循環表示つき浮動小数点演算系

吉田かおる　（慶應義塾大学理工学部）

1. はじめに

ある基数bが与えられると，任意の実数xは

$$x = m \cdot b^e \qquad , |m| < 1$$

のように，仮数mと指数eの対として表わすことができる．この表現形式は浮動小数点（FLP）形式と呼ばれ，一般に，数値計算プログラムにおける"real"宣言は数値がFLP形式で実現されることを意味する．

仮数の実現にあたって，既存の計算機演算系の多くはp-ary（右展開型）位取り記数法を採用し，そしてその基数表現を有限桁で打切っている．基数が正の整数である位取り記数法の下では，ある実数の基数表現は次の性質のうちいずれか一つを示す．

有理数の場合　(1)停止性（→有限小数）又は (2)周期性（→循環小数）
無理数の場合　(3)不規則性

したがって，実際には一部の有限小数しか実現されていない．

一方，計算精度の向上を目的として，分数表現及び，Henselコードとも呼ばれるp-adic（左展開型）位取り記数法による有理数演算系が最近活発に研究されている[3~15]．しかし有限桁において，これらは高価な近似操作を要する．例えば，前者の場合約分と中数丸め[5~9]，後者の場合一次不定方程式の解法[11~13]がそれである．

まず基数表現と分数表現の関係を明らかにしよう．有限小数が循環周期0なる特殊な循環小数であるとみなせば，それは循環小数の性質を調べることに他ならない．基数bに関して，0<x<1内の任意の循環小数xは，

$$x = \frac{u}{v} = \frac{\sum_{i=-(n+r)}^{-(n+1)} d_i \cdot b^{i+n} + \sum_{i=-n}^{-1} d_i \cdot b^{i+n}(b^r-1)}{b^n \cdot (b^r-1)} \qquad : \text{分数表現}$$

$$= .d_1 \cdots d_{-n}\overline{d_{-(n+1)} \cdots d_{-(n+r)}}\ (b) \qquad : \text{基数表現}$$

のように表わされ，分母v，非循環節の長さn，そして循環節の周期rの間に次の関係式が成り立つ．

$$v = v_1 \cdot v_2 \qquad , \ (v_1, v_2) = 1$$
$$v_1 \mid b^n \qquad , \ v_2 \mid (b^r-1) \qquad (*1)$$

このように，基数表現と分数表現の間に介在するのは"周期性の表示"である．本論文は，従来の浮動小数点演算系において有理数を安価に実現するために，仮数部に循環表示を施すことを提案し，これに基づく演算系について述べている．

2. FLP/R* 演算系

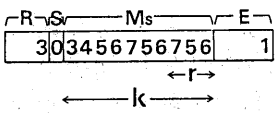
数値計算の高信頼化を目的として，従来の浮動小数点演算系と有理数演算系との融合を図る"循環表示手法"を提案する．本手法は，有限小数のみならず循環小数をも実現するために，仮数部の基数表現に関する周期性を表示するものである．本手法を採用した新しい演算系をFLP/R* (FLoating-Point and Recurring rational) 演算系と称する．

FLP/R*演算系において一つの数は，通常の浮動小数点演算系における仮数部と指数部の他に，仮数部基数表現の循環周期を表示する"循環表示部"を以て表現される (Fig.1)．即ち仮数部全体は，従来の仮数部 (S-仮数部) M_S と循環表示部 R とから構成される．

基数 b で k 桁の基数表現とすると，R 及び M_S はそれぞれ t_R bit, t_M bit を要する．

$$t_R = \lceil \log_2(k+1) \rceil \quad , \quad t_M = k \cdot u \quad ; \quad u = \lceil \log_2 b \rceil \qquad (*2)$$

