A MODIFIED BI-IMAGINARY NUMBER SYSTEM*

Arunas G. Slekys
B-N Software Research Inc.
522 University Avenue
Toronto, Ontario CANADA

Algirdas Avižienis
UCLA Computer Science Department
University of California
Los Angeles, CA 90024 USA

ABSTRACT

In this paper the properties of p-imaginary number systems are reviewed and a modified bi-imaginary number system is introduced as a special case with p = 2. Major properties, including conversion of integer and floating point operands represented in a radix +p system, range, sign and zero tests, and shifting are discussed. The ability to represent the operands as vectors of radix -2 digits suggests advantages in implementing machine-usable arithmetic algorithms.

1. Introduction

Since Knuth [1] first introduced an imaginary number system for representing complex numbers, several articles were published suggesting various algorithms for performing basic arithmetic operations. Nadler [2] presented division and square root methods for complex operand representation in the quater-imaginary number system, following the approach taken by Knuth and extending it to algorithms more realizable within the constraints of a computing machine. Prior to their work, several investigations had been carried out in the area of negative radix number systems and their application to performing arithmetic operations on real numbers. Pawlak and Wakulicz [3] in 1957 were the first to suggest the use and summarize the advantages of employing a negative radix system in the arithmetic unit of a computer. Wadel [4], [5] followed up on their results with a more comprehensive review of negative radix number systems and derived conversion algorithms from conventional positive radix representations. Dietmeyer [6], and then Pongracz-Bartha [7] presented the properties that linked negative and imaginary radix number systems in extensive detail. The latter also derived algorithms for arithmetic operations using imaginary number representations. However, the algorithms were neither faster nor more cost-effective when implemented in hardware than those of equivalent binary arithmetic units. More recently, Sankar [8], Zohar [9], Kanani [10] and Agrawal [11], [12], have presented various negative radix arithmetic algorithms together with hardware realizations that begin to compete in practicality with equivalent function binary units.

*This research was supported in part by the U.S. National Science Foundation, Grant No. MCS72-03633 A05.

This paper reviews the properties of imaginary number systems, showing the relationship with negative radix systems. A number system based on a bi-imaginary radix with certain modifications is described and the major properties illustrated. The ability to represent complex operands as vectors of radix -2 digits in both integer and floating-point formats suggests advantages in implementing simple, machine-useable arithmetic algorithms and therefore in designing hardware arithmetic units.

2. P-Imaginary Number Systems

2.1 Representation

The complex number Z in the p-imaginary number system with radix $i\sqrt{p}$, with an integer $p \geq 2$ and $i = \sqrt{-1}$ is represented by the digit vector $\underline{Z}$, following the notation introduced in [13], where $\underline{Z}$ consists of the set of K+J digits:

$$z_{K-1}\ z_{K-2} \cdots z_1\ z_0 {}^\wedge z_{-1} \cdots z_{-J}$$

The digit values are $0 \leq z_j \leq p-1$, and the symbol $\wedge$ designates the reference point, which serves an analogous function as the radix point. This representation falls under the classification of a non-redundant positional number form, where the weight associated with a digit is determined uniquely by its position in the vector, and there exists only one unique digit vector for a particular number in the complex plane. A vector $\underline{Z}$ represents the unique numerical value denoted as

$$Z = \sum_{k=-J}^{K-1} z_k (i\sqrt{p})^k$$

which can be written as:

$$Z = \sum_{k=0}^{K-1} z_k (i\sqrt{p})^k + \sum_{k=-1}^{-J} z_k (i\sqrt{p})^k = Z_L + Z_R$$

The first term Z_L above describes complex values represented by the digits to the left of the reference point. The second term Z_R describes the complex values represented by the digits to the right of the reference point. All even positions ± k (to the right or left of the reference point) represent real quantities; all odd positions represent imaginary quantities. A convention is adopted whereby all digits to the left of the reference point (representing integers) are designated as having left-side values; all digits to the right of the reference point are designated as having right-side values. This simplifies the problem of description that arises because in the p-imaginary system both integer and fractional parts may be represented either by Z_L or by Z_R alone.

