# ERROR ESTIMATION FOR INDIRECT MEASUREMENTS IS EXPONENTIALLY HARD

Vladik Kreinovich

Computer Science Department
University of Texas at El Paso, El Paso, TX 79968

**Abstract**

For many physical quantities, it is impossible or too costly to measure them directly. In such cases, we measure whatever quantities $x_1, ..., x_n$ we can, and then we use the measurement results $\tilde{x}_1, ..., \tilde{x}_n$ to estimate the value of the desired quantity $y$. Namely, as an estimate, we use $\tilde{y} = f(\tilde{x}_1, ..., \tilde{x}_n)$, where an algorithm $f$ describes how the value $y$ is related to $x_i$. This procedure is called an *indirect measurement.* Measurements of $x_i$ cannot be absolutely precise. The resulting measurement errors $\Delta x_i = \tilde{x}_i - x_i$ cause the estimate $\tilde{y}$ to differ from the actual value $y$. How to estimate this difference (i.e., the error of the indirect measurement)?

Several error estimation algorithms are known for the case when $f$ describes the exact relationship between $x_i$ and $y$, and the only sources of error are the errors of direct measurements (i.e., $\Delta x_i$). In general, the problem of calculating the largest possible value of $\Delta y = \tilde{y} - y$ is proved to be intractable (NP-hard); however, in the realistic case when we can neglect squares of errors $\Delta x_i$, we can use a simple and fast (linear-time) algorithm for error estimation.

In many real-life situations, we know $f$ only approximately. We show that for such situations, the problem of estimating the exact bound for $\Delta y$ is intractable even when we can neglect the squares of errors $\Delta x_i$.

## 1. INTRODUCTION

### 1.1. Indirect measurements are really important

For many physical quantities $y$, it is impossible or too costly to measure them directly. For example, it is extremely difficult to measure directly the amount of oil in a given region. In such cases, we measure whatever quantities $x_1, ..., x_n$ we can, and then use the measurement results $\tilde{x}_1, ...\tilde{x}_n$ to estimate the value of the desired quantity $y$. For example, in the case of oil, we measure the results of reflecting sound waves in different layers, and then apply some geophysical model to reconstruct $y$. This procedure is called an *indirect measurement.*

To undertake an indirect measurement, we must know the relationship between $x_i$ and $y$. In some cases, we know the exact function $f$ that relates them, i.e., a function that transforms the value $x_i$ of the quantities that we measure into the value of the desired quantity $y$: $y = f(x_1, ..., x_n)$. In these cases, as an estimate for

$y$, we use the value $\tilde{y} = f(\tilde{x}_1, ..., \tilde{x}_n)$, where $\tilde{x}_i$ are measurement results.

In other situations, we do not know the function $f$ precisely, but we may know the function $\tilde{f}$ that is close to the actual function $f$. In such situations, we can take $\tilde{y} = \tilde{f}(\tilde{x}_1, ..., \tilde{x}_n)$ as an estimate for $y$.

*Comment.* The problems of indirect measurements form a part of the *theory of measurements*. This is one of the oldest application-related theories. There are so many interesting and relevant results in this area that it is practically impossible to enumerate them here. An interested reader is referred to (Vladimirovich, 1968) or (Rabinovich, 1993). Since this is a computations-oriented journal, in the following exposition, we do not assume that the reader is a specialist in measurement theory. We will explain all relevant problems and give all necessary definitions.

For those readers who *are* specialists in measurement theory, we have a warning: Our definitions will be slightly different from the ones that are traditionally given in measurement theory. The main reason for that difference is that in this paper, our primary concern is the problem that is rarely analyzed in general measurement theory: namely, estimating the *running time* of the algorithms that estimate measurement errors.

### 1.2. Errors of indirect measurements

Measurements of $x_i$ cannot be absolutely precise. The resulting measurement errors $\Delta x_i = \tilde{x}_i - x_i$ cause the estimate $\tilde{y}$ to differ from the actual value $y$. How to estimate this difference (i.e., the error of the indirect measurement)?

### 1.3. How are measurement errors usually described? (see, e.g., (Rabinovich, 1993))

The manufacturer of a measuring instrument must guarantee an upper bound for its measurement error. So, for every $i$, we know an accuracy $\Delta_i$ of the measuring instrument that is used to measure $x_i$ (in other words, we know $\Delta_i$ such that $|\Delta x_i| \leq \Delta_i$).