ゆえに，仮数部全体は $t = t_R + t_M$ bit 要する (Fig.2)．



S : SIGN
Ms: SIGNIFICANT(S) MANTISSA
R : RECURRING(R) INDICATOR
E : EXPONENT

Fig. 1 FLP/R* Number

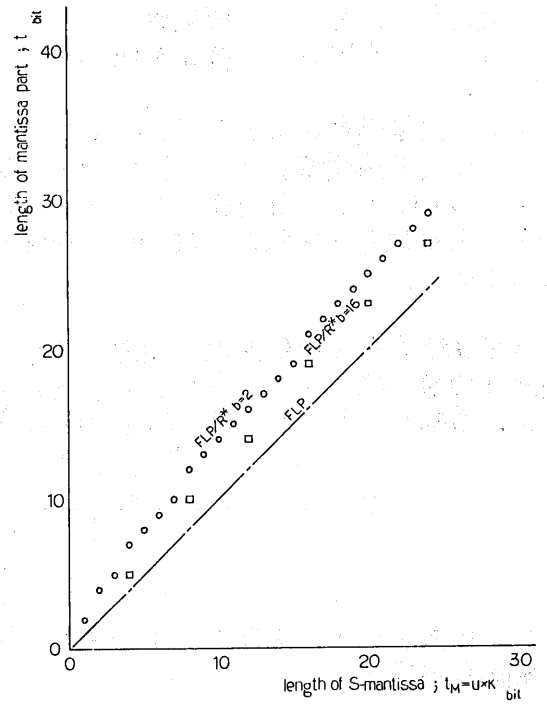


Fig.2 Length of Mantissa Part ; $t = t_R + t_M$

(*1) (a,b) は a と b の最大公約数 (GCD) を表わす．
$a \mid b$ は a が b の約数であることを示す関係式である．

(*2) $\lceil x \rceil$ は x 以上の最小の整数を表わす

3. FLP/R* 数

3.1 FLP/R* 仮数の集合

通常のFLP演算系における仮数（FLP仮数）の集合は有限小数のみから成る．k桁基数表現に関するFLP仮数の集合を R_F^k とする．

$$R_F^k = \{x \mid 0 < x = \frac{u}{v} < 1 \ , \ v = b^k\}$$

FLP/R* 演算系における仮数（FLP/R* 仮数）の集合は有限小数と循環小数から成る．k桁S-仮数部に関するFLP/R* 仮数の集合を R_U^k とする．

$$R_U^k = R_F^k \cup R_R^k$$

ここで，R_R^k はk桁S-仮数部に関する循環小数の集合であり，循環周期 j の循環小数の集合を R_{Rj}^k として次のように表わされる．

$$R_R^k = R_{R1}^k \cup R_{R2}^k \cup \cdots \cup R_{Rj}^k \cup \cdots \cup R_{Rk}^k = \bigcup_{j=1}^{k} R_{Rj}^k$$

$$R_{Rj}^k = \{x \mid 0 < x = \frac{u}{v} < 1, \ v \mid \nu_j\} \quad , \ \nu_j = b^{k-j}(b^k - 1)$$