For a given integer value of p ($p \geq 2$) there exists a unique weight vector $\underline{W}$ associated with the digit vector $\underline{Z}$, where the elements are the weights:

$$w_k = (i\sqrt{p})^k; \quad \text{with } K-1 \leq k \leq -J.$$

The weights w_k belong to four categories, which are identifiable by the values of the modulo 4 residues of the indices k, designated by $4|k$. For any integer j, we have the following weights:

(a) Real-valued, positive when $k = 4j$, or $4|k = 0$:

$$w_k = (i\sqrt{p})^{4j} = p^{2j}$$

(b) Real-valued, negative when $k = 4j+2$, or $4|k = 2$:

$$w_k = (i\sqrt{p})^{4j+2} = (-p)p^{2j}$$

(c) Imaginary-valued, positive when $k = 4j + 1$, or $4|k = 1$:

$$w_k = (i\sqrt{p})^{4j+1} = (i\sqrt{p})p^{2j}$$

(d) Imaginary-valued, negative when $k = 4j + 3$, or $4|k = 3$:

$$w_k = (i\sqrt{p})^{4j+3} = (-ip\sqrt{p})p^{2j}$$

We note a similarity with the weights of the radix -p representation ($p \geq 2$): the weights of the even positions ($k=4j$ or $k=4j+2$) are the same as the weights of the consecutive positions $(2j, 2j + 1)$ in the radix -p system. Multiplying these weights by $i\sqrt{p}$ will give the weights of the odd positions ($k= 4j + 1$, or $k = 4j + 3$) of the p-imaginary representation.

2.2 Relationship with Negative Radix Systems

If Z is rewritten to separate the real and imaginary parts, the relationship with the -p representation becomes obvious.

$$Z = \sum_{k=-J_1}^{K_1-1} z_{2k}(-p)^k + i\sqrt{p} \sum_{k=-J_2}^{K_2-1} z_{2k+1}(-p)^k$$

where $K_1 + K_2 + K$, $J_1 + J_2 = J$ and $\underline{Z}$ is a $K + J$ digits long vector. It is apparent that both the real and imaginary parts of the $\underline{Z}$ vector follow a -p radix representation with the digits placed in the appropriate even and odd positions, respectively. That is, consider representing some complex number Z, with real and imaginary parts denoted Re[Z] and Im[Z] respectively, so that

$$Z = \mathrm{Re}[Z] + i\,\mathrm{Im}[Z]$$

as a p-imaginary K+J digits long vector $\underline{Z}$. First, it is necessary to find the -p radix digit vectors which represent Re[Z] and Im[Z]/$\sqrt{p}$. Let:

$$A = \mathrm{Re}[Z], \quad \text{and} \quad B = \mathrm{Im}[Z]/\sqrt{p}$$

$$\text{where} \quad A = \sum_{k=-J_1}^{K_1-1} a_k(-p)^k, \text{ and } B = \sum_{k=-J_2}^{K_2-1} b_k(-p)^k$$

$$\text{with} \quad 0 \leq a_k,\ b_k \leq p - 1.$$

Observing that, $(-p)^k = (i\sqrt{p})^{2k}$ and multiplying the value B by $i\sqrt{p}$, we get:

$$(i\sqrt{p})B = \sum_{k=-J_2}^{K_2-1} b_k(i\sqrt{p})^{2k+1}$$

Also denoting:

$$z_{2k} = a_k; \quad z_{2k+1} = b_k$$

and recalling:

$$\sqrt{p}\,B = \mathrm{Im}[Z]; \text{ and } A = \mathrm{Re}[Z]$$

then the required digit vector $\underline{Z}$ is derived in the p-imaginary radix, such that

$$Z = \sum_{k=-J}^{K-1} z_k(i\sqrt{p})^k$$

Clearly, the conversion process is reversible. Given the digit vector $\underline{Z}$, the a_k and b_k are readily available, since

$$a_k = z_{2k}$$

$$b_k = z_{2k+1}$$

and hence the digit vectors $\underline{A}$ and $\underline{B}$ can be found.

We note that in deriving the $\underline{B}$ digit vector, the quantity $\mathrm{Im}[Z]/\sqrt{p}$ has to be determined. If $\sqrt{p}$ is a rational number, this need not be a problem. For example, if $p = 4$, and hence the radix is $2i$, the familiar quater-imaginary system results, illustrated in the following example.

