The Accuracy of the Non-continuous I Test for One-Dimensional Arrays with References Created by Induction Variables

Qing Zhang*

Abstract—One-dimensional arrays with subscripts formed by induction variables in real programs appear quite frequently. For most famous data dependence testing methods, checking if integer-valued solutions exist for one-dimensional arrays with references created by induction variable is very difficult. The I test, which is a refined combination of the GCD and Banerjee tests, is an efficient and precise data dependence testing technique to compute if integer-valued solutions exist for one-dimensional arrays with constant bounds and single increments. In this paper, the non-continuous I test, which is an extension of the I test, is proposed to figure out whether there are integer-valued solutions for one-dimensional arrays with constant bounds and non-singular increments or not. Experiments with the benchmarks that have been cited from Livermore and Vector Loop, reveal that there are definitive results for 67 pairs of one-dimensional arrays that were tested.

Keywords—Data Dependence Analysis, Loop Parallelization, Loop Vectorization, Parallelizing/Vectorizing Compilers

1. INTRODUCTION

One, two, and three-dimensional array references approximately account for 56%, 36%, and 8% of the inspected array references [1], respectively. On the other hand, the author [2] indicated that loop normalization makes array references become more complex and brings parallel/vector compilers many difficulties in the source level debugging. Therefore, creating and applying an efficient and precise data dependence testing technique for one-dimensional arrays with constant bounds and non-singular increments is very important.

The data dependence problem is to check if two references to the same one-dimensional array within a nested loop with constant bounds and non-singular increments may refer to the same element of that array [3-7]. This problem in a general case can be reduced to that of examining whether a system of one linear equation with n unknown variables has a simultaneous integer-valued solution that satisfies the constraints for each variable in the system. Assume that a linear equation in a system is written as:

\[ a_1X_1 + a_2X_2 + \cdots + a_{n-1}X_{n-1} + a_nX_n = a_0, \]

(1-1)
where each \( a_j \) is an integer for \( 0 \leq j \leq n \) and each \( X_i \) is a scalar integer variable for \( 1 \leq k \leq n \). Suppose that the constraints to each variable in (1–1) are represented as:

\[
M_k \leq X_k \leq N_k, \quad X_k = M_k + (m-1) \times INC_k \quad \text{and} \quad 1 \leq m \leq P.
\]

(1-2)

Where \( M_k, N_k \) and \( INC_k \) are integers for \( 1 \leq k \leq n \) and \( M_k, N_k \) and \( INC_k \) are lower bound, upper bound, and the increment of a general loop, respectively, and \( P \) is the number of loop iterations in the general loop and \( P = (N_m - M_j) / INC_j \).

The GCD test, the Banerjee test, and Fourier-Motzkin elimination are three basic dependence analysis techniques but are too naive or expensive in practice [3,8-11]. There have been various advanced techniques to extend the above methods for overcoming the disadvantages of them [12-18]. The I test is a refined combination of the GCD and Banerjee tests [14,19-21], which is used to examine the existence of an integer-valued solution as the GCD test and additionally takes limits into account similar as the Banerjee test. However, the I test was originally devised to be employed in the cases that the increment of each loop index variable on an iteration is one. For the cases that the increment of the loop index variables on iteration is not one, the I test cannot be straightforwardly applied. Normalizing the loop index variables and array references to enable the I test to be applied is one way to deal with these cases. However, this creates many difficulties of source level debugging parallel/vector compilers, as already mentioned. Alternatively, we are proposing the non-continuous I test in this paper for these cases. By enabling the I test, our proposed testing technique, which extends the I test to directly manage the non-singular increments of the loop index variables on iterations, can efficiently and precisely determine data dependence for these cases the same as the I test does.

The rest of this paper is organized as follows: in Section 2, we review the fundamental notion of the I test. In Section 3, we present the non-continuous I test, which is an extension of the I test. In Section 4, the experimental results are given. In Section 5, we present our conclusions.

2. FUNDAMENTAL NOTATION OF THE I TEST

The summary accounts of data dependence and the interval equation are briefly introduced in this section.

2.1 Related Work

In this section, we introduce the fundamental notion for the proposed testing techniques based on the I test. The requisite notations are first given and the primary theorems and their application are then offered.

**Definition 2-1:** Let \( a \) be an integer.

\[
a^+ = a \quad \text{if} \quad a \geq 0, \quad 0 \quad \text{otherwise}
\]

\[
a^- = -a \quad \text{if} \quad a \leq 0, \quad 0 \quad \text{otherwise}
\]

**Definition 2-2:** Let \( a_0, a_1, a_2, \ldots, a_n \) be integers. For each \( k, 1 \leq k \leq n \), let each \( M_k \) and \( N_k \) be either an integer or a distinguished symbol ‘*’ (which means an unknown limit), where \( M_k \leq N_k \).
if both $M_i$ and $N_i$ are integers. If $n > 0$, then the equation:

$$a_1X_1 + a_2X_2 + \cdots + a_{n-1}X_{n-1} + a_nX_n = a_0$$

is said to be $(M_1, N_1; M_2, N_2; \ldots; M_n, N_n)$-integer solvable if the integers $j_1, j_2, \ldots, j_n$ exist, such that:

- $a_1 \times j_1 + a_2 \times j_2 + \cdots + a_n \times j_n = a_0$.
- for each $k$, $1 \leq k \leq n$:
  - if $M_k$ and $N_k$ are both integers, then $M_k \leq j_k \leq N_k$
  - if $M_k$ is an integer, and $N_k = *$, then $M_k \leq j_k$
  - if $M_k = *$, and $N_k$ is an integer, then $j_k \leq N_k$

**Definition 2-3:** Let $a_1, a_2, \cdots, a_n, L$ and $U$ be integers. An interval equation is an equation in the form of:

$$a_1X_1 + a_2X_2 + \cdots + a_{n-1}X_{n-1} + a_nX_n = [L, U], \quad (2-1)$$

which denotes the set of normal equations consisting of:

$$a_1X_1 + a_2X_2 + \cdots + a_{n-1}X_{n-1} + a_nX_n = L$$
$$a_1X_1 + a_2X_2 + \cdots + a_{n-1}X_{n-1} + a_nX_n = L + 1$$
$$\vdots$$
$$a_1X_1 + a_2X_2 + \cdots + a_{n-1}X_{n-1} + a_nX_n = U.$$

**Definition 2-4:** Given that the interval equation (2-1) is subject to the constraints as (1-2). Let $a_1, a_2, \ldots, a_n, L$ and $U$ be integers. If $n > 0$, then this interval equation is said to be $(M_1, N_1; M_2, N_2; \ldots; M_n, N_n)$-integer solvable if one or more of the equations in the set that it denotes is $(M_1, N_1; M_2, N_2; \ldots; M_n, N_n)$-integer solvable. If $L > U$, then this set is empty, and the interval equation has no integer-valued solution. If $n = 0$, this interval equation is said to be integer solvable, if and only if, $L \leq 0 \leq U$.