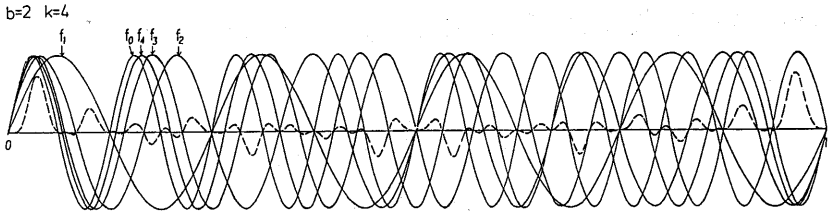


Fig.3 Distribution Function $f(x) = \prod_{j=0}^{k} f_j(x)$

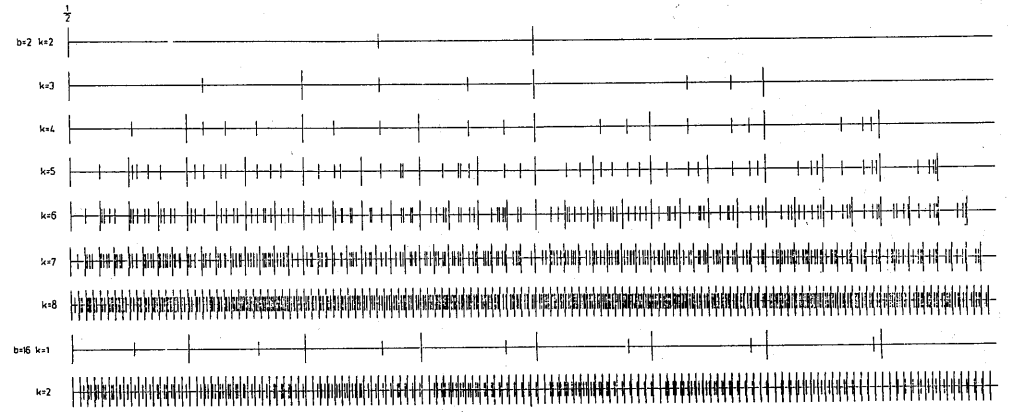


Fig.4 Distribution of FLP/R* Mantissas

ここで，次のような正弦関数 $f_j(x)$ を考える．

$$f_j(x) = \sin(\pi \nu_j x) \quad , \nu_j = \begin{cases} b^k & ; j=0 \\ b^{k-j}(b^j-1) & ; j=1,2,\cdots,k \end{cases}$$

これより，集合 R_T^k，R_R^k，R_U^k を書き直すと，

$$R_T^k = \{x \mid f_0(x) = 0 \quad ; 0<x<1\}$$
$$R_R^k = \{x \mid \prod_{j=1}^{k} f_j(x) = 0 \quad ; 0<x<1\}$$
$$R_U^k = \{x \mid \prod_{j=0}^{k} f_j(x) = 0 \quad ; 0<x<1\}$$

そこで，FLP/R* 仮数の分布状態を Fig.3 及び Fig.4 に示す．

3.2 FLP/R* 仮数の個数

FLP/R* 仮数は ν_j $(j=0,1,\cdots,k)$ のいずれか一つの約数を分母とする有理数である．そこで，FLP/R* 仮数の全個数 $n(R_U^k)$ は次のように求まる．

$$n(R_U^k) = \sum_{d \in D} \phi(d)$$

ただし，$\phi(d)$ ：オイラー関数 (*3)