Example 2.1

Consider representing the quantity $Z = 1 + i$. Then, in the quater-imaginary system,

$$\mathrm{Re}[Z] = 1, \text{ and } \mathrm{Im}[Z] = 1;$$

which gives:

$$\underline{A} = 01\wedge 00; \quad \underline{B} = 01\wedge 20.$$

Hence

$$\underline{Z} = 11\wedge 20.$$

As Knuth [1] points out, using a bi-imaginary radix, with $p = 2$ and radix $\sqrt{2}i$, all digits come from the set $\{0, 1\}$, making this system attractive for implementation in digital hardware. The obvious problem is that the number $\mathrm{Im}[Z]/\sqrt{p}$ will always be irrational if $\mathrm{Im}[Z]$ is rational; for example i cannot be represented in a non-terminating way, producing truncation and rounding errors. This is the major reason why most approaches to date have concentrated on the quater-imaginary number system in deriving algorithms suitable for machine operation, although four digit values, namely $\{0, 1, 2, 3\}$ are possible, instead of the familiar binary set. A means of modifying the bi-imaginary system to make it more suitable for implementation is presented in Section 3.

2.3 Conversion from Positive Radix Systems

Assuming that a number Z is represented as a digit vector in the $+p$ radix system, then a conversion to the $-p$ radix representation is necessary prior to deriving the required $\underline{Z}$ digit vector in the p-imaginary system as described in Section 2.2. Pongracz-Bartha [7] unified the conversion rules from the $+p$ to the $-p$ number system outlined first by Wadel [5] for the case $p = 2$, and by Knuth [1] for $p = 4$. The rules in general are lengthy and are not repeated here. Dietmeyer [6] detailed the rules necessary for the direct conversion of a sign-and-magnitude $+p$ radix-fixed point number into the p-imaginary number system. The complexity of the direct method with the many operations required for ensuring that the number to be converted is within various bounds for different digit vector lengths and values of p, makes it difficult to discuss it in general.

The approach here is to introduce the reader to the references describing the general $+p$ to p-imaginary radix conversion rules and to describe the special case of $p = 2$. An example for the case of $p = 2$ is appropriate to illustrate the two-step conversion.

Example 2.2

Convert the complex number $Z = -5 + 8i$, originally represented in two's complement binary format as the two digit vectors:

$$\mathrm{Re}[\underline{Z}]_2 = 11011\wedge, \text{ and } \mathrm{Im}[\underline{Z}]_2 = 01000\wedge$$

into the bi-imaginary radix represetnation.

Step I: Conversion from the +2 radix to the -2 radix.

$$A = -5; \quad B = \frac{8}{\sqrt{2}} \cong 5.656$$

$$\underline{A} = \mathrm{Re}[\underline{Z}]_{-2} = 01111\wedge 00000$$

$$\underline{B} = \mathrm{Im}[\underline{Z}]_{-2} = 11010\wedge 11111 \ \ldots$$

Step II: Conversion from the -2 radix to the $i\sqrt{2}$ radix.

$$[\underline{Z}]_{i\sqrt{2}} = 1\underline{0}1\underline{1}0\underline{1}1\underline{1}0\underline{1}\wedge 1\underline{0}1\underline{0}1\underline{0}1\underline{0}1\underline{0}$$

where the underlined digits come from $\underline{A}$ and the non-underlined digits come from $\underline{B}$, as discussed in Section 2.2.

2.4 Range, Sign Test, Zero Test, and Shifting

An interesting property of the p-imaginary number system is that the ranges of both left-side and right-side values extend over positive and negative, real and imaginary values. Define the left-side and right-side parts of Z as Z_L and Z_R respectively, so that following the notation of Section 2.1,

$$Z_L = \sum_{k=0}^{K-1} z_k (i\sqrt{p})^k$$

$$Z_R = \sum_{k=1}^{J} z_{-k} (i\sqrt{p})^{-k}$$

Assuming $K_1 + J_1$ real-weighted digits, and $K_2 + J_2$ imaginary-weighted digits, we have:

$$\mathrm{Re}[Z_R] = \sum_{k=1}^{J_1} z_{-2k} (-p)^{-k}$$

$$\mathrm{Im}[Z_R] = \sqrt{p} \sum_{k=1}^{J_2} z_{-2k+1} (-p)^{-k}$$

$$\mathrm{Re}[Z_L] = \sum_{k=0}^{K_1-1} z_{2k} (-p)^k$$

$$\mathrm{Im}[Z_L] = \sqrt{p} \sum_{k=0}^{K_2-1} z_{2k+1}(-p)^k$$

The range of numerical values for each part is found by substituting the values of z_i that give the maximum and minimum values. Detailed derivations are given in reference [14].