In some cases, we know more than that: namely, we know the probabilities of different values of $\Delta_i$. In such cases, estimating an error $\Delta y$ is a problem of mathematical statistics.

In many important cases, however, $\Delta_i$ is all we know. These cases will be considered in this paper.

### 1.4. How to estimate the error of an indirect measurement: case when we know $f$ precisely

Let us first consider the case when we know $f$ precisely. In this case, the only sources of error are the measurement errors $\Delta x_i$. The error $\Delta y = \tilde{y} - y$ of an indirect measurement can be expressed in terms of $\Delta x_i$ as follows:

$$\Delta y = f(\tilde{x}_1, ..., \tilde{x}_n) - f(x_1, ..., x_n) = f(\tilde{x}_1, ..., \tilde{x}_n) - f(\tilde{x}_1 - \Delta x_1, ..., \tilde{x}_n - \Delta x_n). \quad (1)$$

We know the measurement results $\tilde{x}_i$, the measurement accuracies $\Delta_i$, the function $f$. Based on this knowledge, we must calculate the accuracy of an indirect measurement, i.e., the value $\Delta = \sup |\Delta y|$, where $\Delta y$ is given by an expression (1), and supremum is taken over all $\Delta x_i$ for which $|\Delta x_i| \leq \Delta_i$.

### 1.5. In general, this problem is intractable (NP-hard)

It is known (Gaganov, 1985) that the problem of calculating $\Delta$ is intractable (or, to use a precise mathematical term, NP-hard, see, e.g., (Garey & Johnson, 1979)) even in the case when $f$ is a polynomial.

In plain words, this means that unless someone finds an algorithm that solves all possible discrete problems in time that is bounded by a polynomial of an input (which the majority of computer scientists believe to be impossible), any algorithm that calculates $\Delta y$ requires a running time that grows (at least for some inputs) exponentially with $n$ (i.e., as $a^n$ for some $a > 1$).

From the practical point of view, this means that we can still have polynomial algorithms that estimate $\Delta$, but none of these algorithms will give a precise estimate of $\Delta$: in some cases, they will be way off. Such algorithm have been proposed, e.g., in the so-called *interval analysis* (see, e.g., (Moore, 1979)).

### 1.6. If errors are small, then we can have a fast algorithm for error estimation

In the majority of measurements, the measurement errors are small. Actually, they are so small that we can safely neglect the terms that are quadratic (or of higher order) in terms of $\Delta x_i$. For example, if we have a 1% accuracy, then the errors are $\le 0.01$, and therefore, their squares are $\le 0.01^2 = 0.0001 \ll 0.01$.

In such situations, we can simplify the problem of estimating $\Delta$ if we expand the formula (1) (that expresses $\Delta y$ in terms of $\Delta x_i$) in Taylor series, and neglect quadratic and higher order terms. As a result, we get the following expression: $\Delta y = f_{,1}\Delta x_1 + ... + f_n \Delta x_n$, where $f_{,i}$ denotes the partial derivative of $f$ with respect to $x_i$. Since $-\Delta_i \le \Delta x_i \le \Delta_i$, the biggest possible value of $\Delta y$ is attained when each of the terms $f_{,i}\Delta x_i$ has the biggest possible value. If $f_{,i} > 0$, then this is attained when $\Delta x_i = \Delta_i$. If $f_{,i} < 0$, then the maximum is attained when $\Delta x_i$ takes the smallest possible value $-\Delta_i$. In both case, the biggest possible value of the term $f_{,i}\Delta x_i$ is $|f_{,i}|\Delta_i$, and the biggest possible value of the sum is equal to

$$\sum_{i=1}^{n} |f_{,i}|\Delta_i.$$

So, in this case,

$$\Delta = \sum_{i=1}^{n} |f_{,i}|\Delta_i.$$

This new expression for $\Delta$ looks a little bit simpler than the previous one (at least, it has no sup in it), but it is still not yet algorithmic, because we do not yet know how to compute the derivatives. To estimate them, we can use the same assumption as we used to deduce this expression: that the squares of errors are negligible. If we take $s_i$ equal to $\Delta_i$, expand the expression $d_i = f(\tilde{x}_1, ..., \tilde{x}_{i-1}, \tilde{x}_i + s_i, \tilde{x}_{i+1}, ..., \tilde{x}_n) - \tilde{y}$ into a Taylor series, neglect the terms that are quadratic (or of higher order) in $s_i$, we will end up with the formula $d_i = s_i f_{,i}$. Therefore, we can estimate $f_{,i}$ as $d_i/s_i$.

