	4	2	Right
6.0061	4	2	Right
6.0067	1	2	Left
...	...	...	...
6.0076	4	1	Right
6.0082	2	2	Left
6.0075	3	1	Right
6.0077	3	2	Right
6.0061	2	1	Left
6.0063	1	1	Right
6.0063	1	2	Right

improve a manufacturing process in order to produce high quality products in the long run. In practice it is desirable to know, whether there are subsets of influence variables and their values, such that the quality of a process becomes better, if constraining the process by only these parameters. In the following section a non-parametric, numerical approach for identifying those parameters is derived and an algorithm, which efficiently solves this problem is presented.

2.1 Application Example

To illustrate the basic ideas of the employed methods and algorithms, an example is used throughout this paper, including an evaluation in the last section. This example is a simplified and anonymized version of a manufacturing process optimization at a foundry of a premium automotive manufacturer.

In Table 1 an excerpt from the data sheet for such a manufacturing process is shown which is used for further explanations. There are some typical influence variables (i.e., process parameters, relevant for the quality of the considered product) as the used tools, locations and used shafts, each with their specific values for each manufacture specimen. Additionally, the corresponding quality measurement (column "Result") – a geometric property or the size of a drilled hole – is a part of a data record.

2.2 Manufacturing Process Optimization: The Traditional Approach

A common technique to identify significant discrete parameters having an impact on numeric variables like measurement results, is the Analysis of Variance (ANOVA). Unfortunately, the ANOVA technique is only useful if the problem is relatively low dimensional. Additionally, the considered variables ought to have a simple structure and should be well balanced. Another constraint is the assumption that the analyzed data follows a multivariate Gaussian distribution. In most real world applications these requirements are hardly complied with. The distribution of the parameters describing the measured variable is in general non-parametric and often high dimensional. Furthermore, the combinations of the cross product of the parameters are non-uniformly and sparely populated, or have a simple dependence structure. Therefore, the method of Variance Analysis is only applicable in some special cases. What is really needed is a more general, non-parametric approach to determine a set of influence variables responsible for lower or higher quality of a manufacturing process.

3 Rule Extraction Approach to Manufacturing Process Optimization

A manufacturing process X is defined as a pair $(X, \mathbf{Y})$ where $\mathbf{Y}(\omega)$ describes the influence variables (i.e., process parameters) and $X(\omega)$ the corresponding goal variables (measurement results). As we will see later, it is sometimes useful to constrain the manufacturing process to a particular subset of influence variables.

Table 2. Possible sub-processes with support and conditional E_{ci} for the foundry's example

$N_{\mathcal{X}_0}$	$Q_{\mathcal{X}_0}$	Sub-process $\mathcal{X}_0$
123	0.85	Tool in (2,4) and location in (left)
126	0.86	Shaft in (2) and location in (right)
127	0.83	Tool in (2,3) and shaft in (2)
130	0.83	Tool in (1,4) and location in (right)
133	0.83	Tool in (4)
182	0.81	Tool not in (4) and shaft in (2)
183	0.81	Tool not in (1) and location in (right)
210	0.84	Tool in (1,2)
236	0.85	Tool in (2,4)
240	0.81	Tool in (1,4)
244	0.81	Location in (right)
249	0.83	Shaft in (2)
343	0.83	Tool not in (3)

Definition 3. *Let $\mathcal{X}$ describe a manufacturing process as stated in Definition 1 and $\mathbf{Y}_0 : \Omega \to \mathbb{R}$ be a random variable with $\mathbf{Y}_0(\omega) \in \mathcal{S}_0 \subset \mathcal{S}$. Then a sub-process of $\mathcal{X}$ is defined by the pair $\mathcal{X}_0 = (X, \mathbf{Y}_0)$.*

This subprocess constitutes the antecedent (i.e., precondition) of a rule to be discovered. The consequent of the rule is defined by the quality level (as measured by a process capability index) implied by this antecedent. To remain consistent with the terminology of our application domain, we will talk about subprocesses and process capability indices, rather than about rule antecedents and consequents.