It is easy to make out that a linear equation as (1-1) is $(M_1, N_1; M_2, N_2; \ldots; M_n, N_n)$-integer solvable, if and only if, the following interval equation:

$$a_1X_1 + a_2X_2 + \cdots + a_{n-1}X_{n-1} + a_nX_n = [a_0, a_0] \quad (2-2)$$

is $(M_1, N_1; M_2, N_2; \ldots; M_n, N_n)$-integer solvable. While being applied each time, the I test initially operates on a single equation in the form of (1-1), which is subject to the constraint in the form of (1-2). It first applies the GCD test on all of the variable coefficients and then applies the Banerjee test (if the GCD test is successful) on the constant value on the right hand side of the original equation. If both tested results are positive, the I test transforms the original equation into an interval equation in the form of (2-2). We will now introduce the fundamental theorems of the I test to be applied, as shown below.

**Theorem 2-1:** Given that an interval equation as (2-1) is subject to the constraints as (1-2). Let
Let $a_1, a_2, \ldots, a_n, L$ and $U$ be integers. For each $k$, $1 \leq k \leq n - 1$, if $|a_k| \leq U - L + 1$, then the interval equation:

$$a_1 X_1 + a_2 X_2 + \cdots + a_{n-1} X_{n-1} + a_n X_n = [L, U],$$

is $(M_1, N_1; M_2, N_2; \ldots; M_n, N_n)$-integer solvable, if and only if, the interval equation:

$$a_1 X_1 + a_2 X_2 + \cdots + a_{n-1} X_{n-1} = [L-a_n N_n + a_n M_n, U-a_n M_n + a_n N_n]$$

is $(M_1, N_1; M_2, N_2; \ldots; M_n, N_n)$-integer solvable.

**Proof:** Refer to [14].

From Theorem 2-1, the I-test selects an item $a_k X_k$ for $1 \leq k \leq n$, in which the coefficient is small enough to satisfy $|a_k| \leq U - L + 1$. Then, the item is moved from the left hand side of the interval equation to the right hand side to calculate the new integer interval with its low and upper bounds. This process continues until either a definite result is obtained, or there are no more qualified items that can be moved.

**Theorem 2-2:** Let $a_1, a_2, \ldots, a_n, L$ and $U$ be integers. For each $k$, $1 \leq k \leq n - 1$, let each $M_k$ and $N_k$ be either an integer or a distinguished symbol “*”, where $M_k \leq N_k$ if both $M_k$ and $N_k$ are integers. Let $d = \gcd(a_1, a_2, \ldots, a_n)$. The interval equation:

$$a_1 X_1 + a_2 X_2 + \cdots + a_{n-1} X_{n-1} + a_n X_n = [L, U]$$

is $(M_1, N_1; M_2, N_2; \ldots; M_{n-1}, N_{n-1})$-integer solvable, if and only if, the interval equation:

$$
\left(\frac{a_1}{d}\right) X_1 + \left(\frac{a_2}{d}\right) X_2 + \cdots + \left(\frac{a_{n-1}}{d}\right) X_{n-1} + \left(\frac{a_n}{d}\right) X_n = \left\lceil \frac{L}{d} \right\rceil \left\lfloor \frac{U}{d} \right\rfloor
$$

is $(M_1, N_1; M_2, N_2; \ldots; M_n, N_n)$-integer solvable.

**Proof:** Refer to [14]

According to Theorem 2-1, the item $a_k X_k$ for $1 \leq k \leq n$ on the left hand side of the interval equation (2-2) is selected to be moved to the right hand side if its coefficient $a_k$ is small enough (i.e., $|a_k| \leq U - L + 1$). However, something this type of item cannot be immediately found, but may be obtained after transforming the original interval equation to enable all of the variable coefficients to become smaller. This can be achieved by doing something such as dividing the interval equation by the greatest common divisor for all of the variable coefficients. To be applied, the I test theoretically requires the increment of each index variable on an iteration to be one so that when an approved item is moved, it takes all the integers within the lower and upper bounds of the moved item to calculate the new integer interval within which all of the integers are continuous. However, there are many practical cases where the increment of each loop index on an iteration is not one [22-31]. To avoid the troubles caused by the loop normalization, the non-continuous I test has been proposed to cope with these cases. The idea behind the proposed testing technique is to extend the I test so that it can explicitly manage the non-singular increments of the loop index variables on an iteration.
3. The Non-Continuous I Test

For the cases where the increment of each loop index on an iteration is not one, the additional restriction, $INC_k > 1$, will be included in (1-2), where $INC_k$ is the increment of $X_k$ on an iteration. Thus, the constraint on each $X_k$ for $1 \leq k \leq n$ can be mathematically expressed with:

$$M_k \leq X_k \leq N_k, X_k = M_k + (m-1) \cdot INC_k$$

and

$$1 \leq m \leq \frac{N_k - M_k}{INC_k} + 1$$

for $1 \leq k \leq n$ \hspace{1cm} (3-1)

As mentioned, the proposed testing technique extends the I test to directly deal with the constraints on the loop index variable, as represented with (3-1). As such, the interval equation operated in the I test needs to be transformed correspondingly to achieve this. Before the single continuous I test is further discussed, we will first introduce its essential notations in Subsection 3.1.

3.1 Non-Continuous Interval Equation

**Definition 3-1**: Let $a_0, a_1, a_2, \cdots, a_n$ be integers. For each $k, 1 \leq k \leq n$, let each $M_k$ and $N_k$ be an integer, where $M_k \leq N_k$. If $n > 0$. The equation:

$$a_1 \times X_1 + a_2 \times X_2 + \cdots + a_n \times X_n = a_0$$

is then said to be $([M_1, N_1, INC_1, \frac{N_1 - M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2 - M_2}{INC_2} + 1]; \cdots; [M_n, N_n, INC_n, \frac{N_n - M_n}{INC_n} + 1])$-integer solvable if the integers $j_1, j_2, \cdots, j_n$ exist, such that:

- $a_1 \times j_1 + a_2 \times j_2 + \cdots + a_n \times j_n = a_0$.
- for each $k, 1 \leq k \leq n: j_k = M_k + (m-1) \cdot INC_k$, where $m$ is an integer and $1 \leq m \leq \frac{N_k - M_k}{INC_k} + 1$.

**Definition 3-2**: Let $a_1, a_2, \cdots, a_n, L$, and $U$ be integers. A non-continuous interval equation is an equation in the form of:

$$a_1 \times X_1 + a_2 \times X_2 + \cdots + a_n \times X_n = [L, U, INC, \frac{U - L}{INC} + 1]$$

which denotes the set of equations consisting of:

$$a_1 \times X_1 + a_2 \times X_2 + \cdots + a_n \times X_n = L$$
$$a_1 \times X_1 + a_2 \times X_2 + \cdots + a_n \times X_n = L + INC$$

\[ \cdots \]
\[ a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = L + \left( \frac{U-L}{\text{INC}} + 1 \right) \times \text{INC} = U. \]

The transformed interval equation, which is expressed with (3-2), is employed in the proposed testing technique to enable the constraints on the loop index variables, as represented with (3-1), to be directly and consistently operated. Obviously, if \( \text{INC} > 0 \), then the quadruplet, \([L, U, \text{INC}, \frac{U-L}{\text{INC}} + 1]\), represents an integer interval (i.e., \([L, U]\)) within which the actual integers contained are not continuous and is referred to as a non-continuous interval. The transformed interval equation is thus, a non-continuous interval equation. Clearly, the constraint, \([M_k, N_k, \text{INC}_k, \frac{N_k-M_k}{\text{INC}_k} + 1]\), for each index variable \(X_k\) is in itself a non-continuous integer interval.

