MATHEMATICS OF COMPUTATION
VOLUME 64, NUMBER 209
JANUARY 1995, PAGES 265–277

# CONVERTING APPROXIMATE ERROR BOUNDS INTO EXACT ONES

ABRAHAM ZIV

ABSTRACT. In order to produce error bounds quickly and easily, people often apply to error bounds linearized propagation rules. This is done instead of a precise error analysis. The payoff: Estimates so produced are not guaranteed to be true bounds. One can at most hope that they are good approximations of true bounds.

This paper discusses a way to convert such approximate error bounds into true bounds. This is done by dividing the approximate bound by $1 - \delta$, with a small $\delta$. Both the approximate bound and $\delta$ are produced by the same linearized error analysis. This method makes it possible both to simplify the error analyses and to sharpen the bounds in an interesting class of numerical algorithms. In particular it seems to be ideal for the derivation of tight, true error bounds for simple and accurate algorithms, like those used in subroutines for the evaluation of elementary mathematical functions (EXP, LOG, SIN, etc.), for instance. The main subject of this paper is forward a priori error analysis. However, the method may be fitted to other types of error analysis too. In fact the outlines of a forward a posteriori error analysis theory and of running error analysis are given also. In the course of proofs a new methodology is applied for the representation of propagated error bounds. This methodology promotes easy derivation of sharp, helpful inequalities. Several examples of forward a priori error analysis and one of a posteriori error analysis and running error analysis are included.

## 1. INTRODUCTION

In roundoff error analysis of floating-point computations, a local relative error is associated with each input number and with each arithmetic operation, and it is for the analyst to estimate the resulting global, accumulated, relative errors. The challenge that faces an analyst is complicated by the fact that some of the propagation rules of relative error are not linear.

Various techniques were invented that make error analysis possible, in spite of the difficulties (see, e.g., Wilkinson [14], Sterbenz [11], Stoer and Bulirsch [12]). It seems that no single technique matches, in both power and width of applicability, the technique of linearized perturbations, in which terms of second and higher orders are systematically ignored. This technique, however, has one important disadvantage: The bounds it produces are not guaranteed to

Received by the editor October 31, 1991 and, in revised form, October 1, 1993.

1991 *Mathematics Subject Classification.* Primary 65G05; Secondary 65D20, 26D15, 26D20, 33B10.

*Key words and phrases.* Error analysis, differential error analysis, relative error, relative precision, relative distance, error measure, floating point.


0025-5718/95 $1.00 + $.25 per page

be true bounds. They only approximate true bounds. Apparently that is why this technique is not more widely used.

A way to produce true bounds while easing the nonlinearity problem was suggested by Olver [2]. He replaces the familiar relative error by a new error measure, relative precision: If $\bar{x}$ is an approximation of $x$, such that $x = e^{\xi}\bar{x}$ with $|\xi| \leq \alpha$, he says that $x \simeq \bar{x}$ $(\mathrm{rp}\,\alpha)$. This is equivalent to replacing the traditional relative-error measure, $\rho(\bar{x}, x) = |(\bar{x} - x)/x|$, by a new measure, $\mathrm{rp}(\bar{x}, x) = |\ln(\bar{x}/x)|$. The advantages of relative precision follow, so it seems, mainly from the fact that the propagation rules, of its bounds, for multiplication, division, and the power operations, are identical in form to the linearized propagation rules of bounds of relative errors.

Another new error measure, relative distance, was tried by Ziv [15]. Its definition is $d(\bar{x}, x) = |\bar{x} - x|/\max\{|x|, |\bar{x}|\}$, and it eases the nonlinearity problem in much the same way as relative precision does.

The use of relative precision for error analyses of various types is demonstrated in a series of papers of Olver [2, 3, 4, 5, 6] and Olver and Wilkinson [7] (see also Scherer and Zeller [10]). Relative distance was tried for a mixed forward/backward error analysis in Ziv [16]. The idea of defining new, useful error measures was carried also to complex numbers (Olver [4, 5], Ziv [15]) and to linear normed spaces (this includes real and complex, vector and function spaces; see Ziv [15], Pryce [8, 9]).

Both relative precision and relative distance achieve linearity of the propagation rules for multiplication and division. However, both lose the linearity of the propagation rules for addition and subtraction. In Ziv [18] the possibility of defining a new error measure is discussed, that replaces relative error, for which all propagation rules for arithmetic operations are identical in form to the linearized propagation rules of relative error. The result of the discussion is negative. It is shown there that the linearized propagation rules of relative errors, for multiplication and for addition, contradict each other and cannot both be satisfied by any single, reasonable error measure.

In the following sections a method is suggested to go around the difficulty. It is shown that one may perform a standard error analysis, using the linearized propagation rules of the traditional relative error bounds, and convert the final, approximate error bound into a true bound by dividing it by $1 - \delta$ with a small positive $\delta$. Both the approximate error bound and $\delta$ are produced by the same linearized error analysis. The main subject of this paper is a priori error analysis. The methodology, though, may be fitted to other types of analysis. In order to demonstrate this point, we describe, in outline, a theory of forward a posteriori error analysis, including running error analysis.

The suggested method seems to be ideal for the production of rigorous, sharp bounds for simple accurate numerical algorithms like those evaluating elementary mathematical functions (SIN, COS, EXP etc.; see Ziv [17]). This is demonstrated by a simple example in §6.4. For a more interesting example see Ziv [18]. The result described above is achieved by looking into the basic inequalities upon which roundoff error analysis, with relative error, relies.

In §2 we describe the general type of algorithm which is covered by this paper (this is slightly generalized in Note 3.3) and present the linearized propagation rules of relative error. In §3 the main result, Theorem 3.1, is stated. In §4 we

present some basic inequalities; their proofs appear in the appendix. In §5 we use the inequalities to prove the main result of §3. In §6 some examples are given for possible uses of the main result. The examples include error analyses of general multiplication and division algorithms, addition and subtraction algorithms, including the evaluation of Euclidean norm and a simple algorithm for the evaluation of the sine function for small arguments. In §7 we outline a theory of a posteriori forward error analysis, including a short description of a running error analysis theory. These theories are demonstrated on the addition and subtraction algorithm.

