BlakerCI: An algorithm and R package for the Blaker's binomial confidence limits calculation

Klaschka, Jan

2010

Institute of Computer Science
Academy of Sciences of the Czech Republic

BlakerCI: An algorithm and R package for the Blaker's binomial confidence limits calculation

Jan Klaschka

Technical report No. 1099

December 2010

Pod Vodárenskou věží 2, 182 07 Prague 8, phone: +420 266 053 810, fax: +420 286 585 789,
e-mail:klaschka@cs.cas.cz

Institute of Computer Science
Academy of Sciences of the Czech Republic

BlakerCI: An algorithm and R package for the Blaker's binomial confidence limits calculation[1]

Jan Klaschka

Technical report No. 1099

December 2010

Abstract:

This work describes a new algorithm for the calculation of the Blaker's binomial confidence limits. It outperforms the original Blaker's algorithm as regards both accuracy and speed. It will also be shown that it has some advantages in comparison with the recently published algorithm by M.P. Fay.

Keywords:
binomial distribution, Blaker's confidence interval, numerical algorithm

[1]This work was supported by grant 205/09/1079 from The Czech Science Foundation and by the Institutional Research Plan AV0Z10300504.

1 Introduction

This work describes a new algorithm for the calculation of the Blaker's binomial confidence limits and its implementation in an R package. The algorithm performance is tested and compared with that of two existing algorithms – the original Blaker's algorithm, and the recently published algorithm by M. P. Fay. While the drawbacks of the Blaker's algorithm are clear and it is outperformed by the new algorithm without any doubt, the principal advantage of the new algorithm in comparison with the algorithm by Fay is in a markedly higher speed, which becomes important when a large number of confidence intervals has to be computed.

Section 2 introduces the Blaker's confidence interval, including the context of the exact confidence limits and various proposals of less conservative intervals than those by Clopper and Pearson (1934). In Section 3, the original algorithm by Blaker (2000) is presented, together with the results of tests providing a clear evidence of its drawbacks. Section 4 is devoted to the new algorithm – to the theoretical background (Section 4.1), algorithm description (4.2), implementation in R package `BlakerCI` (4.3), and results of tests (Section 4.4). Section 5 informs then of the algorithm by M. P. Fay, including results of testing and comparison with the algorithm of Section 4. The whole work is summarized then in the final Section 6. In order to make the exposition smoother, some extending material (additional graphs and mathematics) is left to Appendices A1, A2, and A3.

2 Blaker's binomial confidence limits

2.1 Binomial distribution, notation

Much of the theory mentioned in this report applies to discrete distributions in general but since our main focus is an algorithm for the *binomial* confidence limits, the exposition will be confined, where useful, to the binomial distribution only.

Throughout this work, the following notation will be used. For integer $n \geq 0$, k, and real $p \in [0,1]$ (define $0^0 = 1$),

$$b_{n,k}(p) = \begin{cases} \binom{n}{k} p^k (1-p)^{n-k} & n \geq 0, 0 \leq k \leq n, \\ 0 & \text{otherwise.} \end{cases}$$

X (with possible subscripts) will denote a random variable with binomial distribution $Bin(n,p)$, i. e. $P_p(X = x) = b_{n,x}(p)$ for integer x and $0 \leq p \leq 1$.

2.2 Clopper-Pearson interval and less conservative alternatives

We will deal with interval estimates of binomial parameter p based on a realization x of $X \sim Bin(n,p)$. (The other parameter n is assumed positive, fixed and known.)

There exists a number of types of binomial confidence intervals and the two most frequently met in applications are the "standard" (or Wald) interval and the Clopper-Pearson interval (see e. g. Brown et al (2001)). The former, published almost 200 years ago by Laplace (1812), is a representative of the approximate confidence intervals, it is based on the normal approximation of the binomial distribution, and the coverage probability (i. e. the probability that the true value of parameter p lies inside the interval) oscillates both above and below the nominal $1-\alpha$ level. The latter, published for the first time by Clopper and Pearson (1934), is derived directly from the binomial distribution (without any approximation) and it is an exact confidence interval in the sense that the coverage probability is always at least $1-\alpha$. (The word "exact" is often used also in the narrow sense, and denotes, then, only the Clopper-Pearson interval. Such a usage of "exact", however, will be avoided here.) While both approximate and exact confidence intervals are advocated by different authors (see Agresti and Coull (1998)), on one hand, and a comment by Casella in the discussion part of Brown et al (2001), on the other hand) we will confine our attention throughout this work to the exact intervals.

The Clopper-Pearson lower and upper $1-\alpha$ confidence limits p_L and p_U, resp., for x successes out of n trials are given as follows:

$$\begin{aligned} p_L &= \inf\{p \in [0,1]; P_p(X \geq x) \geq \alpha/2\}, \\ p_U &= \sup\{p \in [0,1]; P_p(X \leq x) \geq \alpha/2\}. \end{aligned}$$

This guarantees that for all p from $[0,1]$

$$P_p(p < p_L) \leq \alpha/2 \text{ and } P_p(p_U < p) \leq \alpha/2 \tag{2.1}$$

and, consequently,

$$P_p(p < p_L \text{ or } p_U < p) = P_p(p < p_L) + P_p(p_U < p) \leq \alpha, \tag{2.2}$$

so that the interval is exact, indeed.

What is often criticized, is the over-conservativeness of the Clopper-Pearson interval: The coverage probability is, as a rule, greater than the nominal level on the whole interval $[0,1]$ – for an example see Fig. 2.1a. This goes on account of the fact that (2.1) are just sufficient, and not necessary conditions for (2.2).

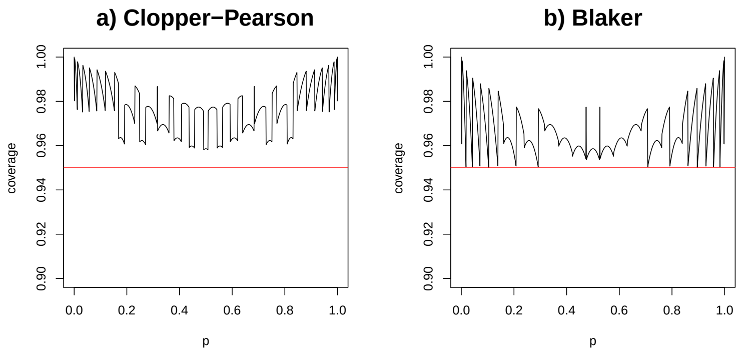

Figure 2.1: Coverage probability for $n = 20$, confidence level $1 - \alpha = 0.95$. a) Clopper-Pearson interval, b) Blaker's interval.

Since 1950's, several proposals of less conservative exact alternatives to the Clopper-Pearson intervals have been published – see Sterne (1954), Crow (1956), Blyth and Still (1983), Casella (1986), and Blaker (2000). All these keep (2.2) valid on $[0,1]$ but violate (2.1), allowing one of the probabilities $P_p(p < p_L)$, $P_p(p_U < p)$ exceed $\alpha/2$. The coverage probability curve then, as a rule, approximates the nominal confidence level better, and its minimum equals typically $1-\alpha$ – for an example see Fig. 2.1b.

The proposal by Blaker (2000) attracted much attention thanks to the following virtues, not possessed (all together) by any of the other previously published interval types.

1. Improving Clopper-Pearson: Blaker's confidence interval is always a subset of the Clopper-Pearson interval (at the same confidence level and based on the same data).

2. Nestedness: When $\alpha_1 > \alpha_2$, the $(1 - \alpha_1)$ confidence interval is always a subset (not necessarily proper) of the $(1 - \alpha_2)$ confidence interval (based on the same data).

3. Easy calculation by a short program in R contained in Blaker (2000).

The Blaker's confidence intervals will be defined and studied in more detail in the following section.

2.3 Blaker's confidence intervals and acceptability functions

The Clopper-Pearson $(1 - \alpha)$ confidence interval for x successes out of n trials might be defined alternatively as $\{p; \beta_{n,x}^{CP}(p) > \alpha\}$ where

$$\beta_{n,x}^{CP}(p) = \min\big(2P(X \geq x), 2P(X \leq x), 1\big). \tag{2.3}$$

Such a definition would be needless if dealing with the Clopper-Pearson interval alone but it is useful for an analogy with the Blaker's interval.

The Blaker's confidence intervals are based on the so called *acceptability function* $\beta_{n,x}$ defined on $[0,1]$ as

$$\beta_{n,x}(p) = \min\big(P_p(X \geq x) + P_p(X \leq x^*), P_p(X \leq x) + P_p(X \geq x^{**}), 1\big) \tag{2.4}$$

where

$$x^* = \max\{y; P_p(X \leq y) \leq P_p(X \geq x)\}, \quad x^{**} = \min\{y; P_p(X \geq y) \leq P_p(X \leq x)\}. \tag{2.5}$$

(Subscripts n, x will be left out when it cannot lead to confusion.)