For right-side values (assuming $J_1 = J_2 = N$) the bounds are, for an even value of N:

$$(-p) \frac{1-p^{-N}}{p+1} \le \mathrm{Re}[Z_R] \le \frac{1-p^{-N}}{p+1}$$

$$(-p\sqrt{p}) \frac{1-p^{-N}}{p+1} \le \mathrm{Im}[Z_R] \le \sqrt{p} \frac{1-p^{-N}}{p+1}$$

For N odd, the bounds are:

$$(-p)\frac{1-p^{-(N+1)}}{p+1} \le \mathrm{Re}[Z_R] \le \frac{1-p^{-(N-1)}}{p+1}$$

$$(-p\sqrt{p})\frac{1-p^{-(N+1)}}{p+1} \le \mathrm{Im}[Z_R] \le \sqrt{p} \frac{1-p^{-(N-1)}}{p+1}$$

In the limit, the upper and lower bounds for infinite-length right-hand sides are obtained by letting N approach an infinite value. The bounds then are:

$$-\frac{p}{p+1} < \mathrm{Re}[Z_R] < \frac{1}{p+1}$$

$$-\frac{p\sqrt{p}}{p+1} < \mathrm{Im}[Z_R] < \frac{\sqrt{p}}{p+1}$$

For left-side values (assuming $K_1 - 1 = K_2 - 1 = N$), that is, N+1 digits each, the bounds are, for an even value of N:

$$(-p) \frac{p^N - 1}{p+1} \le \mathrm{Re}[Z_L] \le \frac{p^{N+2} - 1}{p+1}$$

$$(-p)\sqrt{p} \frac{p^N - 1}{p+1} \le \mathrm{Im}[Z_L] \le \sqrt{p} \frac{p^{N+2} - 1}{p+1}$$

For odd values of N, the bounds are:

$$(-p) \frac{p^{N+1} - 1}{p+1} \le \mathrm{Re}[Z_L] \le \frac{p^{N+1} - 1}{p+1}$$

$$(-p)\sqrt{p} \frac{p^{N+1} - 1}{p+1} \le \mathrm{Im}[Z_L] \le \sqrt{p} \frac{p^{N+1} - 1}{p+1}$$

The sign test is performed separately for the real and the imaginary parts of Z. To test the real part, the leftmost non-zero even-indexed digit z_k is found. If its index k is a multiple of four (i.e., $4|k = 0$), the sign is plus; if it is not (i.e., $4|k = 2$), the sign is minus. To test the imaginary part, the index k' of the leftmost non-zero odd-indexed digit $z_{k'}$ is found. If $4|k' = 1$, the sign is plus; if $4|k' = 3$, the sign is minus. To perform the zero test, we note that the real (imaginary) part has the value zero if and only if all even-indexed (odd-indexed) digits have zero values.

To perform a shift of magnitude f on the p-imaginary digit vector $\underline{Z}$ every digit z_e belonging to position e is reassigned to the position $e \pm f$, the sign being negative for a right shift and positive for a left shift. The value of a shifted number Z* may be written,

$$Z^* = (Z - D_1)\,(i\sqrt{p})^{\pm f} + D_2$$

where

$$Z = \sum_{k=-J}^{K-1} z_k (i\sqrt{p})^k$$

D_1 is the value represented by f discarded digits at one end, and D_2 is the value represented by f inserted (new) digits at the other end.

We note that the conditions $D_1 = 0$ and $D_2 = 0$ must be satisfied when the shift serves as a scaling algorithm. Furthermore, and odd value of f will switch digits between real-weighted and imaginary-weighted positions and is not useful. A multiplication by p^2 (or p^{-2}) occurs when the left (right) shift is by four positions ($f = \pm 4$). A shift by two positions ($f = \pm 2$) will multiply the operand by -p or -1/p for left and right shifts, respectively. Both $f = \pm 2$ and $f = \pm 4$ shifts are discussed further with respect to floating point representation.

2.5 Floating-Point Representation

A convenient floating-point form of the p-imaginary number Z is:

$$Z = (i\sqrt{p})^E \cdot Z_R = (i\sqrt{p})^E \sum_{k=1}^{J} z_{-k}\,(i\sqrt{p})^{-k}$$

where the exponent E is a real-valued integer and the coefficient Z_R is the right-side value of a digit vector $\underline{Z}_R$:

$$\underline{Z}_R = \wedge\ z_{-1}\ z_{-2} \cdots z_{-J}$$

Although there is no restriction on representing E in any radix, including binary, there may be an advantage in retaining compatibility of arithmetic algorithms for both the coefficient and the