**Definition 3-3.** Let \(a_1, a_2, \ldots, a_n, L,\) and \(U\) be integers. For each \(k, 1 \leq k \leq n\), let each \(M_k\) and \(N_k\) be an integer, where \(M_k \leq N_k\). If \(n > 0\), then the non-continuous interval equation:

\[ a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = [L, U, \text{INC}, \frac{U-L}{\text{INC}} + 1] \]

is said to be \(([M_1, N_1, \text{INC}_1, \frac{N_1-M_1}{\text{INC}_1} + 1]; [M_2, N_2, \text{INC}_2, \frac{N_2-M_2}{\text{INC}_2} + 1]; \ldots; [M_n, N_n, \text{INC}_n, \frac{N_n-M_n}{\text{INC}_n} + 1]\))-integer solvable if one or more of the equations in the set that it denotes is \(([M_1, N_1, \text{INC}_1, \frac{N_1-M_1}{\text{INC}_1} + 1]; [M_2, N_2, \text{INC}_2, \frac{N_2-M_2}{\text{INC}_2} + 1]; \ldots; [M_n, N_n, \text{INC}_n, \frac{N_n-M_n}{\text{INC}_n} + 1]\))-integer solvable.

It is easy to make out that an ordinary linear equation:

\[ a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = a_0 \]

is \(([M_1, N_1, \text{INC}_1, \frac{N_1-M_1}{\text{INC}_1} + 1]; [M_2, N_2, \text{INC}_2, \frac{N_2-M_2}{\text{INC}_2} + 1]; \ldots; [M_n, N_n, \text{INC}_n, \frac{N_n-M_n}{\text{INC}_n} + 1]\))-integer solvable, if and only if, the equation:

\[ a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = [a_0, a_0, \text{INC}, 1] \tag{3-3} \]

is \(([M_1, N_1, \text{INC}_1, \frac{N_1-M_1}{\text{INC}_1} + 1]; [M_2, N_2, \text{INC}_2, \frac{N_2-M_2}{\text{INC}_2} + 1]; \ldots; [M_n, N_n, \text{INC}_n, \frac{N_n-M_n}{\text{INC}_n} + 1]\))-integer solvable. According to Definitions 3-2 and 3-3, because the ordinary linear equation only contains one linear equation, \(L\) and \(U\) are both equal to \(a_0\). For the sake of \(L\) being equal to \(U\), the value of the third element in \([a_0, a_0, \text{INC}, 1]\) is set to \(\text{INC}\) and the value does not imply the correctness of the non-continuous interval, \([a_0, a_0, \text{INC}, 1]\), where \(\text{INC}\) is equal to the greatest common divisor of \(\text{INC}_1, \ldots, \text{INC}_n\). Since \(\frac{(a_0-a_0)}{\text{INC}+1}\) is equal to 1, the value of the fourth element is set to 1.

While being applied each time, the non-continuous I test initially operates on a single equation in the form of (1-1), which is subject to the constraints in the form of (3-1). It first transforms the original equation into an interval equation in the form of (3-3). Below, in Subsection 3.2, we present the fundamental theorems of the non-continuous I test to be applied.
3.2 Non-Continuous Interval Equation Transformation

Since the non-continuous I test deals with non-continuous interval equations, we began by considering the generalization of the GCD test to such equations.

**Theorem 3-1**: Let \( a_1, a_2, \cdots, a_n, L, U \) and \( \text{INC} \) be integers, and let \( d = \gcd(a_1, a_2, \cdots, a_n) \). The non-continuous interval equation:

\[
a_1 \times X_1 + a_2 \times X_2 + \cdots + a_n \times X_n = [L, U, \text{INC}, \ \frac{U-L+1}{\text{INC}}]
\]

has an integer solution, if and only if, \( d \times \lceil \frac{L}{d} \rceil \) is one element of the non-continuous integer set \( \{L + (m - 1) \times \text{INC} | 1 \leq m \leq \frac{U-L+1}{\text{INC}}\} \).

**Proof**: According to Definition 3-3 and the theorem that serves as the basis for the standard GCD test, the equation \( a_1 \times X_1 + a_2 \times X_2 + \cdots + a_n \times X_n = [L, U, \text{INC}, \ \frac{U-L+1}{\text{INC}}] \) has an integer solution, if and only if, a multiple of \( d \) belongs to the non-continuous integer interval \( [L, U, \text{INC}, \ \frac{U-L+1}{\text{INC}}] \). Let \( q_L \) and \( r_L \) be the quotient and remainder, respectively, upon dividing \( L \) by \( d \). Now \( \lceil \frac{L}{d} \rceil = \lceil \frac{q_L \times d + r_L}{d} \rceil \), which is equal to \( q_L \) if \( r_L = 0 \), and \( q_L + 1 \) otherwise. So, \( d \times \lceil \frac{L}{d} \rceil \) is equal to \( q_L \times d \) if \( r_L = 0 \), and \( q_L \times d + d \) otherwise.

Thus, \( d \times \lceil \frac{L}{d} \rceil \) is the first multiple of \( d \) that is equal to or greater than \( L \). If \( d \times \lceil \frac{L}{d} \rceil \cup 1 \) element of the non-continuous integer set \( \{L + (m - 1) \times \text{INC} | 1 \leq m \leq \frac{U-L+1}{\text{INC}}\} \), then no multiple of \( d \) is in \( [L, U, \text{INC}, \ \frac{U-L+1}{\text{INC}}] \). If it is one element of the non-continuous integer set \( \{L + (m - 1) \times \text{INC} | 1 \leq m \leq \frac{U-L+1}{\text{INC}}\} \), then there is a multiple of \( d \) in \( [L, U, \text{INC}, \ \frac{U-L+1}{\text{INC}}] \).

Like the I test, the non-continuous I test first applies the GCD test on all of the variable coefficients in the non-continuous interval equation, with each integer belonging to the non-continuous interval that may be examined. If a multiple of the greatest common divisor for all of the variable coefficients belongs to the non-continuous integer interval, for example: \( d \times \lceil \frac{L}{d} \rceil \) \in \( \{L + (m - 1) \times \text{INC} | 1 \leq m \leq \frac{U-L+1}{\text{INC}}\} \); then there may be a \( ([M_1, N_1, \text{INC}_1, \ \frac{N_1-M_1}{\text{INC}_1}] + 1); \cdots; [M_n, N_n, \text{INC}_n, \ \frac{N_n-M_n}{\text{INC}_n}] + 1) \)-integer solution. Otherwise, there is no integer solution.