Blaker (2000) shows that $C_\alpha(x) = \{p; \beta_{n,x}(p) > \alpha\}$ is an exact $(1-\alpha)$ confidence set, i. e. that $P_p\big(p \in C_\alpha(X)\big) \geq \alpha$ for all $p \in [0,1]$. Situation, however, is more complicated than in the Clopper-Pearson case: While $\{p; \beta^{CP}_{n,x}(p) > \alpha\}$ is always an interval, $C_\alpha(x)$ may not be (and often is not) an interval but only a union of disjoint intervals. This is due to the fact that β (unlike β^{CP}) is not necessarily a unimodal function. An example of functions β and β^{CP} is shown in Fig. 2.2.

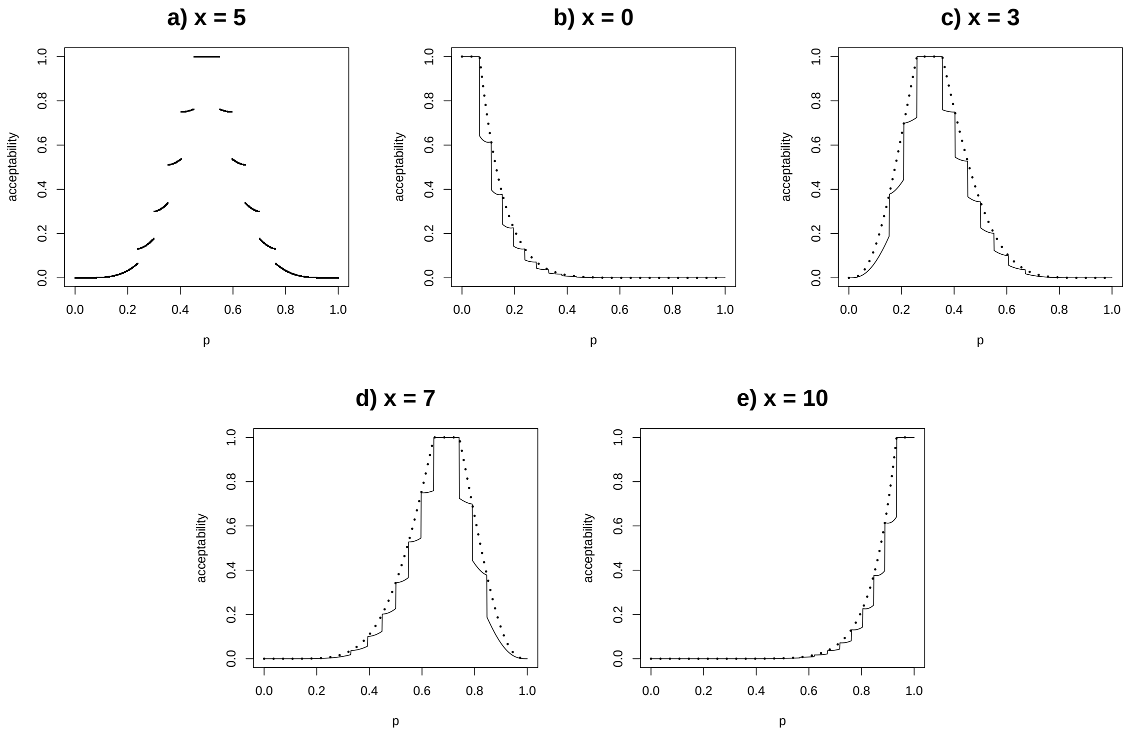

Figure 2.2: Blaker's acceptability function $\beta_{n,k}$ for $n = 10$ and various values of x: a) $x = 5$ (discontinuous plot), b–e) $x = 0, 3, 7, 10$ (discontinuities plotted as vertical lines). Dotted lines: functions $\beta^{CP}_{n,x}$, related to the Clopper-Pearson intervals (see (2.3)).

Since it is reasonable to construct confidence *intervals*, and not disconnected confidence sets, for p, the Blaker's confidence interval is defined as $\mathrm{conv}(C_\alpha(x))$, the convex hull of $C_\alpha(x)$.

Two of the virtues of the Blaker's interval, mentioned in Section 2.2, are easy to see:

1. It is clear from the comparison of (2.4, 2.5) with (2.3) that $\beta_{n,x}(p) \leq \beta_{n,x}(p)$ always holds. Therefore set $C_\alpha(x)$ is a subset of the Clopper-Pearson interval, and so is the convex hull of $C_\alpha(x)$, i. e. the Blaker's confidence interval, too.

2. Clearly, $C_{\alpha_1}(x) \subseteq C_{\alpha_2}(x)$ and, consequently, also $\mathrm{conv}(C_{\alpha_1}(x)) \subseteq \mathrm{conv}(C_{\alpha_2}(x))$ for $\alpha_1 > \alpha_2$. Thus, the nestedness property of the Blaker's intervals is evident.

As regards the third of the virtues mentioned in 2.2, easy calculation, it will be explained in the next section.

3 Blaker's original algorithm

Theoretical developments in Blaker (2000) apply to a broad class of discrete distributions and explicitly studied are interval estimates in several concrete distributions such as Poisson or hypergeometric distributions. The binomial distribution, however, has a prominent position as regards the "calculation recipes". The paper contains a short S+ program[2] designed for the calculation of the binomial confidence limits.

The program consists of two functions, namely `acceptbin`, and `acceptinterval`. The former calculates the acceptability function in a single point p, and the latter realizes a numerical search for the leftmost and rightmost points where the acceptability function exceeds level α.

The numerical search for the lower confidence limit p_L may be described as follows (the search for the upper limit p_U is analogous):

1. Input data: number of trials n, number of successes x, confidence level $1 - \alpha$, tolerance Δ (positive real, default 10^{-4}). Output p_L.
2. If $x = 0$, set $p_L := 0$, output p_L and finish, otherwise continue.
3. Set $p := p_L^{CP}$, the Clopper-Pearson lower limit.
4. While $\beta(p + \Delta) < \alpha$, repeat $p := p + \Delta$.
5. Set $p_L := p$, output p_L and finish.

The description above corresponds, by the way, to a corrected version of the program, published a year after the original paper. The first version of the program, as pointed to by J. Reiczigel, returned as its output the first p where $\beta(p) \geq \alpha$ instead of the last one for which $\beta(p) < \alpha$. Thus, the program produced too short intervals not containing some of those points p where $\beta(p) > \alpha$, and, as a consequence, exactness was violated. When denoting p_L^*, p_U^* the "true" confidence limits given by the theory, then the corrected program tries (though, as will be demonstrated soon, not always with a full success) to provide conservative approximations p_L, p_U of p_L^*, p_U^* such that $p_L \leq p_L^*$ and $p_U \geq p_U^*$. For the output of the uncorrected program, converse inequalities hold.

Even after the correction, however, there remain serious reasons for a criticism:

- The fixed step algorithm implies a poor speed-accuracy tradeoff.
- The program is prone, when set $C_\alpha(x)$ is disconnected, to skipping short intervals and yielding over-optimistic limits (and violating thus exactness).

Fig. 3.1 shows an example of an algorithm failure. The algorithm starts the search for the Blaker's lower confidence limit in the Clopper-Pearson confidence limit $p_L^{CP} = 0.935967$, close to 0.935973, a discontinuity point where the acceptability function "jumps" above the 0.05 level. Though 0.935973 is the correct value of the Blaker's lower limit, the algorithm misses it together with a short interval (length of approx. 5×10^{-5}) and gets by the first step to 0.936067 where β "sinks" again below 0.05. Then, after further 20 steps, the algorithm stops at 0.938067, close to a point where the β function gets above the 0.05 level – unfortunately *not the leftmost one* of such points. The difference between the correct Blaker's limit and the value output by the algorithm is over 0.002, i.e. more than 20 times larger than the tolerance parameter. As a consequence of the failure, the coverage probability is slightly below the nominal level of 0.95 on the erroneously skipped interval.

While Fig. 3.1 demonstrates a single algorithm failure, Fig. 3.2 and Tab. 3.1 show an "overall statistics" of such failures and coverage deficits resulting from them for $\alpha = 0.05$ and n from 1 to 1000.[3] For example, 295 of 1000 coverage probability curves fall locally below the nominal level of 0.95 when the tolerance parameter Δ of the algorithm is set to the default value of 10^{-4}. The smallest n with such a curve is 99. On the 295 curves, there are 386 symmetric pairs of local minima below 0.95, the lowest of them by 1.3×10^{-4} below 0.95. The maximum of errors (differences between

[2] Note that it is, at the same time, an R program, since both R and S+ are implementations of the S language.