The expression $d_i$ is easily computable: we just apply $f$ twice, and subtract the resulting values. Totally, we must thus apply $f$ $n+1$ times (to compute $\tilde{y}$ and $n$ derivatives $f_{,i}$), and then simple computations will give us $\Delta$.

So, we get a linear-time algorithm for estimating error of an indirect measurement (actually, it can be even faster; see, e.g., (Kreinovich et al, 1991)).

### 1.7. What we are going to do

This was all about the case when we know $f$ precisely. But what if we know $f$ only approximately? We are going to prove that for this case, the problem of estimating $\Delta$ is intractable even in the case when we can neglect squares of errors $\Delta x_i$.

## 2. DEFINITIONS AND THE MAIN RESULT

**Denotations.**

- For an arbitrary $n$, by $d(\vec{x}, \vec{y})$ we will denote a usual Euclidean distance

$$\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}.$$

- By $B_r(\vec{x})$, we will denote a ball of radius $r$ with the center in $\vec{x}$, i.e., the set of all the points $\vec{y} \in R^n$ for which $d(\vec{x}, \vec{y}) \leq r$.

**Definition 1.** Assume that a positive integer $n$ is given. It will be called the *number of direct measurements.* By an *indirect measurement procedure*, we mean a pair $(\tilde{f}, \varepsilon)$, where:

- $\tilde{f}$ is an algorithm that inputs $n$ real numbers and returns a real number, and
- $\varepsilon > 0$ is a real number.

The algorithm $\tilde{f}$ will be called an *approximately computed function*, and the number $\varepsilon$ will be called called an *accuracy of computing* $\tilde{f}$.

**Definition 2.** Assume that an indirect measurement procedure is given.

- By the *result of direct measurements*, we mean a pair $(\vec{\tilde{x}}, \vec{\Delta})$ of $n-$dimensional vectors.
- The $i-$th component $\tilde{x}_i$ of a vector $\vec{\tilde{x}}$ is called the *result of $i-$th direct measurement.*
- The $i-$th component $\Delta_i$ of a vector $\vec{\Delta}$ will be called an *accuracy* of $i-$th measurement.
- We say that a real number $x_i$ is a *possible value* of $i-$th quantity if $|\tilde{x}_i - x_i| \leq \Delta_i$.
- The value $\tilde{f}(\vec{\tilde{x}})$ will be denoted by $\tilde{y}$ and called the *result of indirect measurement.*

*Comment.* In this subsection, we are going to prove that the problem of estimating the error of an indirect measurements is exponentially hard even for the case when terms quadratic in $\Delta x_i = \tilde{x}_i - x_i$ are negligible. In view of that, we will only give definitions for this approximately linear case.

**Definition 3.** Let $r > 0$ be a positive real number.

- We say that a function $f : R^n \to R$ is $r-$*possible* if for every $\vec{x} \in B_r(\vec{\tilde{x}})$, the values of $f$ and $\tilde{f}$ are $\varepsilon-$*close* (i.e., $|\tilde{f}(x_1, ..., x_n) - f(x_1, ..., x_n)| \le \varepsilon$).
- We say that for an approximately computed function, *quadratic terms are* $r-$*negligible* if there exists a linear function $f : R^n \to R$ that is $r-$possible.
- If quadratic terms are $r-$negligible for $\tilde{f}$, then we will also say that such a function $\tilde{f}$ is $r-$*approximately linear.* For such functions, the value $r$ will be called a *radius of (approximate) linearity.*
- We say that accuracies $\vec{\Delta}$ are *consistent with linearity* if $d(\vec{0}, \vec{\Delta}) \le r$ (i.e., if

$$\sum_{i=1}^{n} \Delta_i^2 \le r^2).$$

**Definition 4.** Suppose that $(\tilde{f}, \varepsilon)$ is an indirect measurement procedure, and $(\vec{\tilde{x}}, \vec{\Delta})$ are the results of direct measurements. Suppose also that for some $r > 0$, the algorithm $\tilde{f}$ is $r-$approximately linear, and that the accuracies $\Delta_i$ are consistent with linearity.