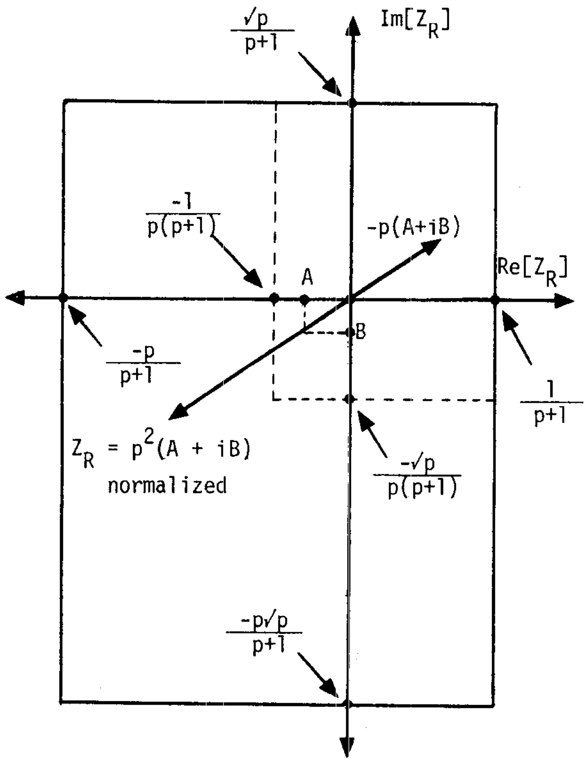

Figure 1. Range of Coefficients Z_R

exponent parts of floating-point operands by using the -p radix to represent E.

The digit vector $\underline{Z}_R$ is defined to be "normalized" in the context of the p-imaginary number system, when at least one of its two leftmost digits (z_{-1}, z_{-2}) has a non-zero value. Under this convention, a normalized $\underline{Z}_R$ may have one of the following three forms:

(i) $\underline{Z}_R = \wedge\, q, q, z_{-3}, \ldots, z_{-J}$

(ii) $\underline{Z}_R = \wedge\, 0, q, z_{-3}, \ldots, z_{-J}$

(iii) $\underline{Z}_R = \wedge q,\ 0, z_{-3}, \ldots, z_{-J}$

where $1 \le q \le (p-1)$, and $0 \le z_k \le (p-1)$ for $-3 \ge k \ge -J$.

Recall that only even shifts (f = 2,4,...) are useful for p-imaginary digit vectors. For this reason a two-position shift is defined to be a unit shift; however, the value of the exponent E is decremented (or incremented) by two for every left (or right) unit shift of the coefficient.

The ranges for infinite-length right sides, as given in Section 2.4, are plotted on the complex plane to show the normalized region. As shown in Figure 1, the smaller area around the origin represents the region of values for unnormalized right-side vectors. An arbitrary unnormalized right-side value, A + i B, as depicted, is unit-shifted left (one pair of positions at a time), performing multiplication times (-p) for each left unit shift, until it falls inside the normalized region. Note that each unit shift causes the vector to change direction by 180 degrees in the complex plane, although the ratio of real and imaginary values remains unchanged. The exponent E is decremented by two for each unit shift. This compensates for the change of sign which takes place when Z_R is multiplied by (-p) as follows:

$$(i\sqrt{p})^{E-2} \cdot Z_R(-p) = (i\sqrt{p})^{E} \cdot Z_R \cdot (i\sqrt{p})^{-2} \cdot (-p)$$

$$= (i\sqrt{p})^{E} \cdot Z_R$$

Conversion of a floating-point complex number initially represented in the +p radix system into the p-imaginary radix system generally follows the two-step procedure discussed in Section 2.3 for fixed-point quantities. However, the ranges of values of both the exponent and coefficient of a number represented in the +p radix system are different from the ranges representable in the -p radix system. A range test must be applied before the conversion process, to assure that the converted number can in fact be represented in the p-imaginary floating-point format already described. A detailed description of floating-point conversion is given in [14].

3. The Modified Bi-Imaginary System.

3.1 Definition and Conversion

The bi-imaginary number system is the p-imaginary number system with p = 2. Hence with radix $i\sqrt{2}$ the complex number Z can be represented as a (K+J) digit vector $\underline{Z}$, where

$$Z = \sum_{k=-J}^{K-1} z_k (i\sqrt{2})^k, \quad \text{and} \quad z_k \in (0,1).$$

As discussed in Section 2.2 and originally suggested by Knuth [1], there may exist advantages to this number representation when designing logic circuitry because of the binary digit values. The problem that Gaussian integers X+iY (where X and Y are integers) are represented in a non-terminating way can be overcome by introducing the following modification of the bi-imaginary system.