[3] The method of coverage testing is described in Section 4.4.

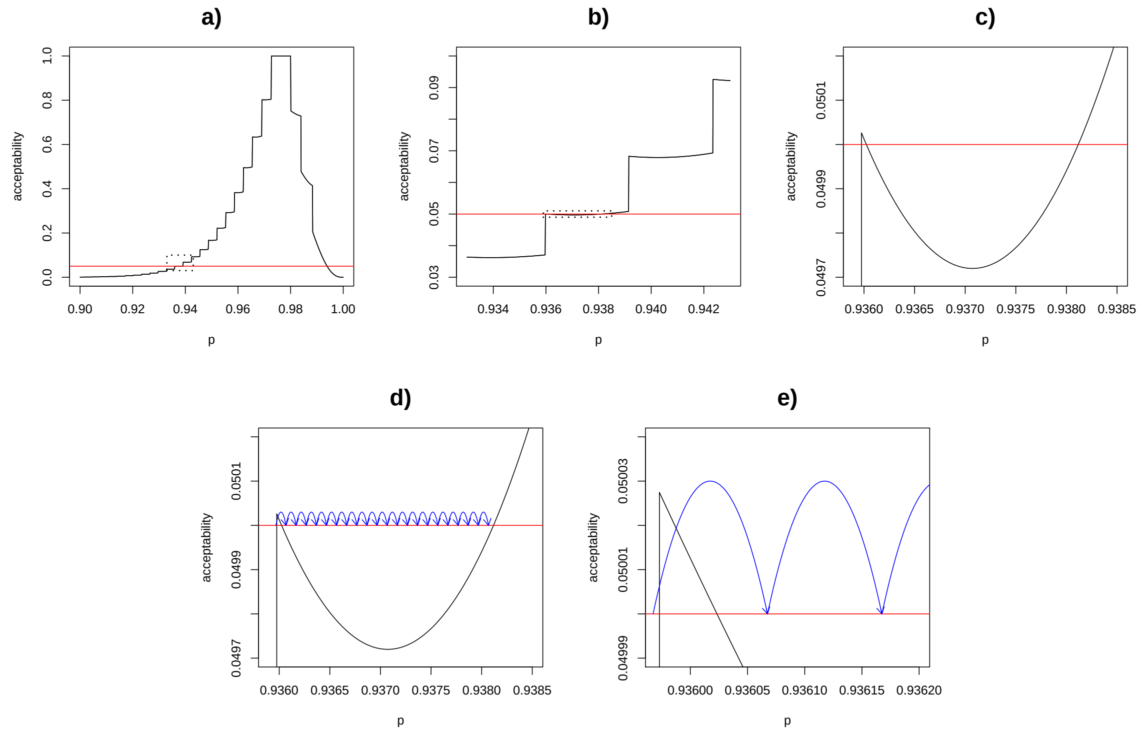

Figure 3.1: Failure of the Blaker's algorithm for $n = 134$, $x = 131$, $\alpha = 0.05$, $\Delta = 10^{-4}$. a) Acceptability function in $[0.9, 1]$; b, c) details of preceding graphs (rectangles with dashed borders from a, b) zoomed at b, c), resp.); d) steps of the search for the lower confidence limit; e) detail of the first steps.

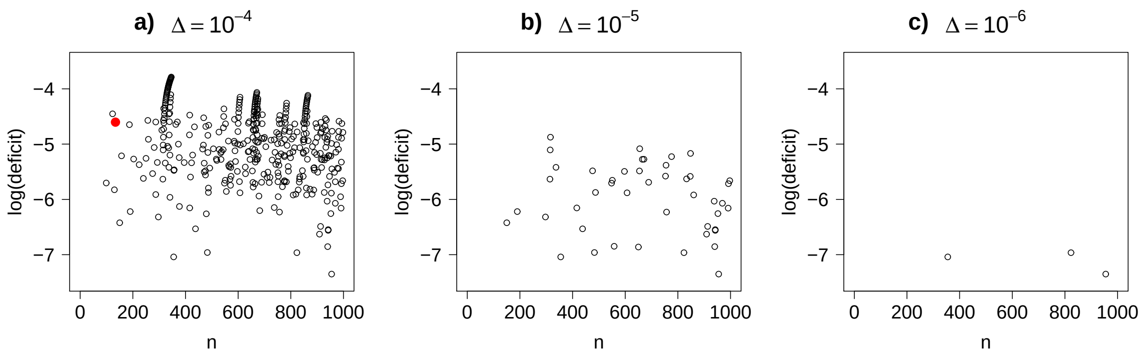

Figure 3.2: Coverage probability deficits of the 0.95 confidence intervals for $n = 1, \ldots, 1000$ and $x = 0, \ldots, n$ calculated by the Blaker's algorithm with the tolerance parameter set to a) 10^{-4}, b) 10^{-5}, and c) 10^{-6}. A point with coordinates (ν, δ) represents a pair of symmetric local minima of $0.95 - 10^{-\delta}$ on the coverage probability curve for $n = \nu$. The full circle in Fig. 3.2a corresponds to the example of Fig. 3.1.

the correct confidence limits and the limits output by the algorithm) is 0.0023 aprox. (23 times the tolerance parameter). When the tolerance parameter is set to 10^{-5} and 10^{-6} (at the price of tenfold and hundredfold slowdown, resp.!), the problems, as we can see, are reduced but do not vanish completely.

Analogous statistics of the algorithm failures for other confidence levels, namely 0.9, 0.99, and 0.999, may be found in Appendix A1.

tolerance	number of n	pairs of minima	min n	max error	max cover. deficit
10^{-4}	295	386	99	2.3×10^{-3}	1.6×10^{-4}
10^{-5}	44	45	150	1.2×10^{-3}	1.3×10^{-5}
10^{-6}	3	3	355	5.5×10^{-4}	1.1×10^{-7}

Table 3.1: Summary statistics of the Blaker's algorithm failures for confidence level $1-\alpha = 0.95$ and $n = 1, \dots, 1000$. Columns: Number of coverage probability curves (out of 1000) whose global minima on $[0,1]$ are below $1-\alpha$; number of pairs (symmetric about 0.5) of local minima below $1-\alpha$ on the curves; the least n for which the minimum of the curve falls below $1-\alpha$; the biggest difference between the correct confidence limit and the limit yielded by the algorithm; the biggest coverage deficit – the difference between $1-\alpha$ and the least coverage probability (where the latter is lower).

Note that if the limitation to a few conventional confidence levels was relaxed, it would be easy to construct examples of failures for an arbitrarily small step Δ (and for considerably smaller n than 99, the minimum of Tab. 3.1).

4 New algorithm

In this section, a new algorithm will be described. It was designed in order to avoid the drawbacks of the Blaker's algorithm, namely a poor speed-accuracy tradeoff, and proneness to skipping short segments of disconnected confidence sets.

4.1 Properties of the acceptability function

The new algorithm is justified by several lemmas on the properties of the acceptability function.

Value in x/n

The acceptability function $\beta_{n,x}$ is majorized by function $\beta^{CP}_{n,x}$, so that the Blaker's confidence limits may be numerically searched for within the Clopper-Pearson interval. Moreover, since $\beta_{n,x}(p^*) = 1$ where $P_{p^*}(X \geq x) = P_{p^*}(X \leq x)$, the point p^* may be used as an upper or lower bound of the numerical search for the lower or upper confidence limit, resp. The following lemma, however, allows for applying, instead of p^* (for which we have no closed formula), another bound, namely x/n.

Lemma 1 $\beta_{n,x}(p) = 1$ *for* $p = x/n$.

Proof Let $p = x/n$. Sufficient (and necessary) conditions for $\beta_{n,x}(p) = 1$ are inequalities

$$P_p(X \geq x) \quad \geq \quad P_p(X < x), \tag{4.1}$$
$$P_p(X \leq x) \quad \geq \quad P_p(X > x). \tag{4.2}$$

These may be, since left and right sides of each of the inequalities sum up to 1, reduced to

$$P_p(X \geq x) \geq \frac{1}{2}, \text{ and } P_p(X \leq x) \geq \frac{1}{2},$$

which, however, holds due to Lemma A1. ■

Discontinuity point as a bound for a confidence limit

Function $\beta_{n,x}$ is piecewise continuous with discontinuities in those points $p \leq x/n$ where $P_p(X \geq x) = P_p(X \leq i)$ for some $i < x$, and in those points $p \geq x/n$ where $P_p(X \leq x) = P_p(X \geq i)$ for some $i > x$. The following lemma shows that the discontinuity points play an extremely important role in locating the confidence limits. In fact, the worst "sin" of the Blaker's algorithm consists in ignoring such a lemma.