## 2. Numerical algorithms and linearized error bounds

Suppose that we have $m$ input real numbers, $x_1, x_2, \ldots, x_m$, and we apply to them a sequence of elementary, binary operations, $\circ_{m+1}, \circ_{m+2}, \ldots, \circ_n$:

(2.1)
$$x_k = y_k \circ_k z_k \quad \text{where } y_k, z_k \in \{x_1, x_2, \ldots, x_{k-1}\} \ (k = m+1, m+2, \ldots, n).$$

We shall limit ourselves to the elementary operations of addition, subtraction, multiplication, division, and power (the exponents of powers are assumed to be errorless constants).

Let $x_k \neq 0$ be a real, exact value, and let $\bar{x}_k$ be its computed approximation, having the same sign. We may express the error by an *error factor* $q_{xk}$: $\bar{x}_k = x_k q_{xk}$, $0 < q_{xk} < \infty$. Usually, $q_{xk}$ is close to 1. The relative error associated with $q_{xk}$ is $\rho_{xk} = q_{xk} - 1$. The elementary operation $\circ_k$ introduces an error factor $\tilde{q}_k$ into $\bar{x}_k$. In analogy with $\bar{x}_k$ and $q_{xk}$ we use the notation $\bar{y}_k, \bar{z}_k, q_{yk}, q_{zk}$ for the computed approximations and the error factors associated with the exact $y_k, z_k$. The *exact* propagation rules for error factors are:

(2.2a) Multiply: $\bar{x}_k = \mathrm{fl}(y_k \times z_k) \Rightarrow q_{xk} = q_{yk} q_{zk} \tilde{q}_{xk}$;

(2.2b) Divide: $\bar{x}_k = \mathrm{fl}(y_k \div z_k) \Rightarrow q_{xk} = (q_{yk} \div q_{zk}) \tilde{q}_{xk}$;

(2.2c) Add/Sub:
$$\begin{cases} \bar{x}_k = \mathrm{fl}(y_k \pm z_k) \Rightarrow q_{xk} = (\theta_{yk} q_{yk} + \theta_{zk} q_{zk}) \tilde{q}_{xk} \\ \quad \text{where } \theta_{yk} = y_k / x_k, \ \theta_{zk} = \pm z_k / x_k \\ \quad (\text{note that } \theta_{yk} + \theta_{zk} = 1); \end{cases}$$

(2.2d) Power: $\bar{x}_k = \mathrm{fl}(y_k^{z_k}) \Rightarrow q_{xk} = q_{yk}^{z_k} \tilde{q}_{xk}$
(note that $\bar{z}_k$ is errorless; namely, $q_{zk} = 1$).

Denote by $l_{xk}, l_{yk}, l_{zk}, \tilde{l}_{xk}$ linearized bounds on the relative errors $\rho_{xk}, \rho_{yk}, \rho_{zk}, \tilde{\rho}_{xk}$. We substitute $q_{xk} = 1 + \rho_{xk}$, $q_{yk} = 1 + \rho_{yk}, \ldots$ in (2.2), ignore terms of second and higher orders, and look for the best possible bounds for $\rho_{xk}$. One gets the *linearized* propagation rules for bounds of relative error:

(2.3a) Mult/Div: $l_{xk} = l_{yk} + l_{zk} + \tilde{l}_{xk}$;

(2.3b) Add/Sub: $l_{xk} = \hat{l}_{xk} + \tilde{l}_{xk}$, where $\hat{l}_{xk} = |\theta_{yk}| l_{yk} + |\theta_{zk}| l_{zk}$,
$\theta_{yk} = y_k / x_k$, $\theta_{zk} = \pm z_k / x_k$;

(2.3c) Power: $l_{xk} = |z_k| l_{yk} + \tilde{l}_{xk}$.

The local, linearized error bounds are actually true bounds, and they must satisfy

(exactly)

$$l_{xk} \geq |\rho_{xk}| = |q_{xk} - 1| \qquad (k = 1, 2, \ldots, m), \tag{2.4a}$$

$$\tilde{l}_{xk} \geq |\tilde{\rho}_{xk}| = |\tilde{q}_{xk} - 1| \qquad (k = m+1, m+2, \ldots, n). \tag{2.4b}$$

There is no guarantee that the bounds $l_{xk}$ $(k = m+1, m+2, \ldots, n)$, which arise from (2.3), (2.4), are true error bounds. One may only hope that they approximate such bounds.

## 3. Converting linearized global bounds into true global bounds

The following theorem is the main theoretical result of this paper.

3.1. **Theorem.** *Let* $\delta$ *equal* $(2n_{\pm} + 1)B$, *where* $n_{\pm}$ *is the total number of* $\pm$ *operations in the algorithm and* $B = \max_{1 \leq i \leq n} l_{xi}$. *If* $\delta < 1$, *then the accumulated relative error satisfies* $|\rho_{xn}| = |q_{xn} - 1| \leq l_{xn}/(1 - \delta)$. □

3.2. *Note.* We assumed that the linearized bounds propagate according to the rules (2.3). Sometimes, however, it is convenient to use simpler but weaker rules. Thus, for instance, if $\theta_{yk}$, $\theta_{zk}$ are both positive, it might be convenient to use the rule $l_{xk} = \max\{l_{yk}, l_{zk}\} + \tilde{l}_{xk}$ in place of (2.3b). It is clear, however, that such weaker rules increase the values of $l_{xn}$ and $\delta$. So Theorem 3.1 remains true if such rules are used. This note emphasizes the fact that a user of Theorem 3.1 may apply freely any reasonable linearized propagation rules in order to produce the sequence $\{l_{xk}\}$, and is not restricted to the tight propagation rules (2.3).

3.3. *Note.* We did not consider, so far, truncation errors. Such errors may be included in Theorem 3.1 by introducing *fictitious steps* into the algorithm: Suppose that the exact $y$ is meant to approximate another exact number, $x$ (for instance, let $R(t)$ be a rational function approximating the transcendental function $T(t)$ and take $y = R(t)$, $x = T(t)$). One may include in the algorithm a transformation of $y$ into $x$. No operation is performed in the passage from $y$ to $x$. So actually $\bar{x} = \bar{y}$. This is why we call such a step *fictitious*. Fictitious steps are helpful in the error analysis. Actually $\bar{x} = \bar{y} = yq_y$. Let us denote $y/x = \tilde{q}_x$. Then we see that $q_x = q_y\tilde{q}_x$. This propagation rule is a special case of the exact propagation rule (2.2d) for the power operation (take $z_k = 1$). Therefore, fictitious steps such as this are permissible and do not affect the validity of Theorem 3.1. One should note, though, that the corresponding $\tilde{\rho}_x = (y - x)/x$ must be considered a local error and, like other local errors, its bound, $\tilde{l}_x$, should be a true bound and not a linearized approximation of a bound. The use of fictitious steps is demonstrated in Example 6.4 below.