- We say that a real number $y$ is a *possible value* of the desired quantity if $y = f(x_1, ..., x_n)$ for some $r-$possible linear function $f$, and for some possible values $x_i$ of $i-$th quantity.
- If $y$ is a possible value of the desired quantity, then the value $e = |\Delta y|$, where $\Delta y = \tilde{y} - y$, is called a *possible value of an error* of the indirect measurement.
- The biggest possible value of $e$ will be denoted by $\Delta(\tilde{f})$ and called an *accuracy* of indirect measurement.

*Comments.*
1. According to this definition, the value that we denoted by $\Delta(\tilde{f})$ actually depends not only on the function $\tilde{f}$, but also on $\varepsilon$, $\vec{\tilde{x}}$, and $\vec{\Delta}$. If we wanted to express all these dependencies, then we would have to use a denotation $\Delta(\tilde{f}, \varepsilon, \vec{\tilde{x}}, \vec{\Delta})$. However, this is a somewhat clumsy denotation. So, we omit $\varepsilon$, $\vec{\tilde{x}}$, and $\vec{\Delta}$ from our denotation. In all our examples, we will explicitly mention what exactly values $\varepsilon$, $\vec{\tilde{x}}$, and $\vec{\Delta}$ we use.

2. For better understanding, let us repeat the definition of an accuracy $\Delta(\tilde{f})$ a little bit more informally. Suppose that we are given the following:

- a function $\tilde{f}$ of $n$ real variables;
- a real number $\varepsilon > 0$;
- $n$ real numbers $\tilde{x}_1, ..., \tilde{x}_n$ (called *results of direct measurements*);
- a real number $r > 0$;
- $n$ real numbers $\Delta_1, ..., \Delta_n$ (called *accuracies of direct measurements*).

First, we compute $\tilde{y} = \tilde{f}(\tilde{x}_1, ..., \tilde{x}_n)$ (this value $\tilde{y}$ is called the *result of indirect measurement*). Then, to compute $\Delta(\tilde{f})$, we must consider all possible values $x_1, ..., x_n$ for which $|x_i - \tilde{x}_i| \leq \Delta_i$, and all possible linear functions $f(x_1, ..., x_n)$ for which $|f(x_1, ..., x_n) - \tilde{f}(x_1, ..., x_n)| \leq \varepsilon$ for all $x_1, ..., x_n$ for which $\sqrt{(x_1 - \tilde{x}_1)^2 + ... + (x_n - \tilde{x}_n)^2} \leq r^2$. For each such $x_1, ..., x_n$ and $f$, we compute $y = f(x_1, ..., x_n)$, and $e = |\Delta y| = |y - \tilde{y}| = |f(x_1, ..., x_n) - \tilde{y}|$.

For different $x_1, ..., x_n$, and $f$, we will, generally speaking, get different values of $e$. The *biggest* of these values is the desired accuracy $\Delta(\tilde{g})$.

3. We want to compute this accuracy $\Delta(\tilde{f})$ with a given accuracy $\delta$. So, we arrive at the following problem:

**Definition 5.** By a *problem of estimating error of an approximately linear indirect measurement*, we mean the following problem:
*Given:*

- a function $\tilde{f} : R^n \to R$ (i.e., to be more precise, an algorithm that computes $\tilde{f}$);
- positive real numbers $\varepsilon$ and $r$;
- vectors $\vec{\tilde{x}} \in R^n$ and $\tilde{\Delta} \in R^n$ such that $\Delta_i > 0$ for all $i$ and $\sum_i \Delta_i^2 \leq r^2$;
- a positive real number $\delta$.

*To compute:* a number $\tilde{\Delta}$ for which $|\tilde{\Delta} - \Delta(\tilde{f})| \leq \delta$, where $\Delta(\tilde{f}) = \sup |\tilde{y} - y|$, $\tilde{y} = \tilde{f}(\vec{\tilde{x}})$, $y = f(x_1, ..., x_n)$, and supremum is taken over all $x_i$ such that $|\tilde{x}_i - x_i| \leq \Delta_i$ and over all linear functions $f : R^n \to R$ for which $|f(\vec{x}) - \tilde{f}(\vec{x})| \leq \varepsilon$ for every $\vec{x} \in B_r(\vec{\tilde{x}})$.