Lemma 2 *Let p_L and p_L^{CP} be the lower Blaker's and Clopper-Pearson confidence limits, resp., on confidence level $1-\alpha$, and let p^* be the leftmost discontinuity point of the acceptability function β to the right of p_L^{CP}. Then $p_L \le p^*$.*

Proof Evidently, $\beta^{CP}(p^*) > \beta^{CP}(p_L^{CP}) = \alpha$. Since β and β^{CP} equal in the discontinuity points of β, $\beta(p^*) > \alpha$ holds, so that $p_L \le p^*$.

Obviously, an analogous lemma on the Blaker's *upper* confidence limit may be obtained by substituting $n - X$ for X.

It should be noted that the search for a discontinuity point or testing the existence of a discontinuity point between a pair of other points are quite simple tasks in the case of the acceptability function, though they might be heavy numerical problems for some other functions.

Piecewise quasiconvexity

When searching for the leftmost/rightmost point where the acceptability function exceeds α, poor results would be obtained if such simple standard numerical methods as interval halving were used "globally": The tests performed by the author have shown that interval halving yields wrong solutions even more frequently than the Blaker's algorithm. The result below, however, justifies the use of interval halving within the intervals where the acceptability function is continuous.

The following lemma concerns (piecewise) quasiconvexity of the acceptability function. The concept of quasiconvexity is well established, but different (though equivalent) definitions appear in the literature. That is why the definition is recalled here.

Definition 1 Function f defined on a convex subset K of a real vector space is *quasiconvex* if for each $u, v \in K$ and $\lambda \in [0,1]$, inequality

$$f\big(\lambda u + (1-\lambda)v\big) \le \max\big(f(u), f(v)\big)$$

holds. If, moreover, the inequality is sharp, whenever $u \ne v$ and $0 < \lambda < 1$, then f is *strictly quasiconvex*. Further, f is *(strictly) quasiconcave*, if $-f$ is (strictly) quasiconvex.

Clearly, when a function defined on a one-dimensional interval is nonincreasing (decreasing) up to a certain point, and then nondecreasing (increasing), it is quasiconvex (strictly quasiconvex). All the monotonous (nonincreasing or nondecreasing) functions on an interval are both quasiconvex and quasiconcave. The strictly monotonous (decreasing or increasing) functions on an interval are both strictly quasiconvex and strictly quasiconcave.

Lemma 3 *The acceptability function is strictly quasiconvex on each of the intervals of continuity, except for the interval where it equals* 1.

Proof When continuous and less than 1 on interval $I \subseteq [0,1]$, $\beta_{n,x}$ can be expressed on I as either $P_p(X \ge x) + P_p(X \le k)$ for some $k < x-1$, or $P_p(X \le x) + P_p(X \ge k)$ for some $k > x+1$, i.e., in the form

$$\beta_{n,x}(p) = 1 - \sum_{i=j}^{k} P_p(X = i)$$

for some j and k ($j \le k$). Due to Lemma A3, the sum on the right-hand side is quasiconcave (on $[0,1]$ and, consequently, on I, too), so that the left-hand side is quasiconvex on I. ∎

The importance of the strict quasiconvexity is in the following fact. When f is strictly quasiconvex on an interval, and inequalities $f(u) < c$, $f(v) > c$ hold for a pair u, v of interval elements and constant c, then there exists exactly one point between u and v where $f(\cdot) - c$ changes signs. (Note that f need not be monotonous between u and v.) Moreover, the point may be easily an safely found by interval halving.

4.2 Algorithm description

Now, we have all the prerequisites needed for the key algorithm. Only the search for the *lower* confidence limit p_L will be described, since the upper limit p_U for x successes out of n trials may be calculated as 1 minus the lower limit for $n - x$ successes.

In the following description, $\kappa(p)$ will denote the biggest one of such integers i that (for given n, x) $P_p(X \leq i) \leq P_p(X \geq x)$.

1. Input data: number of trials n, number of successes x, confidence level $1 - \alpha$, tolerance Δ (positive real, default 10^{-10}). Output p_L.
2. If $x = 0$, set $p_L := 0$, output p_L and finish, otherwise continue.
3. Set $p_{low} := p_L^{CP}$, the Clopper-Pearson lower limit.
4. If $\beta(p_{low}) \geq \alpha$, set $p_L := p$, output p_L and finish, otherwise continue.
5. Set $p_{upp} := x/n$ and $\kappa^{CP} := \kappa(p_L^{CP})$.
6. While $p_{upp} - p_{low} \geq \Delta$, repeat:

   6.1. Set $p_{mid} := (p_{low} + p_{upp})/2$.

   6.2. If $\kappa(p_{mid}) > \kappa^{CP}$ or $\beta(p_{mid}) \geq \alpha$, set $p_{upp} := p_{mid}$, otherwise set $p_{low} := p_{mid}$.
7. Set $p_L := p_{low}$, output p_L and finish.

Remarks on the algorithm:

1. The algorithm follows the Blaker's one (the corrected version of 2001) in one respect: The lower (upper) confidence limit found by the numerical procedure is always a lower (upper) approximation of the "exact" theoretical value. Thus, the results of the algorithm always "keep on the conservative side".
2. The initial choice of x/n as the upper bound of the search is justified by Lemma 1.
3. Since function κ increases by 1 in the discontinuity points of β (up to the lower bound of the interval where β equals 1), the test of $\kappa(p_{mid}) > \kappa^{CP}$ prevents convergence to a point beyond any discontinuity point to the right of p_L^{CP}. Exclusion of such points from the search is justified by Lemma 2.
4. The algorithm would work identically if an "ordinary" interval halving was applied to β^*, a modification of β^*:
$$\beta^*(p) = \begin{cases} \beta(p) & p < p^*, \\ 1 & p \geq p^*, \end{cases}$$
where p^* is the first discontinuity point of β to the right of p_L^{CP}. The algorithm was explained this way at several conferences and seminars. Note that β^* is quasiconvex (though not strictly) in $[0, 1]$, which makes the "global" interval halving safe.
5. In an early version of the algorithm, the discontinuity point p^* was first numerically located, and afterwards p_L was searched for in $[p_L^{CP}, p^*]$. The present algorithm version does not determine p^* explicitly, and it only tests (using values of κ) whether a newly investigated point p_{mid} lies to the left of p^*, or not. This makes the algorithm simpler and does not cause any slowdown.
6. Once the upper bound of the search p_{upp} falls into the same continuous segment of β as p_L^{CP}, the values $\kappa(p_{mid})$ in the newly investigated points p_{mid} remain the same for the rest of calculations. Thus, it might seem uneconomical, in such situations, to calculate $\kappa(p_{mid})$ repeatedly. The effect of avoiding such redundant calculations in one of the tentative versions of the algorithm was, however (quite surprisingly), rather a slowdown than a speedup.

4.3 Implementation: R package BlakerCI

The algorithm described in Section 4.2 has been implemented in R environment (R Development Core Team (2010)) and included in `BlakerCI` – an R extension package, which was made public on the CRAN[4] web pages `http://cran.r-project.org/`.

The `BlakerCI` sources, binaries for Windows and MacOS X, as well as a manual in `pdf` are available at `http://cran.r-project.org/web/packages/BlakerCI`.

The Blaker's confidence limits calculation is realized by functions `binom.blaker.lower.limit` and `binom.blaker.limits`. The former is an internal function (not expected to be directly called by the user), and it performs the numerical search for the lower confidence limit. The latter is a user-level function, and it just calls the former function twice – once with parameters n, x (when computing the lower confidence limit), and then with parameters $n, n-x$ (which yields 1 minus the upper confidence limit).

The input parameters of the `binom.blaker.limits` function are the numbers of trials and successes, confidence level, and the tolerance, which defaults to 10^{-10}. (Note that the tolerance default is much lower that that for the Blaker's algorithm. Tolerance of 10^{-10} would make failures of the Blaker's algorithm, demonstrated in Section 3, extremely rare. Such a low tolerance, which can be easily afforded when using the `BlakerCI` software, would, however, make the calculations, in case of the Blaker's algorithm, very slow.)

4.4 Algorithm performance

Accuracy

Possible sources of inaccuracies may be

- "logical gaps" in the algorithm and/or its implementation,
- limited accuracy of external functions called, namely of R functions `pbinom` (binomial cumulative distribution function) and `qbinom` (binomial quantile function),
- limited accuracy of computer arithmetic.

Therefore, it is desirable to assess the accuracy not only by analyzing the script of the algorithm, but also empirically (experimentally). However, since there is no "golden standard" algorithm available, the algorithm testing was only partial, and mostly focused on the sufficient coverage probability. (Note that the failures of the original Blaker's algorithm, criticized in Section 3, result in deficits of coverage.)