The proof of Theorem 3.1 is given in §§4 and 5. It goes along the following lines: We keep track of error bounds by associating with $q_{xk}$ and $\tilde{q}_{xk}$ two-sided bounds of the form

$$\begin{cases} (1 - \varepsilon_{xk})^{l_{xk}/\varepsilon_{xk}} \leq q_{xk} \leq (1 - \varepsilon_{xk})^{-l_{xk}/\varepsilon_{xk}} & (k = 1, 2, \ldots, n), \\ (1 - \tilde{\varepsilon}_{xi})^{\tilde{l}_{xi}/\tilde{\varepsilon}_{xi}} \leq \tilde{q}_{xi} \leq (1 - \tilde{\varepsilon}_{xi})^{-\tilde{l}_{xi}/\tilde{\varepsilon}_{xi}} & (i = m+1, \ldots, n). \end{cases} \tag{3.1}$$

Here, $l_{xk}$, $\tilde{l}_{xi}$ are the linearized bounds described above, and $\varepsilon_{xk}$, $\tilde{\varepsilon}_{xi}$ are small auxiliary parameters. Theorem 4.1 below establishes the suitability of

such bounds. Theorems 4.1, 4.2 provide basic inequalities that enable one to formulate propagation rules for $\varepsilon_{xk}$ that ensure the validity of (3.1) for all $k$. Theorem 4.3 below makes it possible to convert bounds of the form (3.1), on error factors, into bounds on relative errors. The appropriate value of $\delta$ is deduced from the propagation rules for $\varepsilon_{xk}$.

## 4. Inequalities

The proofs of the theorems in this section are in the appendix at the end of the paper.

4.1. **Theorem.** *The function*

$$u(\lambda, \varepsilon) = \begin{cases} (1-\varepsilon)^{-\lambda/\varepsilon}, & 0 \neq \varepsilon < 1, \ -\infty < \lambda < \infty, \\ e^{\lambda}, & \varepsilon = 0, \ -\infty < \lambda < \infty, \end{cases}$$

*is analytic everywhere in its domain. It is strictly increasing, as a function of* $\lambda$, *for all constant* $\varepsilon < 1$, *strictly increasing, as a function of* $\varepsilon$, *for all constant* $\lambda > 0$, *strictly decreasing, as a function of* $\varepsilon$, *for all constant* $\lambda < 0$ *and a constant,* $1$, *for* $\lambda = 0$. □

*Note.* In what follows we shall use freely the notation $(1-\varepsilon)^{-\lambda/\varepsilon}$ instead of $u(\lambda, \varepsilon)$, even if $\varepsilon = 0$.

4.2. **Theorem.** *Let* $\varepsilon_i$, $\lambda_i$, $\theta_i$ $(i = 1, 2, \ldots, n)$ *be real constants that satisfy* $\varepsilon_i < 1$, $\sum \theta_i = 1$. *Denote* $\lambda = \sum \theta_i \lambda_i$ *and let* $\varepsilon < 1$ *be a real number that satisfies* $\varepsilon \geq \max\{\lambda_i - \lambda + \varepsilon_i, \varepsilon_i, -\lambda\}$ $(i = 1, 2, \ldots, n)$. *Then*

$$(4.1) \qquad \theta_i \lambda_i \geq 0 \ (i = 1, 2, \ldots, n) \Rightarrow \sum_{i=1}^{n} \theta_i (1-\varepsilon_i)^{-\lambda_i/\varepsilon_i} \leq (1-\varepsilon)^{-\lambda/\varepsilon},$$

$$(4.2) \qquad \theta_i \lambda_i \leq 0 \ (i = 1, 2, \ldots, n) \Rightarrow \sum_{i=1}^{n} \theta_i (1-\varepsilon_i)^{-\lambda_i/\varepsilon_i} \geq (1-\varepsilon)^{-\lambda/\varepsilon}. \quad \square$$

4.3. **Theorem.** *Let* $-\infty < \varepsilon < 1$, $-\infty < \lambda < 1$. *Then* $(1-\varepsilon)^{-\lambda/\varepsilon} - 1$ *lies strictly between* $\lambda/(1-\lambda)$ *and* $\lambda/(1-\varepsilon)$, *unless either* $\lambda = 0$ *or* $\lambda = \varepsilon$, *in which cases the three numbers coincide.* □

Note that for small $\lambda_i, \lambda, \varepsilon_i, \varepsilon$ the expressions to be estimated and the bounds in both Theorem 4.2 and Theorem 4.3 differ in terms of second and higher orders. In Theorem 4.1 the effect on $u(\lambda, \varepsilon)$ of a first-order change in $\varepsilon$ is of second order.

## 5. Proof of Theorem 3.1

The proof goes along the lines described at the end of §3; i.e., we have to find first values for $\varepsilon_{xk}$, $\tilde{\varepsilon}_{xi}$ which ensure the validity of (3.1). For the local error factors $q_{xk}$, $\tilde{q}_{xi}$ $(k = 1, \ldots, m, \ i = m+1, \ldots, n)$, we infer from (2.4) that (3.1) is satisfied with

$$(5.1) \qquad \varepsilon_{xk} = l_{xk}, \quad \tilde{\varepsilon}_{xi} = \tilde{l}_{xi} \qquad (k = 1, \ldots, m, \ i = m+1, \ldots, n).$$

For the rest of the error factors, namely for $q_{xk}$ $(k = m+1, \dots, n)$, the following propagation rules for $\varepsilon_{xk}$ are suggested by Theorems 4.1, 4.2:

(5.2a) Mult/Div: $\varepsilon_{xk} = \max\{\varepsilon_{yk}, \varepsilon_{zk}, \tilde{\varepsilon}_{xk}\}$;

(5.2b) Add/Sub:
$$\begin{cases} \varepsilon_{xk} = \max\{|\hat{l}_{xk} - l_{yk}\operatorname{sgn}(\theta_{yk})| + \varepsilon_{yk}, \\ \qquad\qquad |\hat{l}_{xk} - l_{zk}\operatorname{sgn}(\theta_{zk})| + \varepsilon_{zk}, \hat{l}_{xk}, \tilde{\varepsilon}_{xk}\}, \\ \text{where } \hat{l}_{xk}, \theta_{yk}, \theta_{zk} \text{ are defined in (2.3b);} \end{cases}$$

(5.2c) Power: $\varepsilon_{xk} = \max\{\varepsilon_{yk}, \tilde{\varepsilon}_{xk}\}$.