**Lemma 3-1**: Let \( a_1, a_2, \cdots, a_n, L, U \) and \( \text{INC} \) be integers. For each \( k, 1 \leq k \leq n \), let each \( \text{INC}_k, M_k \) and \( N_k \) be an integer, where \( M_k \leq N_k \). If \( a_k > 0 \), \( \text{INC} > 0 \), \( \text{INC}_k > 0 \), \( 0 \leq a_k \times \text{INC}_k \leq U - L + \text{INC} \), and \( a_k \times \text{INC}_k \) is a multiple of \( \text{INC} \). Then, the non-continuous interval equation:

\[
a_1 \times X_1 + a_2 \times X_2 + \cdots + a_n \times X_n = [L, U, \text{INC}, \ \frac{U-L+1}{\text{INC}}]
\]

is \( ([M_1, N_1, \text{INC}_1, \ \frac{N_1-M_1}{\text{INC}_1}] + 1); [M_2, N_2, \text{INC}_2, \ \frac{N_2-M_2}{\text{INC}_2}] + 1); \cdots; [M_n, N_n, \text{INC}_n, \ \frac{N_n-M_n}{\text{INC}_n}] + 1) \)-integer solvable, if and only if, the non-continuous interval equation:
According to the assumption, because 
integer interval \([N_k - (p + 1) \times \text{INC}_k, U - a_k \times (N_k - (p + 1) \times \text{INC}_k)]\) to be integer solvable and 
continuous integer intervals \([N_k - \text{INC}_k, U - a_k \times \text{INC}_k]\) are listed 
in the following sequence in ascending order 
\([N_k - \text{INC}_k, U - a_k \times \text{INC}_k + 1]\). Because \(a_k > 0\), \(\text{INC} > 0\) and \(\text{INC}_k > 0\), these non-continuous 
integer intervals are listed in the following sequence in ascending order of initial element:

\[
[L - a_k \times N_k, U - a_k \times N_k, \text{INC}, \frac{U - L}{\text{INC}} + 1]
\]

\[
[L - a_k \times (N_k - \text{INC}_k), U - a_k \times (N_k - \text{INC}_k), \text{INC}, \frac{U - L}{\text{INC}} + 1]
\]

\[
\ldots
\]

\[
[L - a_k \times M_k, U - a_k \times M_k, \text{INC}, \frac{U - L}{\text{INC}} + 1].
\]

For any two consecutive non-continuous integer intervals \([L - a_k \times (N_k - p \times \text{INC}_k), U - a_k \times (N_k - p \times \text{INC}_k)]\), \(\text{INC}, \frac{U - L}{\text{INC}} + 1\) and \([L - a_k \times (N_k - (p + 1) \times \text{INC}_k), U - a_k \times (N_k - (p + 1) \times \text{INC}_k)]\), \(\text{INC}, \frac{U - L}{\text{INC}} + 1\), there is a gap, in terms of the increment INC, between the two non-continuous integer intervals, if and only if:

\[
U - a_k \times (N_k - p \times \text{INC}_k) + \text{INC} < L - a_k \times (N_k - (p + 1) \times \text{INC}_k).
\]

This inequality reduces to \(U - L + \text{INC} < a_k \times \text{INC}_k\), which is false by the above assumption. Therefore, there is no gap for any two consecutive non-continuous integer intervals.

Suppose that \(L - a_k \times (N_k - p \times \text{INC}_k) + a_k \times \text{INC}_k\) is the first element in the non-continuous integer interval \([L - a_k \times (N_k - (p + 1) \times \text{INC}_k), U - a_k \times (N_k - (p + 1) \times \text{INC}_k)]\), \(\text{INC}, \frac{U - L}{\text{INC}} + 1\).

According to the assumption, because \(a_k \times \text{INC}_k\) is a multiple of \(\text{INC}\) we assume that it is equal to \(q \times \text{INC}\), where \(q\) is an integer variable. Due to \(0 \leq a_k \times \text{INC}_k \leq U - L + \text{INC}\), we can
eventually obtain $0 \leq q \leq \frac{U-L}{INC} + 1$. This implies that two consecutive non-continued integer intervals can be merged as a new non-continued integer interval $[L - a_k \times (N_k - p \times INC_k), U - a_k \times (N_k - (p + 1) \times INC_k), INC, \frac{U-L+a_k\cdot INC_k}{INC} + 1]$. Thus, we have:

$$\frac{N_k-M_k}{INC} \bigcup_{p=0}^{U-a_k \times (N_k-p \times INC_k)} \frac{U-a_k \times (N_k-p \times INC_k), INC, \frac{U-L}{INC} + 1} = [L - a_k \times N_k, U - a_k \times M_k, INC, \frac{U-a_k \times M_k-A_k \times N_k}{INC} + 1].$$

The $z$, mentioned above, is obviously in one element of the set of non-continuous integer intervals $\{(L - a_k \times (N_k - p \times INC_k), U - a_k \times (N_k - p \times INC_k), INC, \frac{U-L}{INC} + 1) \mid 0 \leq p \leq \frac{N_k-M_k}{INC}\}$. Let $r, 0 \leq t \leq \frac{U-L}{INC}$, be the specific integer such that $z = L - a_k \times (N_k - p \times INC_k) + t \times INC$. Then, from $a_1 \times j_1 + \ldots + a_{k-1} \times j_{k-1} + a_k \times j_k + \ldots + a_n \times j_n = z$, we can have $a_1 \times j_1 + \ldots + a_{k-1} \times j_{k-1} + a_k \times j_k + \ldots + a_n \times j_n = L - a_k \times (N_k - p \times INC_k) + t \times INC$. This reduces to: $a_1 \times j_1 + \ldots + a_{k-1} \times j_{k-1} + a_k \times (N_k - p \times INC_k) + a_{k+1} \times j_{k+1} + \ldots + a_n \times j_n = L + t \times INC$.

Since $N_k - p \times INC_k$ is one element in the non-continued integer interval $[M_k, N_k, INC_k, \frac{N_k-M_k}{INC_k} + 1]$ and $L + t \times INC$ is one element in the non-continued integer interval $[L, U, INC, \frac{U-L}{INC} + 1]$, we can obtain that the non-continuous interval equation $a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = [L, U, INC, \frac{U-L}{INC} + 1]$ is $([M_1, N_1, INC_1, \frac{N_1-M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2-M_2}{INC_2} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n-M_n}{INC_n} + 1])$-integer solvable.