The coverage testing was based on Lemma A5. Let us denote by L the set of confidence limits – both lower and upper – corresponding to all the possible realizations of X (i. e. $2n+2$ values in case of no ties). Ordered elements of L divide $[0, 1]$ into intervals, and the coverage probability, as a function of p, is (after passing the test of monotonicity of the confidence limits corresponding to $X = 0, \ldots, n$) quasiconcave in each of them. Since the quasiconcave functions are minimized in the limits, we can estimate the infimum of the coverage probability over $[0, 1]$ by evaluating formula (A3.2) in points $p \in L$. (An important detail: Confidence intervals enter (A3.2) as open.)

The test of minimum coverage based on the above considerations was implemented in R and applied to the confidence intervals calculated by the `BlakerCI` software. The parameters used were $n = 1, \ldots, 1000$, $x = 0, \ldots, n$, $\alpha = 0.1, 0.05, 0.01, 0.001$ and tolerance $\Delta = 10^{-i}, i = 4, 6, 8, 10, 12$.

In none of the settings mentioned, coverage deficits comparable to those presented in Section 3 were observed. The only deficits that have occurred were "microscopically" small (always below 10^{-15}).

As regards the coverage "micro-deficits", their number and parameter combinations yielding them depended on environment (HW, OS, R version). The most systematic testing was performed on two systems, one Linux-based (CentOS 5.5 on Intel Xeon E5430), and one Windows-based (WinXP on Intel Core 2 Duo E6400), both with R version 2.11.1. After having tuned such details, as whether to set α as 0.05, or $1 - 0.95$,[5] almost all "micro-deficits" disappeared (turned into zeros), and only

[4]Comprehensive R Archive Network.

[5]It is not the same in the computer arithmetic – the latter is bigger by approx. 4×10^{-17}.

two (the same on both systems) remained: For $\alpha = 0.001$ and tolerance 10^{-12}, coverage deficits of 1.11×10^{-16} were observed for $n = 884$ and 944. (Note that the size of the deficits is equal or very close to `.Machine$double.neg.eps` – an "almost smallest" one of those quantities whose subtraction from 1 yields a result different from 1 in R.[6]) An exact explanation of these two deficits requires further analysis.

Speed

The speed of the algorithm depends, naturally, on the computer platform and current load, as well as on how "smart" is the calling program. Thus, the numbers are presented here just for a basic orientation or rough comparisons.

The times (in seconds) spent by calculations of the confidence limits tables for $n = 1, \ldots, 1000$ and different values of confidence level and numerical tolerance are given in Tab. 4.1. Computational time grows approximately linearly with the negative decadic logarithm of tolerance (i. e. with the number of exact digits), and the figures are very similar for different confidence levels, as shown in Fig. 4.1. The growth seems to attenuate slightly for the highest confidence levels but the differences are so tiny that it is even hard to say whether they are real, or just a matter of chance. What is almost sure is their little practical importance.

	Confidence level			
tolerance	0.9	0.95	0.99	0.999
10^{-4}	210	213	213	215
10^{-6}	346	350	348	344
10^{-8}	486	487	482	476
10^{-10}	625	626	614	607
10^{-12}	762	762	750	737

Table 4.1: Calculation times (seconds) of the Blaker's confidence limits tables for $n = 1, \ldots, 1000$ (501 500 intervals) – dependence on the confidence level and numerical tolerance. (Environment: Intel Xeon E5430 quad-core, 2.66 GHz, 10 GB RAM, CentOS 5.5, R 2.11.1.)

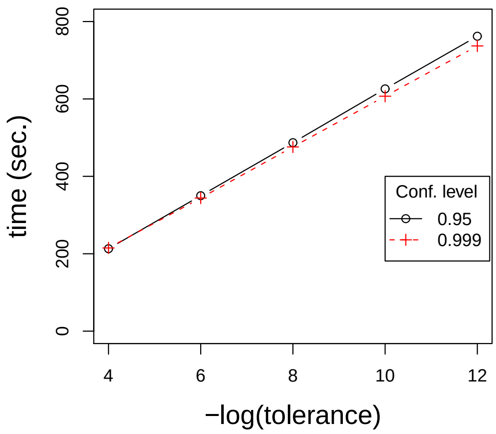

Figure 4.1: Tables of Blaker's confidence limits for $n = 1, \ldots, 1000$ and confidence levels 0.95, 0.999 – approximately linear growth of the computational times with the negative logarithm of numerical tolerance. (Subset of the data of Tab. 4.1.)

[6] It is not exactly the smallest such number. Experimenting suggests that the bound lies at about the half of `.Machine$double.neg.eps`.

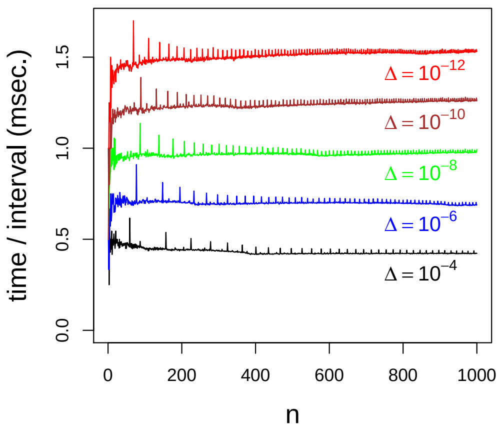

Figure 4.2: Mean computational times (in miliseconds) per confidence interval for $n = 1, \ldots, 1000$ and confidence level $1 - \alpha = 0.95$. Separate lines correspond to the values $10^{-i}, i = 4, 6, 8, 10, 12$ of tolerance parameter Δ.

Fig. 4.2 offers a closer look at the computational times: For each of the 5 tolerance parameter Δ values and fixed confidence level $1 - \alpha = 0.95$, computational times *per interval* were calculated for each n from 1 to 1000 as the means of $n + 1$ times spent by computing $n + 1$ intervals corresponding to $x = 0, \ldots, n$. We can see that for fixed Δ and big n values the times per interval are roughly stable (independent of n).

5 Algorithm by M. P. Fay

Up to now, the report was written as if the algorithm implemented in the `BlakerCI` package was the only existing alternative to the original Blaker's algorithm. Such was my perspective up to the autumn 2010 when, in the midst of preparations of my R package, I found the packages and papers by M. P. Fay.

Fay (2010a,b) criticized the fact that statistical software, including e. g. R function `binom.test`, offers often, when dealing with discrete data, such two-sided exact confidence intervals and tests that do not match each other. Then it may happen that a parameter value rejected by the test lies in the confidence interval or, conversely, a value outside the interval remains not rejected.

As a response to such a "state of affairs", M. P. Fay has written R packages `exact2x2`[7] (first version July 2009), and `exactci`[8] (initial version January 2010). The former package performs exact tests in contingency tables accompanied with confidence intervals for the odds ratio, and the latter covers one-sample exact binomial and Poisson tests together with the confidence intervals for the binomial/Poisson parameters. When applying any of the implemented methods, the user may choose one of three different types of two-sided test and confidence interval. Whichever of the three options is chosen, the test is *matched* with the confidence interval, i. e. when a parameter value is outside (inside) the confidence interval, it is rejected (not rejected) by the test at the same time. Thus, the author avoids the cross-linking of mutually incompatible test and confidence intervals, present in (some) other statistical software.

The binomial confidence intervals resulting from the three options mentioned above are the Clopper-Pearson interval, the interval by Sterne (1954), and the Blaker's interval.

What is important for my work is the fact that, as regards the Blaker's confidence limits calculation,

[7] `http://cran.r-project.org/web/packages/exact2x2`
[8] `http://cran.r-project.org/web/packages/exactci`

M. P. Fay did not rely on the original Blaker's algorithm, and designed, instead, an algorithm of his own, which resembles the algorithm implemented in the `BlakerCI` package in some respects: Fay (2010a, supplementary online material) pointed out the problems with accuracy of the Blaker's algorithm, and justified his procedure with an analysis of the acceptability function discontinuities, as well as of some properties of the continuous segments.

The reasons why I still find the `BlakerCI` package was worth posting (and, possibly, a paper on the matter worth publishing) are as follows.

- The algorithm implemented in `BlakerCI` is simpler, and allows, in my opinion, an easier insight.
- As experiments have shown, the algorithm by Fay is much slower, especially for large n.

The Blaker's binomial confidence limits are calculated by function `exactbinomCI`, an internal function of the package. The function is shared by the Blaker's and Sterne's intervals.

The algorithm may be roughly outlined as follows.