Thus, if $\varepsilon_{xk}$, $\tilde{\varepsilon}_{xi}$ are defined recursively by (5.1) and (5.2), then (3.1) is satisfied for all $k$ and all $i$.

Now we define recursively an auxiliary sequence $\{\varepsilon'_{xk}\}$:

$$\varepsilon'_{x1} = \cdots = \varepsilon'_{xm} = B,$$
$$\varepsilon'_{xk} = \begin{cases} \varepsilon'_{x,k-1} & \text{if } \circ_k \in \{\times, \div, \text{power}\} \\ \varepsilon'_{x,k-1} + 2B & \text{if } \circ_k \in \{+, -\} \end{cases} \quad (k = m+1, \dots, n).$$

We notice that $\{\varepsilon'_{xk}\}$ is not decreasing and therefore $\varepsilon'_{xk} \geq B$ $(k = 1, \dots, n)$. So, from the definition of $B$ (see Theorem 3.1) it is clear that for all $k$ and $j$, $\varepsilon'_{xk} \geq \tilde{\varepsilon}_{xj}$, $\varepsilon'_{xk} \geq l_{xj} \geq \hat{l}_{xj}$. Using this, we can easily prove, by induction, that $\varepsilon'_{xk} \geq \varepsilon_{xk}$ $(k = 1, \dots, n)$. Noting that $\varepsilon'_{xn} = \delta$ (see Theorem 3.1), we infer that $\delta \geq l_{xn}$, $\delta \geq \varepsilon_{xn}$. Also, from (3.1) we have $(1 - \varepsilon_{xn})^{l_{xn}/\varepsilon_{xn}} \leq q_{xn} \leq (1 - \varepsilon_{xn})^{-l_{xn}/\varepsilon_{xn}}$. Theorem 4.3 permits us to infer, therefore, that $|\rho_{xn}| = |q_{xn} - 1| \leq l_{xn}/(1 - \delta)$.

## 6. Examples

The examples below include, more or less, standard, linearized, forward error analyses (compare, e.g., Stummel [13]). The main new element is the factor $1/(1-\delta)$, which transforms the approximate, linearized bounds into exact bounds.

It would sometimes be convenient to use linearized, absolute errors rather than linearized relative errors. We shall introduce therefore the following notation for absolute linearized error bounds: $L$ will denote a linearized bound on absolute error in analogy with $l$, the linearized bound on relative error. Thus, if $x = y \circ z$, then the linearized bounds on absolute errors are related to the linearized bounds on relative errors by $L_x = |x| l_x$, $L_y = |y| l_y$, $L_z = |z| l_z$, $\tilde{L}_x = |x| \tilde{l}_x$.

The following linearized rules for the propagation of absolute error bounds are equivalent (in the exact sense) to those of relative error bounds, given in (2.3):

(6.1a) Mult/Div: $L_x = L_y/|\theta_y| + L_z/|\theta_z| + \tilde{L}_x$;

(6.1b) Add/Sub: $L_x = L_y + L_z + \tilde{L}_x$;

(6.1c) Power: $L_x = |z| L_y/|\theta_y| + \tilde{L}_x$.

The quantities $\theta_y$, $\theta_z$ are defined in (2.2c), (2.3b).

6.1. **Multiplications and divisions.** Suppose that we are given $m$ real numbers, $a_1, \dots, a_m$. We form an expression $E$ by somehow ordering them, inserting a single operation, of either multiplication or division, between each consecutive two, and adding any legitimate set of parentheses (e.g., $(a_3 \div a_1) \div (a_4 \times a_2)$). Let $l_{ak}$ be a true bound on the relative error $|(\bar{a}_k - a_k)/a_k|$ and $\tilde{l}$ a true bound on the relative error produced by any single arithmetic operation. We have $E = x_n$, where

$$x_i = a_i, \quad x_k = y_k \circ_k z_k, \quad \circ_k \in \{\times, \div\} \qquad (i = 1, \dots, m, \ k = m+1, \dots, n, \ n = 2m-1).$$

The linearized error propagation rules give

$$l_{xi} = l_{ai}, \quad l_{xk} = l_{yk} + l_{zk} + \tilde{l} \qquad (i = 1, \dots, m, \ k = m+1, \dots, n).$$

Denote by $I_k$ the set of indices of the $a$'s that compose the subexpression $x_k$. Obviously,

$$l_{xk} = \sum_{i \in I_k} l_{ai} + (|I_k| - 1)\tilde{l} \qquad (k = m+1, \dots, n),$$

where $|I_k|$ is the number of terms of $I_k$. Hence, Theorem 3.1 implies that

$$|(\overline{E} - E)/E| \le l_{xn}/(1-\delta), \quad \text{where } l_{xn} = \delta = \sum_{i=1}^{m} l_{ai} + (m-1)\tilde{l}.$$

6.2. **Additions and subtractions.** We denote $S_k = a_1 + a_2 + \cdots + a_k$ $(k = 1, \dots, m)$ and calculate $S_m$ by the recursion, $S_1 = a_1$, $S_k = S_{k-1} + a_k$ $(k = 2, \dots, m)$. Subtractions may be included by adjusting the signs of the $a$'s.

Again, $l_{ak}$ denotes a true bound on $|(\bar{a}_k - a_k)/a_k|$ and $\tilde{l}$ a true bound on the relative error introduced by a single arithmetic operation. The linearized propagation rules for absolute errors, (6.1), give

$$L_{s1} = |a_1| l_{a1}, \quad L_{sk} = L_{s,k-1} + |a_k| l_{ak} + |S_k|\tilde{l} \qquad (k = 2, \dots, m),$$

from which it follows that

$$L_{sk} = \sum_{i=1}^{k} |a_i| l_{ai} + \sum_{i=2}^{k} |S_i|\tilde{l}, \quad l_{sk} = L_{sk}/|S_k| \qquad (k = 2, \dots, m). \tag{6.2}$$

Theorem 3.1 then yields $|(\overline{S}_m - S_m)/S_m| \le l_{sm}/(1-\delta)$, where

$$l_{sm} = \left( \sum_{i=1}^{m} |a_i| l_{ai} + \sum_{i=2}^{m} |S_i|\tilde{l} \right) \Big/ |S_m|,$$

$$\delta = (2m-1) \max\{l_{ai} | i = 1, \dots, m\} \cup \{l_{sk} \,|\, k = 2, \dots, m\}.$$

If, owing to cancellation, any of the $S_k$'s is very small, then $\delta$ might grow to be larger than 1, in which case Theorem 3.1 is not applicable. This cannot happen if all of the $a_i$'s are of like signs. In such a case, $\delta$ may be simplified. Actually, from (6.2) we get in this case, $l_{sk} \le \max_{1 \le i \le k} l_{ai} + (k-1)\tilde{l}$. So one may take $\delta = (2m-1) \times (\max_{1 \le i \le m} l_{ai} + (m-1)\tilde{l})$.