**Proof:** (only if) Let $a_1 \times j_1 + \ldots + a_{n} \times j_{n} = L + t \times INC$, where $j_1, \ldots, j_n$ satisfy the conditions for $([M_1, N_1, INC_1, \frac{N_1-M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2-M_2}{INC_2} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n-M_n}{INC_n} + 1])$-integer solvable and $0 \leq t \leq \frac{U-L}{INC}$. We can thus obtain $a_1 \times j_1 + \ldots + a_{k-1} \times j_{k-1} + a_k \times j_k + \ldots + a_n \times j_n = L - a_k \times (N_k - p \times INC_k) + (t \times INC)$, where $0 \leq p \leq \frac{N_k-M_k}{INC_k}$. Due to the fact that $L - a_k \times (N_k - p \times INC_k) + t \times INC$ is in the non-continuous integer interval $[L - a_k \times (N_k - p \times INC_k), U - a_k \times (N_k - p \times INC_k), INC, \frac{U-L+a_k\cdot INC_k}{INC} + 1]$ and $\frac{N_k-M_k}{INC} \bigcup_{p=0}^{\frac{U-L+a_k\cdot INC_k}{INC}} [L - a_k \times (N_k - p \times INC_k), U - a_k \times (N_k - p \times INC_k), INC, \frac{U-L+a_k\cdot INC_k}{INC} + 1]$, $L - a_k \times (N_k - p \times INC_k) + t \times INC$ is obviously in the non-continuous integer interval $[L - a_k \times N_k, U - a_k \times M_k, INC, \frac{U-a_k \times M_k-A_k \times N_k}{INC} + 1]$. This implies that the non-continuous interval equation:
The Accuracy of the Non-continuous I Test for One-Dimensional Arrays with References Created by Induction Variables

\[ a_1 \times X_1 + \ldots + a_{k-1} \times X_{k-1} + a_k \times X_k + \ldots + a_n \times X_n = \]
\[ [L - a_k \times N_k, U - a_k \times M_k, INC, \frac{U - a_k \times M_k - L + a_k \times N_k}{INC} + 1] \]

is \( ([M_1, N_1, INC_1, \frac{N_1 - M_1}{INC_1} + 1]; \ldots; [M_{k-1}, N_{k-1}, INC_{k-1}, \frac{N_{k-1} - M_{k-1}}{INC_{k-1}} + 1]; [M_k + 1, N_k + 1, \]
\[ INC_{k+1}, \frac{N_{k+1} - M_{k+1}}{INC_{k+1}} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n - M_n}{INC_n} + 1]) \) - integer solvable. \[ \blacksquare \]

**Lemma 3-2:** Let \( a_1, a_2, \ldots, a_n, L, U \) and \( INC \) be integers. For each \( k, 1 \leq k \leq n \), let each \( INC_k, M_k \) and \( N_k \) be an integer, where \( M_k \leq N_k \). If \( a_k < 0, INC > 0, INC > 0, 0 \leq -a_k \times INC_k \leq U - L + INC \), and \(-a_k \times INC_k \) is a multiple of \( INC \). Then, the non-continuous interval equation:

\[ a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = [L, U, INC, \frac{U - L}{INC} + 1] \]

is \( ([M_1, N_1, INC_1, \frac{N_1 - M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2 - M_2}{INC_2} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n - M_n}{INC_n} + 1]) \) - integer solvable, if and only if, the non-continuous interval equation:

\[ a_1 \times X_1 + \ldots + a_{k-1} \times X_{k-1} + a_k \times X_k + \ldots + a_n \times X_n = \]
\[ [L - a_k \times M_k, U - a_k \times N_k, INC, \frac{U - a_k \times M_k - L + a_k \times N_k}{INC} + 1] \]

is \( ([M_1, N_1, INC_1, \frac{N_1 - M_1}{INC_1} + 1]; \ldots; [M_{k-1}, N_{k-1}, INC_{k-1}, \frac{N_{k-1} - M_{k-1}}{INC_{k-1}} + 1]; [M_k + 1, N_k + 1, \]
\[ INC_{k+1}, \frac{N_{k+1} - M_{k+1}}{INC_{k+1}} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n - M_n}{INC_n} + 1]) \) - integer solvable. \[ \blacksquare \]

**Proof:** Similar to the proof of Lemma 3-1. \[ \blacksquare \]

We will use the example below to show the strength of Lemmas 3-1 and 3-2. Consider the following linear equation:

\[ X_1 - 2 \times X_2 + 3 \times X_3 = 3 \]

(Ex.1)

which is subject to the constraints \( X_1 \in [1, 5, 1, 5], X_2 \in [2, 6, 2, 3] \) and \( X_3 \in [1, 5, 2, 3] \).

First, the greatest common divisor for 1, 2 and 2 is 1, so the value for \( INC \) is equal to 1. Hence, the non-continuous I test transforms the equation (Ex.1) into the following non-continuous interval equation:

\[ X_1 - 2 \times X_2 + 3 \times X_3 = [3, 3, 1, 1]. \]

(Ex.1-1)

By using Lemma 3-1, \( X_1 \) is selected to be moved to the right hand side due to the fact that \( 0 \leq a_1 \times INC_1 \leq U - L + INC \) \((0 \leq 1 \times 1 \leq (3 - 3 + 1))\) and \( a_1 \times INC_1 \) is a multiple of \( INC \) (1 is a multiple of 1). This gives rise to a new non-continuous interval equation of:

\[ -2 \times X_2 + 3 \times X_3 = [-2, 2, 1, 5]. \]

(Ex.1-2)
Then, by using Lemma 3-2, \(-2 \times X_2\) is selected to be moved to the right hand side due to the fact that \(0 \leq -a_2 \times INC_2 \leq U - L + INC\) \((0 \leq -2 \times 2 = 4 \leq (2 - (-2) + 1)=5)\) and \(-a_2 \times INC_2\) is a multiple of \(INC\) \((4\) is a multiple of \(1)\). This results in a new non-continuous interval equation of:

\[
3 \times X_3 = [2, 14, 1, 13].
\]  
(Ex.1-3)

By using Lemma 3-1, \(3 \times X_3\) is selected to be moved to the right hand side, since \(0 \leq a_3 \times INC_3 \leq U - L + INC\) \((0 \leq 3 \times 2 = 6 \leq (14 - 2 + 1) = 13)\) and \(a_3 \times INC_3\) is a multiple of \(INC\) \((6\) is a multiple of \(1)\). This leads to a new non-continuous interval equation of:

\[
0 = [-13, 11, 1, 25].
\]  
(Ex.1-4)

Apparently, 0 is one element in the non-continuous integer interval \([-13, 11, 1, 25]\). Hence, the non-continuous I test proves that there are integer solutions.

### 3.3 Interval Equation Transformation Using the GCD Test

Obviously, as seen in Lemmas 3-1 and 3-2, the proposed method considers justifying the movement of any variable to the right. Any variable in a non-continuous interval equation can be moved to the right if the coefficient for it has small enough values to justify the movement of the variable to the right. If all of the coefficients for variables in the non-continuous interval equation do not have sufficiently small enough values to justify the movements of variables to the right, then Lemmas 3-1 and 3-2 cannot be applied to the immediate movement. While every variable in a non-continuous interval equation cannot be moved to the right, Lemma 3-3 describes a transformation using the GCD test, which enables additional variables to be moved.