- When searching for the lower (upper) confidence limit p_L (p_U), the algorithm investigates the continuous segments of the acceptability function one by one, starting from the leftmost (rightmost) one.
- The limits of the continuous segments (i. e. the discontinuity points of the acceptability function) are calculated using the "standard" R function `uniroot`. (Equations $P_p(X \geq i) = P_p(X \leq j)$ are, for some i, j, solved in p.)
- The lower and upper estimates $\underline{P}$, $\overline{P}$ of the minima and maxima, resp., of the acceptability function on an interval are utilized. The estimates are based on the fact that the acceptability function, when restricted to a continuous segment, is the sum of two known functions, one of which is nondecreasing, and the other nonincreasing.
- Depending on the estimates $\underline{P}$, $\overline{P}$, 3 variants exist:
  A. When $\overline{P} \leq \alpha$, the segment is discarded and the algorithm proceeds to the next one.
  B. When $\underline{P} > \alpha$, the confidence limit p_L (p_U) lies on the lower (upper) segment boundary.
  C. Otherwise the question whether the point being sought lies in the segment remains open.
- In the C variant, the segment is subdivided into several intervals, and the estimates $\underline{P}$, $\overline{P}$ are calculated for each of them. Depending on the set of estimates $\underline{P}$, $\overline{P}$, there are 3 variants again: The whole segment may be discarded, the point being sought may be found on the segment boundary, or it may remain open whether or where the confidence limit lies within the segment. In the last case, some subintervals are excluded from the further search, the rest is subdivided into shorter intervals than in the previous step, and the process is iterated.
- The procedure continues until p_L and p_U are "captured" in a pair of intervals I_L, I_U of lengths at most Δ, a given tolerance.
- The function outputs
  - the limits of I_L and I_U,
  - the "point estimates" of p_L, p_U – the middles of I_L, I_U rounded to the digit above the tolerance level (e. g. to 3 decimal digits when $\Delta = 10^{-4}$).
- The program has a "back door" for the case that short enough intervals I_L, I_U are not found within a given number of iterations. Then, values analogous to those above are output with a warning.

When testing the Blaker's confidence limits yielded by `exactbinomCI`, it is necessary to see that the target is not exactly the same than that of `BlakerCI`: While the `BlakerCI` output is intended to lie always "on the conservative side" of the precise theoretical confidence limits, the "point estimates" output by `exactbinomCI` may approximate the theoretical values from whichever side. As regards a mutual comparison of the confidence limits yielded by the two programs, `exactbinomCI` should be represented by the "outer confidence limits" – the lower bound of I_L and the upper bound of I_U.

For testing purposes, the tables of 0.95 confidence limits were generated, using `exactbinomCI`, for $n = 1, \ldots, 1000$, and tolerances $\Delta = 10^{-i}$, $i = 4, 6, 8, 10, 12$.

All the confidence limits (the "outer" ones) passed the test of coverage probability sufficiency without any deficits.

The "outer" confidence limits from `exactbinomCI` corresponding to all the above tolerance levels were also compared with the "most accurate available" values from `binom.blaker.limits` – the confidence limits calculated with tolerance of 10^{-12}. The results were satisfactory for $\Delta = 10^{-8}$ and smaller – the absolute deviation between the results of the two programs were all smaller than Δ. For $\Delta = 10^{-6}$, the maximum deviation was slightly bigger than Δ – approx. 1.02×10^{-6} – and in the direction to stronger conservativeness. For $\Delta = 10^{-4}$, however, the results are confusing: The maximum deviations in both direction exceed Δ by far – almost 6 times in the "liberal" direction, an almost 14 times in the "conservative" one. That seems to contradict the fact that the output of both programs passed the coverage test. The reasons of such inconsistences (including a possible bug in the testing program) have not been clarified yet, and must be a matter of further work.

As regards the speed of the calculations, the difference between programs `exactbinomCI` and `binom.blaker.limits` is huge.

As demonstrated in Fig. 5.1, computational times per confidence interval, when using function `exactbinomCI`, do not depend too markedly on tolerance, but grow very fast with n. (Only tolerances 10^{-4} and 10^{-12} are shown since the curves for the other tolerance values lay in between and are close to each other, so that the graph would be hard to comprehend.) The growth is approximately (and rather faster than) linear, and for $n = 1000$ calculation of a pair of confidence limits takes, depending on the tolerance parameter, slightly below or above 0.5 second on average. (Compare with Fig. 4.2: `binom.blaker.limits` does the same work in about 1.5 *miliseconds* with the smallest tolerance 10^{-12}.)

Summary computing times of the whole tables – see Fig. 5.2a – are between 44 and 50 hours (compare with approx. 3.5 – 13 minutes shown in Fig. 4.1.) The difference in speed is less dramatic when comparing computational times of smaller tables for $n = 1, \ldots, 100$: Program `exactbinomCI` – see Fig. 5.2b – needs 105 – 135 seconds approx., while `binom.blaker.limits` about 7.5 seconds at maximum.

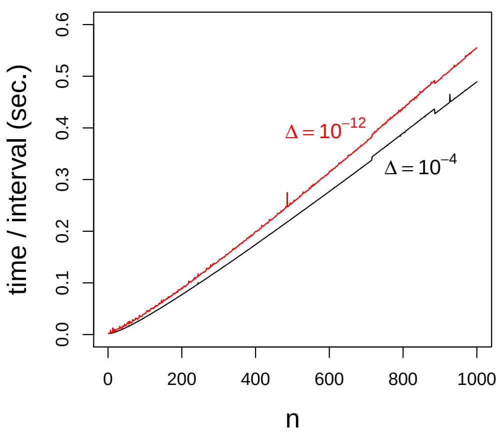

Figure 5.1: Mean computational times (in seconds) per confidence interval calculated by `exactbinomCI` for $n = 1, \ldots, 1000$ and confidence level $1 - \alpha = 0.95$. The lines corresponding to $\Delta = 10^{-4}$ and $\Delta = 10^{-12}$ shown. (Environment the same as that mentioned in the caption of Tab. 4.1.)

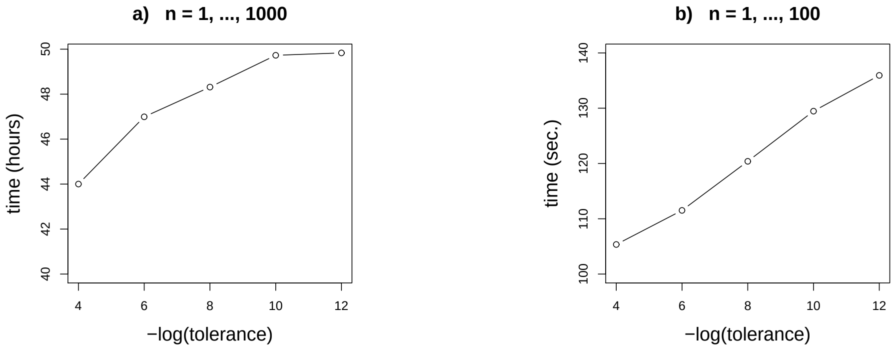

Figure 5.2: Summary computing times of the Blaker's confidence limits tables calculated by `exactbinomCI` for a) $n = 1, \ldots, 1000$, b) $n = 1, \ldots, 100$, confidence level $1 - \alpha = 0.95$, and tolerance $\Delta = 10^{-i}$, $i = 4, 6, 8, 10, 12$.

6 Concluding remarks

The algorithm implemented in the `BlakerCI` package was designed as a response to the drawbacks of the original Blaker's algorithm, and in ignorance of any other existing algorithms. It was demonstrated clearly enough in Section 3 that the Blaker's algorithm fails to guarantee the accuracy of the confidence limits required by the user through the tolerance parameter. Moreover, due to the fixed-step numerical search, attempts to improve the accuracy by setting the tolerance to very low values would result in a dramatic slowdown.

The new algorithm described in Section 4.2 is simple (not much longer and more complicated that the Blaker's one), fast even for as low tolerance as e. g. 10^{-12}, and accurate. (Some unclear results of the accuracy tests, mentioned in Section 4.4, concern only "microscopic" deviations from the required test values, and represent an incomparably smaller problem than those of the Blaker's algorithm.)

The papers and software by M. P. Fay, which appeared in 2010 (plus the related package `exact2x2` of 2009), change the situation markedly but do not, in my opinion, liquidate the *raison d'être* of the `BlakerCI` package. The author of the `exactci` package was aware of the accuracy problems of the Blaker's algorithm, and designed a procedure whose results satisfy similar accuracy criteria as `BlakerCI`. The algorithm by Fay is, nevertheless, much slower, especially for large n. The difference in speed may be neglected when, say, calculating a single confidence interval or a couple of intervals in the framework of everyday practical statistical work. In such circumstances, `exactci` (with more functionalities than calculation of a confidence interval only) is probably the best option. The question of speed, however, may become important when many thousands or even millions of confidence intervals have to be calculated – e. g. when creating large statistical tables or in simulation studies. Then the `BlakerCI` package should be preferred.