Given a manufacturing process $\mathcal{X}$ with a particular realization $(x_i, \mathbf{y}_i), i = 1, \ldots, n$ the support of a sub-process $\mathcal{X}_0$ can be written as

$$N_{\mathcal{X}_0} = \sum_{i=1}^{n} \mathbb{1}_{\{\mathbf{y}_i \in \mathcal{S}_0\}}, \tag{6}$$

and consequently, a conditional PCI is defined as $Q_{\mathcal{X}_0}$. Any of the indices defined in the previous section can be used, whereby the value of the respective index is calculated on the conditional subset $X_0 = \{x_i : \mathbf{y}_i \in \mathcal{S}_0, i = 1, \ldots, n\}$. We henceforth use the notation $\tilde{\mathcal{X}} \subseteq \mathcal{X}$ to denote possible sub-processes of a given manufacturing process $\mathcal{X}$. An extraction of possible sub-process of the introduced example with their support and conditional E_{ci} is given in Table 2.

To determine those parameters which have the greatest impact on quality, an optimal sub-process consisting of optimal influence combinations has to be identified. The first approach could be to maximize $Q_{\tilde{\mathcal{X}}}$ over all sub-processes $\tilde{\mathcal{X}}$ of $\mathcal{X}$. In general, this approach would yield an "optimal" sub-process $\tilde{\mathcal{X}}^*$, which has only a limited support ($N_{\tilde{\mathcal{X}}^*} \ll n$) (the fraction of the cases that meet the constraints defining this subprocess). Such a formal optimum is usually of limited practical value since it is not possible to constrain any parameters to arbitrary values. For example, constraining the parameter "working shift" to the value "morning shift" would not be economically acceptable even if a quality increase were attained.

A better approach is to think in economic terms and to weigh the factors responsible for minor quality, which we want to eliminate, by the costs of removing them. In practise this is not feasible, as tracking the actual costs is too expensive. But it is likely that infrequent influence factors, which are responsible for lower quality are *cheaper* to remove than frequent influences. In other words, sub-processes with high support are preferable over those sub-processes yielding a high quality measure but having a low support.

In most applications, the available sample set for process optimization is small, often having numerous influence variables but only a few measurement results. By limiting ourselves only to combinations of variables, we might get too small a sub-process (having low support). Therefore, we extend the possible solutions to combinations of variables and their values – the search space for optimal sub-processes is spanned by the powerset of the influence parameters $\mathcal{P}(\mathbf{Y})$. The two sided problem, to find the parameter set combining

on one hand an optimal quality measure and on the other hand a maximal support, can be summarized, according to the above notation, by the following optimization problem:

Definition 4.

$$(P_{\mathcal{X}}) = \begin{cases} N_{\tilde{\mathcal{X}}} \rightarrow max \\ Q_{\tilde{\mathcal{X}}} \geq q_{min} \\ \tilde{\mathcal{X}} \subseteq \mathcal{X}. \end{cases}$$

The solution $\tilde{\mathcal{X}}^*$ of the optimization problem is the subset of process parameters with maximal support among those processes, having a quality better than the given threshold q_{min}. Often, q_{min} is set to the common values for process capability of 1.33 or 1.67. In those cases, where the quality is poor, it is preferable to set q_{min} to the unconditional PCIs, to identify whether there is any process optimization potential.

Due to the nature of the application domain, the investigated parameters are discrete which inhibits an analytical solution but allows the use of *Branch and Bound* techniques. In the following section a root cause algorithm (RCA) which efficiently solves the optimization problem according to Definition 4 is presented. To avoid the exponential amount of possible combinations spanned by the cross product of the influence parameters, several efficient cutting rules for the presented algorithm are derived and proven in the next subsection.