$$D = \bigcup_{j=0}^{k} \{d \mid d \mid \nu_j\}$$

FLP/R* 仮数と FLP 仮数の個数の比較を Fig.5 及び Fig.6 に示す．

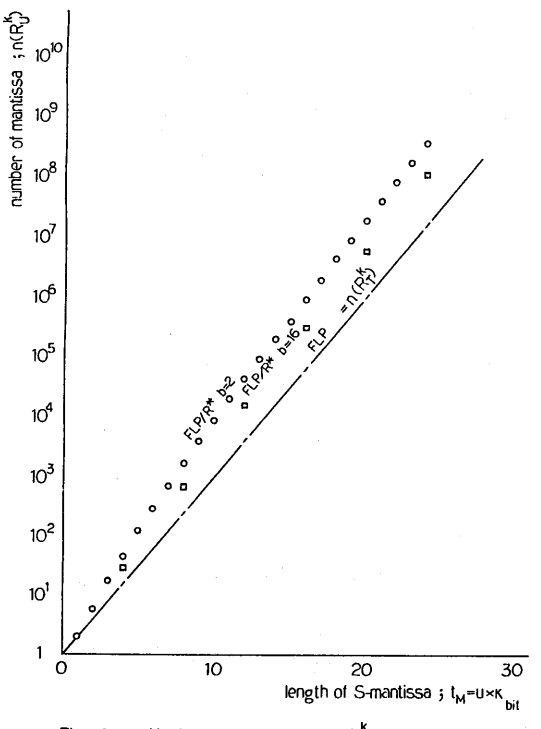


Fig. 5 Number of Mantissas ; $n(R_U^k)$

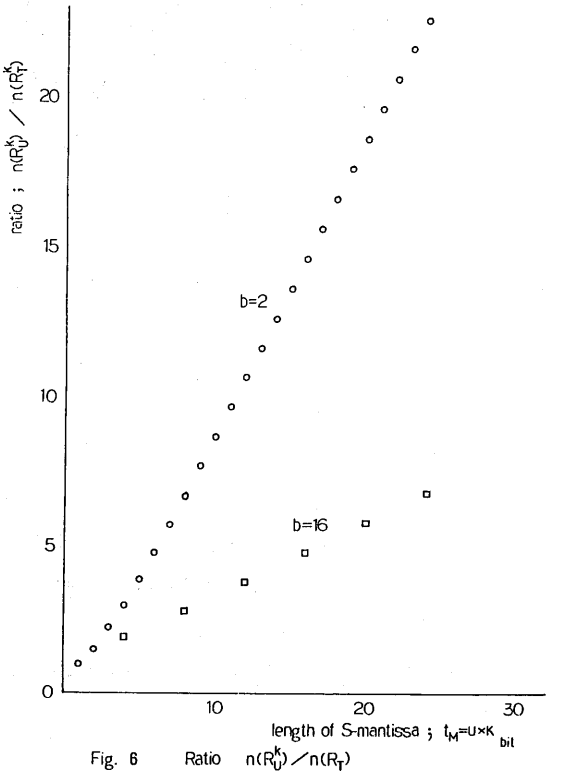


Fig. 6 Ratio $n(R_U^k) / n(R_T)$

(*3) オイラー関数 $\phi(d)$： 正の整数 d に対し，d 以下で d と互いに素なる正の整数の個数 [2]

3.3 FLP/R* 仮数のギャップ分布

FLP仮数の分布は均一であるが，Fig.4 からわかるように，FLP/R* 仮数の分布は不均一である．隣接する FLP/R* 仮数 m_1, m_2 間のギャップを g とすると，

$$m_1 = \frac{u_1}{v_1}, \quad m_2 = \frac{u_2}{v_2}, \quad m_1 < m_2$$

$$g = m_2 - m_1 = \frac{u_2 \cdot v_1 - u_1 \cdot v_2}{v_1 \cdot v_2}$$

m_1, m_2 は FLP/R* 仮数であるから，その分母 v_1, v_2 は各々 v_j $(j=0,1,\cdots,k)$ のいずれか一つの約数である．したがって，最小ギャップ g_{MIN}，最大ギャップ g_{MAX} が次のように求まる．

$$g_{MIN} = \frac{1}{b^k \cdot (b^k - 1)} \quad ; \text{区間} \left[m_1, m_2\right] = \left[\frac{1}{b^k}, \frac{1}{b^k - 1}\right], \left[\frac{b^k - 2}{b^k - 1}, \frac{b^k - 1}{b^k}\right]$$

$$g_{MAX} = \frac{1}{b^k} \quad ; \text{区間} \left[m_1, m_2\right] = \left[0, \frac{1}{b^k}\right], \left[\frac{b^k - 1}{b^k}, 1\right]$$