**Definition 6.** We say that an algorithm *solves* the problem of estimating error of approximately linear indirect measurements, if it computes the value $\tilde{\Delta}$ with the desired property for every indirect measurement procedure with an $r-$approximately linear $\tilde{f}$.

*Remark.* How are these inputs represented in a computer?

- We assume that all the numbers ($\tilde{x}_i, \varepsilon, r, \Delta_i$, and $\delta$) are given (as usual in the computers) by their binary representations (i.e., as a binary-rational numbers).

- The fact that a function $\tilde{f}$ is given, means (in computer terms) that we have a procedure (subroutine) that inputs $x_1, ..., x_n$ and returns $\tilde{f}(x_1, ..., x_n)$.

If we want to use this function in any algorithm, then we must substitute some values into it, and use the result (in computer terms, we must *call* $\tilde{f}$). So, we arrive at the following definition:

**Definition 7.** By a *computational step* of such an algorithm, we will mean either a standard computational step (e.g., $+, -, *, /$), or a call of $\tilde{f}$. By a *computational complexity* of an algorithm, we mean the number of its computational steps.

*Remark.* Of course, by counting the call as 1 computational step, we underestimate the computational complexity of an algorithm, but since we are going to prove an exponential lower bound for the number of calls, we will thus get an exponential lower bound for any other reasonable definition of a computational complexity.

**THEOREM.** *If an algorithm solves the problem of estimating errors of approximately linear indirect measurements, then its computational complexity is* $\geq 2^{n-1}$.

*Remark.* In real-life measurement procedures, we may have up to several thousand direct measurements. For $n = 1000$, the value $2^{n-1}$ is much greater than the total lifetime of the Universe. Therefore, what the Theorem says is that the problem of estimating errors of indirect measurements is intractable even when we restrict ourselves to approximately linear functions $\tilde{f}$.

## 3. PROOF

$1^o$. We will prove this Theorem by reduction to a contradiction. Assume that we have an algorithm $U$ that solves this problem and whose computational complexity for some $n$ is $< 2^{n-1}$. This means that this algorithm will produce the answer in $\leq 2^{n-1} - 1$ computational steps. In particular, it will produce the desired answer with $\leq 2^{n-1} - 1$ calls of $\tilde{f}$.

$2^o$. Let us show that this assumption leads to a contradiction. To show that, we will take $\tilde{x}_i = 0$, $\Delta_i = 1$, $\varepsilon = 1$, and $r = \sqrt{n}$ (we will choose $\delta$ later). As $\tilde{f}$, we will take a function that is identically 0. For this function, $\tilde{y}$ is always equal to 0. This function $\tilde{f}$ is linear, so it is $r-$approximately linear. If $f$ is $\varepsilon-$close to $\tilde{f}$, then $|f(x_1, ..., x_n) - \tilde{f}(x_1, ..., x_n)| = |y| \leq \varepsilon = 1$. Therefore, $|\Delta y| = |\tilde{y} - y| = |-y| = |y| \leq 1$. Hence, the supremum $\Delta(\tilde{f})$ of these values $|\Delta y|$ is also $\leq 1$: $\Delta(\tilde{f}) \leq 1$.

$3^o$. The algorithm $U$ will compute the value $\tilde{\Delta}$ that is $\delta-$close to $\Delta(\tilde{f}) \leq 1$. Therefore, $\tilde{\Delta} \leq \Delta(\tilde{f}) + \delta \leq 1 + \delta$.

$4^o$. This algorithm used $\leq 2^{n-1} - 1$ values of $\tilde{f}$. So, if we use another function $\tilde{g}$ whose values in the tested points $\vec{x}$ are the same, the algorithm $U$ will not notice the difference and thus produce the same result $\tilde{\Delta}$.

$5^o$. There are $2^n$ possible combinations of signs of $n$ values $x_1, ..., x_n$. We can divide them into $2^{n-1}$ pairs of opposite sign combinations $(\varepsilon_1, ..., \varepsilon_n)$ $(\varepsilon_i = \pm 1)$ and $(-\varepsilon_1, ..., -\varepsilon_n)$. Since the algorithm $U$ is applied to $\leq 2^{n-1} - 1 < 2^{n-1}$ different

vectors $\vec{x}$, this means that at least for one of these pairs, none of its combinations of signs appear as

$$(\text{sign}(x_1), ..., \text{sign}(x_i), ..., \text{sign}(x_n))$$

for some $\vec{x}$ to which we apply $\tilde{f}$. Let us choose one of such pairs, and denote one of its sign combinations by $(\varepsilon_1, ..., \varepsilon_n)$.

