ITERATIVE REFINEMENT FOR LINEAR SYSTEMS
IN VARIABLE-PRECISION ARITHMETIC

ANDRZEJ KIĘBASIŃSKI

Abstract.
We define a binary-cascade iterative-refinement process (BCIR) for solving a	nonsingular linear system with prescribed relative accuracy. For very high accuracy
requirements BCIR uses a software-implemented variable-precision arithmetic. The
time-cost and accuracy of BCIR are deduced under various conditions.

1. Introduction.
Consider the problem of solving a nonsingular \( n \times n \) real or complex linear
system
\[
Ax = b
\]
with a prescribed relative accuracy \( \varepsilon \) for the computed solution \( \tilde{x} \); i.e.
\[
\| x^* - \tilde{x} \| \leq \varepsilon \cdot \| x^* \|, \quad x^* = A^{-1}b.
\]
If this accuracy requirement is too high for the standard arithmetic of an available
computer then we may use some software-made extension of this arithmetic.

We assume in the following that a variable-precision floating-point arithmetic,
\( \mathbb{fl}(t) \), with \( t \)-digit binary mantissa is available. For convenience we admit any real
value \( t \) from some interval \([t_0, t_p]\) though in reality \( t \) should be replaced with \([\tilde{t}]\).
We neglect over- and under-flow and assume the best possible representation of
data and the best possible realization of arithmetic operations in \( \mathbb{fl}(t) \). We study
how to use such variable-precision arithmetic to solve (1.1–2) with minimal time-
cost.

In a sequential realization of \( \mathbb{fl}(t) \) the time-cost of arithmetic operations
increases at least linearly with \( t \).

Woźniakowski suggested the idea of using this fact for solving (1.1–2) by an
iterative refinement process, cf. [8]:

- factorize the matrix \( A \) and solve auxiliary linear systems for corrections using
  a low precision arithmetic \( \mathbb{fl}(t_0) \),
- compute only residual vectors using a high-precision arithmetic \( \mathbb{fl}(t_p), \ t_p \gg t_0 \).

For some values of \( t_0, t_p \) and \( n \) this process gives a significant reduction of the
time-cost when compared with the direct methods for (1.1) using \( \mathbb{fl}(t_p) \).
In this paper we define a binary-cascade iterative refinement process (BCIR) which follows Woźniakowski’s idea and reduces further the time-cost by the evaluation of residual vectors using “intermediate” arithmetics $fl(t_j)$, $t_0 \leq t_j \leq t_p$. In fact the whole process is based on using the lowest sufficient precision in the computation of residual vectors.

We show in section 4 that the time-cost of BCIR depends critically on the multiplication time $M(t)$ in $fl(t)$. If $M(t)$ behaves essentially as $t^\sigma$ with fixed $\sigma \in [1, 2]$ then BCIR solves (1.1–2) in time of order

\[ n \left( \frac{c}{c + \tau} \right)^\sigma + \frac{1 - v^{1 - \sigma}}{1 - 2^{1 - \sigma}} n \tau M(c + \tau) \]

where

\[ c \approx \log_2 \text{cond} (A), \quad \tau \approx -\log_2 \varepsilon, \quad v = \min(n, 2\tau/c) \]

(the exact definitions of $c$ and $\tau$ are given in (2.2) and (2.3)).

We know, cf. [8], that the representation errors in $A$ and $b$ in $fl(t)$ may change the solution of (1.1.) by

\[ \|Ax\| \leq \|x^*\|2^{-\tau} \cdot \|x^*\|2^{c+\tau} \cdot \text{cond}(A) \approx \|x^*\|2^{c+\tau} \cdot \text{cond}(A) \]

Therefore the use of $fl(t)$ with $t \geq c + \tau$ for the problem (1.1–2) seems to be necessary. Note that $n^2 \cdot M(c + \tau)$ is essentially the time-cost of computing one residual vector using $fl(c + \tau)$. Hence it seems plausible to conjecture that $n^2 \cdot M(c + \tau)$ is a lower bound on the time-cost of solving (1.1–2).

Implementing in $fl(t)$ the primary-scholl multiplication, cf. [3], we have $\sigma = 2$. The time-cost of BCIR is then of order

\[ n \left( \frac{c}{c + \tau} \right)^2 + 2 \right] n^2 M(c + \tau) \]

For large $\tau$ (say $\tau \geq c/n$) BCIR seems to be a nearly optimal process in this case.

Implementing the optimal (hypothetical) multiplication, cf. [6], [7], we have $\sigma = 1$ and the time-cost of BCIR is of order

\[ n \left( \frac{c}{c + \tau} \right) + \log n \right] n^2 M(c + \tau) \]

We do not know whether BCIR is a nearly optimal process also in this case.

Several problems such as BCIR for sparse systems, adaptive version of BCIR, etc., are discussed in [5] (see also section 5 of this paper).

2. Binary-cascade iterative-refinement process.

We will choose Gaussian elimination with pivoting as a model solution algorithm, cf. [8]. It is well known that for sufficiently large $t$ and for any vector $f$, Gaussian elimination, using $fl(t)$, yields a computed solution $\tilde{d}$, which