6.3. **Euclidean norm.** Suppose that we are given $m \geq 2$ real numbers, $b_1, \ldots, b_m$, and that we want to compute $E = \sqrt{b_1^2 + \cdots + b_m^2}$. The algorithm is: $a_k = b_k \times b_k$, $S_0 = 0$, $S_k = S_{k-1} + a_k$ $(k = 1, \ldots, m)$, $E = \sqrt{S_m}$. We may apply the results of §6.2 in order to write down the linearized error bounds:

$$l_{sk} = \left( \sum_{i=1}^{k} a_i l_{ai} + \sum_{j=2}^{k} S_j \tilde{l} \right) \Big/ S_k ,$$

where $a_i = b_i^2$, $l_{ai} = 2l_{bi} + \tilde{l}$, $S_j = \sum_{i=1}^{j} b_i^2$ $(k = 2, \ldots, m)$.

This yields, after some algebra,

$$l_{sk} = \left[ 2 \sum_{i=1}^{k} b_i^2 l_{bi} + \left( k b_1^2 + \sum_{i=2}^{k} (k - i + 2) b_i^2 \right) \tilde{l} \right] \Big/ \sum_{i=1}^{k} b_i^2 \qquad (k = 2, \ldots, m).$$

The linearized bound on the relative error of the final output is therefore

$$l_E = \tfrac{1}{2} l_{sm} + \tilde{l} = \left[ \sum_{i=1}^{m} b_i^2 l_{bi} + \frac{1}{2} \left( (m+2) b_1^2 + \sum_{i=2}^{m} (m - i + 4) b_i^2 \right) \tilde{l} \right] \Big/ \sum_{i=1}^{m} b_i^2 .$$

Looking for a bound on all of the $l$'s, we get from these results

$$l_{ak} = 2l_{bk} + \tilde{l}, \quad l_{sk} \leq 2 \max_{1 \leq i \leq k} l_{bi} + k\tilde{l}, \quad l_E \leq \max_{1 \leq i \leq m} l_{bi} + \tfrac{1}{2}(m+2)\tilde{l}$$

$$(k = 1, \ldots, m),$$

so, since there are $m - 1$ additions, we may take

$$\delta = (2m - 1) \left( 2 \max_{1 \leq i \leq m} l_{bi} + m\tilde{l} \right).$$

The final result of the error analysis is, by Theorem 3.1,

$$|(\overline{E} - E)/E| \leq \left[ \sum_{i=1}^{m} b_i^2 l_{bi} + \frac{1}{2} \left( (m+2) b_1^2 + \sum_{i=2}^{m} (m - i + 4) b_i^2 \right) \tilde{l} \right] \Big/ \left( (1 - \delta) \sum_{i=1}^{m} b_i^2 \right).$$

This may be simplified into the somewhat weaker result

$$|(\overline{E} - E)/E| \leq \left( \max_{1 \leq i \leq m} l_{bi} + \tfrac{1}{2}(m+2)\tilde{l} \right) \Big/ (1 - \delta).$$

6.4. **Evaluation of** $\sin(t)$ **for a small** $t$. This simplified example demonstrates the use of a fictitious step for the inclusion of the truncation error in the analysis (see Note 3.3). A more complex and interesting example of an algorithm to evaluate $\exp(t)$ is given in Ziv [18].

We use the approximation $\sin(t) \simeq t - t^3/6$, $|t| \leq \alpha = 1/64$ for the evaluation of $\sin(t)$ in single precision of IEEE binary floating-point arithmetic. The relative roundoff error in a single arithmetic operation is bounded, then, by $\varepsilon = 2^{-24}$. The algorithm is given by: $x_1 = t$, $x_2 = 6$, $x_3 = x_1 x_1 = t^2$, $x_4 = x_1 x_3 = t^3$, $x_5 = x_4 / x_2 = t^3/6$, $x_6 = x_1 - x_5 = t - t^3/6$. The fictitious step is $x_7 = \sin(t)$.

For this algorithm, $l_{x1} = l_{x2} = 0$, $l_{x3} = \varepsilon$, $l_{x4} = 2\varepsilon$, $l_{x5} = 3\varepsilon$. Also,

$$l_{x6} = \left|\frac{x_5 l_{x5}}{x_6}\right| + \varepsilon = \left(\frac{3}{6/t^2 - 1} + 1\right)\varepsilon < 1.001\varepsilon$$

and $l_{x7} = l_{x6} + \tilde{l}_{x7}$, where $\tilde{l}_{x7}$ is an exact bound on the relative truncation error $(x_7 - x_6)/x_7$. But

$$\left|\frac{\sin(t) - (t - t^3/6)}{\sin(t)}\right| < \frac{\alpha^5/5!}{\sin(\alpha)} < 4.968 \times 10^{-10} < 0.008335\varepsilon.$$

So, $\tilde{l}_{x7} = 0.008335\varepsilon$ and $l_{x7} < 1.0094\varepsilon$.

Now we may use Theorem 3.1. We see that $\max l_{xk} = 3\varepsilon$ and that there is one addition operation in the algorithm. Hence, $\delta = 9\varepsilon < 5.4 \times 10^{-7}$. So,

$$\left|\frac{\bar{x}_7 - x_7}{x_7}\right| \leq \frac{1.0094\varepsilon}{1 - 5.4 \times 10^{-7}} < 1.01\varepsilon.$$

## 7. A POSTERIORI ERROR BOUNDS AND RUNNING ERROR ANALYSIS

Some of the analyses described in the examples section have characteristics of running error analysis. Thus, for instance, the bound for the additions and subtractions example includes the partial sums, $S_k$, which are natural by-products of the summation algorithm. One should notice, however, that this bound includes the theoretical, exact $S_k$'s rather than the approximate $\bar{S}_k$'s, which are more appropriate constituents of an a posteriori bound. We shall describe, in outline, a linearized analysis method and the proof of an analog of Theorem 3.1, which together constitute a systematic method of producing a posteriori bounds. The details are very similar to those of the a priori analysis described in the previous sections.