$6^o$. Let us now define $l(\vec{x})$ as

$$l(\vec{x}) = \frac{1}{\sqrt{n(n-1)}} \sum_{i=1}^{n} \varepsilon_i x_i,$$

and define $\tilde{g}(\vec{x})$ as follows:

- $\tilde{g}(\vec{x}) = l(\vec{x}) - 1$ if $l(\vec{x}) > 1$;
- $\tilde{g}(\vec{x}) = 0$ if $-1 \le l(\vec{x}) \le 1$;
- $\tilde{g}(\vec{x}) = l(\vec{x}) + 1$ if $l(\vec{x}) < -1$.

$7^o$. Let us prove that this new function $\tilde{g}$ attains the same values (identically 0) for all the vectors, to which $\tilde{f}$ has been applied while the algorithm $U$ was running.

Indeed, let $\vec{x}$ be one of such vectors. Because of our choice of signs $\varepsilon_i$, not all the signs of $x_i$ coincide with $\varepsilon_i$. For those $i$ for which $x_i$ and $\varepsilon_i$ have opposite signs, the product $\varepsilon_i x_i$ is non-positive. Therefore,

$$\sum_{i=1}^{n} \varepsilon_i x_i \le \sum_{i \in P} \varepsilon_i x_i,$$

where by $P$, we denoted the set of all $i$ for which the sign of $x_i$ coincides with $\varepsilon_i$. Since not all of these signs coincide, the number of elements $|P|$ in a set $P$ is smaller than $n$, i.e., $|P| \le n - 1$.

It is well known that the scalar (dot) product $a \cdot b = \sum a_i b_i$ of arbitrary two vectors cannot exceed the product of their lengths: $a \cdot b \le \sqrt{\sum a_i^2}\sqrt{\sum b_i^2}$. Therefore,

$$\sum_{i \in P} \varepsilon_i x_i = \sum_{i \in P} 1 \cdot |x_i| \le \sqrt{\sum_{i \in P} 1}\sqrt{\sum_{i \in P} |x_i|^2}.$$

But

$$\sum_{i \in P} 1 = |P| \le n - 1,$$

and

$$\sum_{i \in P} |x_i|^2 = \sum_{i \in P} \Delta_i^2 \le \sum_{i=1}^{n} \Delta_i^2 \le r^2 = n.$$

Therefore,

$$\sum_{i=1}^{n} \varepsilon_i x_i \le \sum_{i \in P} \varepsilon_i x_i \le \sqrt{n-1}\sqrt{n} = \sqrt{n(n-1)},$$

and

$$l(\vec{x}) = \frac{1}{\sqrt{n(n-1)}} \sum_{i=1}^{n} \varepsilon_i x_i \le 1.$$

Similarly, one can prove that $l(\vec{x}) \ge -1$. Therefore, according to our definition of a new function $\tilde{g}$, we have $\tilde{g}(\vec{x}) = 0$.

$8^o$. So, due to $4^o$, for this new function $\tilde{g}$, the algorithm $U$ will produce the same value for $\tilde{\Delta}$: the value that is $\le 1 + \delta$.

$9^o$. For this new function $\tilde{g}(\vec{x})$, because of the definition, we have $|\tilde{g}(\vec{x}) - l(\vec{x})| \le 1 = \varepsilon$. Therefore, $f = l$ is a possible linear function. The values $x_i = \varepsilon_i$ are possible values of $i-$th quantity, because $|\tilde{x}_i - x_i| = |0 - \varepsilon_i| = |\varepsilon_i| = 1 \le \Delta_i$. Let us now compute the corresponding value of

$$e = |\Delta y| = |y - \tilde{y}| = |f(x_1, ..., x_n) - \tilde{g}(\tilde{x}_1, ..., \tilde{x}_n)|.$$

First,

$$y = f(x_1, ..., x_n) = l(\varepsilon_1, ..., \varepsilon_n) = \frac{1}{\sqrt{n(n-1)}} \sum_{i=1}^{n} \varepsilon_i^2 = \frac{n}{\sqrt{n(n-1)}} = \sqrt{\frac{n}{n-1}}$$

(in terms of Definition 4, this $y$ is a possible value of the desired quantity).