**Lemma 3-3:** Let \(a_1, a_2, \ldots, a_n, L, U\) and \(INC\) be integers. For each \(k, 1 \leq k \leq n\), let each \(\text{gcd}\), \(M_k\) and \(N_k\) be an integer, where \(M_k \leq N_k\). Let \(d = \text{gcd}(a_1, a_2, \ldots, a_n)\) and \(L, U\), and \(INC\) are a multiple of \(d\), respectively. Then the non-continuous interval equation:

\[
a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = [L, U, INC, \frac{U-L+1}{INC}]
\]

is \([M_1, N_1, INC_1, \frac{N_1-M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2-M_2}{INC_2} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n-M_n}{INC_n} + 1])\)-integer solvable, if and only if, the non-continuous interval equation:

\[
\left(\frac{a_1}{d}\right)X_1 + \left(\frac{a_2}{d}\right)X_2 + \ldots + \left(\frac{a_n}{d}\right)X_n = \left[\frac{L}{d}, \frac{U}{d}, \frac{INC}{d}, \frac{U-L}{INC} + 1\right]
\]

is \([M_1, N_1, INC_1, \frac{N_1-M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2-M_2}{INC_2} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n-M_n}{INC_n} + 1])\)-integer solvable.

**Proof:** (if) First, suppose that \(a_1 \times j_1 + a_2 \times j_2 + \ldots + a_n \times j_n = z\), where \(j_1, j_2, \ldots, j_n\) satisfy the conditions of \([M_1, N_1, INC_1, \frac{N_1-M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2-M_2}{INC_2} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n-M_n}{INC_n} + 1])\) integer-solvable, and \(z\) is one element in the non-continuous integer interval \([L, U, INC, \frac{U-L+1}{INC}])\), which is equal to \(L + p \times INC\) for \(0 \leq p \leq
The Accuracy of the Non-continuous I Test for One-Dimensional Arrays with References Created by Induction Variables

\[ \frac{U-L}{INC} \]

By the assumption that \( L, U, \) and \( INC \) are a multiple of \( d \), respectively; then, let \( L = r_1 \times d, U = s_1 \times d \) and \( INC = t_1 \times d \), where \( r_1, s_1, \) and \( t_1 \) are the integers. Subsequently, \( z \) is one element in the non-continuous integer interval \( [r_1 \times d, s_1 \times d, t_1 \times d, \frac{n-n}{t_1} + 1) \) and is equal to \( r_1 \times d + p \times t_1 \times d \) for \( 0 \leq p \leq \frac{n-n}{t_1} \). We thus have \( a_1 \times j_1 + a_2 \times j_2 + \ldots + a_n \times j_n = d \times (r_1 + p \times t_1) \) or

\[ \left( \frac{a_1}{d} \right) \times j_1 + \left( \frac{a_2}{d} \right) \times j_2 + \ldots + \left( \frac{a_n}{d} \right) \times j_n = r_1 + p \times t_1. \]

Because \( r_1 = \frac{L}{d}, s_1 = \frac{U}{d}, t_1 = \frac{INC}{d} \) and \( \frac{n-n}{t_1} = \frac{U-L}{INC} \); then \( (r_1 + p \times t_1) \) is one element in the non-continuous integer interval \( \left[ \frac{L}{d}, \frac{U}{d}, \frac{INC}{d}, \frac{U-L}{INC} + 1 \right] \). Hence, the non-continuous integer equation

\[ \left( \frac{a_1}{d} \right) X_1 + \left( \frac{a_2}{d} \right) X_2 + \ldots + \left( \frac{a_n}{d} \right) X_n = \left( \frac{L}{d}, \frac{U}{d}, \frac{INC}{d}, \frac{U-L}{INC} + 1 \right) \]

is one element in the non-continuous integer interval \( \left[ \frac{L}{d}, \frac{U}{d}, \frac{INC}{d}, \frac{U-L}{INC} + 1 \right] \) and is equal to \( \frac{(L+ p \times INC)}{d} \) for \( 0 \leq p \leq \frac{U-L}{INC} \). We then have

\[ \left( \frac{a_1}{d} \right) \times j_1 + \left( \frac{a_2}{d} \right) \times j_2 + \ldots + \left( \frac{a_n}{d} \right) \times j_n = \frac{(L+ p \times INC)}{d}. \]

By the assumption that \( L, U, \) and \( INC \) are a multiple of \( d \), respectively; then, let \( L = r_1 \times d, U = s_1 \times d \) and \( INC = t_1 \times d \), where \( r_1, s_1, \) and \( t_1 \) are integers. Subsequently, \( z \) is one element in the non-continuous integer interval \( \left[ r_1, s_1, t_1, \frac{n-n}{t_1} \right] \) and is equal to \( r_1 + p \times t_1 \) for \( 0 \leq p \leq \frac{n-n}{t_1} \). We thus have \( \left( \frac{a_1}{d} \right) \times j_1 + \left( \frac{a_2}{d} \right) \times j_2 + \ldots + \left( \frac{a_n}{d} \right) \times j_n = r_1 + p \times t_1 \) or \( a_1 \times j_1 + a_2 \times j_2 + \ldots + a_n \times j_n = d \times (r_1 + p \times t_1) \). Because \( d \times (r_1 + p \times t_1) = L + p \times INC \) and \( \frac{n-n}{t_1} = \frac{U-L}{INC} \), we have the fact that \( d \times (r_1 + p \times t_1) \) is one element in the non-continuous integer interval \( \left[ L, U, INC, \frac{U-L}{INC} + 1 \right] \). Hence, the non-continuous integer equation \( a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = \left[ L, U, INC, \frac{U-L}{INC} + 1 \right] \) is one element in the non-continuous integer interval \( \left[ L, U, INC, \frac{U-L}{INC} + 1 \right] \); \( \left[ M_1, N_1, INC, \frac{N_1-M_1}{INC} + 1 \right] \); \( \left[ M_2, N_2, INC, \frac{N_2-M_2}{INC} + 1 \right] \) integer-solvable.

Consider the following Fortran do-loop in Fig. 1(a). Since the do-loop is an unnormalized Fortran do-loop, it is transformed into the following normalized Fortran do-loop from the do-loop normalization in the parallel/vector compiler, as shown in Fig. 1(b). The data dependence equation for the Fortran normalized do-loop in Fig. 1(b) is shown below.

```
DO1=4,20,4
DO %I=1,5,1
ENDDO
S1: A(I+4)=A(2*I)+N*M
ENDDO
S1: A(4+4*%I)=A(8*%I)+N*M
ENDDO
I=24
```

Fig. 1. A Fortran do-loop with constant bounds and non-one-increment. (a) An unnormalized Fortran do loop. (b) A normalized Fortran do-loop.
4 \times X_1 - 8 \times X_2 = -4, \quad \text{(Ex.2)}

subject to the limits 1 \leq X_1 \leq 5 and 1 \leq X_2 \leq 5.

When the I test is used to deal with the equation (Ex.2), the equation (Ex.2) is transformed into the following interval equation:

\[ 4 \times X_1 - 8 \times X_2 = [-4, -4]. \quad \text{(Ex.2-1)} \]

Because the coefficients for variables \( X_1 \) and \( X_2 \) do not satisfy the condition of the movement, Theorem 2-1 cannot be applied to deal with the interval equation (Ex.2-1). However, \( \gcd(4, -8) = 4 \) from Theorem 2-2, the interval equation (Ex.2-1) is transformed into the following interval equation:

\[ X_1 - 2 \times X_2 = [-1, -1]. \quad \text{(Ex.2-2)} \]

Since the coefficient for \( X_1 \) is 1, it satisfies the condition 1 \(|1| = 1 \leq 1 - (-1) + 1 = 1 \) from Theorem 2-1. Hence, from Theorem 2-1, the interval equation (Ex.2-2) is transformed into the following interval equation:

\[-2 \times X_2 = [-6, -2]. \quad \text{(Ex.2-3)} \]

According to Theorem 2-2, because \( \gcd(-2) = 2 \), the interval equation (Ex.2-3) is transformed into the following interval equation:

\[-X_2 = [-3, -1]. \quad \text{(Ex.2-4)} \]

Since the coefficient for \( X_2 \) is -1, according to Theorem 2-1, the interval equation (Ex.2-4) is transformed into the following interval equation:

\[ 0 = [-2, 4]. \quad \text{(Ex2-5)} \]

Because \(-2 \leq 0 \leq 4\), the I test proves that there are integer-valued solutions.