さらに，平均ギャップ g_{AVE} は前述の $n(R_U^k)$ から

$$g_{AVE} = \frac{1}{n(R_U^k)}$$

となる．ギャップ分布状態を Fig.7 に示す．

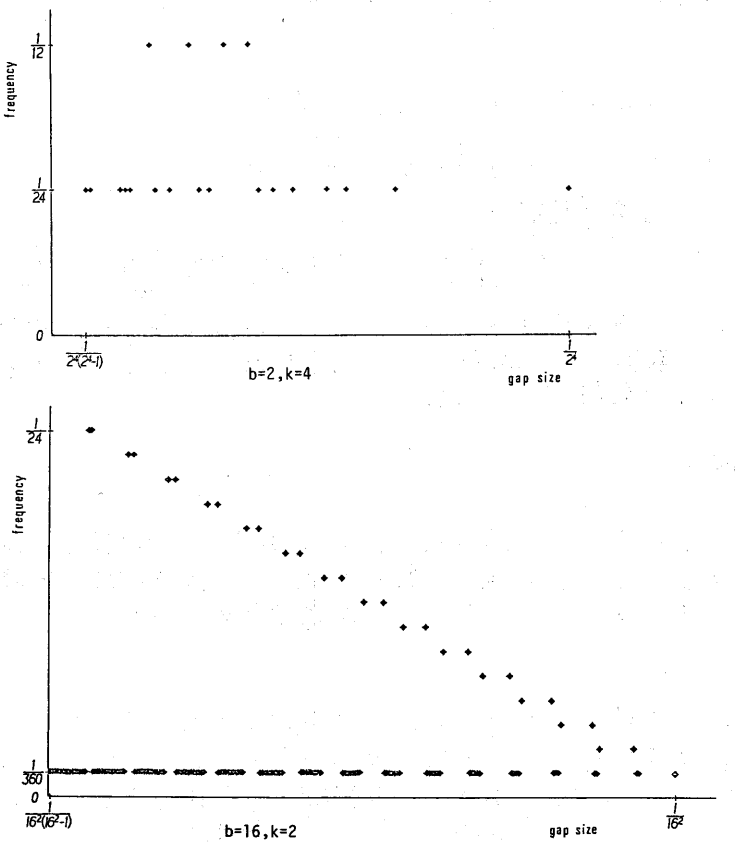


Fig.7 Gap Distribution

4. FLP/R* 演算

FLP/R* 演算系は，周期性の表示が従来のガードディジットの提供 [16,17] に対応するという概念に基づいている．

k桁S-仮数部から成るFLP/R* 仮数の最小ギャップは，2k桁仮数部のFLP仮数のギャップより大きい．即ち，

$$g_{MIN} = \frac{1}{b^k(b^k-1)} < \frac{1}{b^{2k}}$$

これは，すべてのFLP/R* 仮数は倍精度の下で互いに区別できることを意味する．そこで，FLP/R* 演算は倍精度演算を基調とし，その前処理として "循環節展開" 後処理として "循環型丸め" を行なう．

4.1 循環節展開

倍精度演算のためには，オペランドを倍精度（ただし被除数は四倍精度）に展開する必要がある．各オペランドの仮数の基数表現を循環表示部Rの内容を用いて倍精度（又は四倍精度）に展開することを "循環節展開" と称し，その過程をFig. 8 に示す．

循環周期が j である仮数 m のS-仮数部の状態が

$$m = .d_1 \cdots d_{(k)j)} \overline{d_{(kj+1)} \cdots d_k} \quad (b)$$

であるとすると，循環節展開によって

$$m = .\underbrace{d_1 \cdots d_{(kj)}}_{M_S} \underbrace{d_{(kj+1)} \cdots d_k \, d_{(kj+1)} \cdots d_k}_{M_G} \underbrace{d_{(kj+1)} \cdots d_k \cdots}_{} \quad (b)$$

のようになる．ここで基数表現の下位をG-仮数部 M_G と称する．被除数のG-仮数部は各々k桁の G_0-仮数部，G_1-仮数部そして G_2-仮数部から成るが，その他のオペランドのG-仮数部は G_0-仮数部のみから成る．

4.2 循環型丸め

倍精度演算により得られる演算結果はS-仮数部と G_0-仮数部から成る．この倍精度の演算結果を最も近いFLP/R* 仮数に近似して，S-仮数部と循環表示部Rの形式に変換する必要がある．この操作を "循環型丸め" と称し，その過程をFig. 9 に示す．

ある数がFLP/R* 数である時，その G_0-仮数部の状態はS-仮数部の状態に依存する．ゆえに，循環型丸めは次の二つの操作に分けられる．

(1) 候補の生成　：演算結果のS-仮数部から表現できる G_0-仮数部の状態を，近似候補 $\{C_i\}$ として挙げる．

(2) 最近傍数の選択　：候補 $\{C_i\}$ の中から，演算結果の G_0-仮数部に最も近いものを選択する．

今，演算結果 m' が

$$m' = .\underbrace{d_1 \cdots d_{-k}}_{M_S}\underbrace{d_{-(k+1)} \cdots d_{-2k}}_{M_{G_0}}$$