4 Manufacturing Process Optimization

4.1 Root Cause Analysis Algorithm

In order to access and efficiently store the necessary information and to apply Branch and Bound techniques, a multi-tree was chosen as representing data structure. Each node of the tree represents a possible combination of the influence parameters (sub-process) and is built on the combination of the parent influence set and a new influence variable and its value(s). Figure 1 depicts the data structure, whereby each node represents the set of sub-processes generated by the powerset of the considered variable(s). Let I, J be to index sets with $I = \{1, \ldots, m\}$ and $J \subseteq I$. Then $\tilde{\mathcal{X}}_J$ denotes the set of sub-processes constrained by the powerset of $Y^j, j \in J$ and arbitrary other variables $(Y^i, i \in I \setminus J)$.

To find the optimal solution to the optimization problem according to Definition 4, a combination of depth-first and breadth-first search is applied to traverse the multitree (see Algorithm 1) using two Branch and Bound principles. The first, an generally applicable principle is based on the following relationship: by

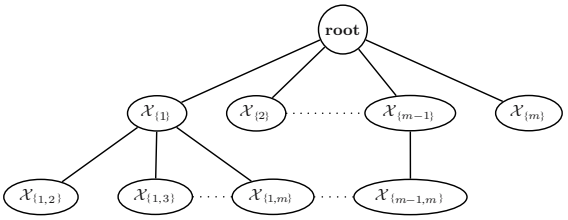

Fig. 1. Data structure for the root cause analysis algorithm

Algorithm 1 Branch & Bound algorithm for process optimization

```
1: procedure TraverseTree(X̃)
2:     X = GenerateSubProcesses(X̃)
3:     for all x̃ ∈ X do
4:         TraverseTree(x̃)
5:     end for
6: end procedure
```

descending a branch of the tree, the number of constraints is increasing, as new influence variables are added and therefore the sub-process support decreases (see Fig. 1). As in Table 2, two variables (sub-processes), i.e., $\mathcal{X}_1$ = Shaft in (2) and $\mathcal{X}_2$ = Location in (right) have supports of $N_{\mathcal{X}_1} = 249$ and $N_{\mathcal{X}_2} = 244$, respectively. The joint condition of both has a lower (or equal) support than any of them ($N_{\mathcal{X}_1,\mathcal{X}_2} = 126$).

Thus, if a node has a support lower than an actual minimum support, there is no possibility to find a node (sub-process) with a higher support in the branch below. This reduces the time to find the optimal solution significantly, as a good portion of the tree to traverse can be omitted. This first principle is realized in the function GenerateSubProcesses as listed in Algorithm 2 and can be seen as the breadth-first-search of the RCA. This function takes as its argument a sub-process and generates all sub-processes with a support higher than the actual n_{max}.

Algorithm 2 Branch & Bound algorithm for process optimization

```
1: procedure GenerateSubProcesses(𝒳)
2:     for all 𝒳̃ ⊆ 𝒳 do
3:         if N_𝒳̃ > n_max and Q_𝒳̃ ≥ q_min then
4:             n_max = N_𝒳̃
5:         end if
6:         if N_𝒳̃ > n_max and Q_𝒳̃ < q_min then
7:             X = {X ∪ 𝒳̃}
8:         end if
9:     end for
10:    return X
11: end procedure
```

The second principle is to consider disjoint value sets. For the support of a sub-process the following holds: Let $\mathcal{X}_1, \mathcal{X}_2$ be two sub-sets with $Y_1(\omega) \in \mathcal{S}_1 \subseteq \mathcal{S}$, $Y_2(\omega) \in \mathcal{S}_2 \subseteq \mathcal{S}$ with $\mathcal{S}_1 \cap \mathcal{S}_2 = \varnothing$ and $\mathcal{X}_1 \cup X_2$ denote the unification of two sub-processes. It is obvious that $N_{\mathcal{X}_1 \cup \mathcal{X}_2} = N_{\mathcal{X}_1} + N_{\mathcal{X}_2}$, which implies that by extending the codomain of the influence variables, the support $N_{\mathcal{X}_1 \cup \mathcal{X}_2}$ can only increase. For the a class of convex process indices, as defined in Definition 1, the second Branch and Bound principle can be derived, based on the next theorem:

Theorem 1. *Given two sub-processes $\mathcal{X}_1 = (X, \mathbf{Y}_1)$, $\mathcal{X}_2 = (X, \mathbf{Y}_2)$ of a manufacturing process $\mathcal{X} = (X, \mathbf{Y})$ with $\mathbf{Y}_1(\omega) \in \mathcal{S}_1 \subseteq \mathcal{S}$, $\mathbf{Y}_2(\omega) \in \mathcal{S}_2 \subseteq \mathcal{S}$ and $\mathcal{S}_1 \cap \mathcal{S}_2 = \varnothing$. Then for the class of process indices as defined in (4), the following inequality holds:*

$$\min_{\mathcal{Z} \in \{\mathcal{X}_1, \mathcal{X}_2\}} Q_{w,\mathcal{Z}} \leq Q_{w,\mathcal{X}_1 \cup \mathcal{X}_2} \leq \max_{\mathcal{Z} \in \{\mathcal{X}_1, \mathcal{X}_2\}} Q_{w,\mathcal{Z}}.$$

Proof. With $p = \frac{P(\mathbf{Y} \in \mathcal{S}_1)}{P(\mathbf{Y} \in \mathcal{S}_1 \cup \mathcal{S}_2)}$ the following convex property holds:

$$\begin{aligned} Q_{w,\mathcal{X}_1 \cup \mathcal{X}_2} &= E\left(w(x) | \mathbf{Y}(\omega) \in \mathcal{S}_1 \cup \mathcal{S}_2\right) \\ &= \frac{E\left(w(x) \mathbb{1}_{\{\mathbf{Y}(\omega) \in \mathcal{S}_1 \cup \mathcal{S}_2\}}\right)}{P(\mathbf{Y}(\omega) \in \mathcal{S}_1 \cup \mathcal{S}_2)} \\ &= \frac{E\left(w(x) \mathbb{1}_{\{\mathbf{Y}(\omega) \in \mathcal{S}_1\}}\right) + E\left(w(x) \mathbb{1}_{\{\mathbf{Y}(\omega) \in \mathcal{S}_2\}}\right)}{P\left(\mathbf{Y}(\omega) \in \mathcal{S}_1 \cup \mathcal{S}_2\right)} \\ &= p \frac{E\left(w(x) \mathbb{1}_{\{\mathbf{Y}(\omega) \in \mathcal{S}_1\}}\right)}{P\left(\mathbf{Y}(\omega) \in \mathcal{S}_1\right)} + (1-p) \frac{E\left(w(x) \mathbb{1}_{\{\mathbf{Y}(\omega) \in \mathcal{S}_2\}}\right)}{P\left(\mathbf{Y}(\omega) \in \mathcal{S}_2\right)}. \end{aligned}$$

Therefore, by combining two disjoint combination sets, the E_{ci} of the union of these two sets lies in between the maximum and minimum E_{ci} of these sets. This can be illustrated by considering Table 2 again. The two disjoint sub-processes $\mathcal{X}_1$ = Tool in (1,2) and $\mathcal{X}_2$ = Tool in (4) yield a conditional E_{ci} of $Q_{\mathcal{X}_1} = 0.84$ and $Q_{\mathcal{X}_2} = 0.82$. The union of both sub-processes yields E_{ci} value of $Q_{\mathcal{X}_1 \cup \mathcal{X}_2} = Q_{\text{Tool not in (3)}} = 0.82$. This value

is within the interval $< 0.82, 0.84 >$, as stated by the theorem. This convex property reduces the number of times the E_{ci} actually has to be calculated, as in some special cases we can estimate the value of E_{ci} by its upper and lower limits and compare it with q_{min}.