Now, we have $\tilde{x}_i = 0$ (see $2^o$). Therefore, according to $6^o$, $l(\tilde{x}_1, ..., \tilde{x}_n) = 0$ and hence, according to our definition of $\tilde{g}$ (see $6^o$), we have $\tilde{y} = \tilde{g}(\tilde{x}_1, ..., \tilde{x}_n) = 0$. Therefore,

$$e = |\Delta y| = |y - \tilde{y}| = |\sqrt{\frac{n}{n-1}} - 0| = \sqrt{\frac{n}{n-1}}.$$

The value of $\Delta(\tilde{g})$ was defined (in Definition 4) as the biggest possible value of $e$ for all possible $r-$possible linear functions $f$ and for all possible values $x_i$ of $i-$th quantity. Formally,

$$\Delta(\tilde{g}) = \sup_{f, x_1, ..., x_n} e,$$

where $f$ runs over all $r-$possible linear functions, and $x_i$ run over all possible values of $i-$th quantity. We have shown the following:

- that $f = l$ is an $r-$possible linear function;
- that for every $i$, $x_i = \varepsilon_i$ are possible values of $i-$th quantity, and
- that for this choice ($f = l$, $x_i = \varepsilon_i$) we have $e = \sqrt{n/(n-1)}$.

Therefore, the supremum $\Delta(\tilde{g})$ of all possible values of $e$ cannot be smaller than this particular value of $e$, i.e., $\Delta(\tilde{g})$ cannot be smaller than $\sqrt{n/(n-1)}$.

$10^o$. Now, we are ready for a contradiction. Let's recall the following statements that we have assumed or proved so far:

- for every given $\tilde{f}$, an algorithm $U$ generates a value that is $\delta-$close to $\Delta(\tilde{f})$ (this is our main assumption);
- we have chosen the function $\tilde{g}$ in such a way that the result of applying $U$ to $\tilde{g}$ is exactly the same as the result $\tilde{\Delta}$ of applying $U$ to $\tilde{f}$ (see $8^o$);
- $\tilde{\Delta} \leq 1 + \delta$ (see $3^o$).

From the first two statements, we conclude that $|\Delta(\tilde{g}) - \tilde{\Delta}| \leq \delta$. In particular, it follows that $\Delta(\tilde{g}) \leq \tilde{\Delta} + \delta$. Using the third statement, we can now conclude that

$$\Delta(\tilde{g}) \leq 1 + \delta + \delta = 1 + 2\delta.$$

This is true for every choice of $\delta$. Let us now recall that in $2^o$, we have decided to postpone a choice of $\delta$ until later. Let us choose $\delta < 1/2\,(\sqrt{n/(n-1)} - 1)$. Then,

$$\Delta(\tilde{g}) \leq 1 + 2\delta < 1 + (\sqrt{\frac{n}{n-1}} - 1) = \sqrt{\frac{n}{n-1}}.$$

This inequality contradicts to $9^o$.

This contradiction shows that our initial assumption that we can have an algorithm $U$ (that solves the problem of estimating errors of approximately linear indirect measurements) with $< 2^{n-1}$ computational steps is wrong. Q.E.D.

## ACKNOWLEDGMENTS

This work was partially supported by an NSF grant No. CDA-9015006 and a Grant No. PF90–018 from the General Services Administration (GSA), administered by the Materials Research Institute. The author is thankful to Sergey Aityan, Pavel Maltsev, and Jiri Rohn for their interest and for their valuable comments. The author is also greatly thankful to the anonymous referees for their thorough analysis of the paper, and for their helpful suggestions.

## REFERENCES

Kreinovich, V., A. Bernat, E. Villa, & Y. Mariscal (1991). Parallel computers estimate errors caused by imprecise data, *Interval Computations*, v. 1, pp. 31-46.

Gaganov, A. A. (1985). Computational complexity of the range of the polynomial in several variables, *Cybernetics*, pp. 418–421.

Garey, M. & D. Johnson (1979). *Computers and intractability: a guide to the theory of NP-completeness*, San Francisco: Freeman.

Moore, R. E. (1979). *Methods and applications of interval analysis*, Philadelphia: SIAM.

Rabinovich, S. (1993). *Measurement errors: theory and practice*, N.Y.: American Institute of Physics.

Vladimirovich, S. (1968). *Theory of Measurements,* Moscow: Radio Publ. (in Russian).