For exact propagation rules of error factors we take (compare to (2.2))

(7.1a) Multiply: $\bar{x}_k = \mathrm{fl}(y_k \times z_k) \Rightarrow q_{xk}^{-1} = q_{yk}^{-1} q_{zk}^{-1} \tilde{q}_{xk}^{-1}$;

(7.1b) Divide: $\bar{x}_k = \mathrm{fl}(y_k \div z_k) \Rightarrow q_{xk}^{-1} = (q_{yk}^{-1} \div q_{zk}^{-1})\tilde{q}_{xk}^{-1}$;

(7.1c) Add/Sub:
$$\begin{cases} \bar{x}_k = \mathrm{fl}(y_k \pm z_k) \Rightarrow q_{xk}^{-1} = \theta'_{yk} q_{yk}^{-1} + \theta'_{zk} q_{zk}^{-1}, \\ \quad \text{where } \theta'_{yk} = \bar{y}_k/\bar{x}_k, \ \theta'_{zk} = \pm\bar{z}_k/\bar{x}_k \\ \quad (\text{note that } \theta'_{yk} + \theta'_{zk} = \tilde{q}_{xk}^{-1}); \end{cases}$$

(7.1d) Power: $\bar{x}_k = \mathrm{fl}(y_k^{zk}) \Rightarrow q_{xk}^{-1} = (q_{yk}^{-1})^{zk} \tilde{q}_{xk}^{-1}$.

The main point to be noticed is that $\theta'_{yk}$, $\theta'_{zk}$ of (7.1c) depend on $\bar{x}_k$, $\bar{y}_k$, $\bar{z}_k$ rather than on $x_k$, $y_k$, $z_k$, as $\theta_{yk}$, $\theta_{zk}$ of (2.2c) do.

For linearized propagation rules of relative error bounds we take (compare to (2.3))

(7.2a) Mult/Div: $l'_{xk} = l'_{yk} + l'_{zk} + \tilde{l}'_{xk}$;

(7.2b) Add/Sub: $l'_{xk} = |\theta'_{yk}| l'_{yk} + |\theta'_{zk}| l'_{zk} + \tilde{l}'_{xk}$;

(7.2c) Power: $l'_{xk} = |z_k| l'_{yk} + \tilde{l}'_{xk}$.

Absolute, a posteriori error bounds are defined in terms of relative error bounds by $L'_{xk} = |\bar{x}_k| l'_{xk}$, $L'_{yk} = |\bar{y}_k| l'_{yk}$, $L'_{zk} = |\bar{z}_k| l'_{zk}$, $\tilde{L}'_{xk} = |\bar{x}_k| \tilde{l}'_{xk}$. Substituting

these relations in (7.2), we get (compare to (6.1))

(7.3a) Mult/Div: $L'_{xk} = L'_{yk}/|\theta'_{yk}| + L'_{zk}/|\theta'_{zk}| + \tilde{L}'_{xk}$;

(7.3b) Add/Sub: $L'_{xk} = L'_{yk} + L'_{zk} + \tilde{L}'_{xk}$;

(7.3c) Power: $L'_{xk} = |z_k| L'_{yk}/|\theta'_{yk}| + \tilde{L}'_{xk}$.

The sequence $\{l'_{xk}\}$ is constructed recursively by (7.2) and/or (7.3). The initial values of the recursion, $l'_{xk}$, $\tilde{l}'_{xi}$ $(k = 1, 2, \dots, m,\ i = m+1, \dots, n)$, are chosen, exactly as in the a priori case, to be true bounds of $\rho_{xk}$, $\tilde{\rho}_{xi}$ (compare to (2.4)). Parameters $\varepsilon'_{xk}$, $\tilde{\varepsilon}'_{xi}$ $(k = 1, 2, \dots, n,\ i = m+1, \dots, n)$ are chosen so as to satisfy the relations (compare to (3.1))

$$(7.4)\qquad \begin{cases} (1-\varepsilon'_{xk})^{l'_{xk}/\varepsilon'_{xk}} \le q_{xk}^{-1} \le (1-\varepsilon'_{xk})^{-l'_{xk}/\varepsilon'_{xk}} & (k = 1, 2, \dots, n), \\ (1-\tilde{\varepsilon}'_{xi})^{\tilde{l}'_{xi}/\tilde{\varepsilon}'_{xi}} \le \tilde{q}_{xi}^{-1} \le (1-\tilde{\varepsilon}'_{xi})^{-\tilde{l}'_{xi}/\tilde{\varepsilon}'_{xi}} & (i = m+1, \dots, n). \end{cases}$$

The initial terms are chosen to be $\varepsilon'_{xk} = l'_{xk}$, $\tilde{\varepsilon}'_{xi} = \tilde{l}'_{xi}$ $(k = 1, \dots, m,\ i = m+1, \dots, n)$ (compare to (5.1)). The remaining $\varepsilon'_{xk}$ $(k = m+1, \dots, n)$ are chosen, relying on the inequalities, to satisfy the recurrence relations (compare to (5.2))

(7.5a) Mult/Div: $\varepsilon'_{xk} = \max\{\varepsilon'_{yk}, \varepsilon'_{zk}, \tilde{\varepsilon}'_{xk}\}$;

(7.5b) Add/Sub: $\varepsilon'_{xk} = \max\{|l'_{xk} - \tilde{l}'_{xk}| + \tilde{\varepsilon}'_{xk},\ |l'_{xk} - l'_{yk}\operatorname{sgn}(\theta'_{yk})| + \varepsilon'_{yk},\ |l'_{xk} - l'_{zk}\operatorname{sgn}(\theta'_{zk})| + \varepsilon'_{zk},\ l'_{xk}\}$;

(7.5c) Power: $\varepsilon'_{xk} = \max\{\varepsilon'_{yk}, \tilde{\varepsilon}'_{xk}\}$.