On the other hand, the data dependence equation for the Fortran unnormalized do-loop in Fig. 1(a) is shown below:

\[ X_1 - 2 \times X_2 = -4, \quad \text{(Ex.3)} \]

subject to the limits \( X_1 \in [4, 20, 4, 5] \) and \( X_2 \in [4, 20, 4, 5] \).

When the non-continuous I test is applied to deal with the equation (Ex.3), the equation (Ex.3) is transformed into the following non-continuous interval equation:

\[ X_1 - 2 \times X_2 = [-4, -4, 4, 1], \quad \text{(Ex3-1)} \]

Where \( \text{INC} = \gcd(4, 4) = 4 \). Since the coefficient for \( X_1 \) is one, according to Lemma 3-1, it satisfies \( 4 \times 4 = 4 \leq 4 \times 4 = 4 \) and 4 is a multiple of 4. Thus, according to Lemma 3-1, the non-continuous interval equation (Ex.3-1) is transformed into the following non-continuous interval equation:
\[-2 \times X_2 = [-24, -8, 4, 5].\]  
(Ex.3-2)

According to Lemma 3-3, \( \text{gcd}(-2) = 2 \) and \(-24, -8\) and 4 are all a multiple of 2, so the non-continuous interval equation (Ex.3-2) is transformed into the following interval equation:

\[-X_2 = [-12, -4, 2, 5].\]  
(Ex.3-3)

Since the coefficient for \(X_2\) is -1, according to Lemma 3-2, it satisfies \(4 \times (-1) \times 4 = 4 \leq 10\) \((-4 - (-12) + 2 = 10\) and 4 is a multiple of 2. Therefore, the non-continuous interval equation (Ex.3-3) is transformed into the following non-continuous interval equation:

\[0 = [-8, 16, 2, 13].\]  
(Ex.3-4)

Because 0 is one element in \([-8, 16, 2, 13]\), the non-continuous I test indicates that there are integer-valued solutions.

The comparison between the I test and the non-continuous I test for solving the same example in Fig. 1 is shown in Table 1. As shown in Table 1, the do-loop normalization of one time is performed for the I test. However, do-loop normalization is not needed for the non-continuous I test. Both the I test and the non-continuous I test perform a computation two times for the Banerjee bound. The I test finishes the GCD test two times and the non-continuous I test. Both tests perform the GCD test one time. It is indicated from the compared results of Table 1 that the non-continuous I test extends the I test to be able to directly deal with a Fortran do-loop with constant bounds and non-singular increments, and that the execution time of data dependence analysis for parallel/vector compilers can be efficiently improved.

| Table 1. The comparison between the I test and the non-continuous I test for solving the same example in Fig. 1 |
|-------------------------------------------------|---------|----------|---------|
| Do-loop normalization | The Banerjee bound | The GCD test |
| The I test | 1 | 2 | 2 |
| The non-continuous I test | 0 | 2 | 1 |

### 3.4 The Algorithm for the Non-Continuous I Test

The following algorithm is used to describe how to implement the non-continuous I test.

**Algorithm I**: The implementation of the non-continuous I test.

**Input**: \((a_0, a_1, \ldots, a_n, INC, M_1, N_1, INC_1; \ldots; M_n, N_n, INC_n)\)

**Output**:  
- **no**: the non-continuous interval equation \(a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = [L, U, INC, \frac{U - L}{INC} + 1]\) is not \([(M_1, N_1, INC_1, \frac{N_1-M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2-M_2}{INC_2} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n-M_n}{INC_n} + 1])-integer solvable.
- **yes**: the non-continuous interval equation \(a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = [L, U, INC, \frac{U - L}{INC} + 1]\) is \([(M_1, N_1, INC_1, \frac{N_1-M_1}{INC_1} + 1]; [M_2, N_2, INC_2, \frac{N_2-M_2}{INC_2} + 1]; \ldots; [M_n, N_n, INC_n, \frac{N_n-M_n}{INC_n} + 1])-integer solvable.
• or **maybe**: the non-continuous interval equation \(a_1 \times X_1 + a_2 \times X_2 + \ldots + a_n \times X_n = [L, U, INC, U - L + 1]\) may be \(([M_1, N_1, INC_1, N_1 - M_1 INC_1 + 1]; [M_2, N_2, INC_2, N_2 - M_2 INC_2 + 1]; \ldots; [M_n, N_n, INC_n, N_n - M_n INC_n + 1])\)-integer solvable.

**Method:**

1. \(L = a_0, U = a_0\) and \(\Phi = \{a_1, \ldots, a_n\}\)

2. While (True)

2a. While (\(\exists a_k \in \Phi\) such that \(|a_k \times INC| \leq U - L + INC\) and \(|a_k \times INC|\) is a multiple of \(INC\))

3. If \((a_k > 0)\) then

3a. \(L = L - a_k \times N_k\) and \(U = U - a_k \times M_k\).

Else

3b. \(L = L - a_k \times M_k\) and \(U = U - a_k \times N_k\).

End If

4. \(\Phi = \Phi - \{a_k\}\).

5. If (\(\Phi = \emptyset\)) then

5a. If (0 is one element in \([L, U, INC, U - L + 1]\)) then

5b. return (yes).

Else

5c. return (no).

End If

End While

6. Compute the greatest common divisor for each element in \(\Phi\) and let \(d\) be equal to the computed result.

7. If \((d \times \lceil L / d \rceil)\) is not an element in \([L, U, INC, U - L + 1]\) then return (no).

8. If \((d \neq 1)\) then

8a. If \((L, U\) and \(INC\) are, respectively, a multiple of \(d)\) then

8b. for all \(a \in \Phi\) \(a = a \div d\).

8c. \(L = L \div d, U = U \div d\) and \(INC = INC \div d\).

8d. Else return (maybe).

End If


End If

End While

End Algorithm

**Theorem 3-2:** The non-continuous I test that is an extension of the I test is an efficient and precise method to figure out whether there are integer-valued solutions for one-dimensional arrays with constant bounds and non-singular increments or not.