In Sections 2.2 and 2.3 the complex number

$$Z = Re[Z] + iIm[Z].$$

was converted to the p-imaginary representation by first finding the -p radix representation of

$$A = Re[Z], \text{ and } B = Im[Z]/\sqrt{p}$$

and then merging the digits a_k and b_k into alternating positions of the p-imaginary digit vector $\underline{Z}$.

To avoid the non-terminating representation of B, we will represent the complex number Z by the complex number Z^*,

$$Z^* = Re[Z^*] + iIm[Z^*] = Re[Z] + i(\sqrt{2}\ Im[Z])$$

Now, to convert to the bi-imaginary representation, we find the radix -2 representations of:

$$A^* = Re[Z], \text{ and } B^* = (\sqrt{2}\ Im[Z])/\sqrt{2} = Im[Z]$$

Every Gaussian integer Z now has a terminating (exact) bi-imaginary representation Z^* which differs from Z by the scale factor $\sqrt{2}$ that has been applied to Im[Z]. The presence of this scale factor causes changes in ranges of representable numbers and requires corresponding modifications of arithmetic algorithms.

It is evident that in addition to p = 2, the proposed modification may be applied to every other value of p, and it is especially useful for those values of p that are not perfect squares, such as p = 8, p = 10, etc. In each case, Im[Z*] represents $\sqrt{p}$ Im[Z], while Re[Z*] represents Re[Z] directly. The modification increases the practicality of using the deci-imaginary representation with the radix $i\sqrt{10}$.

A division by $\sqrt{p}$ must be performed on Im[Z*] in order to get Im[Z] during or after reconversion of the final results into radix +p for output.

To illustrate the conversion, we use the operand Z = -5 + 8i, which was used in Example 2.2.

Example 3.1

Convert Z = -5 + 8i to modified bi-imaginary radix representation

$$\underline{A}^* = Re[Z]_{-2} = 01111\wedge$$

$$\underline{B}^* = Im[Z]_{-2} = 11000\wedge$$

$$\underline{Z}^* = 1\underline{0}1\underline{1}1\underline{0}1\underline{0}1\underline{0}1\wedge$$

where the underlined digits come from $\underline{A}^*$ and the other digits come from $\underline{B}^*$. Contrary to the result of Example 2.2, $\underline{Z}^*$ represents a left-side value only.

3.2 Properties of Modified P-Imaginary Systems

The ranges of representable right-side and left-side values for the p-imaginary systems were presented in Section 2.4. The modified p-imaginary system has exactly the same values for the real parts of Z_L and Z_R. The ranges for the imaginary parts of Z_L and Z_R are reduced by the factor $\sqrt{p}$ because of the modification. We note that in Sec. 2.4 all bounds of the imaginary parts were equal to $\sqrt{p}$ times the bounds of the real parts. The reduction by $\sqrt{p}$ makes the bounds of the imaginary parts equal to the bounds of the real parts. The conclusion is that the bounds which were derived for the real parts $Re[Z_R]$ and $Re[Z_L]$ in Section 2.4 also apply to the imaginary parts $Im[Z_R]$ and $Im[Z_L]$ when a modified p-imaginary system is used. The resulting uniformity of bounds is a useful simplification.

The sign tests and the zero test of the modified p-imaginary system are not affected by the modification and remain the same as those for the p-imaginary system, discussed in Section 2.4.

The left and right shifts for scaling (multiplication or division by -p) also remain unchanged and follow the rules of Section 2.4.

Floating-point representation is affected only because the range of imaginary parts $Im[Z_R]$ is reduced by $\sqrt{p}$. The discussion of Section 2.5 applies throughout, with only one change in Figure 1, where the three boundary values on the $Im[Z_R]$ axis must be divided by $\sqrt{p}$ and become equal to the corresponding boundary values on the $Re[Z_R]$ axis.

3.3 Arithmetic Algorithms for the Modified Bi-Imaginary System

The modified bi-imaginary system appears to be of the most immediate interest because of its close relationship to binary systems. For this reason we go from the general radix $i\sqrt{p}$ to $i\sqrt{2}$ in the following brief discussion of arithmetic algorithms. Detailed discussion of the algorithms is given in [14].

3.3.1. Additive Inverse. As observed by Knuth [1], Songster [15], and others, the additive inverse (negative) of a number represented in the -2 radix cannot be formed merely by a change of sign digit as in the sign-magnitude binary representation.