In the root cause analysis for process optimization, we are in general not interested in one global optimal solution but in a list of processes, having a quality better than the defined threshold q_{min} and maximal support. An expert might choose out of the n-best processes the one which he wishes to use as a benchmark. To get the n-best sub-processes, we need to traverse also those branches which already exhibit a (local) optimal solution. The rationale is that a (local) optimum $\tilde{\mathcal{X}}^*$ with $N_{\tilde{\mathcal{X}}^*} > n_{max}$ might have a child node in its branch, which might yield the second best solution. Therefore, line 4 in Algorithm 2 has to be adapted by *postponing* the found solution $\tilde{\mathcal{X}}$ to the set of sub-nodes $\mathbf{X}$. Hence, the actual maximal support is no longer defined by the (actual) best solution, but by the (actual) n-th best solution.

In many real-world applications, the influence domain is mixed, consisting of discrete data and numerical variables. To enable a joint evaluation of both influence types, the numerical data is transformed into nominal data by mapping the continuous data onto pre-set quantiles. In most of our applications, the 10, 20, 80 and 90% quantiles have performed best. Additionally, only those influence sets have to be accounted for which are successional.

4.2 Verification

As in practice the samples to analyze are small and the used PCIs are point estimators, the optimum of the problem according to Definition 4 can only be defined in statistical terms. To get a more valid statement of the true value of the considered PCI, confidence intervals have to be used. In the special case, where the underlying data follows a known distribution, it is straightforward to construct a confidence interval. For example, if a normal distribution can be assumed, the distribution of $\frac{C_p}{\hat{C}_p}$ ($\hat{C}_p$ denotes the estimator of C_p) is known, and a $(1-\alpha)\%$ confidence interval for C_p is given by

$$C(X) = \left[\hat{C}_p \sqrt{\frac{\chi^2_{n-1;\frac{\alpha}{2}}}{n-1}}, \hat{C}_p \sqrt{\frac{\chi^2_{n-1;1-\frac{\alpha}{2}}}{n-1}} \right]. \tag{7}$$

For the other parametric basic indices, in general there exits no analytical solution as they all have a non-centralized χ^2 distribution. In [2, 10] or [4], for example, the authors derive different numerical approximations for the *basic* PCIS, assuming a normal distribution.

If there is no possibility to make an assumption about the distribution of the data, computer based, statistical methods such as the well known Bootstrap method [5] are used to determine confidence intervals for process capability indices. In [1], three different methods for calculating confidence intervals are derived and a simulation study is performed for these intervals. As result of this study, the bias-corrected-method (BC) outperformed the other two methods (standard-bootstrap and percentile-bootstrap-method). In our applications, an extension to the BC-Method called the Bias-corrected-accelerated-method (BCa) as described in [3] was used for determining confidence intervals for the non-parametric basic PCIs, as described in (3). For the Empirical Capability Index E_{ci} a simulation study showed that the standard-bootstrap-method, as used in [1], performed the best. A $(1-\alpha)\%$ confidence interval for the E_{ci} can be obtained using

$$C(X) = \left[\hat{E}_{ci} - \Phi^{-1}(1-\alpha)\sigma_B, \hat{E}_{ci} + \Phi^{-1}(1-\alpha)\sigma_B \right], \tag{8}$$

where $\hat{E}_{ci}$ denotes an estimator for E_{ci}, σ_B is the Bootstrap standard deviation, and Φ^{-1} is the inverse standard normal.

As all statements that are made using the RCA algorithm are based on sample sets, it is important to verify the soundness of the results. Therefore, the sample set to analyze is to be randomly divided into two disjoint sets: training and test set. A list of the n best sub-processes is generated, by first applying the described RCA algorithm and second the referenced Bootstrap-methods to calculate confidence intervals. In the next step, the root cause analysis algorithm is applied to the test set. The final output is a list of sub-processes, having the same influence sets and a comparable level for the used PCI.