In order to deduce (7.5b), we must use Theorem A.1 (see Appendix) rather than Theorem 4.2, because $\theta'_{yk} + \theta'_{zk} = \tilde{q}_{xk}^{-1} \ne 1$. From this result we get an analog of Theorem 3.1, which is quite similar to it:

7.1. **Theorem.** *Let* $\delta'$ *equal* $(2n_{\pm}+1)B'$, *where* $n_{\pm}$ *is the total number of* $\pm$ *operations in the algorithm and* $B' = \max_{1\le i\le n} l'_{xi}$. *If* $\delta' < 1$, *then the accumulated relative error satisfies* $|\rho_{xn}| = |q_{xn} - 1| \le l'_{xn}/(1-\delta')$. □

Applying this theorem to the additions and subtractions Example 6.2, one gets in place of (6.2)

$$L'_{sk} = \sum_{i=1}^{k} |\bar{a}_i| l'_{ai} + \sum_{i=2}^{k} |\bar{S}_i| \tilde{l}', \quad l'_{sk} = L'_{sk}/|\bar{S}_k| \qquad (k = 2, \dots, m),$$

and the final a posteriori result is

$$|(\bar{S}_m - S_m)/S_m| \le l'_{sm}/(1-\delta'),$$
$$\text{where } \delta' = (2m-1)\max\{l'_{ai} | i = 1, \dots, m\} \cup \{l'_{sk} \,|\, k = 2, \dots, m\}.$$

This may serve as the basis for a running error analysis: Start by setting $\bar{S}_1 = \bar{a}_1$, $l_1 = l_{a1}$, $L_1 = |\bar{a}_1| l_1$, $\delta_1 = l_1$ and then set, for $k = 2, \dots, n$, $\bar{S}_k = \mathrm{fl}(\bar{S}_{k-1} + \bar{a}_k)$, $L_k = L_{k-1} + |\bar{a}_k| l_{ak} + |\bar{S}_k| \tilde{l}$, $l_k = L_k/|\bar{S}_k|$, $\delta_k = \max\{\delta_{k-1}, l_{ak}, l_k\}$, $b_k = l_k/[1-(2k-1)\delta_k]$. Obviously, $\bar{S}_k = \mathrm{fl}(a_1 + \cdots + a_k)$ and $|(\bar{S}_k - S_k)/S_k| \le b_k$. Note that if $b_k$ is to be a true bound, the arithmetic operations leading to it must be exact. However, none of these operations may

cause cancellation. So, a standard finite-precision calculation produces very accurate numbers. In fact, unless $k$ is really large, a slight increase in, say, the third significant decimal digit of the approximate $b_k$ will produce a true bound. A theoretically more satisfactory alternative is inclusion of compensating factors in the evaluation of $b_k$ (see Olver [3]). But this is outside the range of this paper. Note that the recursion relations (7.2), (7.3) may serve for running error analysis in every algorithm. Measures may be taken to reduce the number of operations applied to produce $b_n$. But we shall not discuss this subject here.

## 8. Conclusions

Substitute measures were suggested in the literature, for the traditional relative error, in order to simplify roundoff error analysis. The advantages of these substitutes follow mainly from the simplification they imply on the propagation rules of error bounds. Although these new error measures do simplify error analysis, they do not achieve the degree of simplification achieved by linearized error analysis with the traditional relative error. Linearized error analysis, however, produces only approximate error bounds rather than true bounds.

In order to achieve the simplicity of linearized error analysis, while producing true bounds, a substitute error measure should imply exact error propagation rules which are all identical in form to the linearized propagation rules of relative error. It is demonstrated in Ziv [18] that such a substitute error measure does not exist. Instead we suggest that it is possible to perform linearized error analysis and transform, in a simple way, the approximate error bounds it produces into true bounds. This paper is devoted mainly to the demonstration of the feasibility of this idea for a priori, forward error analysis with scalars. The outlines of a theory of a posteriori error analysis and of running error analysis were described too.

We prove Theorem 3.1, which forms the basis for our method, and then give several examples of error analysis which demonstrate the method. The examples show that sometimes, in cases where catastrophic cancellation occurs, $\delta$ may grow to become larger than 1 in which case Theorem 3.1 is not applicable and the method fails. However, there exist interesting cases where cancellation is not significant. This includes, for instance, algorithms used in subroutines for the evaluation of elementary mathematical functions (see Ziv [17]). In such cases the method is efficient in producing sharp, true error bounds, while enjoying the convenience of linearized error analysis.

## Appendix

This appendix includes the proofs of the inequality theorems of §4.

*Proof of Theorem* 4.1. There is a doubt, as to analyticity, only when $\varepsilon = 0$. The doubt is resolved by the following expansion, which is produced from the binomial expansion for $(1-\varepsilon)^{-\alpha}$:

$$u(\lambda, \varepsilon) = 1+\frac{1}{1!}\lambda+\frac{1}{2!}\lambda(\lambda+\varepsilon)+\frac{1}{3!}\lambda(\lambda+\varepsilon)(\lambda+2\varepsilon)+\cdots, \qquad |\varepsilon| < 1, \ -\infty < \lambda < \infty.$$

The monotonicity properties follow from the identity

$$u(\lambda, \varepsilon) \equiv \exp\left(\lambda \int_0^1 \frac{dt}{1-\varepsilon t}\right), \qquad \varepsilon < 1, \ -\infty < \lambda < \infty. \quad \square$$

*Proof of Theorem* 4.2. In fact, we shall prove a slightly more general theorem. The generalization is necessary for the discussion in §7 of a posteriori analysis. Theorem 4.2 is obtained from the generalization by substituting $\lambda' = 0$, $\varepsilon' \to -\infty$, $q = 1$.