Bibliography

Agresti A, Coull A (1998) Approximate is better than "exact" for interval estimation of binomial proportions. Am Statistician 52:119–126

Blaker H (2000) Confidence curves and improved exact confidence intervals for discrete distributions. Canad J Statist 28:783–798,
Corrigenda: Canad J Statist 29(4):681

Blyth CR, Still HA (1983) Binomial confidence intervals. JASA 78:108–116

Brown LD, Cai TT, DasGupta A (2001) Interval estimation for a binomial proportion. Statistical Science 16:101–133

Casella G (1986) Refining binomial confidence intervals. Canadian J of Statistics 14:113–129

Clopper CJ, Pearson ES (1934) The use of confidence or fiducial limits illustrated in the case of the binomial. Biometrika 26:404–413

Crow EL (1956) Confidence intervals for a proportion. Biometrika 43:423–435

Fay MP (2010a) Confidence intervals that match Fisher's exact and Blaker's exact tests. Biostatistics 11:373–374

Fay MP (2010b) Two-sided exact tests and matching confidence intervals for discrete data. R Journal 2:53–58

Laplace PS (1812) Théorie Analytique des Probabilités. Courcier, Paris

Neumann P (1966) Über den Median der Binomial- und Poissonverteilung. Wissenschaftliche Zeitschrift der Technischen Universität Dresden 15:223–226

R Development Core Team (2010) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, URL `http://www.R-project.org`, ISBN 3-900051-07-0

Sterne TE (1954) Some remarks on confidence or fiducial limits. Biometrika 41:117–129

Appendices

A1 Additional results of the Blaker's algorithm testing

The following figures and tables extend those of Section 3 by showing the statistics of the Blaker's algorithm failures for $n = 1, \ldots, 1000$ and several confidence levels different from 0.95.

Fig. A1.1 and Tab. A1.1 summarize the results of testing for $1 - \alpha = 0.9$, Fig. A1.2 and Tab. A1.2 for $1 - \alpha = 0.99$, and Fig. A1.3 and Tab. A1.3 for $1 - \alpha = 0.999$.

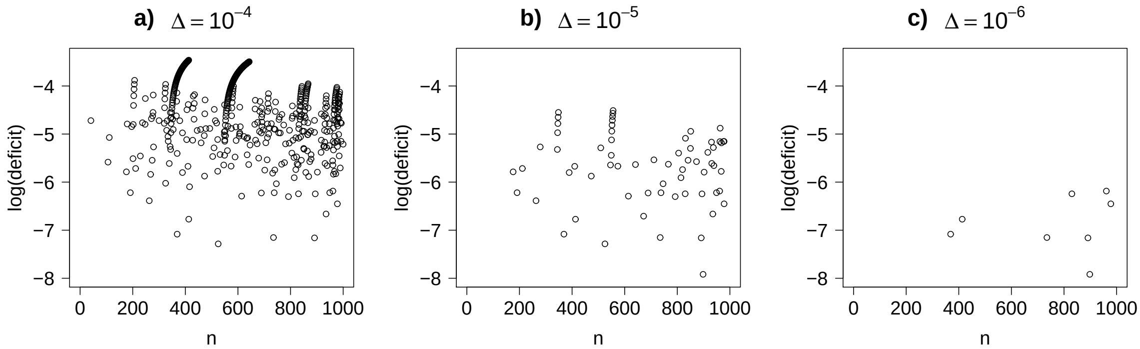

Figure A1.1: Coverage probability deficits of the 0.9 confidence intervals for $n = 1, \ldots, 1000$ and $x = 0, \ldots, n$ calculated by the Blaker's algorithm with the tolerance parameter set to a) 10^{-4}, b) 10^{-5}, and c) 10^{-6}. A point with coordinates (ν, δ) represents a pair of symmetric local minima of $0.9 - 10^{-\delta}$ on the coverage probability curve for $n = \nu$.

tolerance	number of n	pairs of minima	min n	max error	max cover. deficit
10^{-4}	385	491	41	5.1×10^{-3}	3.5×10^{-4}
10^{-5}	64	65	176	1.3×10^{-3}	3.1×10^{-5}
10^{-6}	8	8	369	5.8×10^{-4}	6.5×10^{-7}

Table A1.1: Summary statistics of the Blaker's algorithm failures for confidence level $1 - \alpha = 0.9$ and $n = 1, \ldots, 1000$. Columns: Number of coverage probability curves (out of 1000) whose global minima on $[0, 1]$ are below $1 - \alpha$; number of pairs (symmetric about 0.5) of local minima below $1 - \alpha$ on the curves; the least n for which the minimum of the curve falls below $1 - \alpha$; the biggest difference between the correct confidence limit and the limit yielded by the algorithm; the biggest coverage deficit – the difference between $1 - \alpha$ and the least coverage probability (where the latter is lower).

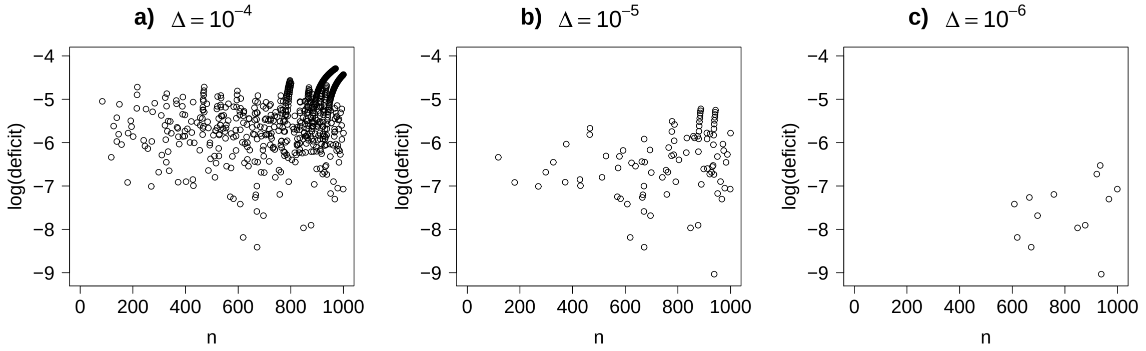

Figure A1.2: Coverage probability deficits of the 0.99 confidence intervals for $n = 1, \dots, 1000$ and $x = 0, \dots, n$ calculated by the Blaker's algorithm with the tolerance parameter set to a) 10^{-4}, b) 10^{-5}, and c) 10^{-6}. A point with coordinates (ν, δ) represents a pair of symmetric local minima of $0.99 - 10^{-\delta}$ on the coverage probability curve for $n = \nu$.

tolerance	number of n	pairs of minima	min n	max error	max cover. deficit
10^{-4}	387	630	84	3.6×10^{-3}	5.1×10^{-5}
10^{-5}	91	95	118	1.9×10^{-3}	6.0×10^{-6}
10^{-6}	13	13	608	3.9×10^{-4}	3.0×10^{-7}

Table A1.2: Summary statistics of the Blaker's algorithm failures for confidence level $1-\alpha = 0.99$ and $n = 1, \dots, 1000$. Columns: Number of coverage probability curves (out of 1000) whose global minima on $[0, 1]$ are below $1 - \alpha$; number of pairs (symmetric about 0.5) of local minima below $1 - \alpha$ on the curves; the least n for which the minimum of the curve falls below $1-\alpha$; the biggest difference between the correct confidence limit and the limit yielded by the algorithm; the biggest coverage deficit – the difference between $1 - \alpha$ and the least coverage probability (where the latter is lower).