5 Experiments

An evaluation of the concept was performed on data from a foundry plant for engine manufacturing in the premium automotive industry (see Sect. 2). Three different groups of data sets where used with a total of 33 different data sets of samples to evaluate the computational performance of the used algorithms. Each of the analyzed data sets comprises measurement results describing geometric characteristics like positions of drill holes or surface texture of the produced products and the corresponding influence sets like a particular machine number or a worker's name. The first group of analyzed data, consists of 12 different measurement variables with four different influence variables, each with two to nine different values. The second group of data sets comprises 20 different sample sets made up of 14 variables with up to seven values each. An additional data set, recording the results of a cylinder twist measurement having 76 influence variables, was used to evaluated the algorithm for numerical parameter sets. The output for each sample set was a list of the 20 best sub-processes in order to cross check with the quality expert of the foundry plant. q_{min} was chosen to the unconditional PCI value. The analyzed data sets had at least 500 and at most 1,000 measurement results.

The first computing series was performed using the empirical capability index E_{ci} and the non-parametric C'_{pk}. To demonstrate the efficiency of the first Branch and Bound principle, an additional combinatorial search was conducted. The reduction of computational time, using the first Branch and Bound principle, amounted to two orders of magnitude in comparison with the combinatorial search as can be seen in Fig. 2. Obviously, the computational time for finding the n best sub-processes increases with the number of influence variables. This fact explains the jump of the combinatorial computing time in Fig. 2 (the first 12 data sets correspond to the first group introduced in the section above). On average, the algorithm using the first Branch and Bound principle outperformed the combinatorial search by a factor of 160. Using the combinatorial search, it took on average 18 min to evaluate the available data sets. However, using the first Branch and Bound principle decreased the computing time to only 4.4 s for C'_{pk} and to 5.7 s using the E_{ci}. The evaluation was performed to a search up to a depth of 4, which means, that all sub-process have no more than four different influence variables. A higher depth level did not yield different results, as the support of the sub-processes diminishes with increasing the number of influence variables used as constraints.

Applying the second Branch and Bound principle reduced the computational time even further. As Fig. 3 depicts, the identification of the 20 optimal sub-processes using the E_{ci} was on average reduced by a factor of 5 in comparison to the first Branch and Bound principle and resulted in an average computational time of only 0.92 s vs. 5.71 s. Over all analyzed sample sets, the second principle reduced the computing time by 80%. Even using the E_{ci} and the second Branch and Bound principle, it still took 20 s to compute, and for the non parametric calculation using the first Branch and Bound principle approximately 2 min. In this special

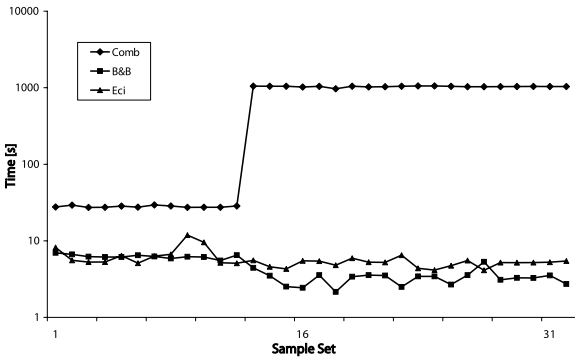

Fig. 2. Computational time for combinatorial search vs. Branch and Bound using the C'_{pk} and E_{ci}

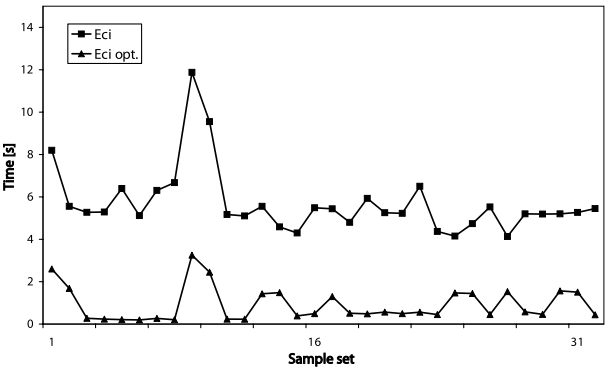

Fig. 3. Computational time first Branch and Bound vs. second Branch and Bound principle using E_{ci}