のように求まっている時，S-仮数部 M_S の状態から最大 $(k+2)$ 個の候補が定まる．

有限小数 ： C_{k+1}（切上げ），C_0（切捨て）

$$C_{k+1} = 10\cdots0 = b^{2k}$$

$$C_0 = \underbrace{0\cdots0}_{\leftarrow k \rightarrow} = 0$$

循環小数 ： $C_1, C_2, \cdots, C_k$ （C_j は循環周期 j の候補）

$$C_j = \underbrace{d_{-(k-j+1)}\cdots d_{-k}}_{\leftarrow\ k\ \rightarrow}\ \underbrace{d_{-(k-j+1)}\cdots d_{-k}\cdots}_{\rightarrow} \quad (b) = \left\lfloor \sum_{i=-1}^{-j} d_{i-(k-j)} \cdot \frac{b^{i+j+k}}{b^j - 1} \right\rfloor \quad (*4)$$

G_0 仮数部の値 V_{G_0} は

$$V_{G_0} = d_{-(k+1)} \cdots d_{-2k} \quad (b) = \sum_{i=-1}^{-k} d_{i+k} \cdot b^{i+k}$$

そこで，$|V_{G_0} - C_j|$ が最小となる候補 C_j を選択し，周期 $\gamma = j \bmod (k+1)$ を循環表示部に格納する．ただし，C_{k+1} が選択された場合，S-仮数部の最下位桁に 1 加算する．

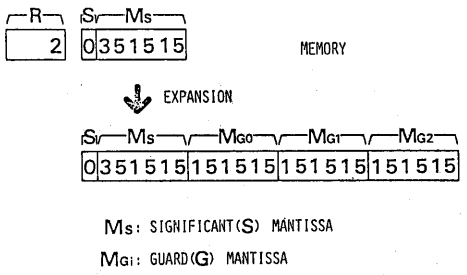


Fig.8 Expansion

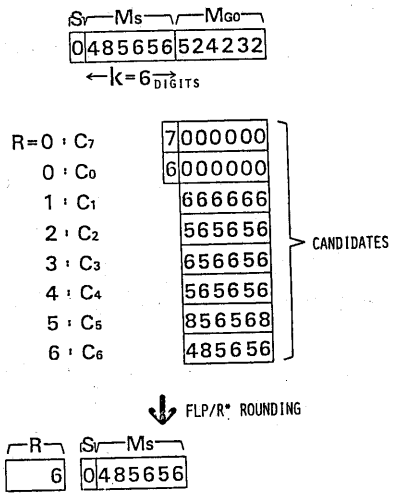


Fig.9 FLP/R* Rounding

EXAMPLE 1

$$\frac{85}{163} + \frac{71}{489} = \frac{2}{3}$$

b = 10

0.5214723···· + 0.1451942···· = 0.6666····

a) FLP (k = 5, t = 4×5 = 20)

52147 + 14519 → 66666

b) FLP/R* (k = 4, t = 4×4+3 = 19)

4 5214 + 0 1452 → 1 6666

EXPANSION ↓ ↑ FLP/R* ROUNDING

5214 5214 + 1452 0000 → 6666 5214

EXAMPLE 2

$$\frac{85}{163} \times \frac{163}{255} = \frac{1}{3}$$

b = 10

0.521472392··· × 0.63921568··· = 0.3333···

a) FLP (k = 5, t = 4×5 = 20)

52147 × 63921 → 33332 38387

b) FLP/R* (k = 4, t = 4×4+3 = 19)