A1. **Theorem.** *Let* $\varepsilon'$, $\lambda'$, $\varepsilon_i$, $\lambda_i$, $\theta_i$ $(i = 1, 2, \ldots, n)$ *be real constants that satisfy* $\varepsilon' < 1$, $\varepsilon_i < 1$, *and let* $\sum \theta_i = q$. *Denote* $\lambda = \lambda' + \sum \theta_i \lambda_i$ *and let* $\varepsilon < 1$ *be a real number that satisfies* $\varepsilon \geq \max\{\lambda' - \lambda + \varepsilon', \lambda_i - \lambda + \varepsilon_i, -\lambda, \varepsilon', \varepsilon_i\}$ $(i = 1, 2, \ldots, n)$. *Then*

(A.1) $q \leq (1 - \varepsilon')^{-\lambda'/\varepsilon'}$, $\lambda' \geq 0$,

$$\theta_i \lambda_i \geq 0 \ (i = 1, 2, \ldots, n) \Rightarrow \sum_{i=1}^{n} \theta_i (1 - \varepsilon_i)^{-\lambda_i/\varepsilon_i} \leq (1 - \varepsilon)^{-\lambda/\varepsilon},$$

(A.2) $q \geq (1 - \varepsilon')^{-\lambda'/\varepsilon'}$, $\lambda' \leq 0$,

$$\theta_i \lambda_i \leq 0 \ (i = 1, 2, \ldots, n) \Rightarrow \sum_{i=1}^{n} \theta_i (1 - \varepsilon_i)^{-\lambda_i/\varepsilon_i} \geq (1 - \varepsilon)^{-\lambda/\varepsilon}.$$

Let us prove (A.1): We have

$$\begin{aligned}
\sum_i \theta_i (1 - \varepsilon_i)^{-\lambda_i/\varepsilon_i} - 1 &= (q - 1) + \sum_i \theta_i [(1 - \varepsilon_i)^{-\lambda_i/\varepsilon_i} - 1] \\
&\leq \int_0^1 \left[ \lambda' (1 - \varepsilon' t)^{-(\lambda' + \varepsilon')/\varepsilon'} + \sum_i \theta_i \lambda_i (1 - \varepsilon_i t)^{-(\lambda_i + \varepsilon_i)/\varepsilon_i} \right] dt \\
&\leq \int_0^1 \left[ \lambda' (1 - \varepsilon' t)^{-(\lambda + \varepsilon)/\varepsilon'} + \sum_i \theta_i \lambda_i (1 - \varepsilon_i t)^{-(\lambda + \varepsilon)/\varepsilon_i} \right] dt \\
&\leq \int_0^1 \left[ \lambda' (1 - \varepsilon t)^{-(\lambda + \varepsilon)/\varepsilon} + \sum_i \theta_i \lambda_i (1 - \varepsilon t)^{-(\lambda + \varepsilon)/\varepsilon} \right] dt \\
&= (1 - \varepsilon)^{-\lambda/\varepsilon} - 1.
\end{aligned}$$

The proof of (A.2) is similar. Only the directions of inequalities should be reversed. □

*Proof of Theorem* 4.3. The case is evident if either $\lambda = 0$ or $\lambda = \varepsilon$. So assume that $\varepsilon \neq \lambda \neq 0$. The discussion is separated into four cases: (i) $\lambda > 0$, $\lambda > \varepsilon$, (ii) $\lambda > 0$, $\lambda < \varepsilon$, (iii) $\lambda < 0$, $\lambda > \varepsilon$, (iv) $\lambda < 0$, $\lambda < \varepsilon$.

In case (i) we get from Theorem 4.1

$$(1 - \varepsilon)^{-\lambda/\varepsilon} - 1 < (1 - \lambda)^{-\lambda/\lambda} - 1 = \frac{\lambda}{1 - \lambda},$$

and using an appropriate identity yields

$$(1 - \varepsilon)^{-\lambda/\varepsilon} - 1 \equiv \lambda \int_0^1 (1 - \varepsilon t)^{-(\lambda + \varepsilon)/\varepsilon} \, dt > \lambda \int_0^1 (1 - \varepsilon t)^{-(\varepsilon + \varepsilon)/\varepsilon} \, dt = \frac{\lambda}{1 - \varepsilon}.$$

The proofs in the three other cases are almost identical. □

## BIBLIOGRAPHY


1. F. L. Bauer, *Computational graphs and rounding error*, SIAM J. Numer. Anal. **11** (1974), 87–96.
2. F. W. J. Olver, *A new approach to error arithmetic*, SIAM J. Numer. Anal. **15** (1978), 368–393.
3. ———, *Further developments of Rp and Ap error analysis*, IMA J. Numer. Anal. **2** (1982), 249–274.
4. ———, *Error analysis of complex arithmetic*, Computational Aspects of Complex Analysis (H. Werner et al., eds.), Reidel, Dordrecht, 1983, pp. 279–292.
5. ———, *Error bounds for polynomial evaluation and complex arithmetic*, IMA J. Numer. Anal. **6** (1986), 373–379.
6. ———, *Error bounds for linear recurrence relations*, Math. Comp. **50** (1988), 481–499.
7. F. W. J. Olver and J. H. Wilkinson, *A posteriori error bounds for Gaussian elimination*, IMA J. Numer. Anal. **2** (1982), 377–406.
8. J. D. Pryce, *A new measure of relative error for vectors*, SIAM J. Numer. Anal. **21** (1984), 202–215.
9. ———, *Multiplicative error analysis of matrix transformation algorithms*, IMA J. Numer. Anal. **5** (1985), 437–445.
10. R. Scherer and K. Zeller, *Shorthand notation for rounding errors*, Computing, Suppl. 2 (G. Alefeld et al., eds.), Springer-Verlag, New York, 1980, pp. 165–168.
11. Pat H. Sterbenz, *Floating-point computation*, Prentice-Hall, Englewood Cliffs, NJ, 1974.
12. J. Stoer and R. Bulirsch, *Introduction to numerical analysis*, 2nd printing, Springer-Verlag, Berlin and New York, 1983.
13. F. Stummel, *Rounding error analysis of elementary numerical algorithms*, Computing, Suppl. 2 (G. Alefeld et al., eds.), Springer-Verlag, New York, 1980, pp. 169–195.
14. J. H. Wilkinson, *Rounding errors in algebraic processes*, Prentice-Hall, Englewood Cliffs, NJ, 1963.
15. A. Ziv, *Relative distance—an error measure in round-off error analysis*, Math. Comp. **39** (1982), 563–569.
16. ———, *A stable method for the evaluation of a polynomial and of a rational function of one variable*, Numer. Math. **41** (1983), 309–319.
17. ———, *Fast evaluation of elementary mathematical functions with correctly rounded last bit*, ACM Trans. Math. Software **17** (1991), 410–423.
18. ———, *Converting approximate into true error bounds*, Technical Report 88.326, July 1992, Science and Technology, IBM Israel.


IBM ISRAEL, SCIENCE AND TECHNOLOGY, MATAM-ADVANCED TECHNOLOGY CENTER, HAIFA 31905, ISRAEL

*E-mail address*: ziv@haifasc3.vnet.ibm.com