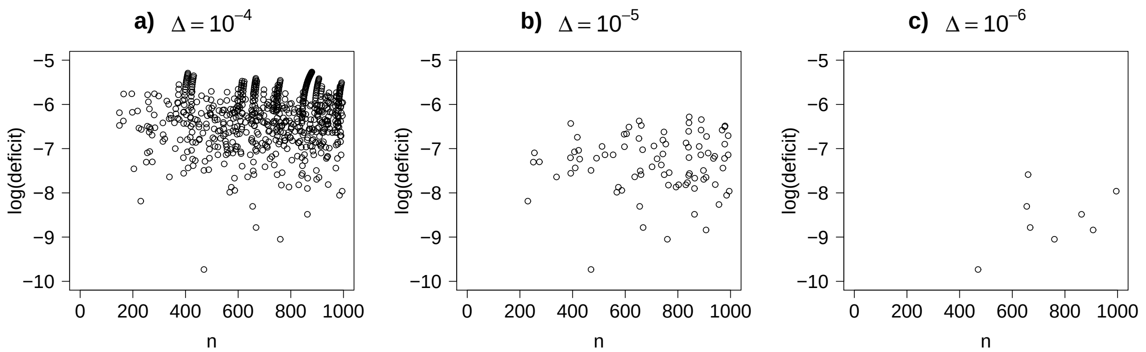

Figure A1.3: Coverage probability deficits of the 0.999 confidence intervals for $n = 1, \dots, 1000$ and $x = 0, \dots, n$ calculated by the Blaker's algorithm with the tolerance parameter set to a) 10^{-4}, b) 10^{-5}, and c) 10^{-6}. A point with coordinates (ν, δ) represents a pair of symmetric local minima of $0.999 - 10^{-\delta}$ on the coverage probability curve for $n = \nu$.

tolerance	number of n	pairs of minima	min n	max error	max cover. deficit
10^{-4}	431	665	149	2.0×10^{-3}	5.4×10^{-6}
10^{-5}	77	86	230	1.1×10^{-3}	5.2×10^{-7}
10^{-6}	8	8	470	4.8×10^{-4}	2.6×10^{-8}

Table A1.3: Summary statistics of the Blaker's algorithm failures for confidence level $1-\alpha = 0.9$ and $n = 1, \ldots, 1000$. Columns: Number of coverage probability curves (out of 1000) whose global minima on $[0, 1]$ are below $1-\alpha$; number of pairs (symmetric about 0.5) of local minima below $1-\alpha$ on the curves; the least n for which the minimum of the curve falls below $1-\alpha$; the biggest difference between the correct confidence limit and the limit yielded by the algorithm; the biggest coverage deficit – the difference between $1-\alpha$ and the least coverage probability (where the latter is lower).

A2 Properties of binomial distribution

Lemma A1 *For integer $n \geq 0$, $x \in \{0, \ldots, n\}$, $p = x/n$, and $X \sim Bin(n, p)$, the following inequalities hold:*

$$P_p(X \geq x) \geq \frac{1}{2}, \text{ and } P_p(X \leq x) \geq \frac{1}{2}.$$

Proof The lemma states, in fact, that the binomial distribution with an integer expected value has the (unique) median that equals the expectation. This, however has been proved by Neumann (1966). ■

Note that notation used in the following lemma was introduced in Section 2.1.

Lemma A2 *For integer $n \geq 1$, k, and real $p \in (0, 1)$,*

$$\frac{\mathrm{d}}{\mathrm{d}\,p}\, b_{n,k}(p) = n\big(b_{n-1,k-1}(p) - b_{n-1,k}(p)\big). \tag{A2.1}$$

Proof When $1 \leq k \leq n-1$,

$$\frac{\mathrm{d}}{\mathrm{d}\,p}\, b_{n,k}(p) = \binom{n}{k} \frac{\mathrm{d}}{\mathrm{d}\,p}\, p^k (1-p)^{n-k} = \binom{n}{k} \big(kp^{k-1}(1-p)^{n-k} + (n-k)p^k(1-p)^{n-k-1}\big),$$

which may be further reduced to

$$n\binom{n-1}{k-1} p^{k-1}(1-p)^{n-k} + n\binom{n-1}{k} p^k (1-p)^{n-k-1} = n\big(b_{n-1,k-1}(p) - b_{n-1,k}(p)\big).$$

When $k = 0$,

$$\frac{\mathrm{d}}{\mathrm{d}\,p}\, b_{n,k}(p) = \binom{n}{0} \frac{\mathrm{d}}{\mathrm{d}\,p}\,(1-p)^n = -\binom{n}{0} n(1-p)^{n-1},$$

which equals to

$$-n\binom{n-1}{0}(1-p)^{n-1} = n\big(b_{n-1,-1}(p) - b_{n-1,0}(p)\big)$$

(note that $b_{n-1,-1}(p) = 0$). Case $k = n$ is analogous.

Finally, when k is outside $[0, n]$, $b_{n,k}(p)$ is zero function (with zero derivative), and $b_{n-1,k-1}(p)$, $b_{n-1,k}(p)$ are zero functions, as well. ■

The following lemma concerns quasiconcavity of functions defined as a sum of consecutive binomial probabilities. As regards the concepts of quasiconvexity and quasiconcavity, see Definition 1 in Section 4.1.

Lemma A3 *Let for $p \in [0,1]$*

$$f(p) = \sum_{i=j}^{k} b_{n,i}(p)$$

where $n \geq 1$, $0 \leq j \leq k \leq n$ and $\{j,k\} \neq \{0,n\}$. Then f is strictly quasiconcave on $[0,1]$.

Proof The derivative of f is

$$\frac{\mathrm{d}}{\mathrm{d}\,p} f(p) = \sum_{i=j}^{k} \frac{\mathrm{d}}{\mathrm{d}\,p} b_{n,i}(p) \tag{A2.2}$$

on $(0,1)$. Due to Lemma A2, the right-hand side of (A2.2) may be reduced to

$$n \sum_{i=j}^{k} \big(b_{n-1,i-1}(p) - b_{n-1,i}(p)\big) = n\big(b_{n-1,j-1}(p) - b_{n-1,k}(p)\big).$$

The derivative of f thus vanishes in p if and only if

$$b_{n-1,k}(p) = b_{n-1,j-1}(p). \tag{A2.3}$$

Let us solve (A2.3) in $(0,1)$.

When $k = n$ or $j = 0$ (note that at most one of these may hold), one side of (A2.3) is zero, while the other is positive for all $p \in (0,1)$. Then, the derivative of f has no zeroes in $(0,1)$.

When $j > 0$ and $k < n$, (A2.3) may be expressed as

$$\frac{(n-1)!}{k!(n-k-1)!} p^k (1-p)^{n-k-1} = \frac{(n-1)!}{(j-1)!(n-j)!} p^{j-1} (1-p)^{n-j}$$

and further reduced (note that $p \notin \{0,1\}$) to

$$\left(\frac{p}{1-p}\right)^{k-j+1} = \prod_{i=j}^{k} \frac{i}{n-i}$$

which, evidently, has a unique solution $p^* \in (0,1)$. Moreover, the equality sign in (A2.3) may be replaced with $<$ or $>$, and the resulting inequations can be solved in the same way as the equation. It becomes clear then that the derivative of f is positive for $p < p^*$, and negative for $p > p^*$.

To summarize, function f is either strictly monotonous on $[0,1]$ (when its derivative has no zeroes in $(0,1)$), or it is increasing on $[0,p^*]$ and decreasing on $[p^*,1]$ for some $p^* \in (0,1)$. In all these cases, f is strictly quasiconcave on $[0,1]$. ■

A3 Coverage probability

The following lemma is well known and widely applied, but most often either used implicitly, or stated without a proof. (It will be formulated in the terms of the binomial distribution, though its generalizeability to a broader class of discrete distributions is obvious.)

Lemma A4 *Let $X \sim Bin(n,p)$, $n \geq 1$, and let $I_x \subseteq [0,1]$, $x = 0, \ldots, n$, be (fixed) intervals. Further, let I_X denote a "random interval" – a random variable whose value is I_x when $X = x$. Denote for $0 \leq p \leq 1$*

$$\gamma(p) = P_p(p \in I_X). \tag{A3.1}$$

Then

$$\gamma(p) = \sum_{x=0}^{n} J_x(p)\, P_p(X = x), \quad 0 \leq p \leq 1, \tag{A3.2}$$

where J_x is the characteristic function of I_x, i. e.

$$J_x(p) = \begin{cases} 1 & p \in I_x, \\ 0 & \textit{otherwise.} \end{cases}$$

Proof Using conditional probabilities, (A3.1) may be expanded as

$$\gamma(p) = \sum_{x=0}^{n} P_p(p \in I_X | X = x)\, P_p(X = x).$$

However,

$$P_p(p \in I_X | X = x) = P_p(p \in I_x | X = x) = J_x(p).$$

■

The following lemma justifies the method of coverage testing mentioned in Section 4.4.

Lemma A5 *Let notation of Lemma A4 hold. Let $I \subseteq [0,1]$ be an interval containing none of the limits of intervals $I_0, \dots, I_n$. Function γ is then continuous on I.*

If, moreover, the lower, as well as upper limits of intervals $I_0, \dots, I_n$ form nondecreasing sequences, and $0 < \gamma(p) < 1$ in at least one $p \in I$, γ is strictly quasiconcave on I.

Proof According to Lemma A4, γ may be expressed on I as

$$\gamma(p) = \sum_{x \in A} P_p(X = x)$$

where $A \subseteq \{0, \dots, n\}$ is a set of indices (fixed, independent of p). Continuity of γ is then evident.

If, moreover, the limits of $I_0, \dots, I_n$ are nondecreasing, and γ is different from the zero and unit constant functions on I, then A is a nonempty set of consecutive indices, and either $0 \notin A$, or $n \notin A$. Strict quasiconcavity of γ on I follows then from Lemma A3. ■