4 5214 × 1 6392 → 1 3333

EXPANSION ↓ ↑ FLP/R* ROUNDING

5214 5214 × 6392 2222 → 3333 2379

(*4) $\lfloor x \rfloor$ は x 以下の最大の整数を表わす．

5. 検討

FLP/R* 仮数の分布は不均一であり，簡単な有理数ほどその隣接ギャップが広くなる傾向にある．したがって，例が示すように，演算結果がこのような簡単な有理数になるべき演算の場合，演算の途中で生ずる中間誤差がギャップに吸収され，打消されるという現象が起る．分数演算系における中数丸めは，簡単な有理数ほど演算における出現頻度が高いという仮定に基づいている [4]．そこで二つの演算系の複雑さ（価格）を比較してみる．

分数演算系の場合，分子又は分母の格納容量 l(bit) を単精度として，四則演算は
(1) 加減算の場合： 単精度乗算 MLT(l)×3，倍精度加算 ADD($2l$)×1
(2) 乗除算の場合： 単精度乗算 MLT(l)×2
を基調とする．そしてその組込み及び後処理として約分及び中数丸めによる近似操作を要する．二数 p, q ; $1 \leqq p, q < n\,(=2^N)$ に対する各操作は
(1) 約分 RED(N)： 二分GCDアルゴリズムに基づく約分アルゴリズム [9]
減算サイクル → ADD(N)×3 : $A_{AVE} = 0.70N - 0.5$, $A_{MAX} = N$ (回)
シフト → SHT(1) : $B_{AVE} = 1.41N - 2.7$ (回)
(2) 中数丸め MRD(N)： ユークリッド互除法による連分数展開打切り [4]
除算サイクル → DIV(N), MLT(N), ADD(N) : $C_{AVE} = \frac{12(\ln 2)^2}{\pi^2} \cdot \left(\frac{N}{2}\right)$ (回)
$C_{MAX} = \lceil 2.078 (\ln 2) \left(\frac{N}{2}\right) \rceil$
と解析されている．

一方，FLP/R* 演算系の場合，基数 b の下で S-仮数部の桁数 k を単精度とし，倍精度演算を基調とすることを先に述べた．その前/後処理は決定的操作である．
(1) 循環節展開 → SHT(u)×k （ただし除算は k） ここで $u = \lceil \log_2 b \rceil$ (bit)
(2) 循環型丸め → 生成 SHT(u)×k，選択 SUB($k \cdot u$)×2： $k+2$ (回)

k と l の対応関係は比較基準（例えば $n(R_S)$, g_{MIN}, t）を何にするかにより変わるが，仮数部全体長 t が等しいとすると

$$2l = k \cdot u + \lceil \log_2 (k+1) \rceil \quad ; \quad u = \lceil \log_2 b \rceil$$

比較例 $b=16$, $k=6$, $l=13$ 加算
(1) 分数 ($2 \times 13 = 26$ bit)
- 演算 : MLT(13)×3, ADD(26)×1
- 約分 組込み型 : RED(13)×1, RED(26)×1 ; RED(13) → ADD(13)×3×A_{AVE} 9, SHT(1)×B_{AVE} 16
- 約分 後処理型 : RED(26)×1 ; RED(26) → ADD(26)×3×18, SHT(1)×34
- 中数丸め : MRD(26)×1 ; MRD(26) → DIV(26), MLT(26), ADD(26)×C_{AVE} 8

(2) FLP/R* ($6 \times 4 + 3 = 27$ bit)
- 循環節展開 : SHT(4)×6
- 演算 : ADD(48)×1
- 循環型丸め : (SHT(4)×6, SUB(24)×2)×8

この例からわかるように，FLP/R* 演算系は演算自体が多少高価でも，近似操作が分数演算系に比べ大変低価格であり，その効果は大きい．さらにこの近似操作は並列化による高速化が可能である．

また，正規化と丸めの順序は誤差に影響を及ぼす問題であり，通常のFLP演算系では正規化→丸めがよいと言われる．しかし循環型丸めの場合，下位桁の上位桁の状態に依存するため，正規化の方向が順序問題に関係する．この解決を含め、演算系全体の設計が今後の課題である．