**Proof:** Refer to Algorithm 1.

If the non-continuous I test returns a result of yes or no, then the result is definitive. For example, a returned value of yes means that the equation is definitively \(([M_1, N_1, INC_1, N_1 - M_1 INC_1 + 1]; \ldots; [M_n, N_n, INC_n, N_n - M_n INC_n + 1])\)-integer solvable, and a returned value of no means that
the equation is definitively not \((|M_1, N_1, INC_1, N_1 - M_1|_{INC_1} + 1] \ldots [M_n, N_n, INC_n, N_n - M_n|_{INC_n} + 1])\)-integer solvable. On the other hand, a returned value of \textit{maybe} means that the equation has a solution that satisfies the limits on all the variables that the non-continuous I test has managed to move to the right hand side, and might still have a solution that satisfies the limits on the rest of the variables.

If the non-continuous I test returns a result of \textit{maybe} because there are no longer any coefficients with small enough values for Lemmas 3-1 and 3-2 to justify their movement to the right, then, it is very clear that the ‘step-by-step Banerjee test’ should be performed (i.e., to finish the computation of the Banerjee bounds). A negative result means that no solution exists. Performing the ‘Banerjee test residue’ also ensures that the non-continuous I test is always at least as accurate as the Banerjee test.

### 3.5 The Time Complexity of the Non-Continuous I Test

The main phases of the non-continuous I test to detect whether integer solutions exist for a non-continuous interval equation (3-2) satisfying the constraints of (3-1) are as follows: (1) finding a qualified item to be moved to the right hand side of the non-continuous interval equation (3-2); (2) calculating the new non-continuous integer interval on the right hand side of a non-continuous interval equation (3-2), due to the movement of the qualified item; and (3) applying the non-continuous interval-equation GCD test on all of the coefficients for each variable in the new non-continuous interval equation.

The time complexity of finding a qualified item to be moved is \(O(n)\), where \(n\) is the number of variables in a non-continuous interval equation. Thus, the time complexity of moving all of the items (if they are all qualified) is \(O(n^2)\), which is due to the fact that there are at most \(n\) moves. To calculate the new non-continuous integer interval on the right hand side of a non-continuous interval equation due to the movement of the qualified item is actually equivalent to applying a single \textit{Banerjee inequality} [17]. Applying a single Banerjee inequality to calculate the lower bound and the upper bound of the new non-continuous integer interval needs a constant time of \(O(1)\). Thus, the time complexity of the non-continuous I test to calculate each new non-continuous integer interval is \(O(n)\) because there are at most \(n\) moves. In the absolute case, the non-continuous I tests involve \(n\) GCD tests. In actual practice, it usually requires far fewer time, and normally no more than \(O(1)\). Hence, the time complexity of the non-continuous I test to be able to determine data dependence for one-dimensional arrays with constant bounds and \textit{non-singular increments} is \(O(n^2)\), which is similar to the results obtained by using the I test [14].

### 4. Experimental Results

We tested the I test and the non-continuous I test and performed experiments on the codes abstracted from the following four numerical packages: Vector Loop, Livermore, MDG (Perfect Benchmarks), and MG3D (Perfect Benchmarks) [8,32,33]. One-hundred and forty pairs of one-dimensional array references were observed to have subscripts with non-singular increments. If lower bounds, upper bounds, and non-singular increments were unknown variables (at the time of compilation), then assume that they were 1, 39, and 2, respectively [34]. After manual do-loop normalization for 140 pairs of one-dimensional array references was performed, the I test
was used to figure out if there were integer-valued solutions for the normalized do-loops. Simultaneously, the non-continuous I test was also applied to compute whether there were integer-valued solutions for the original 140 pairs of one-dimensional array references. The experimental results for the I test and the non-continuous I test for solving the same problems are shown in Table 2.

Table 2. The experimental results for the I test and the non-continuous I test for solving the same problems

<table>
<thead>
<tr>
<th></th>
<th>Manual do-loop normalization</th>
<th>The Banerjee bound</th>
<th>The GCD test</th>
<th>The number of integer-valued solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>The I test</td>
<td>140</td>
<td>310</td>
<td>160</td>
<td>140</td>
</tr>
<tr>
<td>The non-continuous I test</td>
<td>0</td>
<td>310</td>
<td>0</td>
<td>140</td>
</tr>
</tbody>
</table>

As can be seen in Table 2, manual do-loop normalization was performed one time for each case tested for the I test. However, manual do-loop normalization for every case checked was not needed for the non-continuous I test. Because do-loop normalization made the coefficient for each variable in any tested data dependence equation become larger, for 87.5% of the tested cases the I test additionally needed to perform one GCD test and for the other 12.5% of checked cases the I test additionally needed to perform two GCD tests. For any original cases examined without do-loop normalization, the coefficient for every variable was 1 or −1, so the non-continuous I test did not need to additionally perform one GCD test. The total number for computation of the Banerjee bound for all of the cases tested was 310 times for both the I test and the non-continuous I test. As indicated in Table 2, the I test and the non-continuous I test obtained the same precise results for the cases that were tested. As shown in Table 2, the non-continuous I test extended the I test to directly deal with a Fortran do-loop with non-singular increments. Simultaneously, the execution time of data dependence analysis for parallel/vector compilers could be efficiently improved.

5. CONCLUSIONS

The research in [10] stated the following: (1) the cost of scanning array subscripts and loop bounds to build a dependence problem was typically 2 to 4 times the copying cost (the cost of building a system of dependence equations) for the problem; and (2) the dependence analysis cost for more than half of the simple arrays tested was typically 2 to 4 times the copying cost. However, the dependence analysis cost for other simple arrays and all of the regular, convex, and complex arrays tested was more than 4 times that of the copying cost. Based on these results we can conclude that for simple arrays, the analysis cost of data dependence for a parallelizing/vectorizing compiler generally occupies about 29% to 57% of the total compilation time. But, for complex arrays, the analysis cost of dependence testing takes more than 57% of the total compilation time. Therefore, enhancing the performance of dependence testing may result in a significant improvement on the compilation performance of a parallelizing/vectorizing compiler.

The Power test is a combination of the Fourier-Motzkin variable elimination method with an extension of Euclid’s GCD algorithm [11]. The Omega test combines new methods for eliminating equality constraints with an extension of the Fourier-Motzkin variable elimination method [10]. The two tests currently have the highest precision and the widest applicable range
in the field of data dependence analysis for testing arrays with linear subscripts. Wolfe [11] found that using the Fourier-Motzkin variable elimination method for dependence testing takes from 22 to 28 times longer than the Banerjee test. Wolfe also indicated that the Lambda test is a very precise and efficient method for testing two-dimensional coupled arrays with constant bounds. The authors [3,16,17,20,21,35] also indicated that the Omega test is a precise method. The Range test [6] and the access range test [7,18] currently have the highest precision and the widest applicable range for checking nonlinear arrays in the field of data dependence testing.

The non-continuous I test can be viewed as involving the term-by-term computation of the Banerjee bounds. The Banerjee bound computation component of the non-continuous I test costs, at most, the same as a single Banerjee test. Depending on the application domains and environments, the non-continuous I test can be applied independently or together with other well-known methods to analyze the data dependence for linear-subscript array references.

**REFERENCES**


Qing Zhang

She received the M.S. degree in Computer Science from Dalian Maritime University in 2009. She is currently a lectureship in Dalian Shipping College. Her research interests include parallel processing, distributed systems, and multimedia processing.