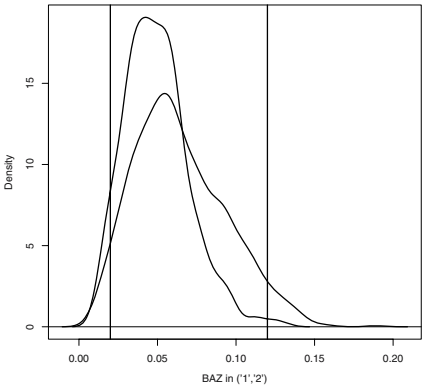

Fig. 4. Density plot for optimal sub-process (*narrower plot*) and its original process (*broader plot*) using E_{ci}

case, the combinatorial search was omitted, as the evaluation of 76 influence variables with four values each would have taken too long.

5.1 Optimum Solution

Applying the identified sub-processes to the original data set, the original, unconditional PCI is improved. More precisely, considering for example the sub-process $\mathcal{X}$ = Tool in (1,2) and using the E_{ci} the index improves from 0.49 to 0.70. As Fig. 4 shows, the quality of the sub-process (narrower distribution plot) clearly outperforms the original process (broader distribution plot), having less variance and a better process location.

On the test set, the performance of the optimum solution, characterized by Q_{Test} is over its lower bound determined by the bootstrap procedure on the training set, as shown in Table 3.

Table 3. Results for the process optimization for one data set

Index	N_{Test}	Q_{Test}	N_{Train}	C_B^l
E_{ci}	244	0.85	210	0.84

6 Conclusion

We have introduced an algorithm for efficient rule extraction in the domain of root cause analysis. The application goal is the manufacturing process optimization, with the intention to detect those process parameters which have a major impact on the quality of a manufacturing process. The basic idea is to transform the search for those quality drivers into an optimization problem and to identify a set of optimal parameter subsets using two different Branch and Bound principles. These two methods allow for a considerable reduction of the computational time for identifying optimal solutions, as the computational results show.

A new class of convex process capability indices, E_{ci}, was introduced and its superiority over common PCIs is shown with regard to computing time. As the identification of major quality drivers is crucial to industrial practice and quality management, the presented solution may be useful and applicable to a broad set of quality and reliability problems.

References

1. M. Kalyanasundaram, S. Balamurali. Bootstrap lower confidence limits for the process capability indices c_p, c_{pk} and c_{pm}. *International Journal of Quality and Reliability Management*, 19:1088–1097, 2002.
2. A.F. Bissel. How reliable is your capability index. *Applied Statistics*, 39(3):331–340, 1990.
3. B. Efron, R.J. Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall, New York, 1993.
4. G.S. Wasserman, L.A Franklin. Bootstrap lower confidence limits for capability indices. *Journal of Quality Technology*, 24(4):196–210, 1992.
5. J.S. Urban Hjorth. *Computer Intensive Statistical Methods*, 1st edition. Chapman and Hall, New York, 1994.
6. S. Kotz, N.L. Johanson. Process capability indices – a review, 1992–2000. *Journal of Quality Technology*, 34(1): 2–19, 2002.
7. Douglas C. Montgomery. *Introduction to Statistical Quality Control*, 2nd edition. Wiley, New York, 1991.
8. W. Pearn, K. Chen. Capability indices for non-normal distributions with an application in electrolytic capacitor manufacturing. *Microelectronics Reliability*, 37:1853–1858, 1997.
9. K. Vännman. A unified approach to capability indices. *Statistica Sina*, 5:805–820, 1995.
10. G.A. Stenback, D.M. Wardrop, N.F. Zhang. Interval estimation of process capability index c_pk. *Communications in Statistics. Theory and Methods*, 19(12):4455–4470, 1990.