5. 結論

有理数実現のための新しい浮動小数点演算系FLP/R* について述べた．不均一な仮数分布から起る中間誤差の打消し現象は大変興味深く，しかも分数演算系に比べ近似操作が非常に低価格であることがこの演算系の大きな特徴である．

謝辞

本研究に際し御指導下さった慶應義塾大学理工学部相磯秀夫教授に深く感謝いたします．また日頃有益な御助言を授けて下さる同大学同学部所真理雄講師に心より感謝いたします．そして発表を勧めて下さり貴重な御意見を下さった京都大学数理解析研究所一松信教授に厚く御礼申し上げます．

参考文献

[1] Yoshida,K. : "A Research on Error Free Machines", (M.S. Thesis), Computer Science Report ARITHMETIC-82-1, Faculty of Science and Technology, Keio University, March 1982

[2] 高木貞治 : "初等整数論講義", 共立出版, 1931

[3] Mahler,K. : "Introduction to P-adic numbers and their functions" , CAMBRIDGE UNIV. PRESS (1973)

[4] Knuth,D.E. : "The Art of Computer Programming", SECOND EDITION, Vol.2 / Seminumerical Algorithms ADDISON WESLEY (1981)

[5] Matula,D.W. : "Fixed-Slash and Floatig-Slash Rational Arithmetic" , Proc. the 3rd IEEE Symp. Computer Arithmetic (1975), pp.90-91

[6] Matula,D.W. and Kornerup,P. : "Feasibility Analysis of Binary Fixed -Slash Number System", Proc. the 4th IEEE Symp. Computer Arithmetic (1978) pp.29-38

[7] Kornerup,P and Matula,D.W. : "Feasibility Analysis of Fixed-Slash Rational Arithmetic", Proc. the 4th IEEE Symp. Computer Arithmetic (1978) , pp.39-47

[8] Kornerup,P. and Matula,D.W. : "An Integrated Rational Arithmetic Unit", Proc. the 5th IEEE Symp. Computer Arithmetic (1981), pp.233-240

[9] Irwin,M.J. and Smith,D.W. : "A Rational Arithmetic Processor", Proc. the 5th IEEE Symp. Computer Arithmetic (1981), pp.241-244

[10] Hwang,K. and Chang,T.P. : "An Interleaved Rational/Radix Arithmetic System for High Precision Computations", Proc. the 4th IEEE Symp. Computer Arithmetic (1978), pp.15-24

[11] Krischnamurthy,E.V.,Rao,T.M. and Subramanian,K. : "Finite Segment p-adic Number Systems with applications to Exact Computation", Proc. Indian Academy of Science vol.81A NO.2 (1975), pp.58-79

[12] Krischnamurthy,E.V.,Rao,T.M. and Subramanian,K. : "P-adic Arithmetic Procedures for Exact Matrix Computations", Proc. Indian Academy of Science vol.82A No.5 (1975), pp.165-175

[13] Krischnamurthy,E.V. : "Matrix Processors Using p-adic Arithmetic for Exact Linear Computations",
Proc. the 3rd IEEE Symp. Computer Arithmetic (1975) pp.92-97
IEEE Trns. Computer C-26 No.7 (July 1977), pp.633-639

[14] Horspool,R.N.S. and Hehner,E.C.R. : "Exact Arithmetic Using A Variable-Length p-adic Representaion", Proc. the 4th IEEE Symp. Computer Arithmetic (1978), pp.10-14

[15] Hehner,E.C.R. and Horspool,R.N.S. : "A New Representation of The Rational Number for Fast Easy Arithmetic", SIAM J. Computing vol.8 No.2 (May 1979), pp.124-134

[16] Cody,W.J. : "Static and Dynamic Numerical Characteristics of Floating Point Arithmetic", IEEE Trans. Computer vol.C-22 (June 1973) pp.508-601

[17] Kuck,D.J., Paker,D.S. and Sameh,A.H. : "Analysis of Rounding Methods in Floating-Point Arithmetic", IEEE Trans. Computer vol.C-26 (July 1977) pp.643-650