A convenient method to find the additive inverse of a number N represented as a -2 radix digit vector $\underline{N}$, is to shift $\underline{N}$ one digit to the left, getting (-2)N and then add it to $\underline{N}$ to get the representation of -N. The analogous procedure for a complex number Z, represented as the digit vector $\underline{Z}^*$ in the modified bi-imaginary radix, so that

$$Z^* = Re[Z] + i\sqrt{2}\ Im[Z]$$

is to shift the entire vector $\underline{Z}^*$ two places to the left, thus getting the value $-2\underline{Z}^*$ and then to add the shifted vector to $\underline{Z}^*$. This gives the new digit vector $\underline{X}$ such that

$$X = Z^* - 2Z^* = - \text{Re}[Z] - i\sqrt{2}\ \text{Im}[Z]$$

We note that this method is not suitable for other values of p.

3.3.2. Complex Conjugate. As an extension of the discussion of the additive inverse of a complex number represented in the modified bi-imaginary system, we derive the algorithm for the complex conjugate Y^* of Z^* which is defined as:

$$Y^* = \text{Re}[Z] - i\sqrt{2}\ \text{Im}[Z]$$

To obtain the digit vector $\underline{Y}^*$, we must shift only the imaginary part two places to the left and then add the shifted digits to $\underline{Z}^*$, thus obtaining:

$$\begin{aligned} Y^* &= \text{Re}[Z] + i\sqrt{2}\ \text{Im}[Z] - 2i\sqrt{2}\ \text{Im}[Z] \\ &= \text{Re}\ [Z] - i\sqrt{2}\ \text{Im}[Z]. \end{aligned}$$

This algorithm depends on the existence of selective shifting, i.e., provisions to select only the digits of the imaginary part for the shift operation. We note that the algorithm is based on the additive inverse and therefore specific for p = 2.

3.3.3. Addition and Subtraction. The bi-imaginary addition and subtraction algorithms [1], [7] are not affected by the modification of the representation. The real and the imaginary parts are added or subtracted separately as operands in the negabinary representation, giving:

$$S^* = X^* \pm Z^* = (\text{Re}[X] \pm \text{Re}[Z]) + i\sqrt{2}(\text{Im}[X] \pm \text{Im}[Z])$$

The carry-save principle can be directly extended to implement multi-operand summation for three or more operands. Details of implementation for two-operand "lookahead" fast addition and for multioperand addition are presented in [14]. A convenient implementation of the subtraction $D^* = X^* - Y^*$ is obtained by using a three-operand adder to add the additive inverse (see 3.3.1. above) of Z^* to X^*:

$$D^* = X^* + (Z^* - 2Z^*)$$

3.3.4. Multiplication. The modification of the p-imaginary representation leads to a change in the complex multiplication algorithm. A direct multiplication of two complex operands represented in the modified p-imaginary system would yield:

$$(A + i\sqrt{p}B)\cdot(C + i\sqrt{p}D) = (AC - pBD) + i\sqrt{p}(AD + BC)$$

The imaginary part has the required value, but the real part should be (AC - BD); therefore the term (p-1) BD must be added to the real part obtained as above. In the case of p=2, the required correction is the addition of BD to the product formed by the ordinary bi-imaginary multiplication algorithm. A convenient implementation of this correction results when the terms representing BD are identified as they are formed during the bi-imaginary multiplication and are immediately added to the developing product [14]. Using this approach, multi-operand summation and multiplier recoding techniques are very well adaptable to implement fast multiplication in the modified bi-imaginary representation [14].

3.3.5. Division. It was pointed out by Knuth [1] that the conventional remainder-generating division algorithm in which successive quotient digits are selected was poorly suited for his proposed quater-imaginary system. Nadler [2] later described an adaptation of a convergence scheme which he had called "the method of radixes". In the present investigation it was concluded that the most effective method of division for modified bi-imaginary operands is an adaptation of the convergence algorithm as implemented in the floating-point unit of the IBM System 360 Model 91 computer [16]. This approach is based on the existence of a high-speed multiplication algorithm and is discussed in detail in [14].

3.3.6. Floating-Point Algorithms. The logic design of a floating-point modified bi-imaginary arithmetic unit has been performed [14]. The implementation of floating-point algorithms did not encounter any unexpected difficulties. The range of the coefficients Z_R, when limited to right-side representations (as described in Section 2.5) is for the normalized part:

$$-\frac{p}{p+1} < \{\text{Re}[Z_R],\ \text{Im}[Z_R]\} < -\frac{1}{p(p+1)}$$

which gives a range between -2/3 and - 1/6 for the modified bi-imaginary operands. We note that a normalized number, as defined in Section 2.5, may have only the real part, only the imaginary part, or both parts in the normalized range. A range closer to the customary 1/2 to 1 range can be obtained by allowing two (or four) digits to the left of the reference point. With p=2, we then have the range between 4/3 and 1/3 (or - 8/3 and - 2/3). The sign of the floating-point number is controlled by the sign of $(i\sqrt{2})^E$; therefore the normalized parts of the coefficients are either all positive or all negative. During shifting the exponent E is incremented or decremented by even values only. For this reason the values of E for input operands can be restricted to even values.

4. Conclusions

The results of this investigation of p-imaginary number representations show that the proposed modification eliminates the major disadvantage of inexact representation of integer imaginary parts for the bi-imaginary, deci-imaginary and other p-imaginary representations in which p is not a perfect square. A floating-point representation has been defined and its major properties have been identified. On the basis of the results reported here and detailed in [14], it is concluded that the modified bi-imaginary and other modified p-imaginary representations fall into the set of potentially useful candidates for the implementations of arithmetic units for complex arithmetic.

5. References

[1] Knuth, D.E., "An Imaginary Number System", Comm. of the ACM, Vol. 3, No. 4, pp. 245-247, April 1960.

[2] Nadler, M., "Division and Square Root in the Quater-Imaginary Number System", Comm. of the ACM, Vol. 4, No. 4, pp. 192-193, April 1961.

[3] Pawlak, Z. and Wakulicz, A., "Use of Expansions with a Negative Basis in an Arithmometer of a Digital Computer", Bulletin de l'Academie Polonaise des Sciences, Cl. III Vol. V, No. 3, pp. 233-236, March 1957.

[4] Wadel, L.B., "Negative Base Number Systems", IRE Trans. Electronic Computers, Vol. EC-6, No. 2, p. 123, June 1957.

[5] Wadel, L.B., "Conversion from Conventional to Negative Base Number Representation, IRE Trans. Electronic Computers, Vol. EC-10, No. 4, p. 779, December 1961.

[6] Dietmeyer, D.L., "Conversion from Positive to Negative and Imaginary Radix", IEEE Trans. Electronic Computers, Vol. EC-12, No. 1, pp. 20-22, February 1963.

[7] Pongracz-Bartha, E., "Properties of the Imaginary Number System and Their Application to Digital Computing", Master of Science Thesis in Engineering, University of California, Los Angeles, 1965.

[8] Sankar, P.V., et al., "Arithmetic Algorithms in a Negative Base", IEEE Trans. Computers, Col. C-22, No. 2, pp. 120-125, February 1973.

[9] Zohar, S., "Negative Radix Conversion", IEEE Trans. Computers, Vol. C-19, No. 3, pp. 222-226, March 1970.

[10] Kanani, D.V. and O'Keefe, K.H., "A Note on Conditional-Sum Addition for Base -2 Systems", IEEE Trans. Computers, Vol. C-22, No. 6, p. 626, June 1973.

[11] Agrawal, D.P., "Negabinary Carry-Look-Ahead Adder and Fast Multiplier", Electronics Lett., Vol. 10, pp. 312-313, 1974, and ibid, Vol. 10, p. 468, 1974.

[12] Agrawal, D.P., "Negabinary Parallel Counters", Digital Processes, Vol. 1, No. 1, pp. 75-85, 1975.

[13] Avižienis, A., "Digital Computer Arithmetic: A Unified Algorithmic Specification", Proceedings of the Symposium on Computers and Automata, (Polytechnic Inst. of Brooklyn, April 13-15, 1971), MRI Symposium Proceedings, Vol. XXI; Polytechnic Press, 1971; pp. 509-526.

[14] Slekys, A.G., "Design of Complex Number Digital Arithmetic Unit Based on a Modified Bi-Imaginary Number System", Ph.D. Dissertation in Engineering, University of California, Los Angeles, 1976.

[15] Songster, G.F., "Negative-Base Number-Representation Systems", IEEE Trans. Electronic Computers, Vol. EC-12, No. 3, pp. 274-277, June 1963.

[16] Anderson, S.F., et al., "The IBM System/360 Model 91: Floating-Point Execution Unit", IBM Journal of Research and Development, Vol. 11, No. 1, pp. 34-53, January 1967.